SIEMENS

SIMATIC S5

Positioning Module
IP 247
for Stepper Motors

Manual

Order No.: 6ES5998-5SB22
Release 02

Contents

Siemens Aktiengesellschaft

6ES5998-5SB22
Elektronikwerk Karlsruhe

Printed in the Federal Republic of Germany

Guidelines for Handling Electrostatically Sensitive Devices (ESD)

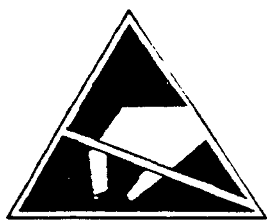
1 What is ESD?

VSLI chips (MOS technology) are used in practically all SIMATIC S5 and TELEPERM M modules. These VLSI components are, by their nature, very sensitive to overvoltages and thus to electrostatic discharge:

They are therefore defined as
"Electrostatically Sensitive Devices"

"ESD" is the abbreviation used internationally.

The following warning label on the cabinets, subracks and packing indicates that electrostatically sensitive components have been used and that the modules concerned are susceptible to touch:



ESDs can be destroyed by voltage and energy levels which are far below the level perceptible to human beings. Such voltages already occur when a component or a module is touched by a person who has not been electrostatically discharged. Components which have been subjected to such overvoltage cannot, in most cases, be immediately detected as faulty; the fault occurs only after a long period in operation.

An electrostatic discharge
- of 3500 V can be felt
- of 4500 V can be heard
- must take place at a minimum of 5000 V to be seen.

But just a fraction of this voltage can already damage or destroy an electronic component.

The typical data of a component can suffer due to damage, overstressing or weakening caused by electrostatic discharge; this can result in temporary fault behavior, e.g. in the case of

- temperature variations,
- mechanical shocks,
- vibrations,
- change of load.

Only the consequent use of protective equipment and careful observance of the precautions for handling such components can effectively prevent functional disturbances and failures of ESD modules.

2 When is a Static Charge Formed?

One can never be sure whether the human body or the material and tools which one is using are not electrostatically charged.

Small charges of 100 V are very common; these can, however, very quickly rise up to 35000 V.

Examples of static charge:

- Walking on a carpet	up to 35000 V
- Walking on a PVC flooring	up to 12000 V
- Sitting on a cushioned chair	up to 18000 V
- Plastic desoldering unit	up to 8000 V
- Plastic coffee cup	up to 5000 V
- Plastic bags	up to 5000 V
- Books, etc. with a plastic binding	up to 8000 V

3 important Protective Measures against Static Charge

. Most plastic materials are highly susceptible to static charge and must therefore be kept as far away as possible from ESDs.

. Personnel who handle ESDs, the work table and the packing must all be carefully grounded.

4 Handling of ESD Modules

One basic rule to be observed is that electronic modules should be touched by hand only if this is necessary for any work required to be done on them. Do not touch the component pins or the conductors.

Touch components only if

- the person is grounded at all times by means of a wrist strap

or

- the person is wearing special anti-static shoes or shoes with a grounding strip.

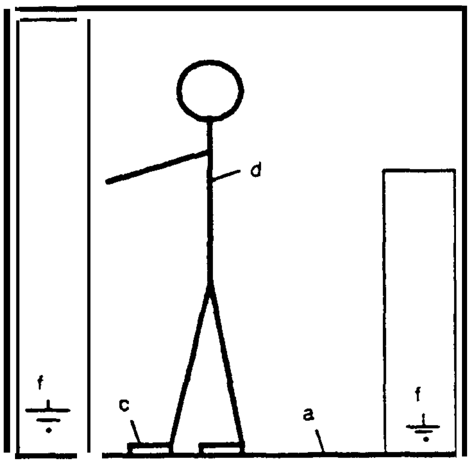
Before touching an electronic module, the person concerned must ensure that (s)he is not carrying any static charge. The simplest way is to touch a conductive, grounded item of equipment (e.g. a blank metallic cabinet part, water pipe, etc.) before touching the module.

Modules should not be brought into contact with insulating materials or materials which take up a static charge, e.g. plastic foil, insulating table tops, synthetic clothing, etc.

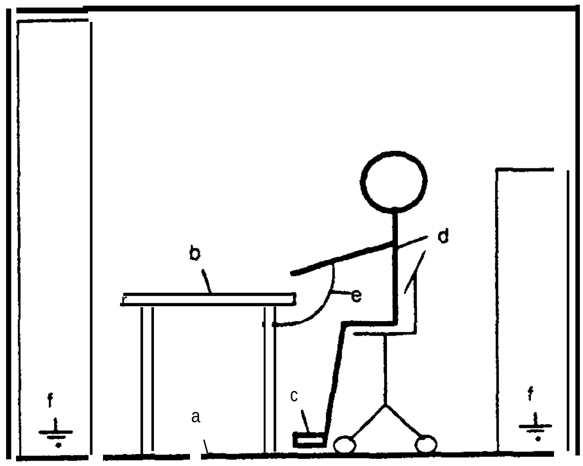
Modules should only be placed on conductive surfaces (table with anti-static table top, conductive foam material, anti-static plastic bag, anti-static transport container).

Modules should not be placed in the vicinity of monitors, TV sets (minimum distance from screen > 10 cm).

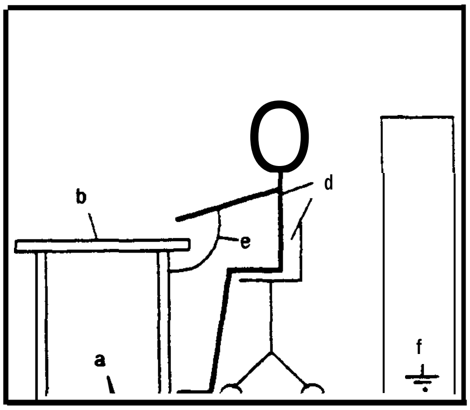
The diagram below shows the required protective measures against electrostatic discharge.



Standing position



Standing/sitting position



Sitting position

a Conductive flooring
b Anti-static table
c Anti-static shoes
d Anti-static coat
e Grounding wrist strap
f Grounding common of the cabinets

5 Measurements and Modification to ESD Modules

- Measurements on modules may only be earned out under the following conditions:

  The measuring equipment is grounded (e.g. via the PE conductor of the power supply system) or

  when electrically isolated measuring equipment is used, the probe must be discharged (e.g. by touching the metallic casing of the equipment) before beginning measurements.

- Only grounded soldering irons may be used.

6 Shipping of ESD Modules

Anti-static packing material must always be used for modules and components, e.g. metalized plastic boxes, metal boxes, etc. for storing and dispatch of modules and components.

If the container itself is not conductive, the modules must be wrapped in a conductive material such as conductive foam, anti-static plastic bag, aluminium foil or paper. Normal plastic bags or foils should not be used under any circumstances.

For modules with built-in batteries ensure that the conductive packing does not touch or short-circuit the battery connections; if necessary cover the connections with insulating tape or material.

Contents

1 Notes

1.1 Notes on Using the Manual

This manual describes a system for position control of three independent drives.

The system comprises the following components:

- IP247 positioning module
- COM 247 communications software
- standard function blocks FBI 64 and FBI 65

The IP247 positioning module represents the link between your plant and the programmable controller (PC). The standard function block FBI 64 is used for operating and monitoring, and FB165 for assigning parameters to the IP247. With the programming package COM247, you can generate, save and print machining programs and machine data. COM247 is also used to test the IP247 online with the plant connected.

This manual refers to the following products:

The **module IP247**

- The version for ventilated operation, single width, without additional heat sink, order number6ES5247-4UA31.
- The version for non-ventilated operation, double width, with additional heat sink, order number6ES5247-4 UA41.

The communications software COM247

- From release A02.0,
  order number 6ES5 895- 5SB22.

The standard function blocks FB164 and FB165

- For the S5-115,
  order number 6ES5 845- 8TA01.
- For the S5-135 with CPU922 or 928,
  order number 6ES5 842- 8TB01.
- For the S5-150,
  order number 6ES5 844- 8TA01.
- For the S5-155,
  order number 6ES5 846- 8TA01.

The manual is structured to allow you to become familiar with the system and can later be used as a reference work to look up specific points.

Part 2: ‘Fundamentals of Positioning” introduces terms you require to work with the positioning module, e.g.:

- machine data,
- machining programs,
- axis attributes,
- messages,

By familiarizing yourself with these terms, you will also gain a better understanding of the functions and concept underlying the IP247.

Part 3: “Hardware” deals with the hardware requirements necessary to use the IP247 in a variety of situations. This covers the following topics:

- connections,
- jumper settings,
- switch settings,

Part 4: ‘Functions” introduces you to the operating concept of the IP247. This is based on the following:

- operating functions,
- monitoring functions,

These functions and their effects are described in this part,

Part 5: **"COM247 Communications Software"** explains how to assign parameters to the I P247 and how to test it using this software package. The following aspects of COM247 are covered:

- generating machine data and machining programs,
- saving the generated data,
- storing the data in the memory of the PG,
- printing machine data records and machining programs,
- testing the IP247 with the plant connected,

Part 6: **“Standard Function Blocks FB164 and FB165"** describes the assignment of parameters, operation and monitoring of the IP247 by the CPU. This description discusses the following:

- FBI **64** for operating and monitoring the IP247,
- FBI 65 for assigning parameters to the IP247,
- the structure of the machine data and machining programs in a STEP 5 block,
- examples of parameter assignment for an axis.

Part 7: ‘Planning, Installation and Service” contains the following:

- notes on planning the drive and the machine data,
- guidelines for installing and starting the module,
- an overview of troubleshooting routines,
- instructions for diagnosing problems.

Part **8:** “Index” lists the most important terms used in the handbook.

1.2 Important Notes on Safety

Note

Before starting up the system, the plant must be equipped with emergency stop limit switches , which directly affect the power supply.

If the plant is operated from the PG, the emergency stop switch, to switch off the whole plant must be accessible from the PG.

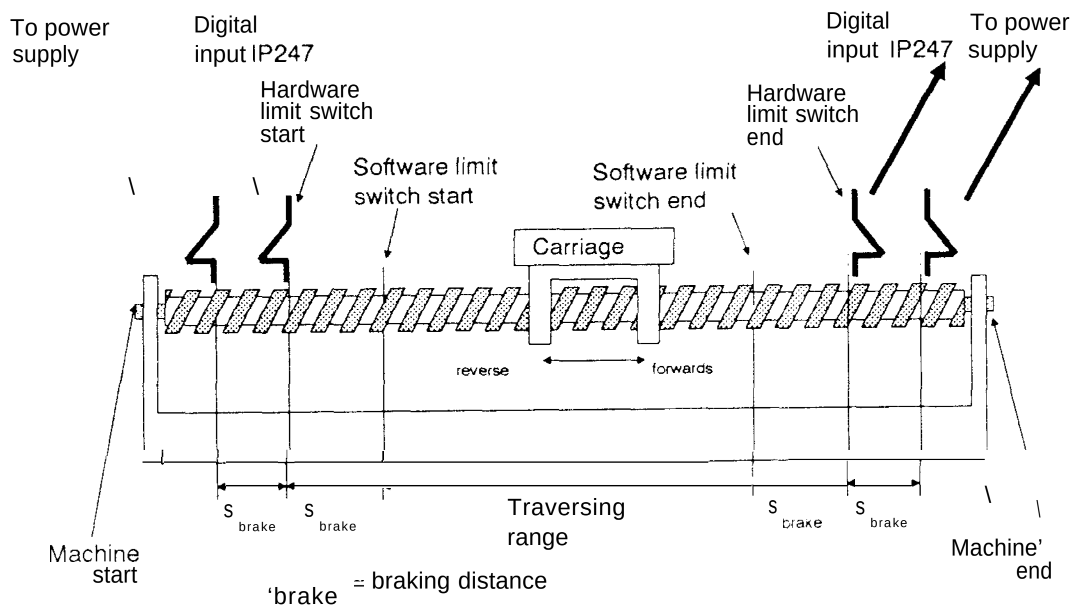
If the positioning module is linked into the programmable controller, an emergency stop switch must be integrated in the control panel used for operation.

Despite extensive measures both in development and production to achieve the high reliability of SI MATIC S5, errors can never be completely excluded. Whenever an error could lead to damaged equipment or even personal injury, all measures must be taken to ensure a safe configuration according to the pertinent regulations.

The commissioning and starting up of a drive always demands particular care, The possibility that the drive might start moving unexpectedly for whatever reason can never be fully excluded. Such a movement can, for example, result from accidental triggering of commands or from faults in the electronics.

To be able to stop the, in some cases, enormous energy of a moving drive, emergency stop limit switches at the ends of the traversing range, which switch off the power supply directly, must always be present, Depending on the type of drive, these limit switches must also be combined with mechanical brakes and buffers to prevent any possible damage.

The positioning module has inputs for two limit switches for each axis, however, these can never be a substitute for emergency stop limit switches directly connected to the power supply.



Fig, 1/1 Linear axis with limit switches

The positioning module also has a further stop input.

There is a digital input “external start/stop" for each axis, which must be wired up before starting the system. This allows an axis to be stopped at any time regardless of whether it is being operated from the PC or from the PG. Once again, this input cannot be regarded as a substitute for emergency stop limit switches. Remember that depending on prior operation, this input can also have the effect of an internal start signal.

When operating the plant with a PG, remember the following points:

If an axis is started from the programmer during installation and initial start-up, it continues to move even if you exit the test display in which the start was triggered or switch off the programmer. Depending on the particular operation, the axis only stops when the target or a limit switch is reached. It is therefore strongly advised that you remain in the test display while the axis is traversing.

When characters are entered at the programmer keyboard, they are written to a buffer. If characters are entered more quickly than they can be processed, they are stored temporarily in this buffer. This can become noticeable in the test display of COM247 when entering commands, if, for example, the commands “forward” and “reverse” are entered in quick succession. The execution then lags behind the input. A **stop command is therefore only executed, when all the commands stored before it in the character buffer have been processed.**

The last job is completely executed unless it contradicts the second to last job.

2 Fundamentals of Positioning

2.1 Introduction

This part introduces you to the IP247. It provides you with certain information about positioning and briefly describes the function of the IP247 positioning module and its firmware, which represents the heart of the module.
The following terms, which must be familiar when working with the IP247, are then explained:

- machine data,
- machining programs and
- axis attributes.

Finally, this part provides you with information about the digital inputs and digital outputs made available by the IP247 and an explanation of the limit switch concept and its effects with the IP247.

2.2 A Brief Introduction to the IP247

Using the positioning module IP247, you can move and position three independent axes. From the positioning jobs and the machine data, the module calculates pulse trains which are output to the connected stepper motor power unit, The number of pulses decides the distance travelled, the pulse frequency corresponds to the speed, A direction signal is also output to specify the direction of the movement,

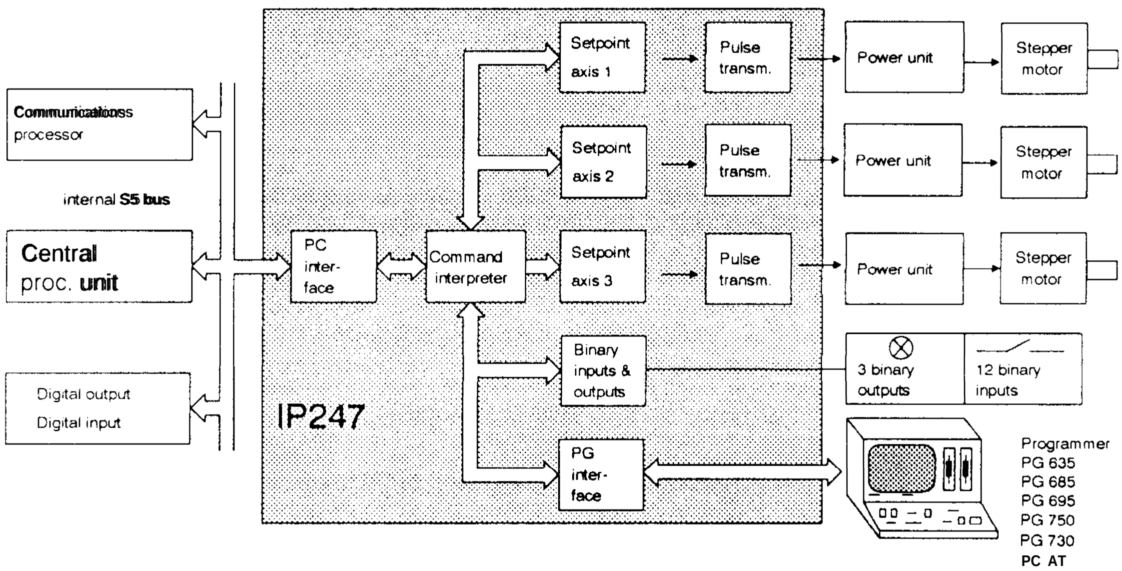


Fig. 2/1 The IP247 in the SIMATIC S5system

Due to its adaptability, the module must have parameters assigned to it. Parameter assignment is simple and is performed at the monitor of a programmer (PG) using the software package COM247. You can assign parameters to the IP247 via the PC interface, however, without the user-friendly support of the COM247 software package,

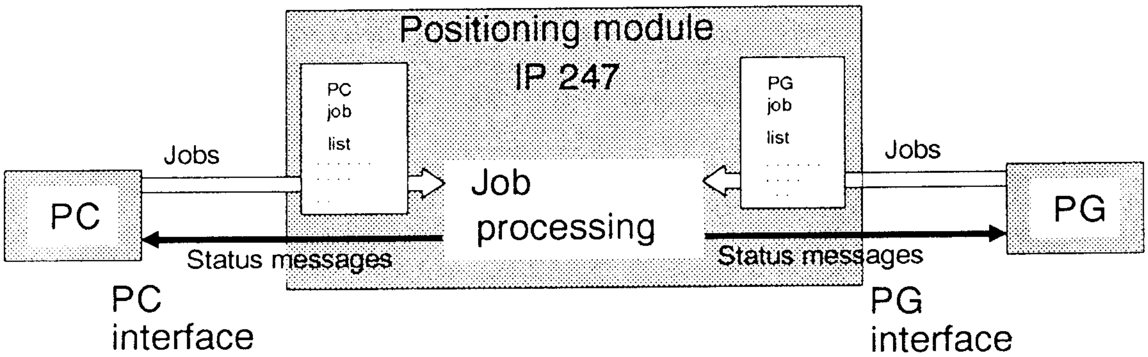
In the test mode, the COM247 software package allows you to test all the functions of the IP247 and therefore the positioning functions of your plant.

Two standard function blocks FBI 64 and FBI 65 are used to incorporate the functions of the IP247 in a user program, allowing all the functions of the IP247 to be executed from the CPU. These are stored in an EPROM cartridge in the CPU. The data handling blocks for communications processors are subordinate to the function blocks.

The module can be operated both from a PG and from a PC, however, the functions of the interfaces differ from each other. With the software package COM247, the PG is used for convenient parameter assignment, starting up and testing the module. The PC interface is used to execute the functions of the IP247 during normal plant operation.

Jobs can be sent to the IP247 via the PG interface and via the PC interface simultaneously.

When requested, the IP247 sends status messages via both interfaces.



Fig, 2/2 Communication with the PC and PG

Each positioning operation of the IP247 is based on a machine data record specific to the axis, which must be transferred to the memory of the I P247 via one of the two interfaces. An axis is only functional when a correct machine data record exists on the module. By making entries in this data record, you stipulate the electrical and mechanical limits of your plant. These include the maximum rate of frequency increase of the axis, the maximum pulse frequency, the permitted traversing range of your axis and the type of axis (linear or rotary).

With the I P247, positioning jobs can be issued in two ways, as follows:

- machining programs, i.e. a connected series of traversing jobs, dwell times, corrections and switchovers, which are stored in the memory of the IP247,

- single jobs, sent to the IP247 via an interface.

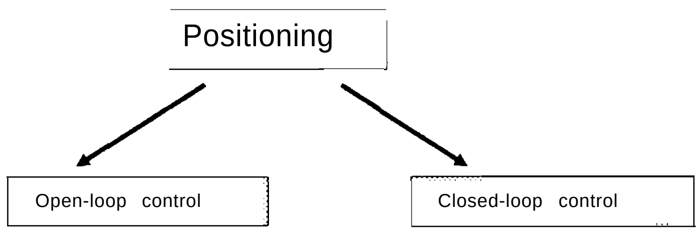
You can input and delete a machining program both via the PG and the PC interface.

It is possible to take into account changing tool lengths and to execute zero point offsets.

2.3 Positioning Axes

2.3.1 What is Positioning?

Positioning means approaching a previously specified point or previously specified coordinate automatically following a procedure established by parameter assignment. Such an operation can be controlled by either closed-loop or open-loop control systems.



Fig, 2/3 Types of positioning

When using closed-loop control, the physical variable to be controlled is measured and compared and matched to another value.

Once parameters can be assigned for the positioning operation, a setpoint generator is necessary, regardless of whether closed-loop or open-loop control is to be used. This setpoint generator supplies an output value, which depends both on the difference between the current position of the axis and the required target point, as well as on the parameters, e.g. speed, acceleration or deceleration. The more opportunities for parameter assignment and for modifying parameters during the positioning operation provided by the setpoint generator, the more complex and comprehensive is its structure. In the simplest version, the output value of the setpoint generator is switched on and off. Specifying the maximum speed and maximum acceleration and deceleration according to the mechanical capabilities of the plant improves the efficiency of the operation.

Positioning control with the IP247 is open-loop.

The actual position of the drive is not monitored. The actual position specified by the IP247 is calculated from the axis data and number of pulses output.

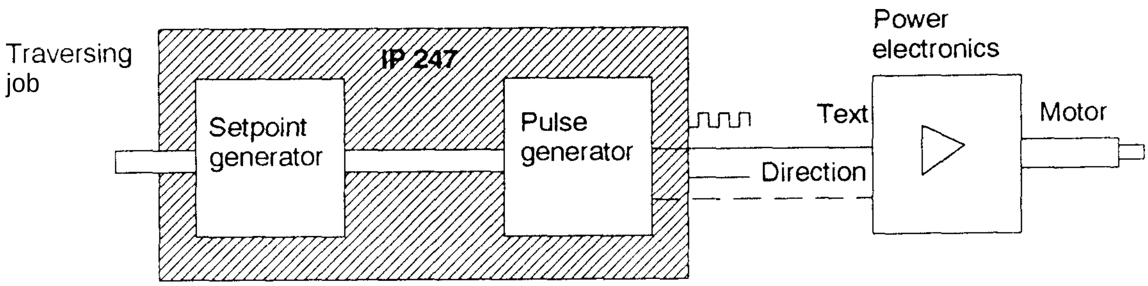


Fig. 2/4 Open-loop position control

Stepper motors are drives which rotate by going through a sequence of individual step angles. If the stepper motor receives a pulse, it revolves through a fixed angle; if the number of pulses and their frequency is increased, a continuous rotation is gradually achieved.

A pulse train is applied to the stepper motor power circuitry; the number of pulses determines the distance travel led, the frequency of the pulses determines the speed.

Example

- Stepper motor with 500 steps/revolution, per step the motor travels through an angle of 0.72°.
- If 10,000 pulses are output, the stepper motor rotates through 20 complete revolutions.
- If the pulses are output at a frequency of 1 kHz, the motor requires 10 seconds for the 20 revolutions.

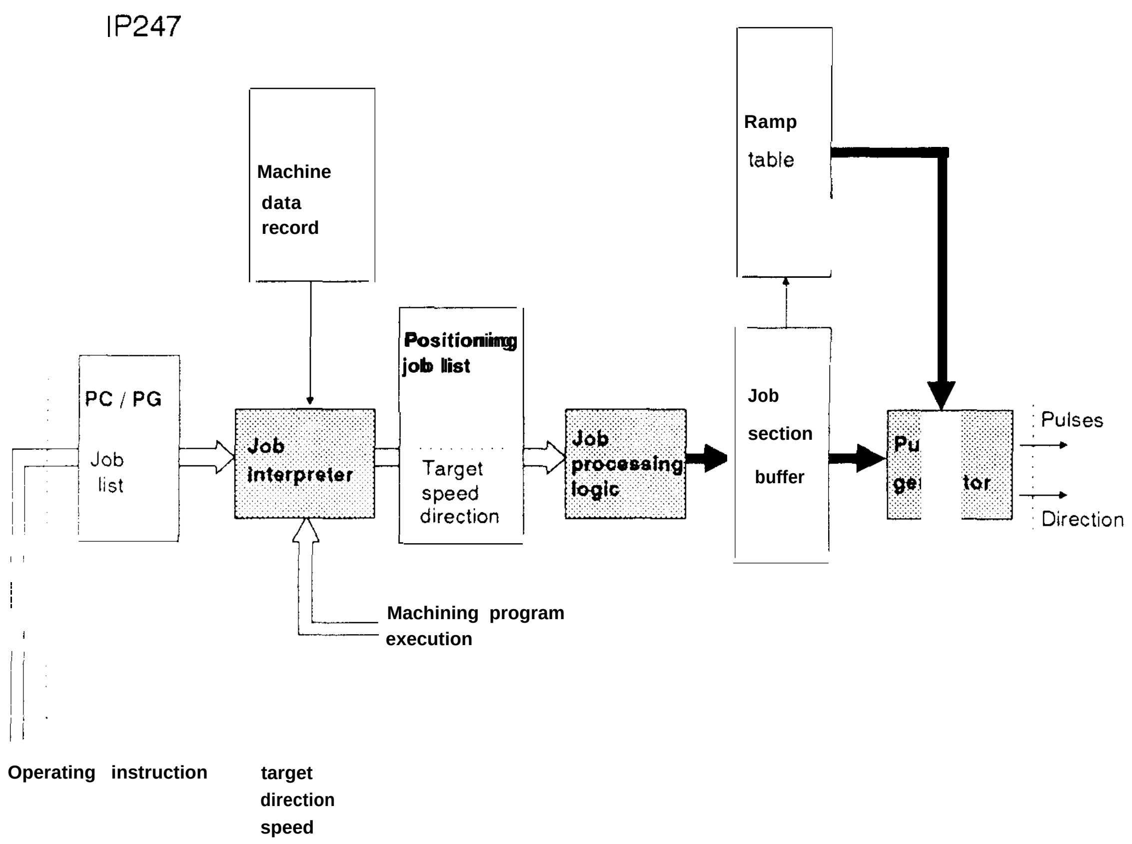
A direction signal is required to control the direction.

Based on the technical data of the plant (machine data) and the required traversing job (target position, speed), the IP247 positioning module supplies a corresponding pulse train and the required direction of travel.

The stepper motor drive (stepper motor and power circuitry) converts these pulse trains into a traversing movement.

The advantage of positioning with stepper motors is that the motor remains at a fixed position when it is at a standstill, In contrast to this, the drive in closed-loop systems always oscillates slightly.

2.4 How Does the IP247 Execute a Positioning Job?



Fig, 2/5 Structure of job processing

Since there is no feedback of the physical actual value of the system, and it is therefore not possible to compensate for any step losses, it is extremely important that stepper motors are correctly dimensioned.

> ⚠ **Note**
>
> Incorrect dimensioning of the stepper motor can lead to a loss of steps and therefore to incorrect positioning.

2.5 Machine Data and their Structure

Before a positioning module such as the IP247 can execute a positioning operation automatically, it must be provided with information **about the connected drive.** This information is known as machine data. Machine data is stored in a data block along with other parameters. This has a constant length. Machine data can be divided into the following parameter groups:

- specific to the power unit
- specific to the stepper motor
- specific to the plant
- specific to the operations
- specific to the machining program

Using the COM247 software package, machine data records can be generated efficiently and easily at the programmer and transferred to the positioning module. Once on the module they can be read again, corrected or deleted. Both COM247 and the module perform consistency checks. If machine data are sent to the IP247 via the PC interface, they are only checked by the IP247. It is therefore possible to assign bad machine data to an axis on the module.

Bad in this case means that either data in the machine data record exceed the stipulated limit values, or that certain combinations of machine data are not permitted.

If a bad machine data record is transferred to the positioning module, the IP247 signals the error "error in machine data" via the PG and PC interface. The type of error itself, e.g. "wrong axis/module number" is stored by the firmware of the positioning module in the machine data block (=> Part 7, "Planning, Installation and Service"). If you enter the machine data using the COM247 software package, the type of machine data error is displayed in plain text in the error message line on the PG. The message "error in machine data" is then overwritten.

If you wish to position using all the axes of the IP247, a machine data block (DB) must be stored on the module for each axis. You can assign a machine data block with the same DB number to different axes.

If no correct machine data are stored on the IP247 for an axis, the axis is not operational. If operating instructions are sent to the axis, the job is rejected with the error message "wrong or no machine data".

If you edit the machine data record using the COM247 software package, all the required machine data are requested in plain text using a menu technique. Following the input field, the default dimensional unit and the permitted range of values is displayed. This is explained in detail in Part 5"COM247 Communications Software".

Since no special software is available for planning the machine data in the CPU, the following description of the individual machine data includes the data formats as required for entry or storage in the CPU. A table in Part 6 "Standard Function Blocks FBI 64 and FB1 65" provides an overview of these formats.

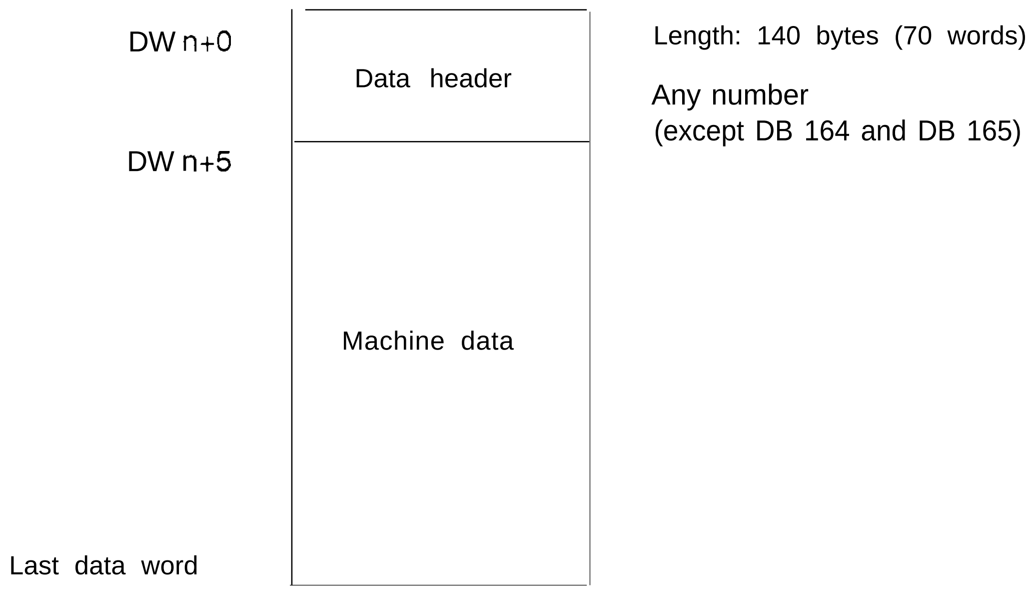
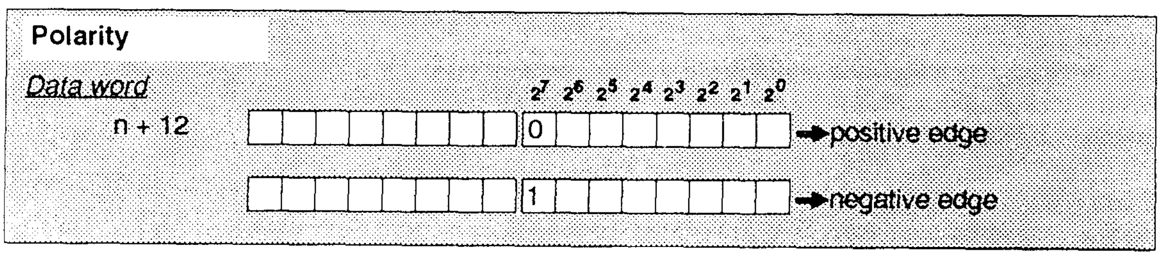


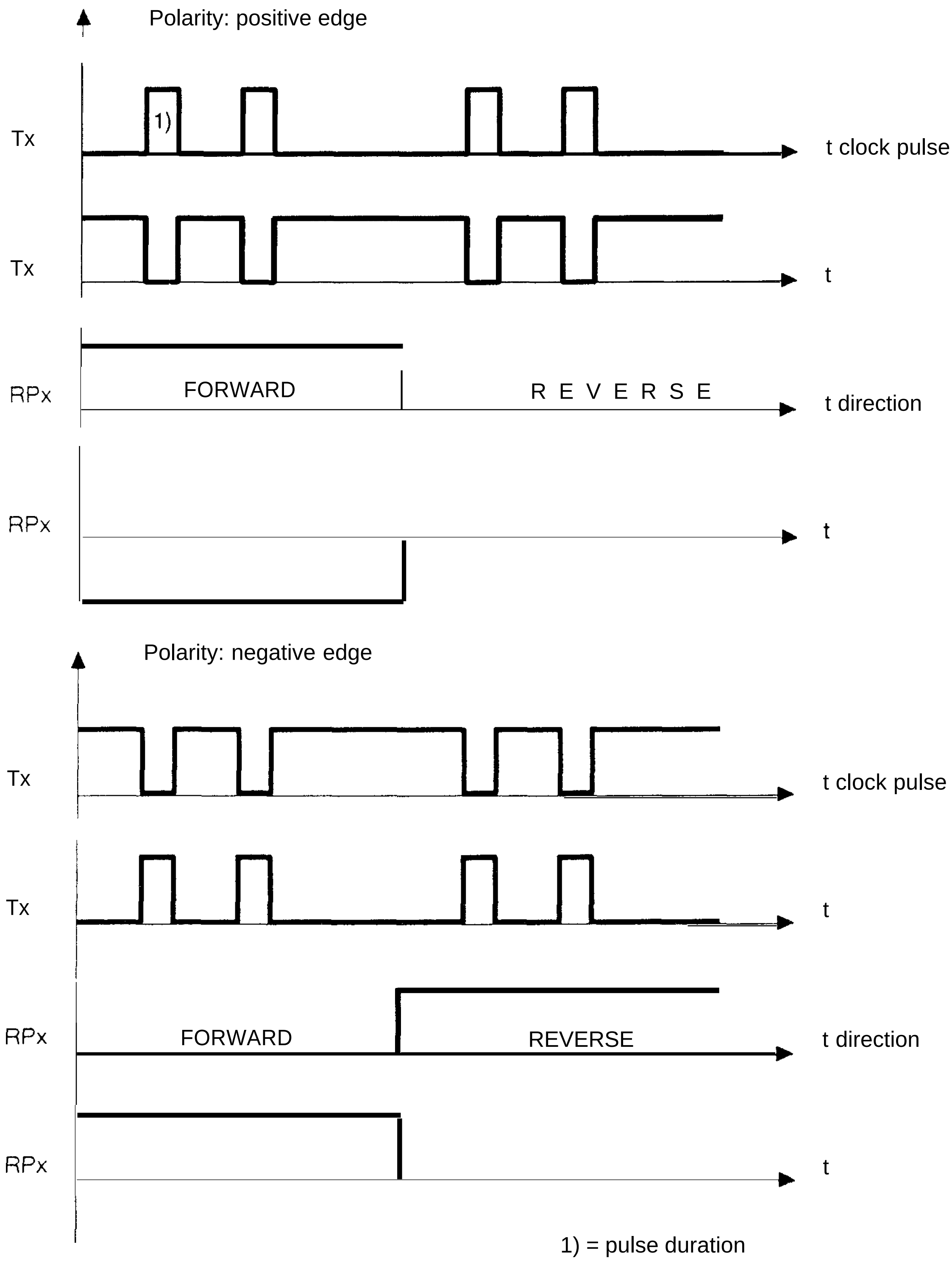
Fig. 2/6 The machine data record in the **CPU**

2.5.1 Machine **Data for the Power Unit**

2.5.1.1 **Polarity**



The manual for the power unit will tell you whether or not the power unit reacts to the negative or positive edge at its pulse input. With this information, you can set the level (active high or active low) of the outputs of the **IP247** using the machine data “polarity”.



Fig, 2/7 Output level

By stipulating the polarity, you also decide the direction of forward and reverse movements, At the end of the axis which is approached in a “forward” direction, there are software and hardware end limit switches; at the end approached in the “reverse” direction there are software and hardware start limit switches.

Note

Once the drive has been installed and started up correctly, this machine data must not be changed, otherwise the wiring of the limit switches and parameters for the software limit switches must also be changed.

2.5.1.2 Pulse Duration

Pulse duration		
Data word n + 11	☐☐☐☐☐☐☐☐ ☐☐☐☐☐☐☐☐	Range of values 1....31

The manual for your power unit specifies the minimum pulse duration required for trouble-free operation.

The pulse duration can be set in the intervals

1 μs ≤ minimum pulse duration < 0.5 x period of the maximum frequency

or

1 μs ≤ minimum pulse duration < 31μs.

2.5.2 Machine Data for the Stepper Motor

An important characteristic of a stepper motor is the mode “full step or half step”. This is usually a hardware setting on the power unit. This characteristic is not specified a separate machine data, but is taken into account in the “pulse/revolutions” and “pulse pattern number” machine data,

2.5.2.1 Pulses per Revolution

Pulses/revolution		
Data word n + 14	☐☐☐☐☐☐☐☐ ☐☐☐☐☐☐☐☐	Range of values 12...1000

This machine data specifies the number of steps of the motor, The number of steps for the full step mode is usually specified on the rating plate of the motor. If this is not the case, you can calculate the number of steps for the **full step mode from** the step angle, as follows:

pulses per revolution = 360/step angle

I n the **half step mode,** you must double this number.

2.5.2.2 Number of Excitation Patterns

No. of excitation patterns

Data word

n + 13 Range of values 4...40

The phases of a stepper motor must be excited in a sequence which the rotor can follow step-by-step.

The number of possible phase excitations is calculated as follows for the **full step mode:**

number of excitation patterns = 2 x number of phases

For the **half step mode** this number must be doubled,

2.5.3 Machine Data for the Plant

2.5.3.1 Axis Type (Linear or Rotary Axis)

Axis type

Data word

Data word		2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
n + 53		1								→ Rotary axis
n + 53		0								→ Linear axis

All three axes of a module can be assigned parameters as linear axes or as rotary axes independently, From version A02.0 onwards, the software package COM247 supports rotary axes.

The assignment of the following parameters depends on whether you selected a linear axis or a rotary axis as the ""axis type"

- the software limit switches or range limits and
- the resolution.
-

Operator input also depends on the type of axis

- with incremental approach,
- with zero or tool length offsets and
- in the automatic mode,

These differences are explained in detail in the appropriate section, Some fundamental aspects are, however, discussed below.

2.5.3.2 The Linear Axis

A linear axis or open axis is an axis with a limited traversing range. The traversing range of a linear axis is limited with the IP247 by assigning the software limit switches. This is effective only when the reference point exists.

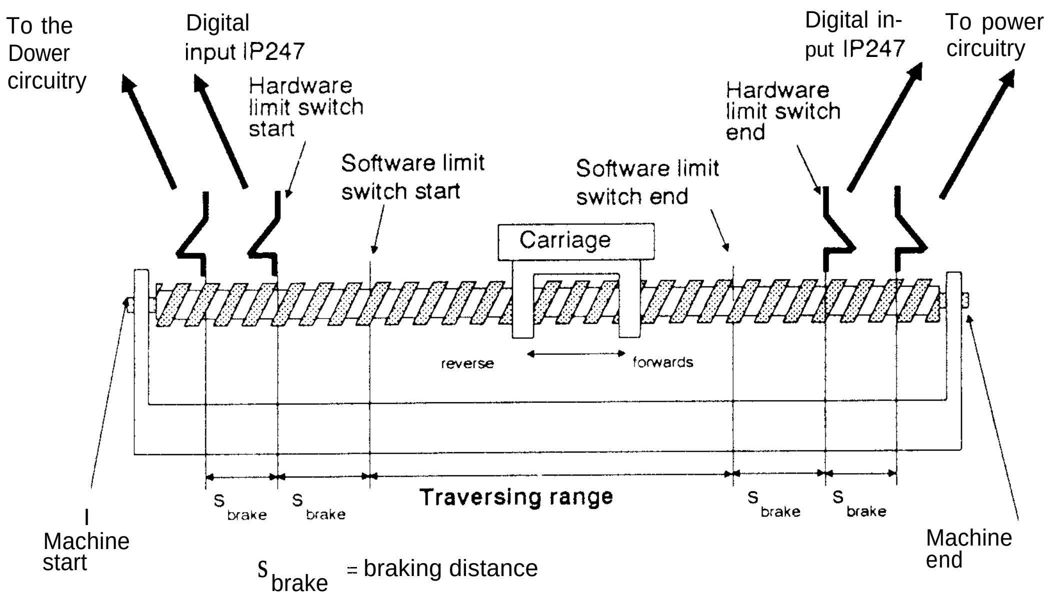


Fig. 2/8 Linear axis with limit switches

If the permitted traversing range of a linear axis is exceeded, the equipment will almost always be damaged. For this reason, particular care must be taken that the axis type and the assignment of limit switches are correct.

> **Note**
>
> If the axis type “rotary axis” is accidentally selected instead of a linear axis, the values assigned in the data double words DD n+29 and DD n+31 (machine data in the CPU) will not be evaluated as limit switches. The data double words are then only used to identify the display range for the actual value. If these values are exceeded, the drive is not stopped.

2.5.3.3 **The Rotary Axis**

A rotary axis or closed axis is an axis without restrictions in terms of the traversing range. This might be, e.g.

- a round table (e.g. 360 degree divisions),
- continuous tape which can be divided into metric units or
- a tape winder.

With a rotary axis, the start of the range and end of the range are physically the same point *on* the axis (closed axis). If degrees are used as the dimension, the traversing range is not limited to 360 degrees.

The traversing range must be a whole multiple of the positioning resolution. If the reference point must be reproducible, the following must apply:

Traversing range = whole multiple of $\frac{pulses}{revolution}$ x travel resolution

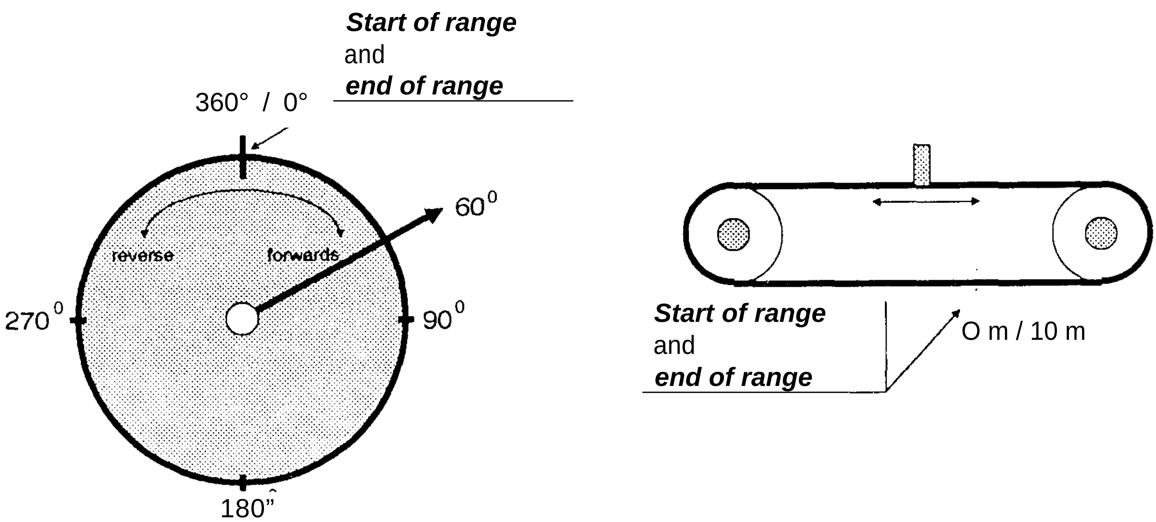


Fig. 2/9 The rotary axis and range limits

The absolute traversing range of a rotary axis is between the start of the range and the end of the range. If the actual position value exceeds the end of the range, the actual value indication is automatically set again to the coordinate of the start of the range.

Absolute targets must remain within the specified traversing range. If, however, a traversing movement is specified relative to the current actual value (e.g. 500 degrees forward), distances greater than the traversing range can also be travelled.

With the rotary axis, there is no limitation of the traversing range by software limit switches. The digital inputs of the hardware limit switches are, however, evaluated and can be used to limit the traversing range to values less than 1 revolution or as additional safety switches.

2.5.3.4 TransmissionRatio

Transmission ratio

Data double word

n + 15		Range of values
n + 16		0.012...400.000

The transmission ratio describes the distance travelled per motor revolution, Within this data, for example, the lead screw pitch of an axis is taken into account. The distance travelled is in the set dimension.

The positioning resolution is obtained from the quotient of the transmission ratio and the pulses per revolution.

Positioning resolution [unit/pulse] = transmission ratio/pulses per revolution

The maximum positioning resolution is:	0.001 [mm/pulse], 0.0001 [inches/pulse] or 0,001 [degrees/pulse]
The minimum position resolution is:	33.333 [mm/pulse] 3.333 [inches/pulse] or 33.333 [degrees/pulse]

Note

The positioning resolution is directly proportional to the distances travelled and to the speed. To ensure that the distances travelled correspond exactly to the specified distances, the positioning resolution must correspond exactly to the technical “reality”.

2.5.3.5 Maximum Frequency

Maximum frequency

Data double word

n + 5		Range of values
n + 6		0.012...100.000 kHz

This is not the maximum possible frequency which the motor or power unit can cope with!
The maximum frequency is the frequency output when the axis is intended to move at maximum speed, At this frequency, the motor must still have sufficient torque to move its load.

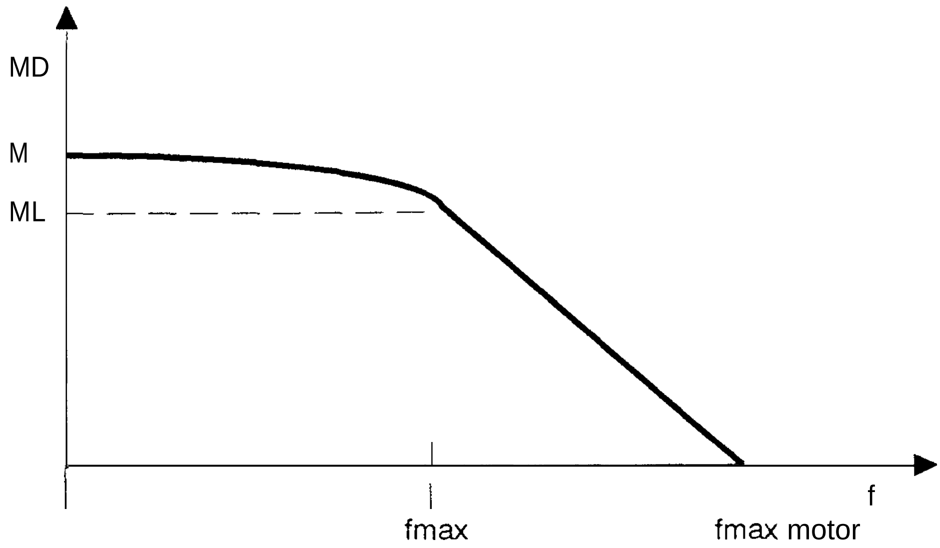


Fig. 2/1 O Torque characteristic curve of a stepper motor

The torque characteristic curve is specified by the stepper motor manufacturer for full and half step operation. f_{max} should be determined with the appropriate curve. You should allow for sufficient reserve.

2.5.3.6 Start-Stop Frequency

Start-stop frequency

Data double word		
n + 7		Range of values
n + 8		0.001...10.000kHz

The start-stop frequency is the frequency to which the motor can jump under load without disengaging and stopping.

The start-stop frequency f_{ss} is entered in the torque characteristic curve for the unloaded motor. The value of f_{ss} depends on the inertia of the load. The simplest way of determining this is by trial and error.

2.5.3.7 Rate of Frequency Increase

Rate of freq. increase

Data double word		
n + 9		Range of values
n + 10		0.020...2599.999Hz

When the machine data is input, an acceleration and deceleration ramp are generated.

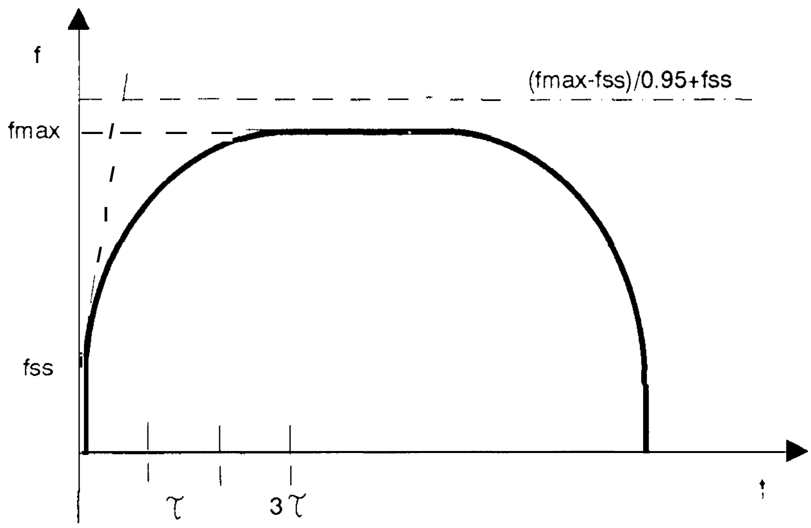
The acceleration ramp is generated using the formula

$$f = F \times (1 - e^{(-tb/\tau)}) + f_{ss}$$

Definition of the variables used:

f_{ss}	: start-stop frequency
f_{max}	: maximum frequency
F	: theoretical maximum frequency = $(f_{max} - f_{ss})/0.95$
t_b	: acceleration time [0...3 τ]
τ	: constant for the ramp up time

The deceleration ramp is the mirror reflection of the acceleration ramp.



Fig, 2/1 1 Acceleration and deceleration ramp of a job

To generate the ramps, three machine data are required:

Maximum frequency f_{max}: this frequency is output at the maximum traversing speed.

Start-stop frequency f_{ss}: this frequency is the maximum frequency at which the stepper motor can start up from stationary (taking into account the load and half and full step modes) and from which it can brake immediately to become stationary,

Rate of frequency increase a: quotient of the theoretical maximum frequency F and the ramp up time constant τ

$$a = F/\tau \text{ [Hz/ins]}$$

τ is a third of the required acceleration time t_b.

The acceleration time t_b is the time allowed to accelerate from a stationary position to f_{max}.

The maximum acceleration time is 7.8 seconds. The minimum acceleration time is 15 ms. This means that τ must be between 5 ms and 2.6 seconds, The maximum and minimum frequency increase is obtained as follows:

$$a_{min} = \frac{(f_{max} - f_{ss})/0.95}{2.6s}$$

$$a_{max} = \frac{(f_{max} - f_{ss})/0.95}{5ms}$$

With all traversing movements carried out at maximum speed (frequency), the full acceleration and deceleration ramp is used for acceleration and deceleration. With traversing movements at lower speeds, the acceleration is continued only until the required speed (frequency f_v) is achieved.

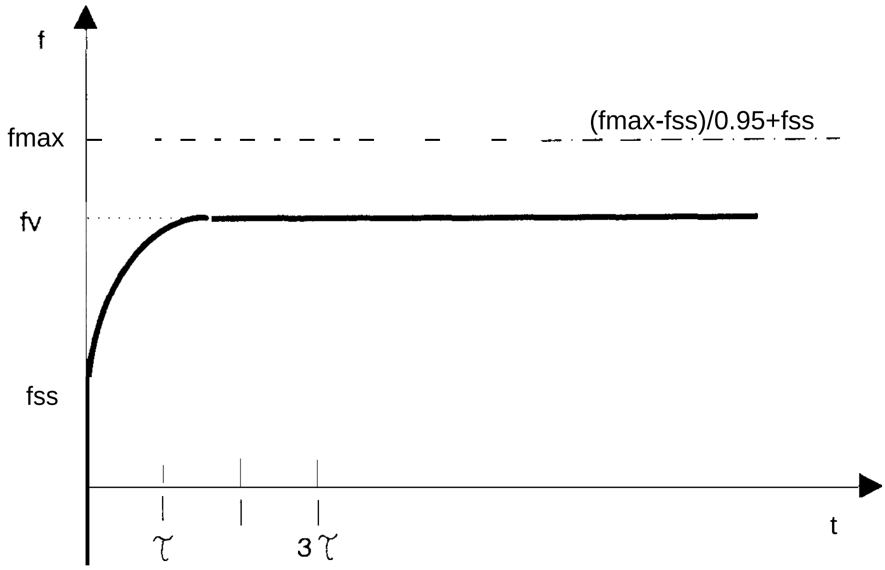


Fig. 2/12 Acceleration to frequencies less than f_{max}

The advantage of this acceleration and deceleration ramp is that a greater distance is travelled than with linear acceleration within the same time.

Note

To avoid the drive being damaged when the traversing movement is aborted, e.g. by a software or hardware limit switch, there must be sufficient braking distance after the limit switches.

The braking distance for any speed (frequency f.) can be calculated as followed:

Ramp up time t_v to traversing frequency f_v

$$t_v = -\tau \times \ln(1 - f_v / F)$$

The acceleration and braking distance can then be calculated as follows:

Distance = positioning resolution $\times (F \times (t_v + \tau \times (e^{(-t_v/\tau)} - 1)) + f_{ss} \times t_v$

> **Note**
>
> From now on, in the representation of traversing movements in the speed/time or speed/distance diagram a linear frequency increase will be used instead of the exponential increase, to help simplify the representation.

If the distance to be travelled is shorter than the sum of the acceleration and braking distances, the traversing frequency f. is reduced for the positioning job until it is certain that the positioning job not only includes the acceleration and deceleration ramp but also a section to be traversed at a constant speed.

2.5.3.8 Range Limits (Software Limit Switches)

Start of range (software start limit switch)

Data word

n + 29

n + 30 Range of values +- 99999.999 mm

End of range (software end limit switch)

Data word

n + 31

n + 32 Range of values +- 99999.999 mm

All distances are specified in the dimensional unit selected in the "measurement system" parameter. The traversing range is characterized by the following:

- the start of the traversing range (software start limit switch), X_A
- the end of the traversing range (software end limit switch), X_E

For a linear axis, the start of the traversing range is the software start limit switch, the end is the software end limit switch. Targets can only be approached within this range. If a software limit switch is tripped during operation, the axis is braked.

The following must apply: $X_A < X_{ref} < X_E$

For a rotary axis, the traversing range must be a whole multiple of the value

$$\text{pulses/revolutions} \cdot \text{positioning resolution}$$

to ensure that the reference point is reproducible.

Example:

Pulses/revolutions = 400 (half step mode)
Positioning resolution = 4 mdeg
Traversing range = 360,000 mdeg

$$\frac{\textit{Traversing range}}{\textit{pulses/rev. x pos. res.}} = \textit{225} \text{ (whole number)}$$

Example **of a rotary axis**
A round table is divided into 360 degrees. The start of the range is at O degrees, and the end of the range at 360 degrees. O and 360 degrees are the same point on the round table and can both be specified as the target coordinate.

When using degrees, the traversing range is not necessarily from O degrees to 360 degrees. It may, e.g. be from 400 degrees to 800 degrees. The only restrictions are the numerical range of the individual parameters and the rule that the coordinate for the start of the range must be smaller than the coordinate for the end of the range.

2.5.3.9 Backlash Compensation

Backlash compensation

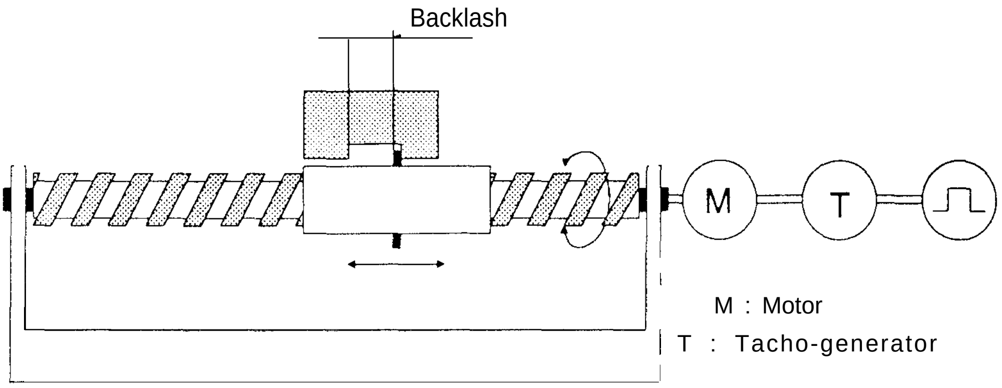
Data word n + 36

Data word n + 37

Range of values 0.000...64.999 mm

The backlash compensation value is used to compensate mechnical backlash (play in the drive).

If there is backlash greater than zero, there is a discrepancy between the detected actual value and the real distance travelled whenever the direction is reversed. The actual value of the axis position is displaced by the amount of the backlash. Using the backlash compensation parameter, this error can be adjusted, providing the backlash is measured exactly. Whenever the direction is changed, the positioning module includes the backlash in the distance to be travelled and therefore eliminates the backlash of the mechanical equipment. Since the actual position of the axis does not change until the backlash has been taken up when the direction is reversed, the actual value also remains unchanged in this area, although the motor is turning.



Fig, 2/1 3 Backlash

The backlash can be a value between O and 64,999 mm (0,1 inches or degrees).

The backlash can only be taken into account when the carriage can be moved directly by the drive, This is always the case when a distance greater than the backlash has been travel led. In a reference point approach this is always fulfilled, (=> Section 4.2.3,1 “Reference Point Approach”). After “setting” the reference point you must make sure that this movement takes place (to take up any play),

The backlash compensation value is ignored until a distance is travelled which is equal to or greater than the selected backlash compensation value. Correct detection or display of the actual position is not affected.

A backlash less than the positioning resolution cannot be compensated. The backlash compensation value is always a multiple of the resolution.

2.5.3.10 **The Polarity of the Hardware Limit Switches**

Polarity HW limit switches

Data word	$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$	
n + 33	0	→ positive (normally open)
n + 33	1	→ negative (normally closed)

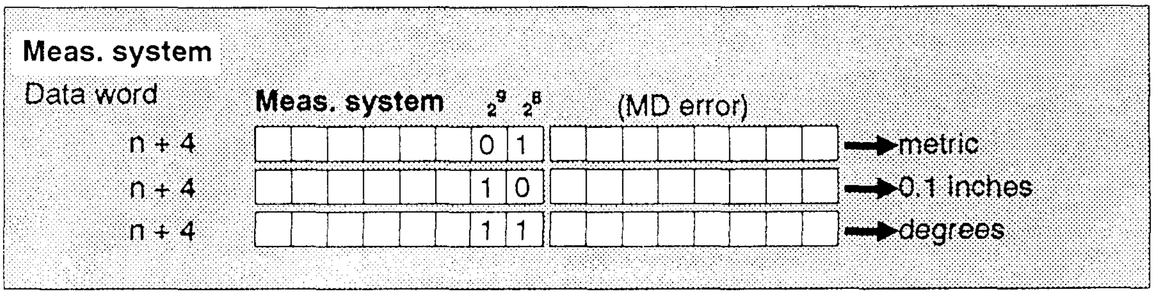
In addition to the software limit switches, two hardware limit switches are also evaluated via digital inputs, These should normally be after the software limit switches. If they are in front of the software limit switches, they limit the traversing range instead of the software limit switches,

These hardware limit switches can either both be stipulated as normally closed (“1”) or both as normally open (“O”) using the parameter “polarity HW limit switches”. A normally open switch generates a positive edge at the corresponding digital input and is therefore high-active, **A** normally closed switch generates a negative edge and is therefore low-active. For safety reasons, you should use normally closed switches as the hardware limit switches. The IP247 then recognizes a wire break during operation as the tripping of a limit switch and stops the movement. The tripping of a hardware limit switch is, however, only recognized when the axis is moving or should move in the direction of the activated switch.

When machine data are transferred to the module, the assignment of parameters for the hardware limit switches is checked. The IP247 can only detect incorrect parameter assignment when neither of the hardware limit switches is active when the machine data are input. If a rotary axis has been selected, the digital inputs for hardware limit switches are also evaluated, If, however, no limit switches are connected, the polarity must be set to “normally open” (COM247: "pos", PC interface "O").

2.5.4 Machine Data for Operation

2.5.4.1 Measurement System

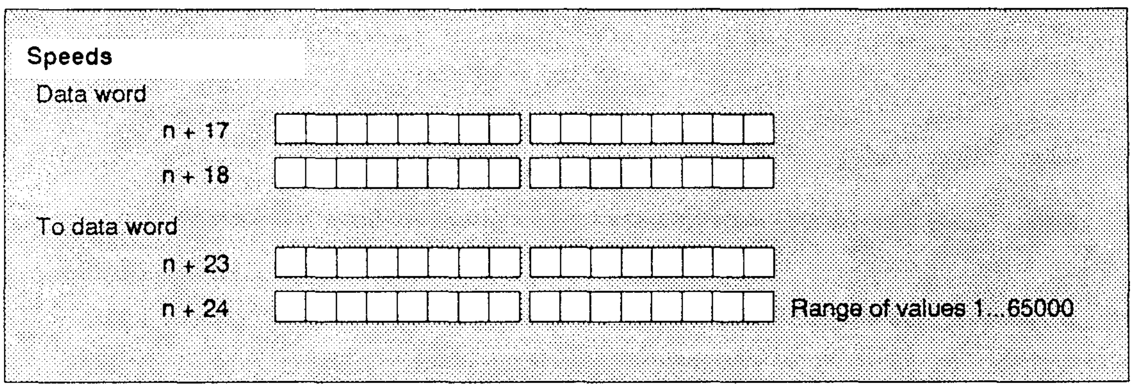


Regardless of whether you operate your axis as a linear or rotary axis, the IP247 allows the following dimensional units to be used:

- metric system with a basic unit of 0.001 mm,
- inches with a basic unit of 0.0001 inches and
- degrees with a basic unit of 0.001 degrees.

The basic units are the smallest values permitted in machine data, machining programs and command inputs. All positions, speeds and the resolution relate to the dimensional unit selected for the axis.

2.5.4.2 Speeds



Speeds for the various modes are assigned in the machine data record. The speeds must be selected as follows, depending on the dimensional unit:

- in mm/min for metric input,
- in 0.1 inches/rein for dimensions in inches or
- in degrees/rein for dimensions in degrees

The maximum range of values is 1 to 65000.

The starting point is the maximum speed. This is the speed at which the drive travels when the pulse generator outputs the maximum frequency to the power unit. This speed and frequency must be determined exactly from the technical specifications of the drive.

Speed [dimensional unit/rein] = frequency x positioning resolution . 60

Example

f_{mex}	= 30 kHz
Transmission ratio	= 1 mm/revolution
Pulses	= 500 pulses/revolution
Positioning resolution	= 1 mm/500 pulses = 0.002 mm/pulse
Maximum-speed [mm/min]	
	= 30000 x 0.002 x 60= 3600 mm/min

> **Note**
>
> The positioning resolution is directly proportional to the traversing distances and the speed. To ensure that the distances travelled and the speeds correspond exactly to those selected, the positioning resolution must correspond exactly to the technical reality.

Minimum speed

The minimum speed v_{min} is calculated as follows:

$$V_{min}\left[\frac{mm}{min}\right] = f_{min}\,[Hz]\; x\, res.\,[\mu m]\; x\, \frac{60}{1000}$$

V_{min} must be greater than 1 mm/min.
f_{min} is within the interval [1 ...15 .25 Hz]

The value of f_{min} is determined when the ramp table is generated from the machine data. Maximum frequency, start-stop frequency and rate of frequency increase.

No speed can exceed the maximum speed.

The speeds to be specified areas follows:

- JOG speed 1 for the first JOG mode,
- JOG speed 2 for the second JOG mode,

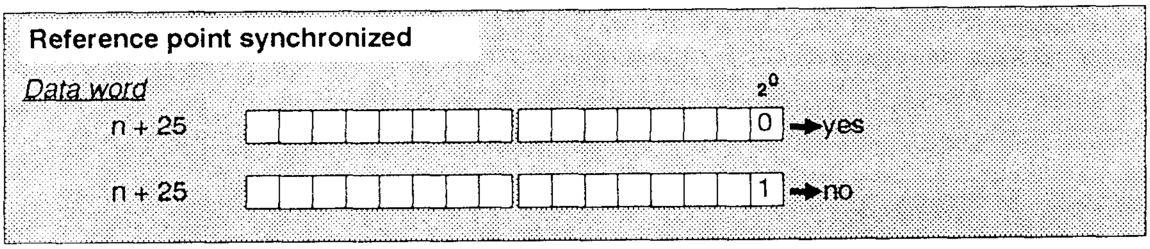
  the incremental speed for the modes absolute and relative incremental approach
- reference speed for the reference point approach,

The reference speed is the speed at which the axis travels to the reverse point and from therein the reference direction to the start of the precontact. The reverse point can be a hardware limit switch or the precontact itself.

The reference speed must not exceed the maximum speed and must be greater than the speed achieved at the start-stop frequency.

In the modes “JOG” and “incremental approach”, the speeds contained in the machine data are only used if a “O” is transferred in the speed parameter of the job.

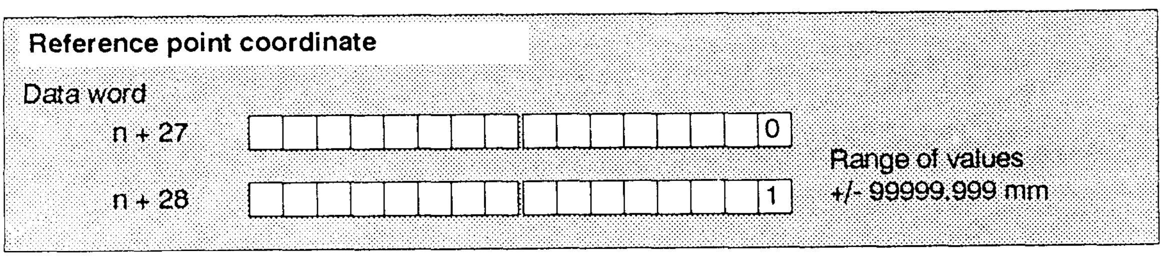
2.5.4.3 Reference Point Synchronized



This machine data decides whether or not the zero reading of the excitation pattern counter should be taken into account when locating the reference point during a reference point approach (see “reference point approach/set reference point”). This then specifies the type of reference point approach.

The location of the reference point when synchronization is set is to some extent dependent on the dispersion of the edge of the precontact.

2.5.4.4 Reference Point Coordinate



This machine data contains the position of the reference point in the current coordinate system. This coordinate can be assigned to the current position in the "set reference point" mode or can be assigned to a point on the axis determined by a precontact, the reference direction and type of reference point approach in the "reference point approach" mode (see Section 2.5.4.5 "Reference Direction").

2.5.4.5 Reference Direction

Reference direction

Data word	High byte	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
n + 26				0						➡ "forward"
n + 26				1						➡ "reverse"

The reference direction specifies whether the reference point is to be approached in a forward direction ("0"; for a rotary axis, in a clockwise direction), or in a reverse direction ("1"; for a rotary axis, anti-clockwise) If the axis is not exactly on the precontact at the beginning of the reference point approach, the axis first travels in the opposite direction to the specified reference direction, as far as the reverse point and then in the reference direction until the reference point is detected (=>Section 4.2.3.1 "Reference Point Approach").

> ⚠ **Note**
>
> The reference point can only be reproduced when it is always determined in the same direction.

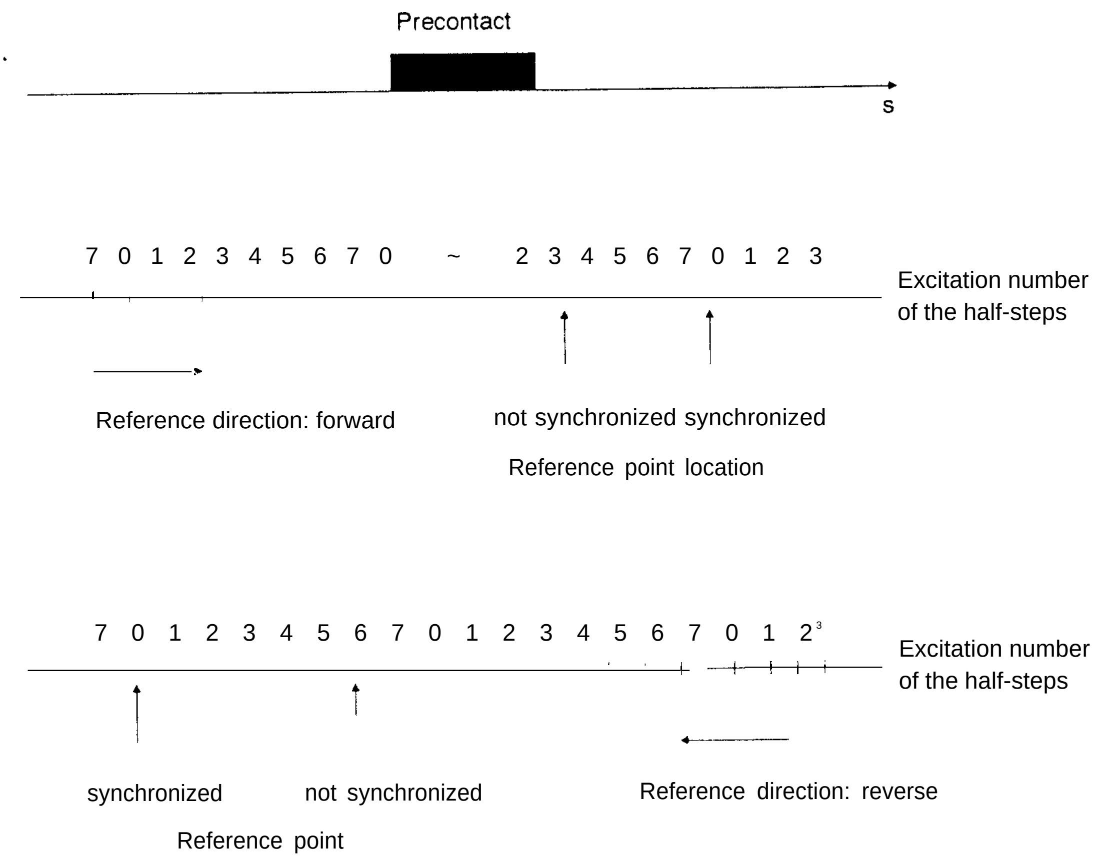
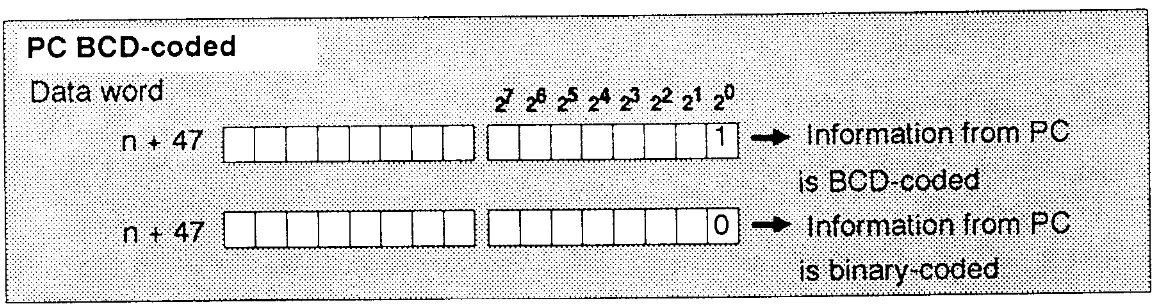


Fig. 2/1 4 Reference direction

2.5.4.6 Target Information from PC is BCD-Coded



Using the parameter “PC BCD-coded", you can inform the module whether target information, tool offsets and zero offsets (in the modes “JOG“, “incremental approach” also speeds) sent by the PC to the I P247 are in binary or BCD format.

Remember that each speed, each tool length offset and each zero offset of the corresponding axis is interpreted in the selected format by the I P247 and this selected format will be used until a different coding is specified in the machine data record.

> **Note**
>
> The machine data does **not** influence the output of the actual value or the distance to go in the function block.

A double word (32 bits) is available for each piece of information, Each digit in a BCD-coded number requires four bits, and the sign in STEP 5 format also requires four bits. The maximum representable value in BCD format is therefore 9999999 pm.

Speeds use the byte and word parameter.

For a binary coded speed, the value between O and 65000 is only transferred in the word parameter, the byte parameter is ignored.

With BCD coded speeds, the byte parameter informs the module whether the actual speed or a tenth of the speed is located in the word parameter.

Speed	Byte parameter	Word parameter
9999	0	9999
50000	1	5000

If a 1 is entered in the byte parameter, the IP247 multiplies the value in the word parameter by 10,

If actual values (actual position value, distance to go), transferred by the IP247 via the PC interface to the CPU are to be output in BCD format, you must set this in FBI 64 (⇒Part 6 "Standard Function Blocks FBI 64 and FBI 65").

2.5.5 Machine Data for Machining Programs

2.5.5.1 Tool Length Offset

Tool length offset

Data word n + 34	☐☐☐☐☐☐☐☐ ☐☐☐☐☐☐☐☐	Range of values +- 99999.999 mm
Data word n + 35	☐☐☐☐☐☐☐☐ ☐☐☐☐☐☐☐☐	Range of values +- 99999.999 mm

With a linear axis, the theoretical range of values for the tool length offset is from O to +/- 99.999999 m. For a rotary axis, the tool length offset is limited to values less than the traversing range fixed by the range limits. The following points must also be taken into account:

- the coordinates of the software end limit switch (end of range) plus the tool length offset must be less than or equal to + 99.999999 m and
- the coordinates of the software start limit switch (start of range) plus the tool length offset must be greater than or equal to -99.999999 m.

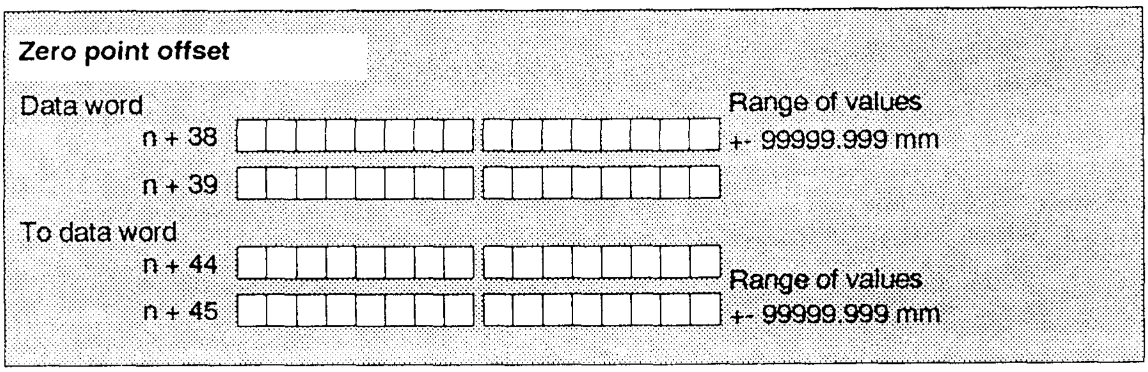
Remember that the tool length offset has a sign.
The tool length offset assigned in the machine data can be called in machining programs with G43 ("positive tool length offset on") or G44 ("negative tool length offset on"), and is then added to an already existing tool length offset or subtracted from it. This can be repeated. At each call, the system checks whether the new tool length offset will exceed the limits outlined above, If either of these limits would be exceeded, the machining program is stopped and an error message output.

Using G40 ("clear tool length offset"), you can clear all the active tool length offsets in the whole machining program.

An active tool length offset means that the tip of the tool approaches the specified position, inactive tool length offset means that the tool holder (e.g. drill chuck) approaches the required position,

An overall positive value for tool length offset means that the positioning module reduces the setpoint by such an amount that the position is reached with the length of the tool. When the software limit switches are checked, the tool length offset is, however, not taken into account, i.e. the tool holder can use the same traversing range as without an offset, (=> Section 4.3.8 "Tool Length Offset".)

2.5.5.2 Zero Point Offset



A total of four zero point offsets can be assigned in the machine data record and can be called in machining programs using the G-functions (=> Section 2,6.6 "The G-Functions"). These can have values throughout the whole traversing range (+/- 99.999999 m).

If a zero offset is executed, all coordinates (software limit switches, reference point coordinates and actual value) are corrected by the amount of the offset.

When the machine data are entered, the IP247 checks whether one of the four assigned offsets exceeds the permitted traversing range of +/- 99.999999 m. Such offsets are not executed, and cause an error message.

The zero offsets are called in machining programs using functions G54 to G57 and are cancelled with G53. They can only be enabled as alternatives, If an offset has already been executed with the zero offset modes (=> Part 4 “Functions”), the offsets activated by G54 to G57 are added to those already existing. For this reason, a check is made during the execution of a machining program to ensure that the activation of a zero offset does not exceed the maximum range. If the maximum range would be exceeded by the offset, the machining program is stopped.

2.5.6 OtherParameters

This section discusses the following parameters:

- number of the machine data record,
- module number,
- axis (number), for which the machine data record is valid,
- machine data errors,

Number of the machine data record			
Data word (DR)		DB number	
n + 2	0 1 0 0 0 0 1 0 (ASCII "B")		Range of values 0...255

A DB number from 0...255 can be assigned to a machine data record.
The left byte (DL n+2) must always have the bit pattern shown above.

Module number/axis			
Data word	Module number	Axis	Module number Range of values 0...99
n + 3			Axis Range of values 1...3

Each positioning module can control three axes, with the fixed designation axis 1, axis 2 and axis 3. Each module is *also* assigned a module number between O and 99. This information uniquely identifies an axis, To be able to assign a machine data record to an axis, this must contain both values,

Machine data errors			
Data word	Meas. system	MD error	
n + 4			Range of values 0...max. 255

Each machine data record also contains an error variable. Some of the possible input errors made when generating the machine data record on the programmer are detected by the software package CC) M247. Further checks are made by its firmware when the machine data are entered into the positioning module. If an error is detected, the corresponding error number is written to the error variable of the machine data record and the error message “error in machine data” is output.

2.6 Machining Programs and their Structure

2.6.1 General

A machining program is a connected series of traversing jobs, dwell times and offsets. Machining programs are made up of individual statements. Each statement is itself a complete and feasible job for the positioning module. The machining programs can be stored in the RAM of the positioning module, from there they are executed either as a series of statements or in the single statement mode. The machining programs accepted by the positioning module generally correspond in terms of their structure to a subset of the representation described in DI *N 66025,* Only this subset is explained here. COM247 provides you with user-friendly support when generating a machining program. Deviations from the permitted subset of DIN 66025 are signalled immediately when generating the program.

> **Note**
>
> Machining programs are independent of the axes. A machining program can be executed simultaneously on all three axes. It is of little importance whether the particular axis is linear or rotary.
>
> Machining programs do not include dimensions. Position information and speeds are always interpreted in the unit assigned to the axis in the machine data record,
>
> Machining programs can be interrupted and continued from the same point.

The programs consist of a sequence of ASCII characters. The following restrictions apply:

- a maximum total of 6000 ASCI I characters can be stored on the I P247
- these can be divided into 255 programs
- a maximum of 1023 ASCII characters are permitted per program.

Repeat loops and subroutines are possible in the programs up to a common nesting depth of 5.

If a statement is inserted in an existing machining program using the machining program editor of COM247, or if a statement is appended to a program, 50 characters are reserved for this statement. If the maximum length of a machining program would be exceeded by this addition, COM247 generates an error message.

A machining program created with COM247 cannot be loaded in the CPU directly. If you wish to store a machining program in the CPU, you must transfer the machining program to the IP247 and then to an **S5** data block on the CPU. The machining program number is entered in the machining program header. Program numbers 0...255 are permitted.

A machining program is structured as follows:

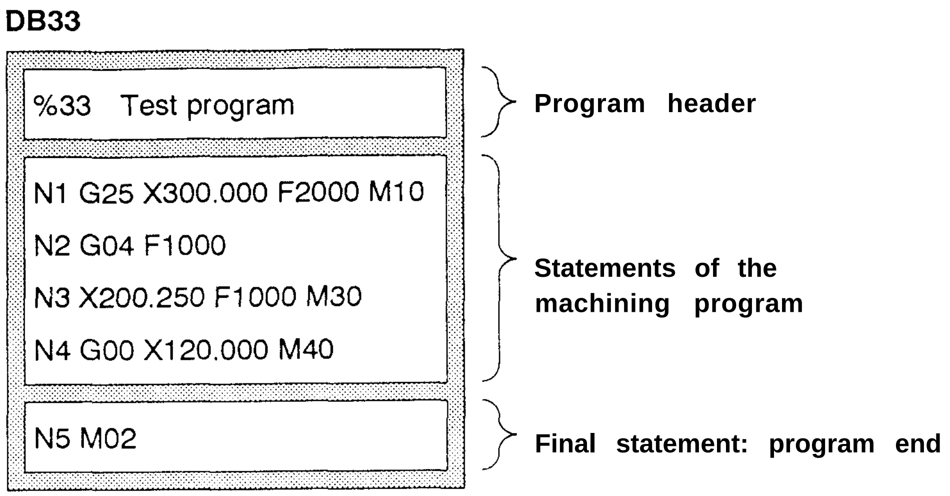


Fig. 2/1 5 Structure of a machining program with program number 33

The machining program always has a program header and a final statement. The final statement has the special identifier M02 at the end. The length of individual statements can vary.

2.6.2 **Program Header**

The program header is generated automatically by COM247 when the machining program is created at the PG.

The header includes the following:

- The program identifier:
  % = main program
  L = subroutine,
- The program number (maximum three characters) . DB number of the data block
- A text with a maximum of 58 characters (selected as required)
- <LF> (line feed) to complete the header.

Example:

%5	this is a main program in DB5 <LF>
L12	this is a subroutine in DB12 <LF>

The difference between main programs and subroutines is simply of documentary interest, the module does not distinguish between them, so that a program can be used both as a main program and a subroutine.
Recursive **or reciprocal program calls are, however, not permitted.**

2.6.3 **ProgramStatements**

A statement in a machining program consists of a series of functions which have a fixed order and must be separated by at least one blank. Each statement must be completed by a line feed (<LF>). The length of a statement is limited to 50 characters, including <LF>. Blanks before and after the line feed are not necessary, but permitted. Blanks following a line feed are included in the length of the next statement.

The following functions are available:

- N-function statement type and statement number
- L-function subroutine call
- G-function preparation of traversing conditions
- X-function target function
- F-function speed, time, loop execution
- M-function auxiliary function

It is not necessary to include all functions in a statement, however, they must not occur more than once in a statement. All the functions used must be in the order listed here. Some functions must be the last in a statement or can only be followed by certain other functions.

The N-function in a statement and the completion by <LF> are obligatory, as is the function M02 in the final statement of the machining program. No further statement can follow this.

Example

%9	program example
N10 G74 M10	approach the already known reference point M10 is output at the start of the statement
N20G24 F5	beginning of a repetition loop with five repetitions
N30L36	call subroutine 36
N40X50F2000	approach point 50 mm at 2000 mm/min *
N50G20	end of the repetition loop
N60 M02	final statement, program end

*The example applies to the presets “dimensional unit mm” and “target specification absolute”.

2.6.4 The N-function

The N-function is the first function in a statement and specifies the number of the statement. This function is obligatory and consists of the character ‘N’, followed by a maximum three digit number between O and 999.

The statement numbers can be entered in any order, and can be used more than once in a machining program.

> **The execution of the statements is always in the order in which they are entered in the machining program.**

All statements are treated as “normal statements” according to DIN 66025. The statement identifiers “/N” for skippable statement and ":N" for main statement are permitted, but are of *no* significance.

2.6.5 The L-Function

A different program can be called as a subroutine in a program statement. This call must follow the N-function immediately, The function consists of the character ‘L’ followed by the machining program number of the program to be called.
No further functions can follow the L-function and the statement is completed with <LF>.

Examples

N1O L123 <LF>	call subroutine 123
N20 L5 <LF>	call subroutine 5

Subroutines can be nested. The nesting of loops and subroutines must not exceed a nesting depth of 5.

2.6.6 The G-Functions

A G-function can follow an N-function. It is identified by the letter ‘G’, followed by a two digit number. Only one G-function is permitted in a statement. Only the following G-functions are permitted:

G00: rapid traverse
G04: dwell time
G1 O: flying change
G20: loop end
G24: loop start
G25: approach target by shortest route (*)
G26: approach target in clockwise direction (*)
G27: approach target in anti-clockwise direction (*)
G40: clear tool length offset
G43: positive tool length offset on
G44: negative tool length offset on
G53: clear offsets
G54: offset 1 on
G55: offset 2 on
G56: offset 3 on
G57: offset 4 on
G70: dimensions in 0,1 inches (*)
G71: dimensions in mm (*)
G74: reference point approach
G90: position specifications absolute (*)
G91: position specifications incremental (*)

(*) = latching (retentive) functions

At the beginning of a program the following G-functions are automatically active:

- G25: approach target by shortest route
- G90: position specifications absolute

If the machine data of the axis on which the machining program is to be executed are in mm, G71 (dimensions in mm) is also the default. If the machine data are in 0.1 inches, then G70 (dimensions in 0.1 inches) is the default.

If degrees are selected, neither G70 nor G71 are defaults, since it is not possible to change the dimensional unit.

With the G-functions implemented on the IP247, the following preparatory conditions, offsets or switchovers can be executed.

2.6.6.1 G00: Rapid Traverse

The target position programmed in this statement is approached at the maximum speed (see machine data). Specifying the speed using the F-function is then not permitted.

Example

N50 G00 X1 000 M23	output the auxiliary function M23 at the beginning of the statement.. At maximum speed to target point 1000.

2.6.6.2 G04: **Dwell** Time

A dwell time is executed in this statement. The duration can be set using the F-function in units of 100 ms.

Example

N38G04 F1O M34	output of auxiliary function M34 at the beginning of the statement. Wait for 10 x 100 ms

2.6.6.3 **G10: Flying Change**

The statement following the statement containing G1 **O** is carried out without stopping the axis. The following can therefore be achieved:

- speed changes during a traversing movement (example 1) 01
- changing the M-function during a continuous traversing movement (example 2).

Example 1: initial point at program start x = O

N30(GIO)X50F1000 M30	to target point 50 mm at 1000 mm/min
N32 (Gl O) Xl 00 F500 M31	**to** target point 100 mm at **500** mm/min
N34X150F1 000 M32	to target point 150 **mm at** 1000 mm/min
N36M02	**final** statement/program **end**

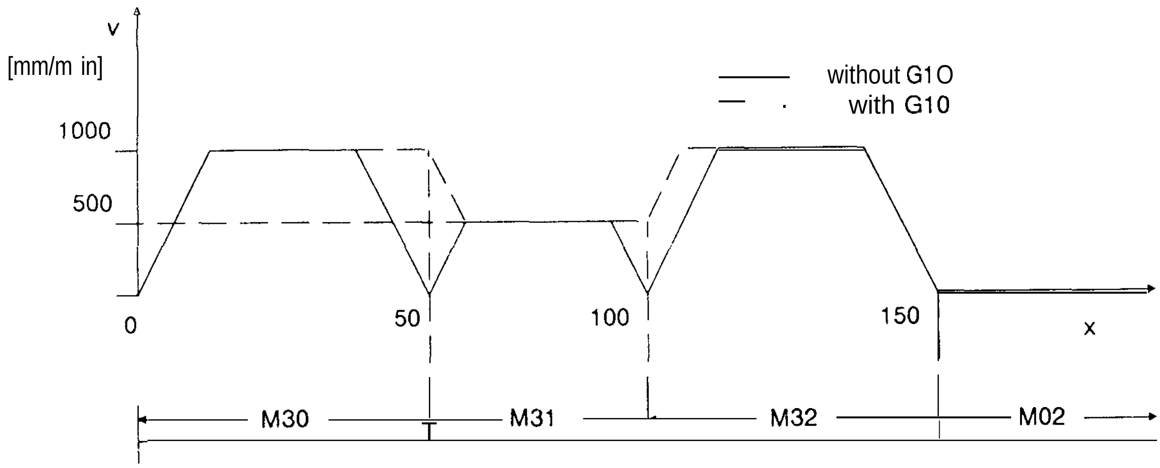


Fig. 2/1 6 Flying change with speed change

Example2: initial point at program start x = O

N40 (G10) X50 F1 000 M40	to target point 50 mm at 1000 mm/min
N42(GIO)XI00F1000 M41	to target point 100 mm at 1000 mm/min
N44 XI 50 F1OOO M42	to target point 150 mm at 1000 mm/min
N45M02	final statemerrt/program end

If no different M-functions were required, the movement could be brought together in one statement (e.g. N1OX150 F1 000 M40).

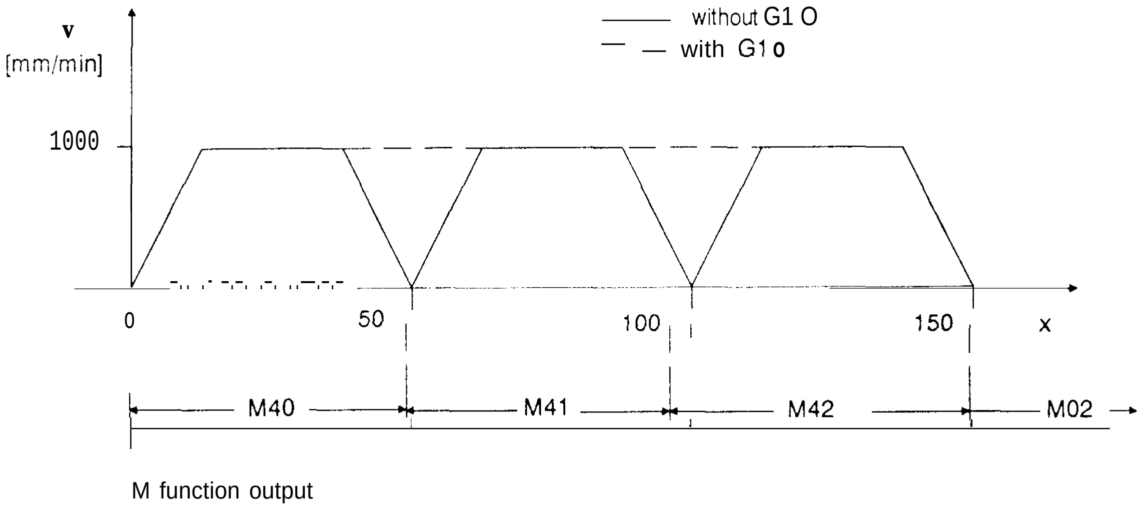


Fig. 2/1 7 Flying change without speed change

Note

A machining program may be aborted sporadically on an axis with the error message “statement not yet fully interpreted”, when:

the IP247 has a high workload (e.g. machining programs with G10 active on all axes

a statement with a short processing time is followed by a statement with flying change

Special features of “flying change”

The “position reached” message (=> Section 2.7 “Axis Attributes”), is **not** set on completion of a statement with G1 O.

Statements connected by the flying change are treated as one statement in the mode BA9 “automatic single statement” (execution of the machining program statement by statement). This means that there is no stoppage between these statements. If G1 O and an MOO (“programmed halt”) are programmed in one statement, MOO has priority.

A “flying change” cannot be executed under the following conditions. The program is then stopped with the error message “flying change could not be executed”:

- when the statement following the flying change specifies the opposite direction,
- when the statement following the flying change contains a dwell time,
- when the statement following the flying change only contains an M02,
- when the traversing distance following the flying change is shorter than the braking distance of the previous statement,
- when the statement following the flying change is too short to achieve the required final speed,
- when the statement following the execution of the statement containing G1 O could not be interpreted or
- when the statement following the flying change contains a switchover, tool offset or zero offset.

* Note the following points with a rotary axis*

If individual target positions cannot all be reached via the shortest route, then the direction in which the flying change is to be executed should be stipulated with G26 or G27. If you do not do this, the flying change is aborted with the message “Change of direction illegal with flying change”.

2.6.6.4 **Loops**

Loops can be nested within each other. Subroutines containing further loops can be called in loops. The nesting depth for subroutines and loops must not exceed a total value of 5. Closed (endless) loops can only be programmed at the highest level, A closed loop cannot therefore be included in a program called with an L-function.

G20: loop end

A statement containing G20 is the end of a repetitive loop and must not contain any other functions.

Example

N80G20 end of the repetitive loop.

G24: **loop start**

A statement containing G24 is the start of a loop. The number of repetitions is specified by the F-Functions. FO means a closed loop, The statement must not contain any further functions, including M-functions,

Example

N1 O G24 FO	start of a closed loop
N20G74	approach reference point
N30G24 F5	start of a loop with 5 repetitions
N40L30	call subroutine 30
N50G04 F1 O	wait one second
N60L30	recall subroutine 30
N70G20	end of the inner loop
N80G20	end of the closed loop
N90 M02	final statement/program end

2.6.6.5 Direction of Approach to the Target Point with a Rotary Axis

With a rotary axis, absolute target points can either be approached by the shortest route (G25) or clockwise (G26) or anti-clockwise (G27). If machining programs containing these G- functions are executed on a linear axis, they are ignored.

G25: approach target by shortest route (default at program call)

With a rotary axis, the function G25 means that all absolute targets are approached by the shortest route. The module itself determines the direction of approach. If the distance to the target is the same both in a clockwise and anti-clockwise direction, the clockwise direction will always be selected (= preferred direction). When deciding the direction of approach, backlash compensation is ignored.

Example

A backlash compensation value was selected in the machine data.
The target is to be approached by the shortest route (G25). The current position is O degrees.
Ignoring the reversal backlash, the travel distance is the same in both directions.
The direction of the previous job was anti-clockwise.

=> The distance travelled is longer owing to the backlash. The traversing movement therefore takes longer in the preferred direction than in the opposite direction.

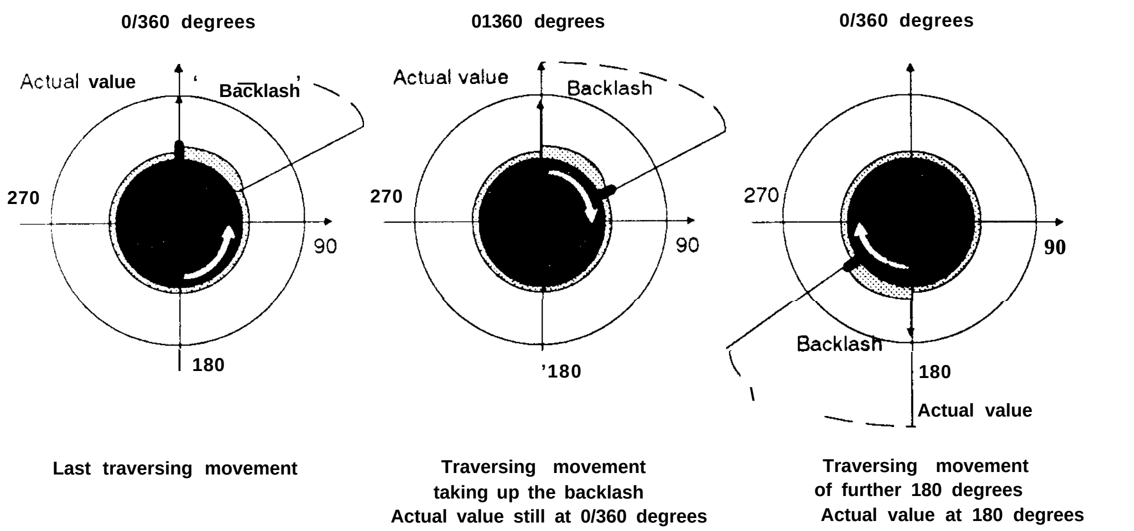


Fig. 2/1 8 Reversal backlash with a rotary axis

G26: approach target in clockwise direction

All **absolute targets are** approached in a clockwise direction (forward) when G26 is selected. On a linear axis, a G26 is ignored and does not cause a stoppage of the machining program.

G27: approach target in anti-clockwise direction

All **absolute targets are** approached in an anti-clockwise direction (reverse) when G27 is selected. On a linear axis, a G27 is ignored, and does not cause a stoppage of the machining program.

Example

Rotary axis, traversing range O degrees to 360 degrees

N1 OG74 Ml O	approach the reference point by the shortest route
N20G27	change direction: reverse
N30X180F1 000M30	approach 180 degrees in reverse direction
N40G25XOF500	approach O degrees/360 degrees by the shortest route, here the preferred direction forwards
N50G26	change direction: forwards
N60X0F1000M60	one revolution of the rotary axis forwards
N70G25X360F500	switchover to the shortest route. No traversing movement since shortest route.
N80M02	final statement/program end

Note

The G-functions G26 and G27 are only effective when “position specifications absolute” (G90) is set.

2.6.6.6 Tool Length Offset

By using a tool length offset in the machining program, a change in the length of the tool during execution of the program (usually wear on the tool) can be taken into account. This is added to a tool length offset executed with mode BA15 (“tool length offset”).

The value of the tool length offset used in the machining program is stored in the machine data. Each time a tool length offset is called in the machining program, the value stored in the machine data record is added to the already existing offset. The following limit values apply to the resulting tool length offset:

For a linear axis:

- value of the offset maximum 100 m,
- software end limit switch + offset value < 100 m and
- software start limit switch + offset > -100 m

For a rotary axis:

- offset less than the traversing range. (Range end - range start), If the tool length offsets implemented by a machining program during its execution are not reset with G40 (“clear tool length offset”), they are retained on completion of the machining program. The offset implemented in the machining program can then only be cleared using modeBA16 (“tool length offset off”). However, a basic tool length offset activatedbyBA15 (“tool length offset”) is also deleted. If a new tool offset is activated by BA15 when a machining program is completed, the cumulative tool length offset achieved during the machining program is no longer effective,

	Tool length offset BA 15/16	Tool length offset machining program	Total
Tool change			
↓			
BA 15; 10Omm, forwards	1 00mm	0	1 00mm
↓			
Machining program start			
G44	00mm	-1 Omm	90mm *
G44	00mm	-20mm	80mm *
Machining program end			80mm
↓			
Tool change			80mm
↓			
BA 15; 200mm, forwards	200mm	o	200mm

* In the machine data record: tool length offset =10mm

Fig. 2/1 9 Tool length offset

G40: clear tool length offset

A statement containing G40 switches off all the active, positive or negative tool length offsets in this machining program. This also applies to subroutines. The G-function G40 does not affect the tool length offset set with mode BA15 (“tool length offset on”).

G43: positive tool length offset on

A statement containing G43 causes a tool length offset in a forwards direction by the length specified in the machine data (=> Section 4.3.8 “Tool Length Offset”). This occurs each time the function is executed.

G44: negative tool length offset on

A statement containing G44 causes a tool length offset in a reverse direction by the length specified in the machine data (=> Section 4.3.8 “Tool Length Offset”). This occurs each time the function is executed.

Example

– The tip of a tool with a basic length of **40** mm must approach coordinate O. During each machining operation, the tool is reduced in length by **5 mm, The tip of the tool is at position -65 mm before** the first machining operation. The home position of the tool holder is -105 mm.

The following must be programmed:

– in the machine data: tool length offset = +5mm
in the machining program: N1 0X0F1000
N15G44X-65F2000

This means that the tip of the tool is at the same position following each machining operation

In this example, the tool holder does not return to the home position when it is retracted.

If the tool holder must always return to the basic position when it is retracted (e.g. owing to interlocks), you should set the reference point at this position, In a reference point approach, or with G74 in an automatic program, the tool holder always returns to the same position both with or without offsets. However, the corrected value (coordinate of the tip of the tool) is displayed as the actual value.

The machining program is then as follows:

N1 0X0F1000
N15G74
N20G44

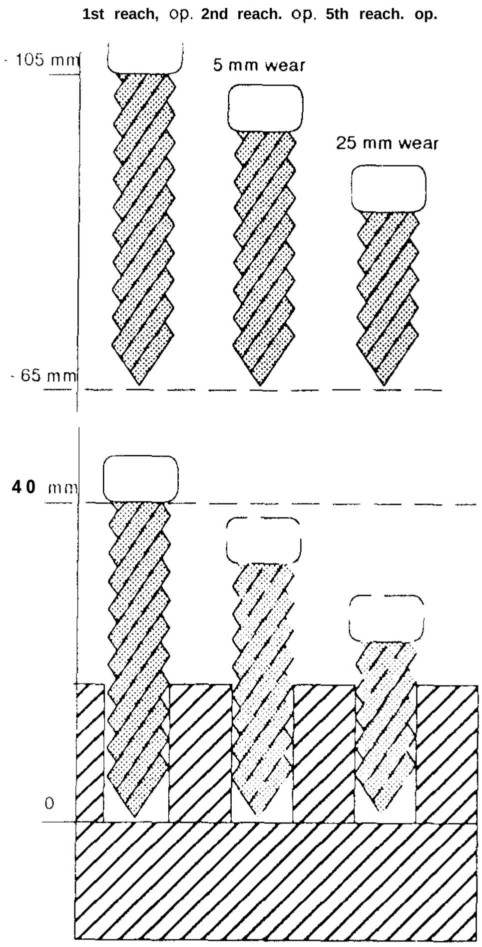


Fig. 2/20 Tool length offset

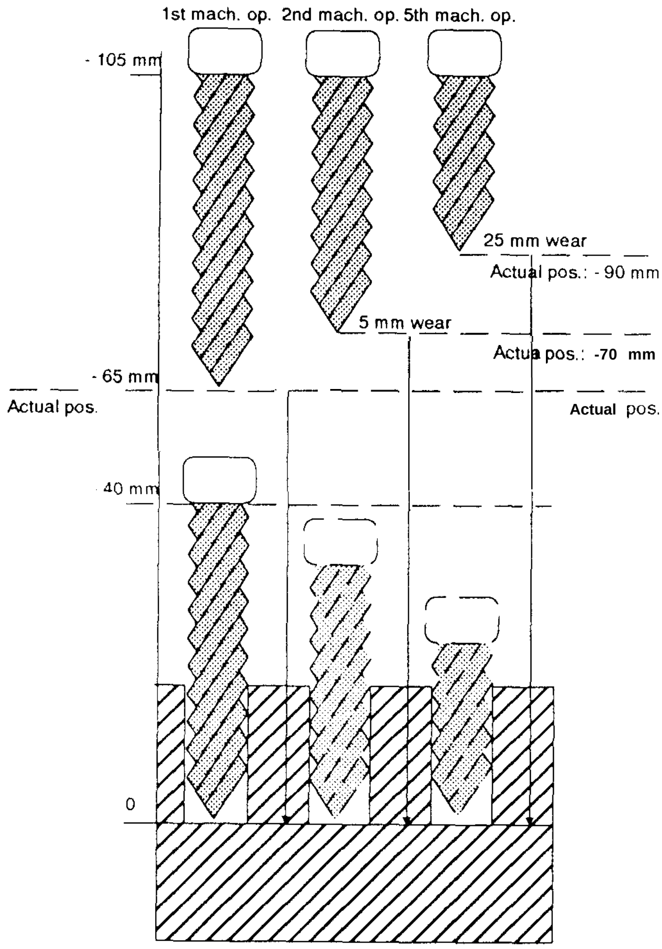


Fig. 2/21 Tool length offset

2.6.6.7 Zero Point Offset

You can program a relative displacement of the coordinate system of your axis during a machining program. This offset is added to offsets executed with mode BA12 (“zero offset absolute”) or BA13 (“zero offset relative”).

I n machining programs, only one of the four offset values selected in the machine data can be activated (G54...G57). If a second zero offset is activated, the first is no longer effective.

The direction of the offset depends on the sign in the machine data,

On completion of a machining program, the offsets activated in the machining program are automatically switched off again, This is, however, not the case if the machining program stops owing to an error message or because of a stop command. In this case, the basic coordinate system can only be established again by clearing all offsets with operating mode BA14 (“clear zero offset”), G53 also clears offsets executed in subroutines.

Zero offsets executed in a subroutine are not cleared following the return to the main program. They are only reset on completion of the main program.

A zero offset changes the limits of the traversing range, the reference point and the actual position value according to the value of the offset, With a positive zero offset, the zero point of the coordinate system is displaced in a positive direction, i.e. the individual points on the axis have a more negative value. A negative zero offset has the opposite effect.

G53: clear offsets

G53 deactivates ail the zero offsets active in the machining program. Offsets set with the mode “zero offset absolute or relative” (=> Sections 4,3.5 or 4,3.6 “Zero Offset Absolute or Relative”), are not changed.

G54 -G57: offsets 1-4 on

A statement containing one of the G-functions G54-G57 executes a relative zero offset.

G54 ⇒ zero offset 1
G55 ⇒ zero offset 2
G56 ⇒ zero offset 3 and
G57 ⇒ zero offset 4

The following example contains both types of zero offset, After stipulating the coordinate system

Mode BA 5	Parameter approach	Start command,

a relative zero offset 10 mm forwards is executed.

Mode BA 13 (zero offset relative)	Parameter 10000 µm	Command forward

The actual position value is displaced from O mm to -10 mm. Following this, a traversing movement to point O mm is executed.

Mode 9A 6 (incremented absolute)	Parameter O mm	Start command

Here, machining program 1 is started.

Mode BA 8 (automatic)	Parameter 1;	Start command

Subroutine 9 is called in this machining program. By means of Ml O, this program controls the drilling of three holes (at O mm, 10 mm and 20 mm). The three coordinates are specified as absolute values. Following each execution of subroutine 9, coordinate 40 mm is approached in machining program 1 and a relative zero offset of 40 mm is executed via G54, G55 and G56. The values of the offsets are assigned in the corresponding machine data record. Subroutine 9 is called a total of three times. Before the end of the program, the tool holder is brought to its home position at the reference point by G74. Since the offsets are still effective, the actual position value is now displayed as -130 mm. At the end of the main program, all the zero offsets activated in the machining program are cleared again. An offset of +1O mm remains, which was executed at the beginning with BA13.

Example

Values in the machine data:

zero offset 1 : 40 mm
zero offset 2: 80 mm
zero offset 3: 120 mm

Machining programs:

```
%1 main program
N1 L9               call subroutine 9
N2X40.000F500       to coordinate 40 mm at 500 mm/min
N3G54               zero offset by 40 mm
N4 L9               call subroutine 9
N5X40.000F500       to coordinate 40 mm at 500 mm/min
N6G55               zero offset by 80 mm / G54 no longer effective
N7 L9               call subroutine 9
N8X40.000F500       to coordinate 40 mm at 500 mm/min
N9G56               zero offset by 120 mm/ G55 no longer effective
N1OG74              approach reference point
N11 M02             program end

L9 subroutine
N1 XOF100 Ml O to coordinate O mm at 100 mm/min
N2 G04 F50 Ml O     wait 5 sec
N3X1OF100           to coordinate 10 mm at 100 mm/min
N4 G04 F50 Ml O     wait 5 sec
N5X20F100           to coordinate 20 mm at 100 mm/min
N6 G04 F50 Ml O     wait 5 sec
N7 M02              program end
```

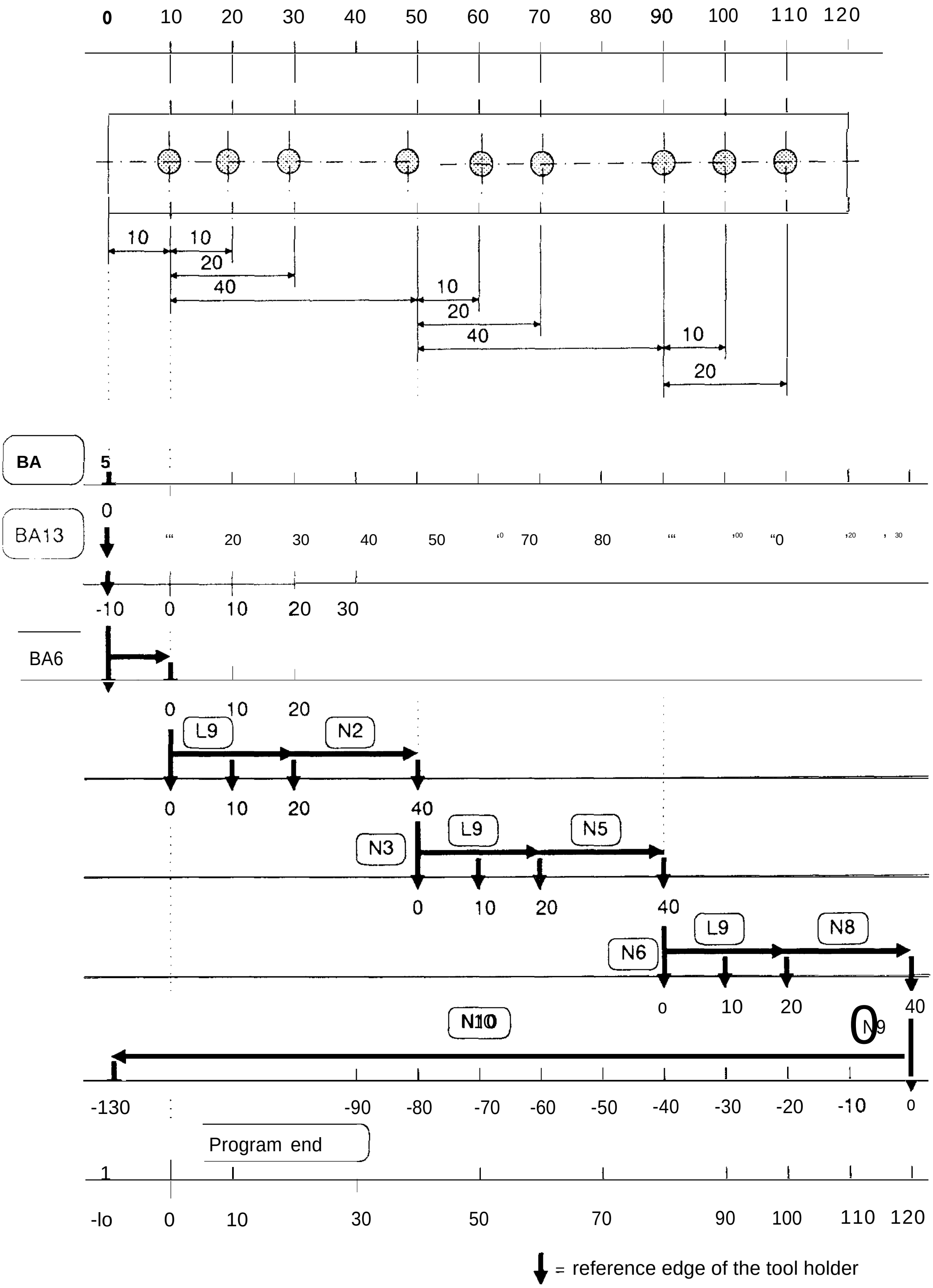


Fig. 2/22 Example of zero offsets

2.6.6.8 Dimensional Units in Machining Programs

The IP247 positioning module interprets machining programs in the dimensional unit specified in the machine data, i.e.:

machine data in 0,1 inches = >G70 default
machine data in mm = >G71 default
machine data in degrees = >G70 and G71 disabled

G70: dimensions **in 0.1 inches**

Following the function G70, all further distances are interpreted in 0.1 inches and all further speeds as 0.1 inches/rein.

G71: dimensions **in mm**

Following the function G71, all further distances are interpreted in dimensional unit mm and all further speeds as dimensional unit mm/min.

2.6.6.9 Reference Point

A statement containing G74 moves the carriage or tool holder to the known reference or home point at **the incremental** speed. No reference point approach for calibration purposes is executed, The carriage always moves to the physically stipulated reference point, regardless of whether the coordinate has been changed by a zero offset. Following the movement, the display of the actual position value takes into account both a zero offset and a tool length offset.

Example

reference point coordinate	. 0 mm
effective zero point offset	. +500 mm and
effective tool offset	= 20 mm

After G74 is executed, the axis is positioned at the physically specified reference point. The tip of the tool juts out 20 mm from this point. -480 mm is indicated as the actual position.

2.6.6.10 Absolute and Relative Dimensions

G90: position specifications absolute (default at program call)

All target information (X-functions) following G90 is interpreted as absolute until G91 is entered.

G91: position specifications relative

All target information (X-functions) after G91 is interpreted as relative until G90 is entered.

2.6.7 The X-Function

The X-function is the target function of the statement. It consists of the character “X”, followed by an optional sign and a number **which specifies a distance in the units** mm, 0.1 inches or degrees. The number consists of five digits and three decimal places.

The maximum range of values is

X-99999,999 .. X+99999.999

Examples

X50, X50., X-.5, X+12345.678,...

If the decimal point is missing, it is assumed to beat the last place in the number,

2.6.8 The F-Function

The F-function describes one of the following:

- the speed,
- a dwell time or
- the number of loop repetitions.

it consists of an ‘F’ and a maximum five digit, signless whole number.

- As a speed, it indicates the units mm/min, 0.1 inches/rein or degrees/rein. The range of values is then 1 .,. 65000.
- As a dwell time, it specifies a multiple of 100 ms and can have values between 1 and 65000,
- If the F-function is interpreted as a number of loop repetitions, the whole range of O to 65000 is possible. If O is specified, the loop is repeated continuously.

2.6.9 The M-Function

The M-function consists of the character ‘M’ and a **two digit number. Permitted values are 0...99.** The significance of the number is only fixed for M02 and MOO.

An M-function is only output in conjunction with a traversing job (X-function or G74) or a dwell time (G04) to the programmable controller and to the programmer. M-functions standing alone in a statement or alone with switchovers or offsets in a statement are ignored (exception: MOO). and not output

Each M-function is output at the beginning of the execution of a statement (traversing job or dwell time) and remains valid until the next M-function at the start of the next statement (traversing job or dwell time) containing an M-function is output.

In the control program, M-functions can be used to trigger user- specific actions, e.g. the switching on and off of plant during the traversing movement of the axis.

If several statements with consecutive different M-functions are programmed with the help of the "flying change" (G I O) the new M-function is output after the transition.

If statements without M-functions are programmed at the beginning of a machining program, M255 is output. This is output until a statement (traversing job or dwell time) containing an M-function is executed. If a machining program does not contain any M-functions except for M02, then M255 will be output during the whole program.

Examples

N05 G91 M-function not feasible, output M255

N05G91	M-function not feasible, output M255
N1 OG74	output M255
N15G24F3	M-function not allowed, output M255
N20X1000F2000M1O	output M 10
N30G57	M-function not feasible, output Ml O
N35X-500F2000	output Ml O
N45G20	M-function not allowed
N50 G74M20	output M20
N60X500F1 00M30	output M30
N65 G26	M-function not feasible, output M30
N70X1000 F2000 M60	output M60
N75M02	final statement/program end

Certain special factors apply to M02 and MOO as follows:

M02

M02 means "program end". Main programs and subroutines are completed with M02. It must be specified in the final statement of the machining program. Following this, no further statements can be appended to the machining program. The final statement can simply consist of the N-function and M02. If M02 is specified in a traversing statement, no further M-function can be specified in the statement.

MOO

MOO means "programmed halt". A statement with a MOO has the effect that the next programmed traversing movement (X-function or G74) or dwell time (G04) is only executed following an enter command. Offsets (e.g. G57 or G43) and switchovers (e.g. G91) following an MOO and before a traversing movement or dwell time are, however, executed before the halt.

N1OX1 OOF1 000M00	(! = break point)
N20 ! G04 F200 MOO	program halt before the dwell time
N30 ! X200 F500	program halt before the traversing movement
N40X1OOFI 000M00	
N50G57 ! X200 F500	break point after the zero offset

If MOO "programmed halt" and G10 "flying change" are programmed in one statement, the programmed halt has priority.

N10G1OX100 F500 MOO	**(! = break point)**
N20 **! X200** F1 000	separated owing to MOO

MOO can stand alone or alone with an offset or a switchover following the N-function in a statement. In this case, MOO acts as if it is programmed in conjunction with a traversing job with traversing distance O. This means that several enter commands may be required following a stop to start the next traversing movement or dwell time.

N1O G10 **X1OO F1OOO MOO**	**(!** = break point)
N20 ! MOO	
N30 G56 ! MOO	
N40 ! X500 F1 000	three enter commands are required from Xl 00 to X500

If in conjunction with a traversing movement or a dwell time only offsets or switchovers follow the MOO or if M02 (program end) follows directly in the next statement, the machining program is no longer stopped.

N1 0X100F1000M00	
N20M02	MOO no longer effective

I n the mode BA9 ("automatic single statement") MOO has no further significance, since in this mode each traversing movement and each dwell time is always started by an enter command. The stop does not need to be acknowledged twice.

N10X100F1000M00	(! = break point)
N1 O ! G04F1O	in BA9 only one enter command necessary to continue processing

Remember, however, that each MOO either alone or alone with an offset or a switchover following the N-function in a statement is handled as a traversing job with a traversing distance of O. This means that although the MOO in mode 9 is ignored, the traversing job with the traversing distance O must still be started with an enter command.

N1OX100 F1 000	(! = break point)
N20 ! MOO	halt due to BA9. After enter, execution of the traversing job with traversing distance O
N30 G57 ! MOO	halt due to BA9. After enter, execution of the traversing job with traversing distance O

2.6.10 Programming Restrictions and Syntax Diagram

To generate feasible statements in machining programs, there are several restrictions and relationships between the functions which are automatically checked by COM247.

- Following an L-function (subroutine call) only the end of the statement is permitted. In the DIN representation, this means that no further entry can be made in this line, in the text representation, only the selection of another statement is possible.
- The X-function (target) must follow the function G00 (rapid traverse) directly.
- No X-function can follow the function G04 (dwell time), an F-function (time) is required.
- No further function can follow G20 (end of loop) in the statement,
- The X-function (target) is not permitted with *G74* (reference point approach).
- If an X-function is programmed without G00, the F-function (speed) must follow.
- Unless an X-function, G04 (dwell time) or G24 (repetition) is programmed, no F-function can be used.

The statement syntax is represented in the following diagram.

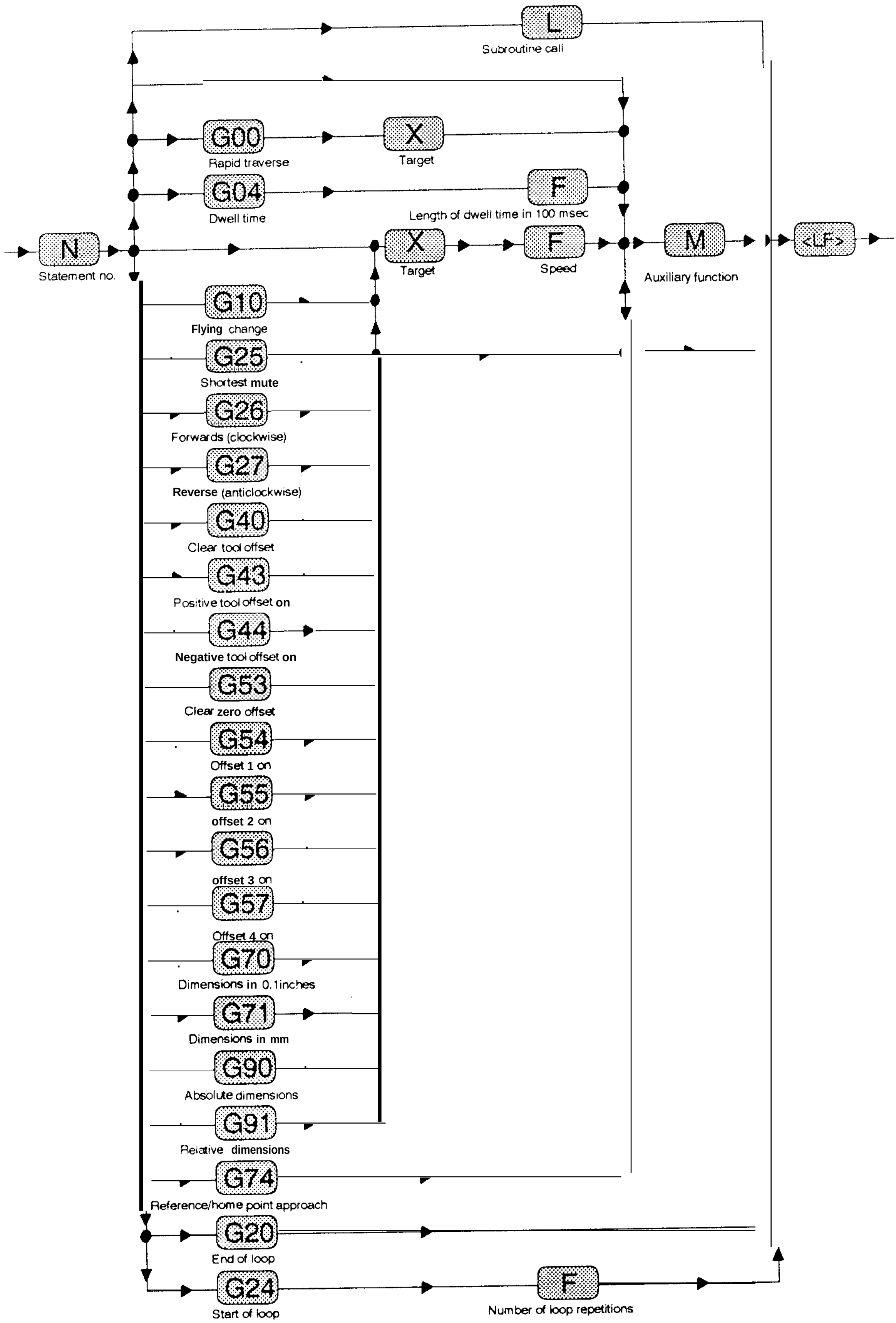


Fig. 2/23 Syntax diagram

2.7 Axis Attributes

The axis attributes contain up-to-date information about the axis as follows:

- the dimensional unit selected for position encoding,
- whether the required position is reached or not,
- (this signal is also output via a digital output of the IP247),
- whether the reference point location is synchronized or not,
- whether the teach-in mode is on or off,
- the existence of the reference point,
- the existence of the machine data on the axis,
- the axis status (“finished” or “running”).

The axis attributes are passed onto the control system via FBI 64 in the checkback signals. (DL (n+1 2) of the axis DB).
(=>Section 6.2.7.2 “Structure of the Axis Data Block”),

Bit	Metric	Inches	Degrees
0	0	0	1
1	1	0	0

Bit		
2	1: In position	0: Not in position
3	1: Ref. point synchronized	0: Ref. point not synchronized
4	1: Axis in teach-in	0: Axis not in teach-in
5	1: Ref. point does not exist	0: Set
6	1: Machine data does not exist	0: Machine data exists
7	1: Job finished	0: Job running

Fig. 2/24 The axis attributes

Apart from the axis attribute which indicates whether the required position has been reached or not, all axis attributes are indicated directly in the test axis selection and modes display in COM247 (=> Section 5.8 “Test”).

2.7.1 **Machine Data does not Exist**

The bit indicating that machine data does not exist is only cleared in the axis attributes (check-back signals) when the machine data for an axis is transferred. The machine data which is then located on the module, may, however, still contain errors. This is ignored at this point. Operating instructions only cause the axis to move when the machine data is free of errors.

2.7.2 MeasurementSystem

In the test display of COM247, the dimensional unit is displayed beside the actual position value and the distance to go.

2.7.3 Reference Point does not Exist

Movements to an absolute target are only possible when a coordinate system has been fixed. The coordinate system is fixed using mode BA5, “reference point approach” or “set reference point”, Following this, the axis attribute indicating the absence of the reference point is reset.

2.7.4 **Teach-in On**

The axis attribute “teach-in on” indicates that the current actual position values of the axis can be stored in a machining program as target information (X-functions). Teach-in is activated with mode BA1 O and deactivated with mode BA11.

2.7.5 Reference Point Synchronized

This axis attribute indicates that the counters of the excitation pattern in the power unit and on the IP are to be synchronized in the “reference point approach” mode. This information is stored in the machine data. The counters are synchronized when the IP247 and power unit are switched on together. If the IP247 recognizes that the power unit has been switched off, (monitoring input on the power unit) the synchronization is lost. Once the I P247 recognizes that the power unit has been switched on again, the synchronization is re-established.

2.7.6 Axis Status ’’Finished”or “Running”

An axis can only be switched from one mode to another in the axis status “finished”. Providing the job itself is correct, the axis status is changed from “finished” to “running”. In this respect, there is no difference between traversing jobs and jobs which do not lead to a traversing movement, e.g. coordinate transformations, or data transfer. On completion of the current job, the axis status once changes to the “finished” status. In automatic operation, the axis status is only set to “finished” on completion or interruption of the machining program (see next section).

2.7.7 “Position Reached” Message

The “position reached” axis attribute is closely related to the axis status. In positioning jobs with absolute or relative target specifications, the “position reached” message signals the correct completion of the job. The “position reached” message is set when the target is reached.

One exception is to be found in automatic operation. While the axis status in automatic operation only changes from “running” to “finished” on completion of the whole machining program, the “position reached” message is generated after each traversing statement and each dwell time.

Response to abnormal termination of positioning jobs
If a job with absolute or relative target information is terminated before the target is reached, this axis attribute is not set. The remaining distance to go to the actual target point remains indicated. It is updated if you subsequently execute a tool offset, You can now send a relative traversing job with the indicated “distance to go” to the module. The originally required target is then reached.

2.8 Digital Inputs/Outputs and their Effects

2.8.1 Inputs and Outputs to the Power Unit

The I P247 positioning module has digital inputs and outputs via which it is connected to the power units and to the plant. It has one input connected to the PC, via which the BASP signal (block command output) can be received from the CPU.

Control and ready signals are exchanged with the power unit.

Control signals:

- Positioning pulses Tx, $\overline{Tx}$ x = (axis 1,2, 3)
- Direction RPx, RPx
- Reset RSx, RSx

Ready signals:

- +24 V from the module to the power unit BBxL+ x = (axis 1, 2, 3)
- Ready input feedback of the 24 V from the power unit to the IP BBx

Significance of the control signals:
The outputs “positioning pulses” and “direction” can be assigned parameters in the software. Using the machine data “polarity”, the active pulse edge and therefore also the inactive level and the signal level for the direction of rotation can be selected, (See machine data “polarity”.)

The output “reset” is used to disable the power unit when the axis is not installed, and to synchronize the power unit and the module for a synchronized reference point approach, As long as there is no machine data on the module, this output carries a high signal. As soon as there is valid machine data on the axis, this output changes to low, At the beginning of the reference point approach, a high signal is applied to this output for 100 ms. Providing the power unit has a reset input, this synchronizes the power unit and the IP247.

Note:

If the reference point approach is to be synchronized and the power unit does not have a reset input, you must make sure that the module and power unit are switched on and off at the same time. The IP247 assumes that the excitation pattern counter of the power unit is at zero when it is switched on. This is always the case after the power unit has been switched on or reset.

Ready signal **BBx:**
The power unit can be monitored for overload and power down via the digital input BBx. To do this, you can loop the 24 V available at output BBxL+ via a floating contact of the power unit to the input BBx. When the power unit is switched off, the contact must close.

2.8.2 The ”Position Reached” Message

The “position reached” message is supplied both to the CPU as well as to a digital output. You can find a detailed description of the “position reached” message in Section 2.7 “Axis Attributes”,

2.8.3 The Digital Inputs for Hardware Limit Switches

The hardware limit switches are evaluated regardless **of the axis type. With a rotary axis, hard-** ware limit switches are generally not required, but can be used as an additional safety measure.

The polarity of the two hardware limit switches can be assigned in the machine data record. You can select both limit switches as normally closed or both as normally open switches using the “polarity HW limit switches” parameter.

The hardware limit switches are only detected during a traversing movement.

If the module recognizes that a hardware limit switch has tripped, the traversing movement is stopped and the current traversing job is terminated.

Note

A hardware limit switch is only detected if the traversing movement is towards it. If a limit switch responds, further movement in the direction of the activated limit switch is not possible. This traversing direction is only enabled again

- when the module has detected that the axis has left the hardware limit switch in the opposite direction or
- when a hardware limit switch has been tripped and is tripped again in the opposite direction.

If a hardware limit switch is tripped either manually or by some other external event, movement in this direction is blocked. This direction can be released again by starting a traversing job in the opposite direction, tripping the hardware limit switch again and then returning it to the neutral position.

A blocked direction is also released when the axis reaches the precontact.

After completing the parameter assignment and starting up your system, each axis (linear axis) has two software limit switches. These should always be assigned so that the hardware limit switches can never be reached during operation. Since the IP247 **only starts to brake when a** software limit switch is reached, the hardware limit switches should be set far enough away from the software limit switches to allow for the maximum braking distances.

The maximum braking distance can be calculated for $t_v = 3\tau$ as follows:

$$S_{brake} = \text{pos. resolution} \cdot (F \cdot (t_v + -c \times (e^{(+tv/\tau)} \cdot 1)) + f_{ss} \times t_v)$$

Where:

f_{ss} : start-stop frequency
f_{max} : maximum frequency
F : theoretical maximum frequency = $(f_{max} - f_{ss})/0.95$
t_v : acceleration time $[0...3\tau]$
τ : ramp-up constant = F/a
a : rate of frequency increase

2.8.4 External Start/Stop

The digital input "external start/stop" has two functions. A signal change from "O" to"1" serves as an "external stop", the change from "1" to "O" serves as an "external start enable".

External stop
During the processing of a traversing job, a signal change from "O" to"1" at this digital input causes the error message "external stop received", the traversing job and current mode are terminated. If the external stop is received while processing a machining program (automatic mode), the machining program is interrupted.

External start
If a"1" is set at this digital input before the start of a traversing job, the traversing job is interpreted by the IP247 but is not executed. If the job is permissible, you will obtain the message "motor waiting for external start". The negative edge of the signal at the digital input causes the traversing job to be executed.

Only one single job can be waiting for execution. Other jobs during the waiting time are not allowed. If a further operating instruction is sent to the IP247, the job currently waiting for execution is deleted, A stop command in conjunction with any mode causes error-free termination. The message "motor waiting for external start" is reset. Any command other than "stop" leads to the error message "job not permitted".

The external start enable is also effective in the automatic mode with traversing jobs and dwell times. If the signal "1" is set at this digital input before the start of the automatic mode (mode 8), the machining program starts with the start command and is executed up to the first traversing job or the first dwell time. The dwell time or traversing job then causes the message "motor waiting for external start". The negative edge at the digital input enables the traversing movement or dwell time. A further statement within the started machining program cannot be blocked with the "external start enable", since the positive edge of the signal at the digital input is then evaluated as "external stop". (Exception: module waiting for "enter signal" after "programmed halt" (M OO).)

If a"1" is set at this digital input before the start of the "automatic single statement" mode (mode 9), the start command also leads to the execution of the machining program up to the first programmed traversing job or first dwell time. The "enter command" then causes the message "motor waiting for external start". The first traversing job or the first dwell time and all the offsets and switchovers programmed after it are then executed on the negative edge of the signal at the digital input, If signal "1" is set again at the digital input during the execution of the traversing job or dwell time, this acts as “external stop”; the machining program is terminated. If, following the completion of the traversing movement (“position reached” message set), a”1” is set at the digital input, the machining program is also interrupted. The message “FC1 (65) machining program waiting to continue” is output. After “enter”, the message “motor waiting for external start” appears. If the IP247 then detects a negative edge change, the statement is executed.

Example: N1 X1OO F2500 MOO
N2 X200 F1000 M20
N2 .,..

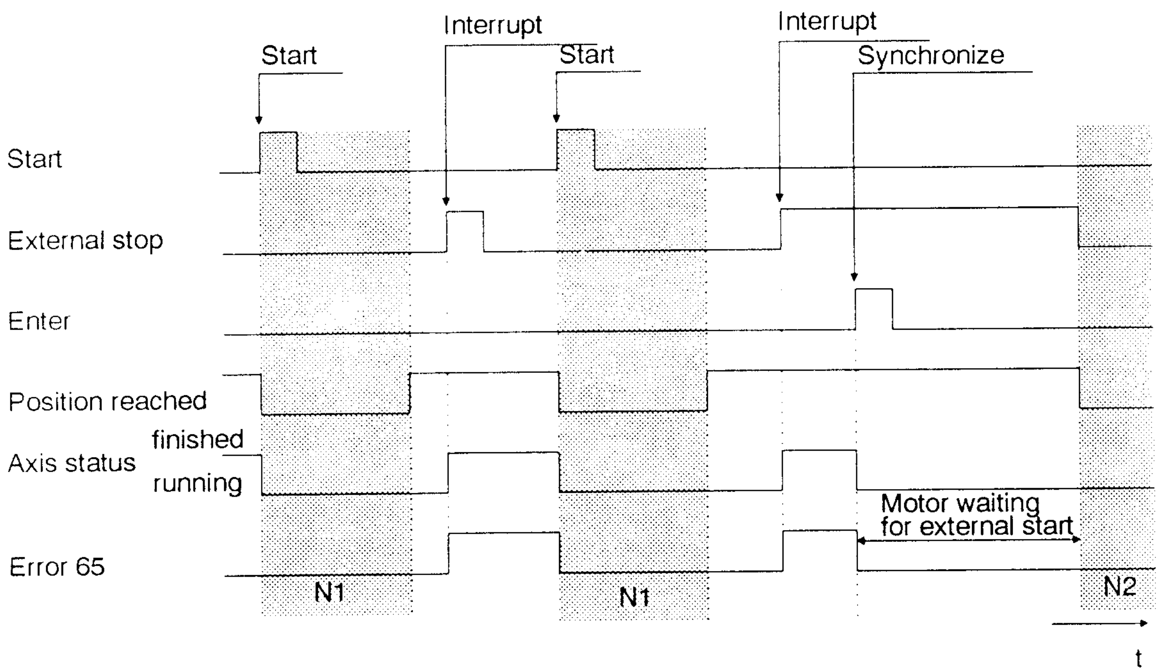


Fig. 2/24 External start-stop

2.9 BASP Signal

Whether or not this signal is evaluated depends *on* a jumper setting on the IP247 (see Instructions). If the signal is active and is evaluated, the PEx outputs are switched to low, the traversing jobs on the axes are aborted and the message “PC failure” is output.

3 Hardware

3.1 Technical Description

3.1.1 Mode of Operation

The IP247 as an intelligent I/O module controls positioning equipment driven by stepper motors. The IP247 outputs pulse trains to the connected stepper motor power unit corresponding to the target position and the traversing speed, The number of pulses output determines the distance travelled, the frequency of the pulses determines the speed of travel. A direction signal to indicate the direction of travel is also output.

The module has the following features:

- 16-bit microprocessor with internal timer and interrupt controller (801 86)
- 16 Kbyte local RAM, backed up by the PC battery
- EPROM cartridge interface for loading the firmware
- dual-port RAM, backed up by the PC battery
- bus interface to S5 programmable controllers
- serial interface to S5 programmers
- three interfaces to stepper motor power units
- twelve 24 V digital inputs
- three 24 V/l 20 mA digital outputs
- two status LEDs.

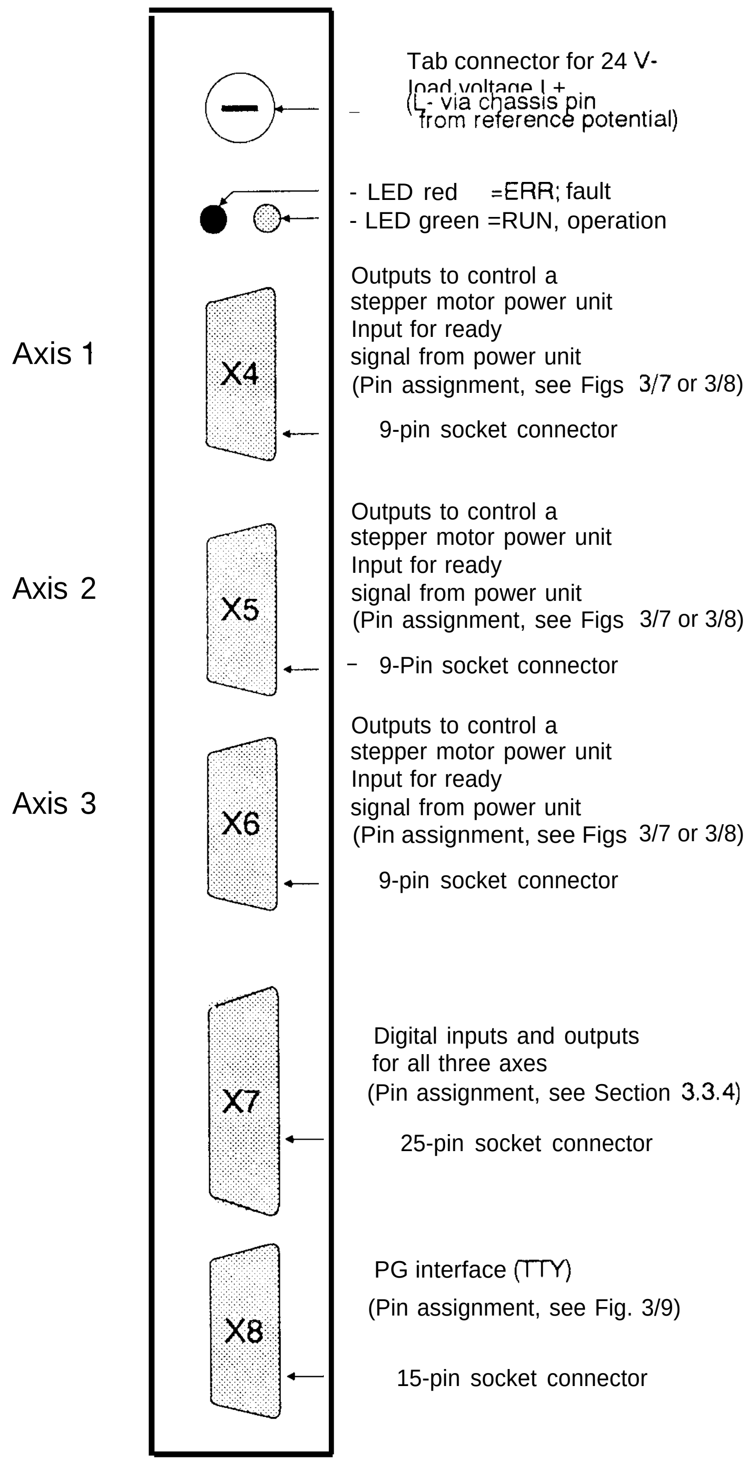
The operation of the positioning module is controlled by the microprocessor according to the operating program (firmware) stored in an exchangeable EPROM cartridge. The parameter assignment, programming and start- up are performed via the PG interface using the software package COM 247. If the data is stored in the CPU, parameter assignment and programming can also be performed via the PC interface, e.g. when exchanging a module. Providing the module remains plugged into a battery-backed PC frame, the machine data and machining programs stored in the RAM of the IP247 are retained if a power failure occurs.

Communication with the programmable controller is via the S5 bus interface and a dual-port RAM with a capacity of 4 Kbytes.

To connect stepper motor power units, the I P247 module has three identical interfaces with outputs at connectors X4,X5 and X6. You can connect both power units with optocoupler inputs (5 V/20 mA, 24 V/20 mA, 15 V/20 mA if an external voltage of 5...24 V is supplied) as well as 5 V differential inputs. You select the type of input by setting jumpers on the module. When operating with voltages between 5 V and 24 V, you must apply this voltage to connector X7.

Connectors on the front panel

6ES5247-4UA31/4UA41



Fig, 3/2 Front panel

3.1.4 Technical Data

Interfaces to stepper motor drives (front panel connectors X4, X5, X6)

Output signals (per axis)
(n = axis number 1,2 or 3)

Clock pulse	Tn	
Clock pulse inverted	$\overline{\text{Tn}}$	
Direction level	RPn	
Direction level inverted	$\overline{\text{RPn}}$	
Reset	RSn	
Reset inverted	$\overline{\text{RSn}}$	
Output voltages		
with + **5** V supply:	signal O	max. 0.4 V
	signal 1	min 4.5 V
with L+ = 24 V supply:	signal O	max. 0.4V
	signal 1	min. L+ -0.4 V
with U_s = **15** V supply:	signal O	max. 0.4 V
	signal 1	min. Us -0.4 V
Output current		20 mA
Input for ready signal		BBn
Isolated		no
Input voltage	signal O	-33 V...+ 3 v
	signal 1	10.5V...33V
Input current		typ. **7** mA
Voltage for contact BBn+ (ready signal)		24 V (from backplane connector X2)
Load current		max. 20 mA (short-circuit **proof)**
Permitted cable length		100 m **(screened)**

Digital inputs (front connector X7)

Rated input voltage		24 V
Number of inputs per axis		4
Isolated		no
Input voltage	signal O	-33 V...3.6 V
	signal 1	13V.,.33V
Input current		typ. 9.5 mA

You can use two-wire BEROS with a supply voltage of 22 V...33 V.

Digital outputs (front connector X7)

Rated supply voltage L+	24 V
Number of outputs per axis	1
isolated	no
Range of supply voltage	20 v to 30 v
Switching current	max. 120 mA, short-circuit proof
Max. total load of the outputs at 60°C	10070

Power supply

Supply voltage from system bus	+5 v $\pm$ 5%
Current consumption	approx. 0.8A

supply voltage L+ (front connector)

Rated value	24 V
Ripple U_{pp}	3.6V
permitted range (including ripple)	20 v to 30 v

Special voltage U_s (applied if necessary via X7, ground via M_{ext} contact)

Rated value	15V
Permitted range	5to 30 v

Current consumption without load

from L+ (24 V)	typ. 50 mA
from U,(15 V)	typ. 35 mA

Battery voltage (back-up)		2.7...5.25V
Current from the battery		typ. 5 µA; max. 250 µA

Safetytest

Surge voltage test according to IEC 255-4 inputs and outputs to L-	U_s = 1 kV; 1.2/50 µs
Interference voltage test according to IEC 255-4 inputs and outputs to L-	U_s = 1 kV, 1 MHz

Mechanical data

Dimensions (W x H x D)		
version with forced ventilation	(-4UA31)	20 mm x 233 mm x 160 mm
self -ventilated version	(-4UA41)	40 mm x 233 mm x 160 mm
Weight		approx. 0.4 kg

Ambientconditions

Operating temperature		
version with forced ventilation	(-4UA31)	0...60°C
self-ventilated version	(-4UA41)	0...55°C
Storage and transport temperature		-40...+70°C
Relative humidity		max. 95% at 25°C

3.2 Installation

3.2.1 Inserting and Removing the Module

The module can only be plugged into the slots interided for CPS in the PC or EU. **The module may only be removed when the programmable controller or the expansion unit is switched off.**

3.2.2 Connecting the Signai Lines

The signal lines are connected via the connectors on the front panel. The braided shield is connected to the metallized part of the connector cover.

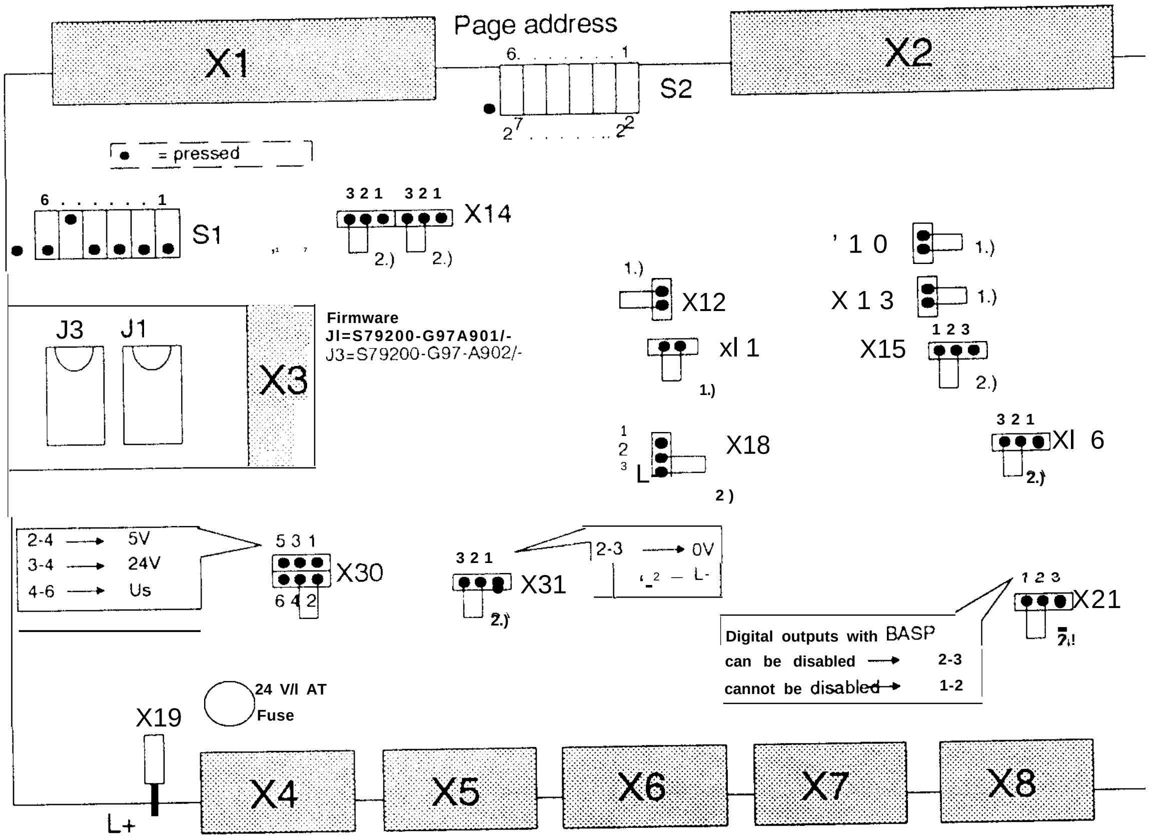
Connecting cables to the power units should be laid with shield clamps at the device reference potential, as recommended in the Installation Instructions C79000-B8576-C452, Section 7.7.

Note

With the exception of the PG interface, the insertion and removal of the front connectors during operation of the module is not permitted.

3.3 Operation

3.3.1 Position of the Jumpers and Switches



1.) Test points: jumpers XI O, XI 1, XI 2 and XI 3 must always be plugged in.
2.) Jumpers inserted at the factory.

Fuse: GWK-No. W79054-M1041-T100

Fig. 3/3 Jumpers and switches for the IP247-4UA31/-4UA41

3.3.2 Setting the Module Address

Data is exchanged between the CPU and the IP247 via the S5 bus interface and a dual-port RAM with a memory capacity of 4 Kbytes, divided into four “pages”. Each axis to be controlled is assigned one page. The fourth page is used to transfer machining programs.

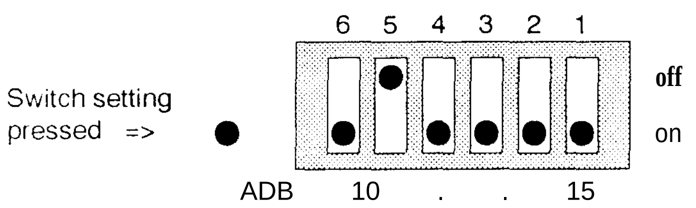
Thepagesfor all IP247s are in the address area from OF400H to OF7FFH (61 Kbytes to 62 Kbytes -1), which is set at the factory. You must simply set the page number for the first page (first axis), 0,.,252 (in multiples of four).

The four pages of a module must have consecutive numbers. The addresses for the following pages are calculated automatically by the IP247, after you have set the base address.

When supplied, each module is set with the same address area for the page number (switch S1 and jumpers Xl 4, Xl 5 and Xl 6).

Address area OF400H to OF7FFH (61 Kto62 K-1)

Switch S1

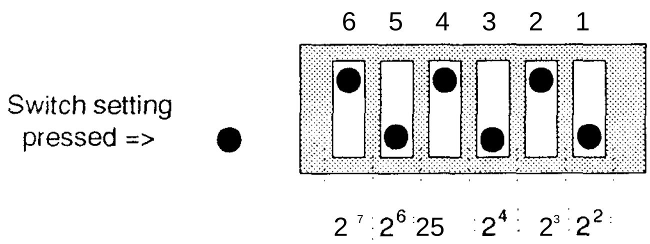


Jumpers X14 / 2-3
xl 5 / 1-2
Xl 6 / 2-3
Xl 7/2-3
Xl 8/2-3

Fig, 3/4 Switch setting at switch S1

You must set the page number of the module (even-numbered base address of the first axis) be- tween O and 252 in steps of four, using switch *S2.*

Switchs2



Example: page number 84
for the first axis

$2^2 + 2^4 + 2^6 = 4 + 16 + 64 = 84$

Fig, 3/5 Switch setting at switch S2

The page addresses 85, 86, 87 for the following pages are automatically decoded by the module.

The first page address of the next module can then be set to 88.

Disable Command Output

The BASP signal (disable command output), which is triggered by the PC (e.g. when it changes to STOP or if the load voltage drops below 15 V), can be used to disable the digital outputs on the module.

Jumper X21/1 -2 inserted	digital outputs are not disabled when BASP is output
Jumper X21/2 -3inserted	digital outputs are disabled when BASP is output

3.3.3 Connecting Stepper Motor Power Units

Three stepper motor power units can be connected to the module (X4, X5, X6). The signals "clock pulse" (T), "direction level" (RP) and "reset" (RS) are supplied via special output stages, which can be operated with 5 V, 24 V or with a special voltage U_s (5 V to 24 V). This allows power units with 5 V differential inputs (RS 422) or optocoupler inputs (5 **V/20** mA, **24 V/20** mA) to be connected. If a special voltage U_s (5 V to 24 V) is used via connector X7, the outputs of the module can also be operated with this voltage,

The three interfaces must be operated with the same voltage.

The output circuit is shown schematically in the following figure for an output signal (e.g. clock pulse 1).

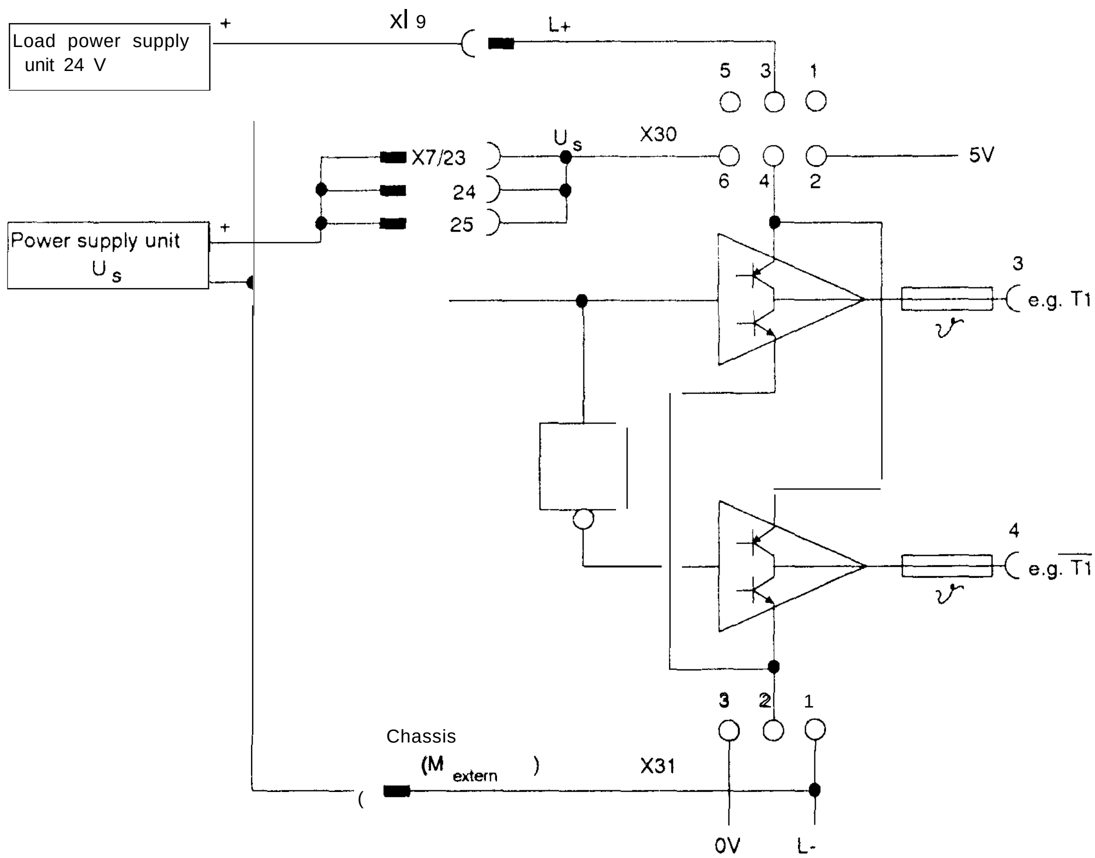


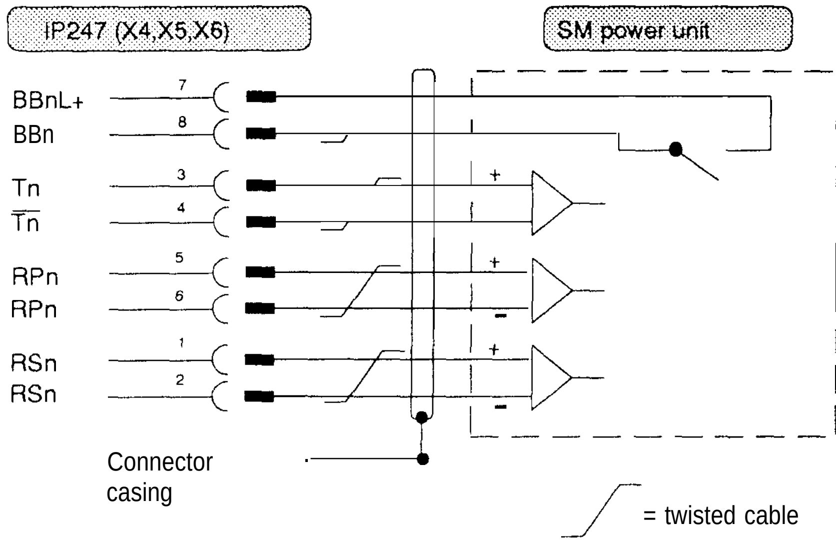
Fig. 3/6 Output circuit for controlling power units

Connection of power units with 5 V differential inputs

You must set the jumpers on the module as follows:

jumper X30/2-4 inserted (5 V)
jumper X31/2-3 inserted (O V)

You must connect the power unit as shown in Fig. 3/7.



Fig, 3/7 Connection of power units with 5 V differential inputs to connectors X4/X5/X6 of the IP247 module

To reset the power unit, the module outputs a high-active pulse for each axis, If a low-active pulse is required, you must change over the connections at **pins 1 and** 2 of the connector. The polarity of the clock pulse and direction level can be programmed.

Connecting power units with 5V optocoupler inputs

On the module, you must make the same jumper setting as for 5 V differential signals:

jumper X30/2-4 inserted (5 V)
jumper X31/2-3 inserted (0 V)

You must connect the power unit as shown in Fig. 3/8.

Connecting power units with 24V optocoupler inputs

You must make the following jumper setting on the module:

jumper X30/3-4 is inserted (24 V)
jumper X31/1 -2 is inserted (L-)

You must connect the power unit as shown in Fig. 3/8.

Connecting power units with 5...24 V optocoupler inputs

You must make the following jumper setting on the module:

jumper X30/4-6 inserted (US)
jumper X31/1 -2 inserted (L-)

If you are using a special voltage, the voltage Us must be supplied via connector x7/23, 24, 25 (see Fig. 3/6).

You must connect the power units as shown in Fig. 3/8.

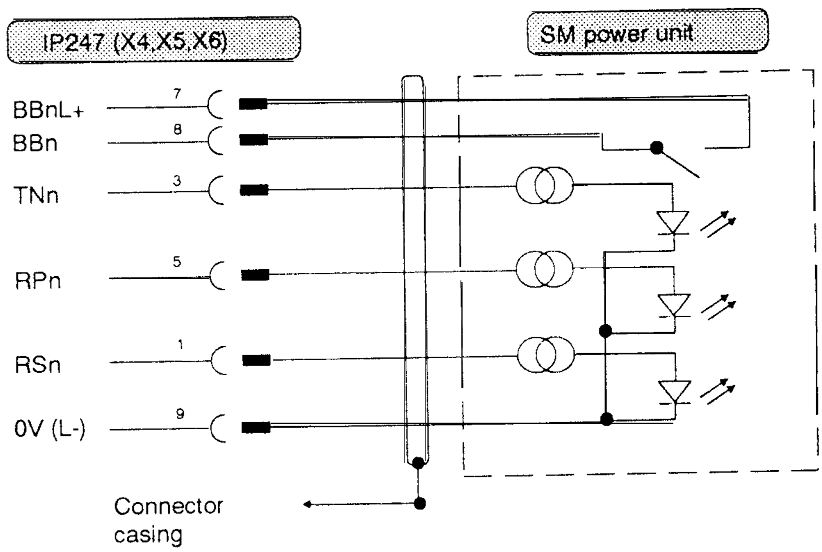


Fig. 3/8 Connection of power units with opto-coupler inputs to the connectors X4/X5/X6 on the IP247 module.

Pin assignment of the connectors for connecting power units (X4, X5, X6)

The pin assignment of the three connectors for axes 1 (X4), 2 (X5) and 3 (X6) is the same.

Socket	Connection for:	Axis 1 (X4)	Axis 2 (X5)	Axis 3 (X6)
1	Reset signal	RS1	RS2	RS3
2	Inver. reset signal	RS1	RS2	RS3
3	Clock pulse	T1	T2	T3
4	Inv. clock pulse	T1	T2	T3
5	Direction level	RP1	RP2	RP3
6	Inv. direction level	RP1	RP2	RP3
7	24 V output for BB contact	BB1L+	BB2L+	BB3L+
8	Ready signal	BB1	BB2	BB3
9	Chassis	L-	L-	L-

3.3.4 Digital Inputs/Digital Outputs

The digital inputs/outputs for all three axes are connected to the 25-pin connector X7 on the front panel. You can connect current sourcing switches (contacts or two-wire BEROs) to the inputs.

The function signals (position reached) are output via short-circuit proof digital outputs.

Pin assignment of connector X7 for digital inputs/digital outputs

Socket	Connection for:		
1	Limit switch axis 1	I	ANF1
2	Reference switch axis 1	I	BERO1
3	Limit switch axis 1	I	END1
4	External start/stop axis 1	I	START_N/STOP1
5	Limit switch axis 2	I	ANF2
6	Reference switch axis 2	I	BERO2
7	Limit switch axis 2	I	END2
8	External start/stop axis 2	I	START_N/STOP2
9	Limit switch axis 3	I	ANF3
10	Reference switch axis 3	I	BERO3
11	Limit switch axis 3	I	END3
12	External start/stop axis 3	I	START_N/STOP3
13	Position reached axis 1	Q	PE1
14	Position reached axis 2	Q	PE2
15	Position reached axis 3	Q	PE3
16		Q	
17...22	Not used		
23	Special voltage	I	U_s
24	Special voltage	I	U_s
25	Special voltage	I	U_s

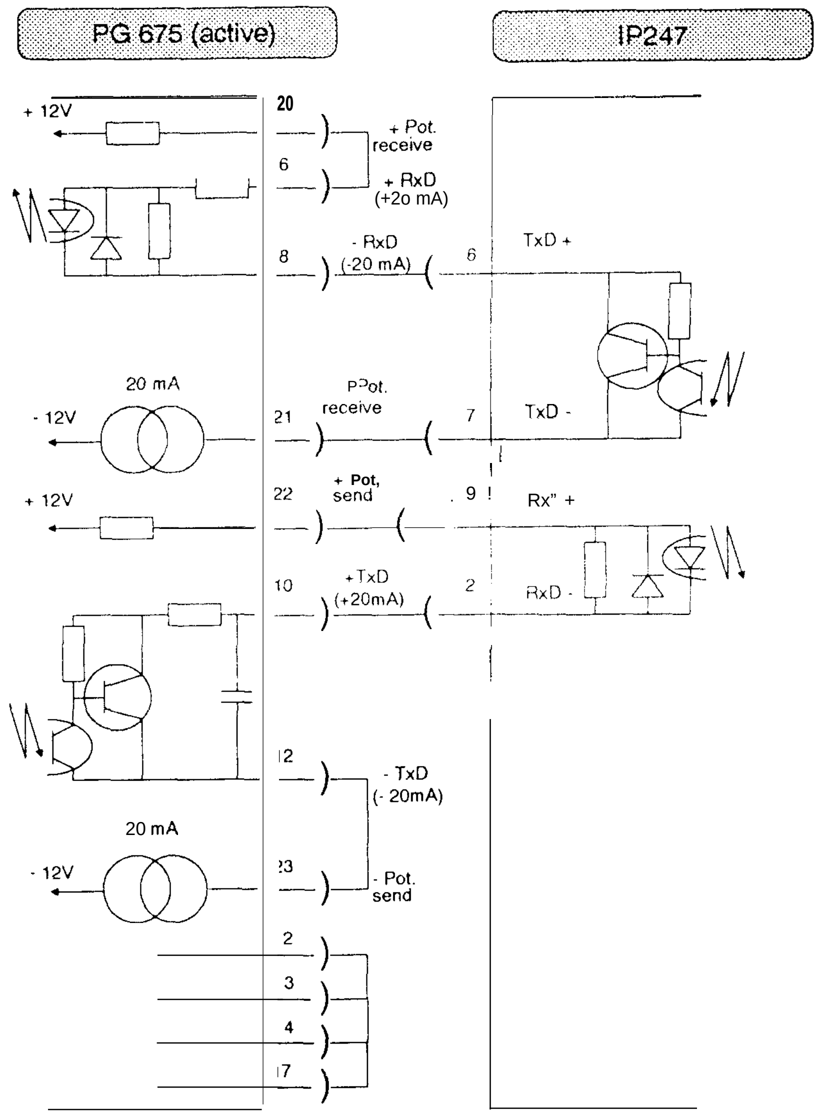
I = input; Q = output

Note

The chassis for the special voltage is supplied via the M_{ext} contact; i.e. the minus pole of the special voltage must be applied to the common chassis pole.

3.3.5 PG Interface 20 mA

The programmers PG 635, PG 675, PG 685, PG 695, PG 730 and PG 750 can be connected to the I P247 at connector X8 via connecting cables (e.g. 6ES5 731- 1.. .0).



Connecting cable 6ES5 731-1 ---4

Fig, 3/9 Connecting the programmer to the IP247

To set the transmission rate of the programmer (PG), you must connect pins 2, 3, 4, 17 together in the connector on the PG side (transmission rate 9600 bps). When using standard cables, this speed is already set.

In the programmable controllersS5-135U and S5-1 55U, you can also use the PG interface via the backplane bus. To do this, you must insert the module in a suitable slot. The module is then operated via the coordinator module (for details, refer to the Coordinator Instructions).

Pin Assignment of Connector X8 for the PG Interface

Socket	Connection for:
1	Shield
2	RxD -
3	---
4	24 V
5	---
6	TxD +
7	TxD -
8	Shield
9	RxD +
10	Ground
11	-20 mA/transmitter
12	---
13	-20 mA/receiver
14	---

Backplane Connectors X1/X2 and Memory Cartridge Connector X3

	d	b	z
2		Ground	+5 V
4	UBATT		
6	ADB 12	ADB O	
8	ADB 13	ADB 1	/MEMR
10	ADB 14	ADB 2	/MEMW
12	ADB 15	ADB 3	/RDY
14		ADB 4	DB O
16		ADB 5	DB 1
18		ADB 6	DB 2
20		ADB 7	DB 3
22		ADB 8	DB 4
24		ADB 9	DB 5
26		ADB 10	DB 6
28	/DSI	ADB 11	DB 7
30		BASP	
32		Ground	

Fig. 3/1 O Backplane connector Xl

	d	b	z
2		Ground	+5 V
4			
6			
8			
10			
12			1 - . .
14			/NAU
16 -			
18			
20			
22	TxDSn		
24			
26		RxDSn	
28			
30	—	Gnd 24V	Gnd24V
32		Ground	+24 V

Fig. 3/1 1 Backplane Connector X2

	d	b	z
1	SADB12	Ground	+5 v
2	SADB0	SADB1	SADB2
3	SADB3	SADB4	SADB5
4	SADB6	SADB7	SADB8
5	SADB9	SADB 10	SADB11
6	SADB13	SADB14	/RD
7	+5Volt	SDB8	SDB9
8	SDB 10	SDB 11	SDB 12
9	SDB 13	SDB 14	SDB 15
10	SDB O	SDB 1	SDB 2
11	SDB 3	SDB 4	SDB 5
12	SDB 6	SDB 7	
13	/UCS	/UCS	
15	SADB15		
16	+5 V	Ground	

Fig, 3/1 2 Memory Cartridge Connector X3

3.4 Connecting Cables

To make the connection of power units and digital inputs/outputs easier, connecting cables are available with one end open.

Connecting cable for power units 6ES5 704-4...0

(cable end open, . . . = length key for connecting cables)

Signal	Pin	Wire color	Identifier
RS	1	bl	
RS	2	rd	
T	3	gr	
$\overline{T}$	4	ye	1 Ring
RP	5	gn	
RP	6	br	
BB L+	7	wt	
BB	8	bk	
	9	bl	2 Rings
	Casing	Shield	

Fig, 3/1 3 Connecting cable for power units

Connecting cable for digital inputs/outputs 6ES5 704-5...0
(cable end open, ...= length key for connecting cables)

Signal	Pin	Wire color	Identifier
I ANF 1	1	bl	
I BERO 1	2	rd	
I END 1	3	gr	
I START/STOPP1	4	ye	1 Ring
I ANF 2	5	gn	
I BERO 2	6	br	
I END 2	7	wt	
I START/STOPP2	8	bk	
I ANF 3	9	bl	
I BERO 3	10	rd	
I END 3	11	gr	
I START/STOPP3	12	ye	2 Rings
Q PE 1	13	gn	
Q PE 2	14	br	
Q PE 3	15	wt	
	16	bk	
I U_s	23	bl	
I U_s	24	rd	
I U_s	25	gr	

I = input, Q = output

Fig, 3/1 4 Connecting cable for digital inputs/outputs

4 Functions

4.1 Principle of Operation

The module is operated by means of commands and instructions, regardless of whether they are sent to the IP247 by the CPU or by a programmer. Commands are divided into two basic groups:

instructions for "operating" and
commands for "monitoring".

Operating instructions are used for the following:

- to setup an axis (input of machine data),
- to set (change) modes,
- to start the execution of a mode,
- to abort the execution of a mode

The reaction to an operating instruction depends on various factors.

- The instruction must be feasible.
- It must be feasible in the currently set mode and during execution of the mode.
- It must not contradict the "axis attributes" which determine whether or not an operating mode is permissible at a given time (=> Section 2.7 "Axis Attributes").
- It must not contradict the mode of the other axes, if they have already been set by previous operating instructions (e.g. "teach-in" or "delete program").

If all these conditions are fulfilled, the operating instruction will be processed, otherwise an error message is output and with a few exceptions processing is terminated.

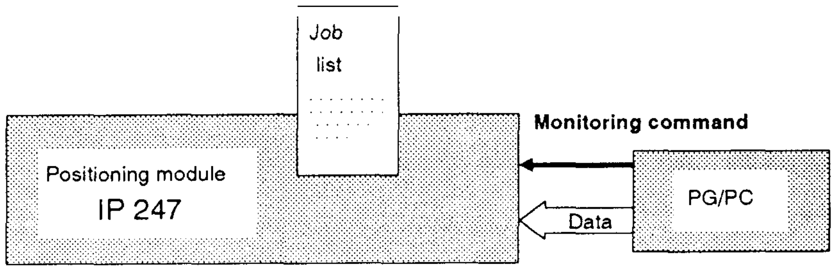
Operatinginstructions are entered in the appropriate PC or PG job list on the IP247 in the order in which they are received. In each IP247 cycle an attempt is made to fetch and interpret the oldest valid job in this list. If the job is permitted in the current axis status, it is executed immediately. If it contradicts the current mode, the mode is terminated and an error message output.

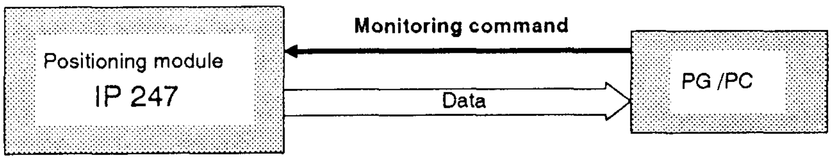
> **Note**
>
> An operating instruction which causes a mode to start should only be transferred to the module when the previous job is complete.

While an axis is braking, further jobs are accepted, however, not interpreted since they would trigger a stop and the module is already braking. If several jobs are sent to the IP247 during this phase, it is possible that an entry cannot be made in the job list. These jobs are then lost. The IP247, however, outputs the error message "PC (or PG) job list is full".

With certain operating instructions (transferring a machine data to the IP247), data are also sent to the module along with the instruction. The reactions in this case are explained in the description of the modes.

Monitoring commands are used to fetch the axis attributes (checkback signals), the module errors and information about the actual value and distance to go from the module cyclically. They are independent of the operating instructions and can be sent to the module at any time. They are processed immediately. The next monitoring command can only be input to an interface when the previous command has been processed. Monitoring commands can be sent to the





positioning module simultaneously by both interfaces, without the commands interfering with each other.

Fig, 4/1 Operating instructions and monitoring commands

Error messages resulting from incorrect operation or an external event (e.g. limit switch responded) are not reset until they are acknowledged by the input of a new operating instruction at one of the two interfaces. Between the error message and the acknowledgement, any number of monitoring commands can be entered: the unacknowledged error will continue to be output along with the monitoring information.

Each of the axes is always in one of the operating modes. After power on, the “axis off” mode (see below) is set. Within each mode, an axis can be active or idle. This is expressed by the “axis status” which can have the values “running” or “finished” (=> Section 2.7 “The Axis Attributes”). The axis status can be interrogated via both interfaces. The axis status appears in the test display on the PG; on the PC side it can be read in the checkback signals using FBI 64. (=> “Standard Function Blocks FB164 and FB165").

In the axis status “finished”, an axis can be changed from one operating mode to any other operating mode, unless prevented by the restrictions mentioned above. The operating instruction includes the required operating mode number and a “command”. The command can be “start”, “stop”, “forward”,” reverse” or “enter”. The axis then changes to the required mode, The mode is, however, only executed when the set mode and the command represent a feasible combination, i.e. “jog speed 1, forward”. The relationship between commands and individual modes is explained in more detail in the description of the modes.

If the input is correct, the axis begins to execute the mode. The axis status changes from “finished” to “running”. Once the mode is completed, the axis returns to the status “finished” and can be started again.

If the execution of a mode is to be terminated, you once again send an operating instruction specifying a mode and the command “stop” or the mode “axis off” and “start” to the module. The axis then changes to the “finished” status of the terminated mode, Modes used for data transfer or coordinate transformation cannot be terminated, since this could lead to inconsistencies.

In the “jog” and “incremental” modes (BA 1,2,6 and 7) it is also possible to specify a speed at the start which differs from the speed in the machine data. The speed must be in the valid range from 1-65000 mm/min (or 1 -650000.1 in/rein or 1-65000 deg/min) and must not exceed the maximum speed (max. frequency) programmed in the machine data record. If the maximum speed would otherwise be exceeded, the speed is changed to the maximum upper or lower limit and the error message “speed range exceeded” is output. If the value “O” is transferred, the speed selected for this mode in the machine data will be used.

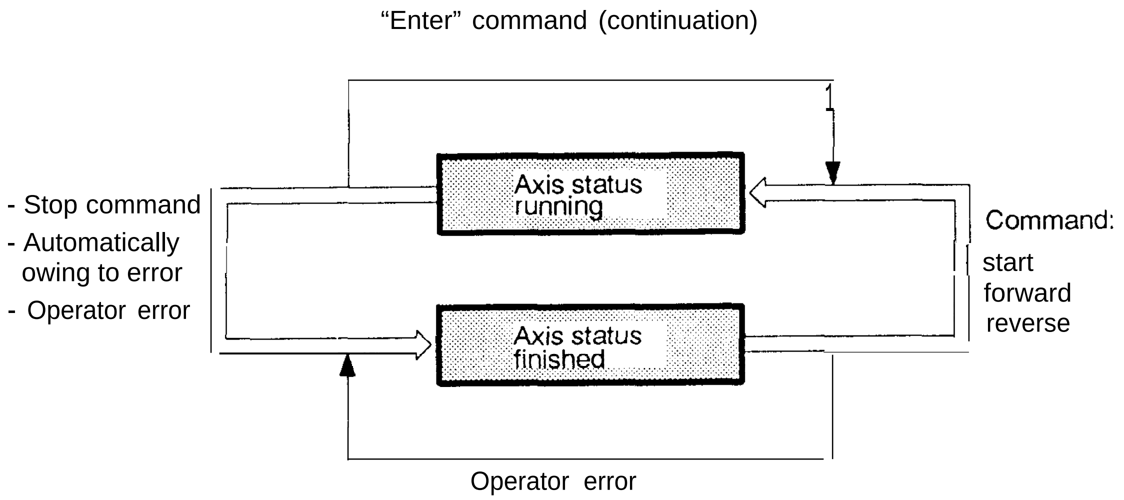
4.1.1 **Operating Instruction**

An operating instruction consists of the following parts:

Select axis	Mode	Parameter	Command
- Axis 1,	e.g. jog	- According to the mode	- Start
- Axis 2 or	incremental	e.g. speed	- stop
- Axis 3	automatic		- Forward
	etc.		- Reverse
			- Enter

Fig, 4/2 Structure of an operating instruction

The relationship between modes, operating instructions and the axis status can be seen in the following diagram:



Fig, 4/3 Relationship between operating instruction and axis status

You can only change or start a mode in the axis status "finished", This is achieved by the start, forward or reverse command in conjunction with the required mode.

Operating instructions entered while a mode is running result in the error message "job not permitted" and the current action is terminated. If an operating instruction is simply incorrect, the axis remains in the "finished" status.

The axis can change from the "running" status to the "finished" status for a number of reasons.

These include the following:

- a stop command in a mode,
- a start command in the "axis off" mode,
- an operator error (e.g. "enter" command with a different mode),
- an error resulting from an external event (e.g. external stop command or a limit switch being tripped) or
- the correct completion of a job (e.g. approach to a particular target point or entry of machine data).

The enter command is required for the following tasks:

- to trigger a single traversing movement in the "automatic single statement" mode,
- to acknowledge a "programmed halt" in the "automatic" or "automatic single statement" modes,
- to continue an interrupted machining program,
- to store statements in the teach-in mode.

Note

When entering instructions at the PG, remember that everything you enter faster than can be processed by the PG or by the COM247 software is written to a key-board buffer in the PG. If all the stored inputs are feasible and correct, they will be entered in the PG job list in the order in which they were input and then Processed by the IP247. This-can lead to a stop command being delayed.

The modes of the IP247 can be selected both by the PC and PG interfaces.

The operating modes of the IP247.

The following modes can be called directly by COM247 and by the PC via FBI 64:

- BA 1- Jog speed 1
- BA 2- Jog speed 2
- BA 3- Free
- BA 4- Axis off
- BA 5- Reference point (approach or set)
- BA 6- Incremental (target approach) absolute
- BA 7- Incremental (target approach) relative
- BA 8- Automatic
- BA 9- Automatic single statement
- BA 10- Teach-in on
- BA 11- Teach-in off
- BA 12- Zero offset absolute (set actual value)
- BA 13- Zero offset relative (offset coordinate system by value specified)
- BA 14- Clear zero offset
- BA 15- Tool length offset
- BA 16- Tool length offset off
- BA 17 - Clear error

The following modes are used by COM247 automatically in the test mode and can be called by the PC via FBI 64:

- Modes BA 71, BA 73, BA 74: (for monitoring modes, see Section 4.4 "Description of the Individual Monitoring Commands")

The following modes can be called indirectly by COM247 by means of function keys and by the PC via FB1 65:

- BA 20- Enter machine data
- BA 21- Delete machine data
- BA 22- Enter machining program
- BA 23- Delete machining program
- BA 24- Enter SYSID (module identifier)
- BA 64- Read machine data directory
- BA 65- Read machining program directory
- BA 66- Read actual values (monitoring mode)
- BA 67- Read machine data
- BA 68- Machine data overview
- BA 69- Read machining program
- BA 70- Read SYSID (module identifier)

4.2 Description of the Individual Operating Modes

In this description of the modes, it is assumed that you are familiar with the terms "machine data", "machining program" and "axis attributes". You can read a detailed description of these terms in Part 2 "Fundamentals of Positioning" in the Sections:

- 2,5 Machine Data and their Structure,
- 2.6 Machining Programs and their Structure and
- 2.7 Axis Attributes.

Note

In the following graphics, the representation has been simplified, so that exponential functions are represented as ramps,

4.2.1 JOG Speeds 1 and *2* (Modes 1,2)

In these two operating modes, you can move an axis at a constant speed. The basic speeds themselves are contained in the machine data. You can traverse at JOG speed 1 or 2 by entering a "0" in the speed parameter.

After selecting one of the two operating modes, you can start an axis moving in the required direction by setting the commands “forward” or “reverse”. By pressing the stop key on the PG you can stop the axis again. From the programmable controller’s side, FB164 provides a special feature. On the signal edge of the “forward” or “reverse” command from O to 1, the axis is moved in the selected direction and is stopped again when the signal changes from 1 to O. The axis also stops if a stop command is entered (=> Section 6.2.9.2 “Special Features of the Parameters VORWandRUCK").

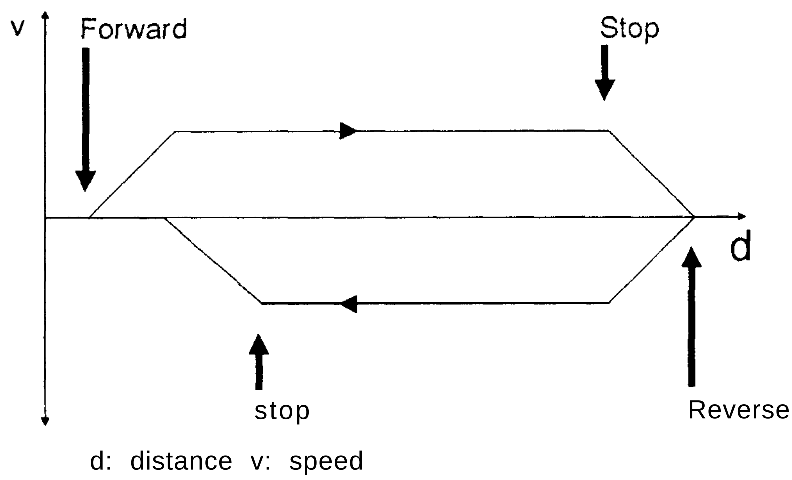


Fig. 4/4 Traversing in the jog mode

You can also switch from one jog speed to the other while the axis is moving. The axis then stops and continues its movement in the new jog mode.

In the jog mode (BA 1 and 6A 2) and in the incremental mode (BA 6 and 6A 7), you can traverse at speeds different from those in the machine data, by entering a value between 1 and 65000 in the speed parameter, Values outside this range are restricted to the limit values. The traversing movement is then executed at the limit speed. You **cannot change the speed while the axis is** moving.

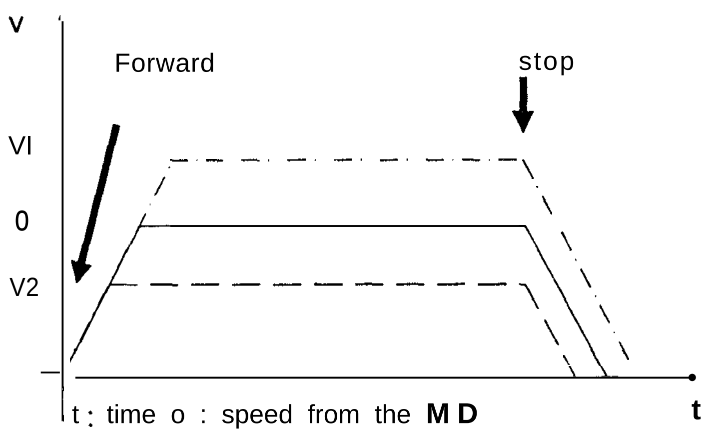


Fig. 4/5 Traversing in the jog mode with variable speeds

Note

Traversing speeds specified in the speed parameter must not exceed the maximum speed achieved at maximum frequency. The limits are 1-65000 mm/min (or 1-650000,1 in/min or 1-65000 deg/min). If this speed range would otherwise be exceeded, the speed is changed to the lower or upper limit and the error message “speed range exceeded” is output.

4.2.2 **Axis Off (Mode** 4)

After the IP247 starts up, this is the default mode.

In this mode, COM247 can only enter a start command. FB 164 can enter any commands. These commands are converted to a stop command for the current mode by the IP247. This means that any positioning job can be aborted by a command in this mode. Aborting a mode with “axis off” does not change the IP247 to the “axis off” mode. The COM247 test display and FB 164 still contain the aborted mode along with the stop command.

4.2.3 **Reference Point (Mode** 5)

Mode 5 is used to calibrate the axis. This means the following:

- Reference point approach: the reference point is located by a calibration run. The reference point precontact (e.g. BERO) and the zero reading of the excitation pattern counter are identified (synchronization = yes).

- Set reference point: the current position of the axis (at rest) is assigned the coordinate of the reference point stored in the machine data. The excitation pattern counter is **not** reset.

In each case, an error-free machine data record is required on the module. The coordinate of the reference point is stored in the machine data. The direction of approach to the reference point and the speeds for approaching the reference point are also contained in the machine data record.

Note
On correct completion of mode 5, the checkback signal “reference point set” is sent. (=> Section 2.7 “AxisAttributes”).

If the reference point is not set, the software limit switches stipulated in the machine data are not evaluated and the following operating modes are blocked:

“incremental absolute” (mode 6),
“automatic” (mode 8),
“automatic single statement” (mode 9) and
“teach-in on” (mode 10).

Zero offsets and tool length offsets which were active before the calibration of the axis (=> Section 4.3.5 or Section 4.3.6 “Absolute/Relative Zero Offset” and Section 4.3.8 “Tool Length Offset”) are retained and are included in the calculation of the reference coordinate.

Example

A zero offset of 100 mm in the reverse direction was executed.

The mode “set reference point” was executed. The coordinate of the reference point in the machine data is O mm.

The actual value following “set reference point” is indicated as 100 mm.

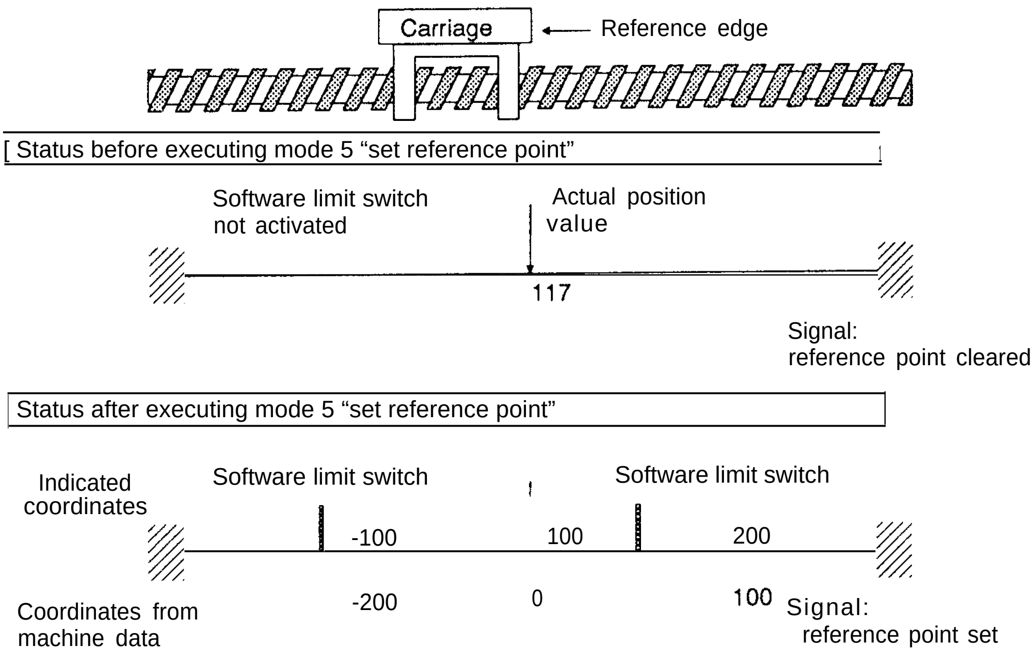


Fig. 4/6 Mode 5 with a zero offset

The reference point is lost when the positioning module is switched on and must be calculated again,

The calibration of the axis is triggered by specifying operating mode 5 and the command “start”. You must also decide whether the reference point is to be established using a reference point approach or by setting the reference point.

By starting mode 5, an existing reference point is cleared or overwritten

4.2.4 Reference Point Approach

Hardware requirements:

- A reference signal generated by an NO contact (usually BERO) which has its falling edge in the “reference point direction”.
- Possibly hardware limit switches, which restrict the traversing range and trigger the reversal of direction during the reference point approach.

- For “reference point synchronized” the excitation pattern counter on the module must be synchronized with the counter in the power unit.

Synchronization

- “Reference point synchronized” has been selected with “yes” in the machine data.

- The power unit is capable of being monitored. When the power unit is switched off, afloating contact is closed.
  If the IP247 recognizes that this contact has closed (See digital inputs/outputs), it sets its excitation pattern counter to “O”, The counter in the power unit is set to “O” when it is switched on.

- The power unit can be reset.
  Before starting the “reference point approach” mode, the I P247 outputs a reset signal for 100 ms (see digital inputs/outputs). This signal resets the excitation pattern counter of the power unit. The counter on the IP247 is also set to “O”.

- The power unit cannot be reset and cannot be monitored.
  In this case, the IP247 and the power unit must always be switched on and off together.

Sequence of the reference point approach
A reference point approach goes through the following steps: (see Fig. 4/7 “Reference point approach” with reversal at the limit switch.)

BA 5 (reference point); run; start

1) Select “reference point” (mode 5) with the parameter “run”.

2) Send the start command.

3) The drive traverses in the opposite direction from the reference direction at the reference speed.

4) The direction is reversed at the hardware limit switch, the axis traverses at reference speed until after the precontact.

5) After leaving the precontact, the axis brakes and traverses in a direction opposite reference direction to the precontact. This movement is at the speed corresponding to the start-stop frequency.

6) Once the precontact is recognized, the axis stops and then leaves the precontact in single steps in the reference direction.

7) Depending on whether or not you have selected synchronization, the reference point approach is completed at different positions:

   Synchronization: no
   Once the module recognizes that the precontact has been left, the reference point approach is completed, the coordinate of the reference point is entered as the actual value and the reference point is marked as existing (axis attribute).

Synchronization yes:
In this case, the reference point is only located after the axis has left the precontact and the excitation pattern counter has reached zero.

Note

The precontact is monitored. If the axis has not left the contact following 2500 single steps, the reference point approach is aborted and the error message "FBB (59) reference cam switch defective".

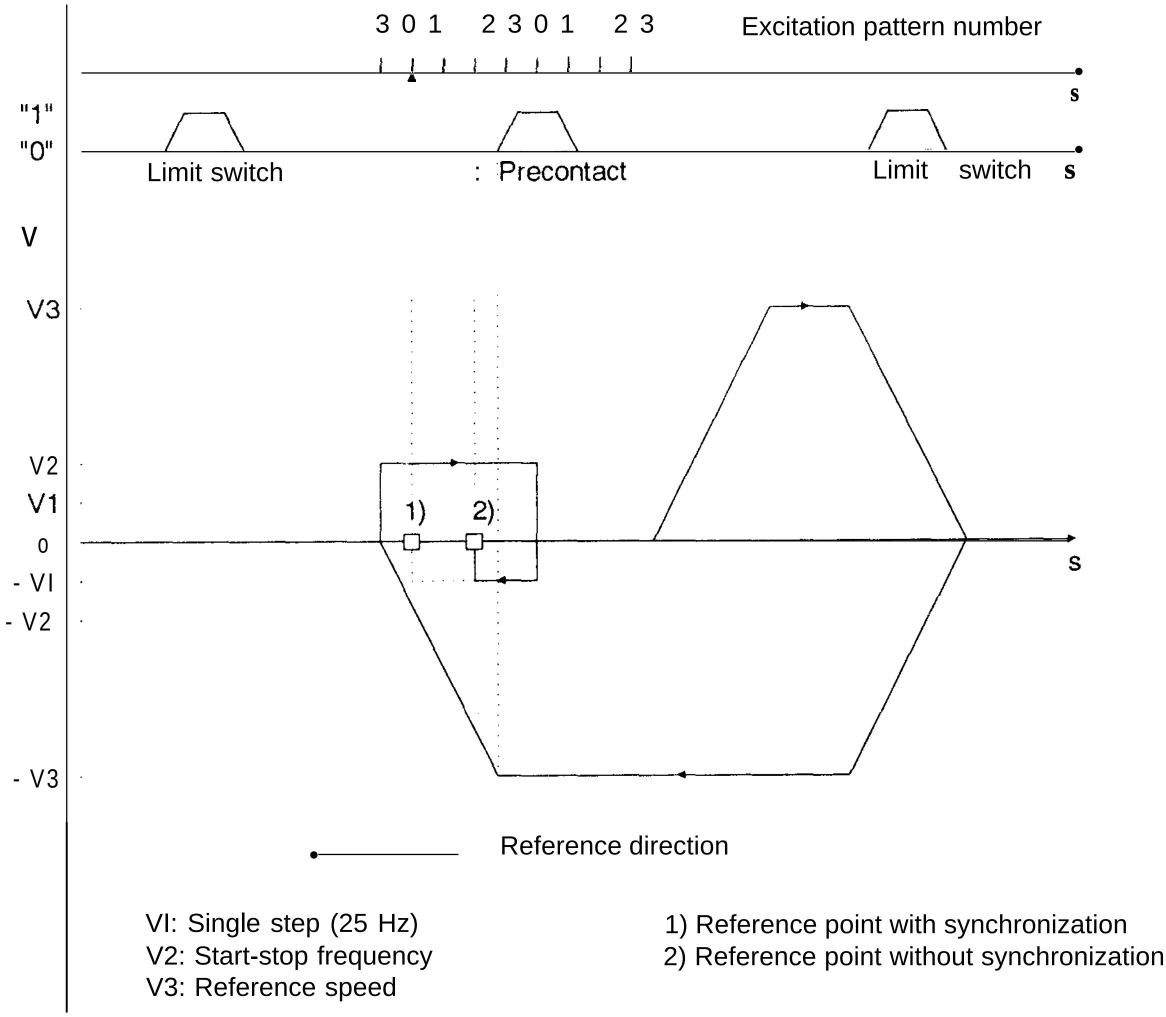


Fig. 4/7 Reference point approach with reversal at the limit switch

Special cases with reference point approach

Depending on the position of the drive before the reference point approach is executed, there are three special situations which affect the sequence of movement. These can be seen in the following diagrams. In the opposite approach direction, these movements are reversed.

- **Special case 1**

  If the IP247 detects the precontact before reaching the hardware limit switch, the direction is reversed at the end of the precontact.

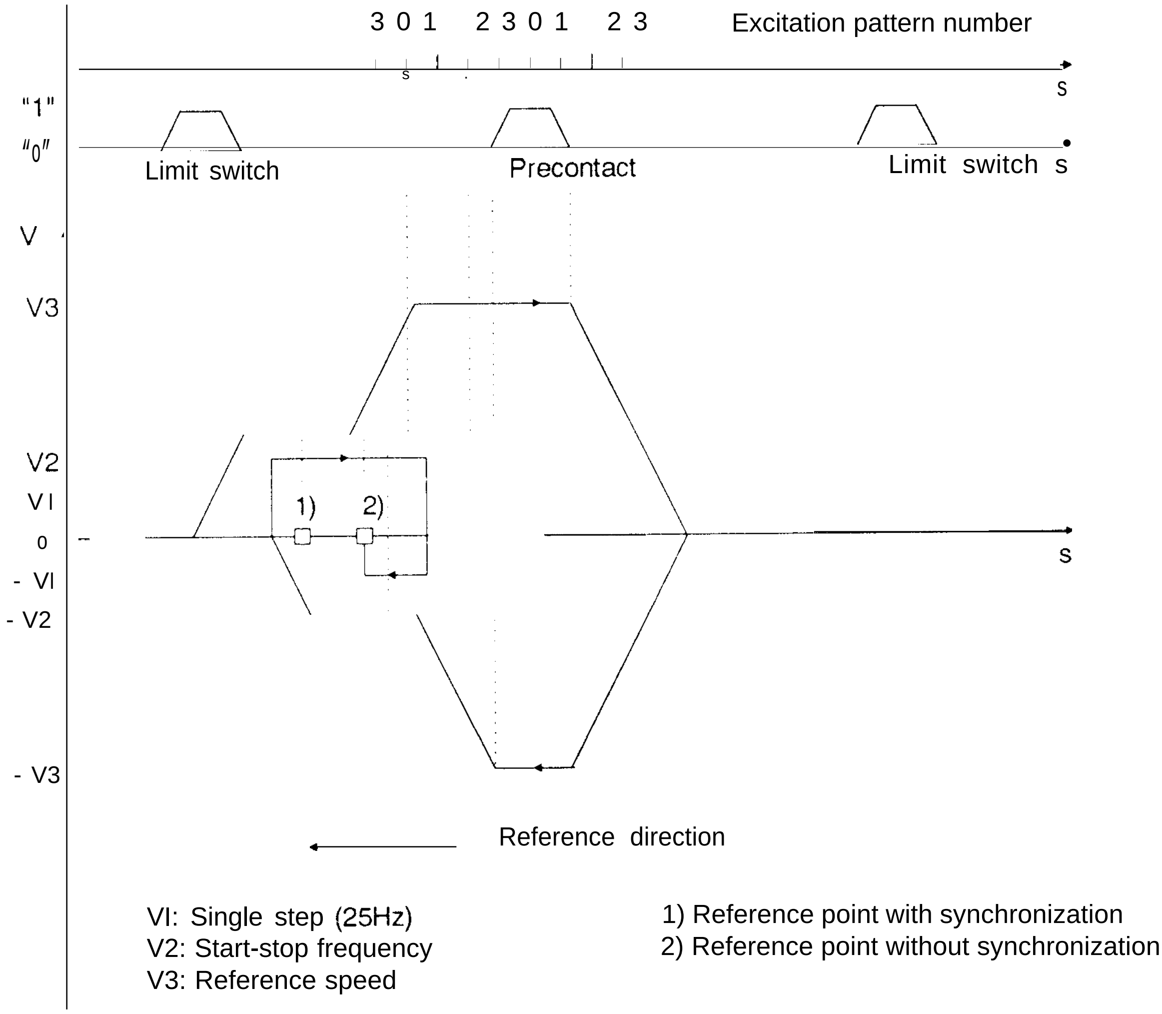


Fig. 4/8 Reference point approach with reversal at the BERO

- Special **case** 2

  If the appropriate limit switch is activated when the approach is started, the drive starts immediately in the reference point direction.

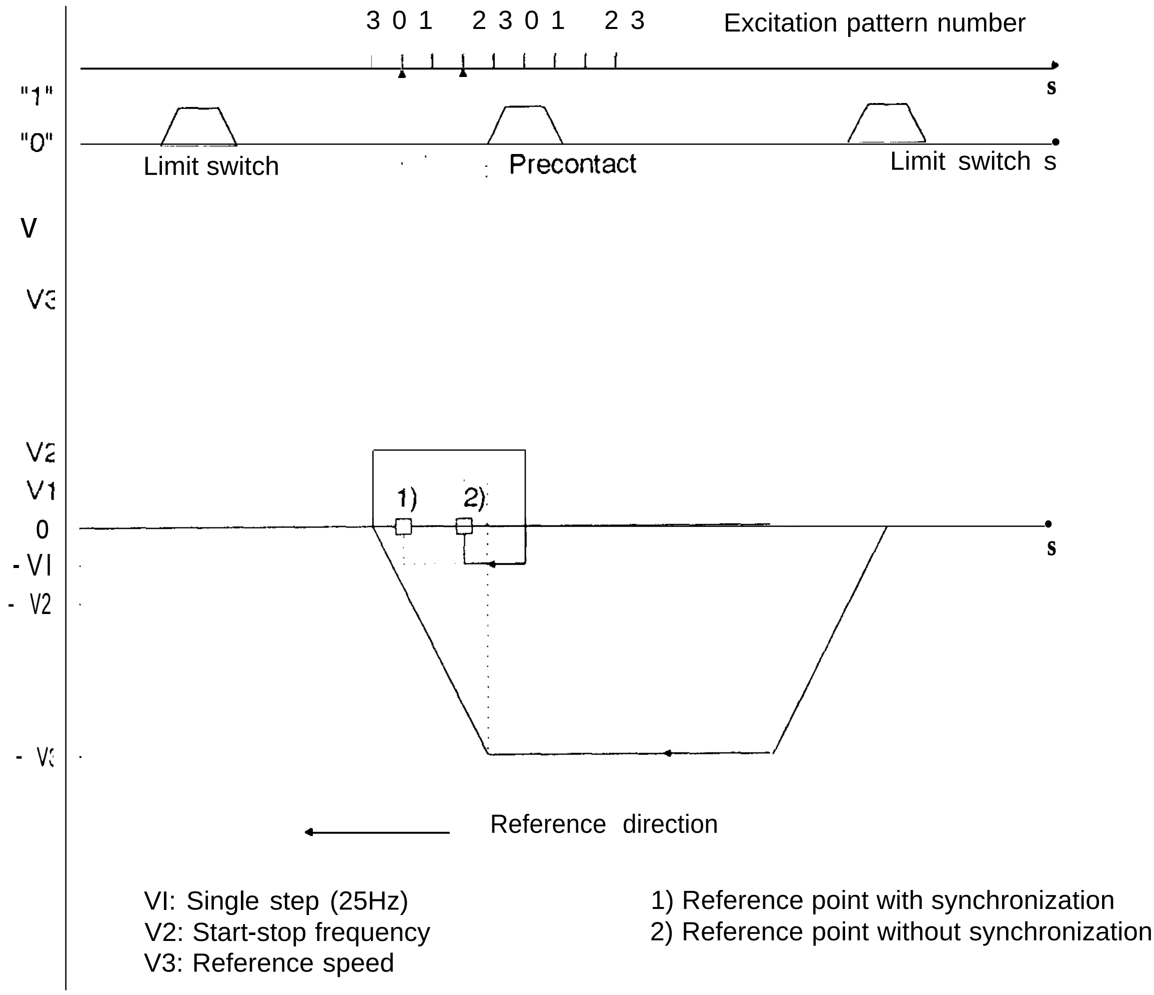
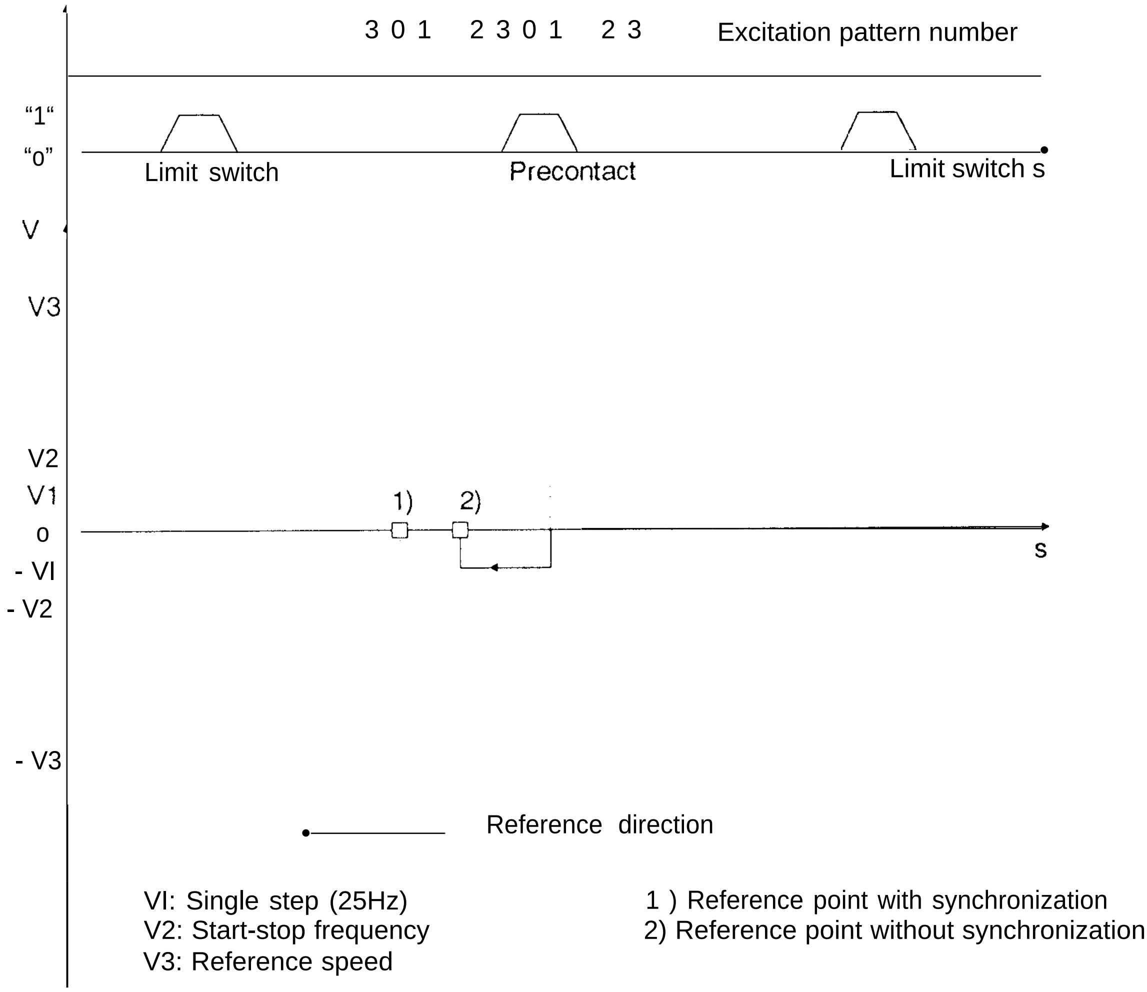


Fig. 4/9 Reference point approach with start at the reversal limit switch

- Special **case** 3

  If the precontact is activated when the reference point approach starts, the drive moves immediately in the reference point direction in single steps.



Fig, 4/1 O Reference point approach with start at the BERO

If one of the two limit switches is "out of bounds", i.e. must not be reached, you must specify the reference direction in the machine data so that the reversal of direction only occurs at the other limit switch. If both limit switches are prohibited, the axis must be positioned in front of or on the precontact (BERO) before the reference point approach is started, so that special case 1 or 3 comes into effect.

If the axis is already on the precontact at the beginning of the reference point approach, you can be sure that no limit switch will be tripped. There is also no reversal of direction in the reference point approach.

If the reference point approach is abandoned, there is no reference point even if there had been one previously. The mode must be restarted and completed.

4.2.5 Set Reference Point

With the “set reference point” function, the axis is calibrated without movement. No hardware limit switches and no precontact are required. The point at which the axis is located (actual position) at the start of the “set reference point” function is assigned the reference point coordinate programmed in the machine data record. Tool length and zero offsets are taken into account.

A reference point can be set at any axis position, even outside the hardware limit switches. You must, therefore, make sure that your axis is in a permissible position within the hardware limit switches before executing the “set reference point” function. Remember that the programmed software limit switches may, under certain circumstances, be outside the hardware limit switches mounted on the axis and therefore have no effect.

If a backlash compensation value is assigned other than O, the reference point must only be set when there is no play in the drive. The first traversing movement (traversing distance greater than/equal to the assigned backlash) must be in the direction *in* which there is no play, since the backlash is not yet taken into account with this traversing movement (=> Section 2,5.3.9 “BacklashCompensation”).

4.2.6 Incremental Approach Absolute (Mode 6)

in this mode, a target specified in absolute coordinates is approached. If you select “O” in the speed parameter, the speed of the approach is the speed specified in the machine data "incremental speed". You can vary this speed by specifying a value from 1 to 65000. The resulting speed must, however, not exceed the maximum speed, (See “jog” modes 1, 2,) The target position must be within the software limit switches or range limits, The mode can **only** be executed when there is a reference point.

With a **linear axis** the movement is triggered by the start command.

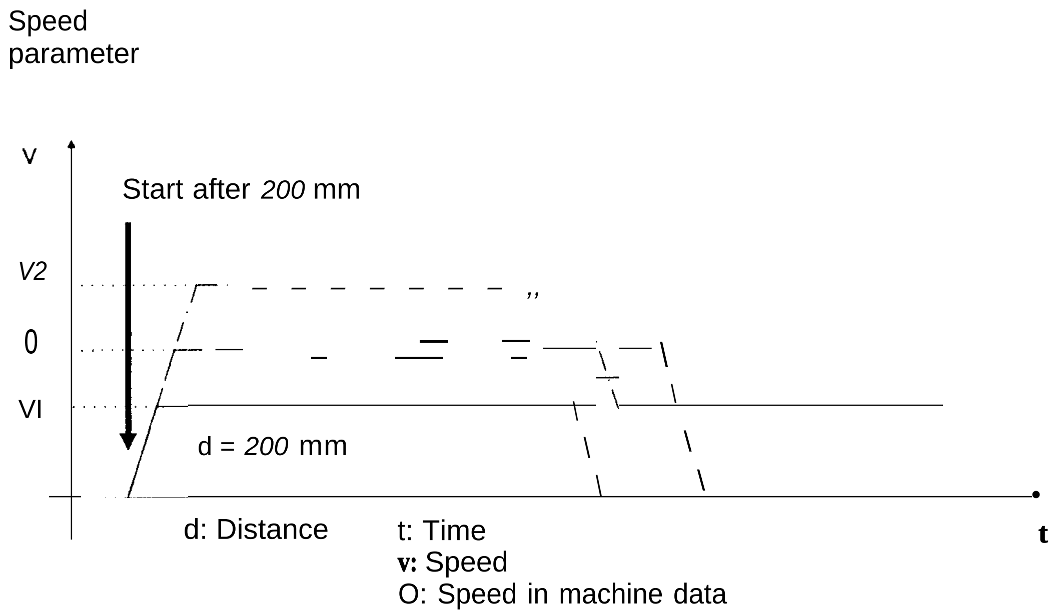
When operating a **rotary axis** the terms below have the following meaning:

- “Start” = approach the target by the shortest route.
  If the distance is the same in both directions, the direction forwards (clockwise) has priority, Reversal backlash is in this case not taken into consideration. If the axis is already on the target position, no movement is executed (=> Section 2,6.6.5 “Direction of Approach to the Target Point with a Rotary Axis”).

- “Forward” = approaching the target in a forwards direction (clockwise direction), If the axis is already on the target position, the total traversing range is covered once.

- “Reverse” = approaching the target in a reverse direction (anti-clockwise direction). If the axis is already on the target position, the total traversing range is covered once.

Changing the target while the axis is moving is not possible.



Fig, 4/1 1 Traversing in the incremental absolute mode

4.2.7 **Incremental Approach Relative (Mode 7)**

In this mode, a preset distance forwards or backwards is travel led from the current actual position. The same conditions apply to the speed as for mode 6 “incremental approach absolute”.

This mode can also be executed when the reference point is deleted. The travel direction is determined by the operating instruction “forward” or “reverse”.

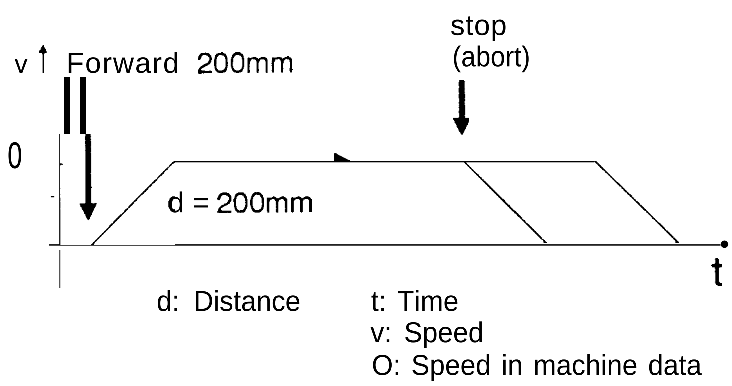


Fig. 4/1 2 Traversing in the “incremental approach relative” mode

With a linear axis, the distance to be travelled must be such that the resulting target position with a reference point set remains within the traversing range between the two software limit switches (also taking into account zero point offsets). If this is not the case, the job is aborted and the error “traversing range exceeded” is displayed. With a rotary axis, the distance travel led is limited to +/-200 m (+/-20 000 inches, +/-200 000 degrees). If this limit is not adhered to, the job is also aborted and the error message “illegal dist. spec. ” is generated.

4.3 Executing Machining Programs

4.3.1 Automatic (Mode 8)

A series of traversing movements, dwell times and loops can be stored on the module as a machining program. The structure and effects of machining programs or of functions in machining programs is discussed in Section 2.6 “Machining Programs and their Structure”. To execute a machining program, you must specify the parameter “program number” in the operating instruction. The machining program is executed with the “start” command. You can interrupt the program at any time with the “stop” command. The distance to go then remains unchanged until the next traversing movement, You can start again from the first statement, (start command) or from the point at which the program was interrupted (enter command). The program is terminated by a further stop command or by starting a different mode. For the automatic mode, the reference point must be set. While a machining program is being executed you can not change or delete this program.

Machining programs stored on the module are not assigned to a particular axis. They can be used by both axes simultaneously, If the machine data or your plant contradict the requirements of the machining program, the IP247 recognizes the error while executing the machining program. The machining program is then terminated for the corresponding axis and an error message is displayed.

On the module, a machining program interpreter evaluates the individual statements of the machining program. This interpreter is normally several statements ahead of the statement currently being executed. This means that an error such as “flying change could not be executed” may be signalled before the illegal statement in the machining program has been executed.

Zero offsets can be programmed in machining programs, which are then automatically cancelled again after the program is completed. If the program is aborted before it is completed, these offsets are not automatically cleared, They must be eliminated with BA14 “clear zero offset” (=> Section 4.3,7 “Clear Zero Offset”).

Tool length offsets activated in the automatic mode are retained after the machining program is completed or aborted.

If a machining program is executed on a rotary axis, the approach to the target by the shortest route is the default. Since the IP247 calculates the shortest route itself and therefore determines the direction, you must make sure when programming a flying change that the flying change is permitted and does not lead to an error.

Note

- With a programmed halt (MOO) and a flying change (GI O) in one statement, the programmed halt has priority.
- In mode 8 (“automatic”) an enter command is required for each “programmed halt”.
- The point at which the program is halted is always immediately before the next traversing movement or dwell time.

In the following example, a number of unnecessary MOO functions have been used. Before approaching target 200, the enter command must be given three times although the interrupt point is directly before the movement.

```
N10 X1OO F1000 MOO
N15 ! MOO
N20 G56 ! MOO
N25 G43 ! X200 F1OOO M02        (! = break point)
```

4.3.2 Automatic Single Statement (Mode 9)

This mode runs, in principle, in exactly the same way as automatic. However, you must supply the “enter” command before a traversing movement or dwell time is executed. Only one traversing movement or one dwell time in the machining program is executed, any further movement or dwell time must be triggered separately. The break point is always immediately before the next traversing movement or dwell time.

The command sequence with “automatic signal statement” is as follows:

BA 9 (automatic single statement); program number; start.

- The axis changes to the “automatic single statement” mode.
- The axis searches for the machining program with the specified program number.
- Within the machining program the statements are executed until the first dwell time or first traversing movement.
- The axis then waits for the enter command.

BA 9 (automatic single statement); program number; enter

- When the enter command is received, all functions are executed until the next traversing movement or dwell time.

etc.

BA 9 (automatic single statement); program number; stop

With this command, the “automatic single statement” mode is interrupted. This can occur both between the execution of two statements or during a traversing movement or dwell time.

Special features

- Statements connected with a flying change are treated as one statement

  N10 G10 X1000 F100 M10
  N20 X2000 F500 => treated as one statement

  Exception:

  If “flying change” and “programmed halt” are used in one statement, “programmed halt” has priority.

  N10 G10 X1000 F100 MOO
  N20 ! X2000 F500 => separated owing to MOO

- Offsets or switchovers are executed following the program start, following the previous traversing movement or dwell time and following the programmed halt.

  N10 X100 F1000 M10
  N11 G56! X200 F1000 MI 1 => The execution of the program is only interrupted after G56 (! = break point)

- A “programmed halt” is suppressed in conjunction with a traversing movement and dwell times to avoid two enter commands being required.

  N10 X100 F1000 MOO
  N11 G56 ! X200 F1000 => Despite “programmed halt” and single statement execution, only one enter command is required (! = break point)

  N10 G74 M10
  N15 G90 ! X100 F1000 MOO
  N20 G56 ! X200 F1000 M20 => Only one enter command is required at both break points

  In the following cases, the "enter" command is required twice:

  Apart from the statement number, only MOO was programmed (N10 MOO)

  In the statement with MOO, only an offset or switchover was programmed.

4.3.3 Interrupting and Continuing Machining Programs in BA 8 and BA *9*

You can interrupt and then continue a machining program processed by the IP247 positioning module. In the automatic modes (mode 8 or mode 9), you interrupt the machining program as follows:

- a stop command for any mode,
- external stop or
- an operator error during the automatic mode.

> **Note**
>
> Data transfer jobs, e.g. inputting machine data, sent to the module while an automatic mode is active, are not handled as an operator error and do not cause the machining program to be interrupted. Since the axis is in operation, data manipulation is not possible, This is rejected with the error message “axis active => entry not possible”,
>
> Warnings or messages occurring during the processing of a machining program (e.g. “axis active => entry not possible”), are only displayed for your information (cf. Section 7.2.2.2 “Module Errors and Possible Causes”). These messages do not influence the current operation.

If the program is interrupted, the error message “machining program waiting to continue” (PC: 65, COM247: FC1) is output. The axis status changes from “running” to “finished” (checkback signals), In this status, only the enter command in conjunction with the interrupted mode can continue the machining program.

Any other input, axis errors and the external stop change this status and delete or overwrite the error message, If the new input is permitted and is consistent, it will be executed once the interrupt status of the machining program is exited. The interrupted machining program can, however, no longer be continued. Zero offsets and tool length offsets already executed in the machining program remain effective and must, if necessary, be cleared with mode 14 “clear zero offset” or with mode 16 “tool length offset off”. If a machining program is interrupted and then started again, the error message “machining program waiting to continue” is cleared and the machining program is started from the beginning.

Only dwell times (G04) and traversing jobs (X function, G74) can be interrupted. An interruption is, however, possible between two jobs (dwell times or traversing jobs), e.g. if the program is waiting to start the next statement during a programmed halt (MOO) or in mode 9 “automatic single statement”.

Offsets (e.g. G57 or G43) and switchovers (e.g. G91) cannot be interrupted.

> **Note**
>
> **Closed loops without traversing jobs and without dwell times are illegal.**

The following pulse diagrams represent traversing movements as speed overtime, The distance travelled at any point in time corresponds to the area below the curve. To simplify matters, acceleration and deceleration phases are assumed to be linear.

Interruption during a dwell time

If a machining program is interrupted during a dwell time, the system assumes that the dwell time has elapsed. The dwell time is aborted, i.e. after the enter command to continue the machining program, the next traversing job or the next dwell time is processed.

When a dwell time is interrupted, all offsets (e.g. G57 or G43) and switchovers (e.g. G91) programmed before the next dwell time or next traversing job are executed. After this:

- the error message “machining program waiting to continue” is set
- “position reached” is signalled
- the axis status changes to “finished”.

When a machining program is interrupted during a dwell time, the break point is therefore always directly before the next dwell time or before the next traversing job.

N1O G04 F1000 Ml O = interrupt during this dwell time
N20 G56 ! X200 F500 M20 (! = break point in the program)

Interruption during a single traversing movement

A machining program can be interrupted at any phase of a single traversing movement. If the program is interrupted by a stop command or by an operator error, the axis is braked.

A traversing movement can be interrupted as follows:

1, while the axis is accelerating or traveling at a constant speed or

2. in the deceleration phase of the traversing movement.

Situation 1

If the machining program is interrupted while the axis is accelerating or traveling at a constant speed, the axis is braked. Since the target of the job is not reached, the following occurs:

- “Position reached” is not set.
- The axis status changes to “finished”.
- The error message “machining program waiting to continue” is output.

The distance to go is displayed. This interrupted job can be continued.

Situation 2

If the machining program is interrupted by a stop command or an operator error during the deceleration phase of a traversing movement, the target of the job is reached. There is no distance to go. The following then occurs:

- “Position reached” is set.
- The axis status changes from “running” to “finished”.
- The error message “machining program waiting to continue” is output.

In this case, offsets and switchovers are handled in the same way as described for dwell times.

If the program is interrupted by a stop command or by an operator error during the deceleration phase, the break point is always directly before the dwell time or before the next traversing job.

N10Xl00F1000 Ml O = machining program interrupted during the deceleration phase
N20 G57 ! X200 F500 M20 (! = break point in program)

Interruption of a machining program during traversing jobs linked by “flying changes”

If a machining program is interrupted during a traversing job which is followed by a further traversing job with a flying change (Gl O), the axis is braked.

- “Position reached” is not set, even if the “intermediate target” has been reached exactly.
- The axis status changes from “running” to “finished”,
- The error message “machining program waiting to continue” is output.

The following different situations can therefore arise:

1. The axis stops before the intermediate target and the distance to go is sufficient to achieve the programmed speed when the program is continued (see Fig. 4/13).

2. The axis stops before the intermediate target. The distance to go is not sufficient to achieve the programmed speed (see Fig. 4/1 4).

3. The axis stops after the intermediate target (see Fig. 4/15).

Situation 1

The distance to go following the interruption is sufficient to achieve the programmed speed when the program resumes. The flying change is executed normally.

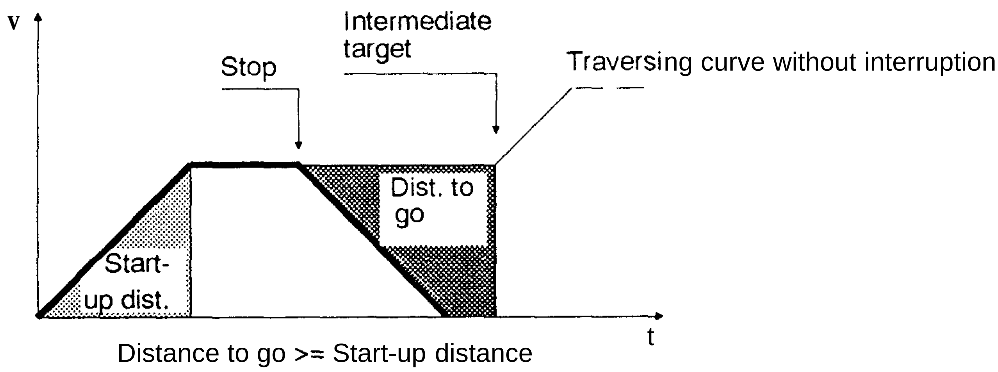


Fig. 4/1 3 Sufficient distance to go

Situation 2

Following the interruption, the distance to go is not sufficient to achieve the programmed speed. The distance to go of the interrupted job is added to the next traversing job.

Note

When the program continues, the M-function of the next job is valid.

The distance to go displayed after the enter command is the difference between the new target and the current actual position.

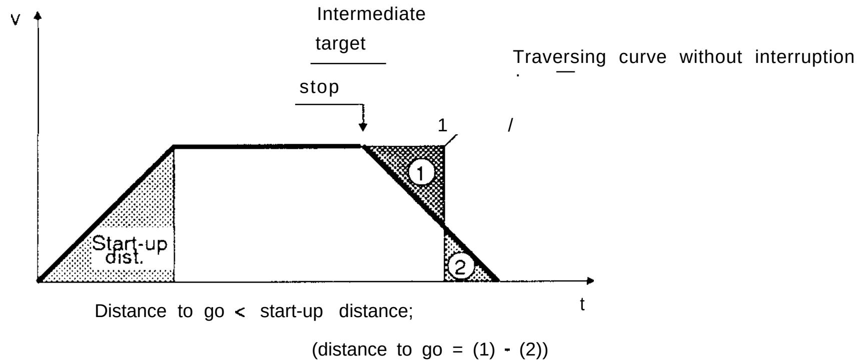


Fig. *4/1 4* Distance to go positive and less than the start-up distance

Situation3

When the interruption occurs, the braking distance is already greater than the current distance to go to the intermediate target. This means that the intermediate target of the interrupted job is overrun, This therefore leaves a negative distance to go.

Note

Although the axis overruns the intermediate target when decelerating, the M-function of the interrupted job is still output, The M-function of the next job is only output when the program is continued.

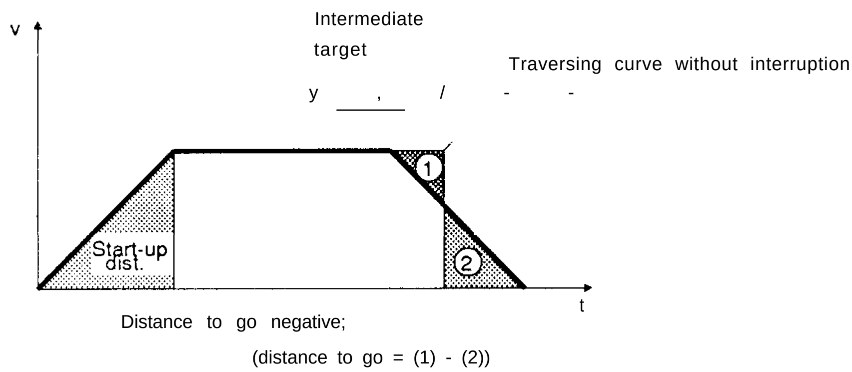


Fig. 4/1 5 Negative distance to go

Note

In situations *2* and 3, if the start-up distance is greater than the distance to go to the next target, or if the distance to go is negative, jobs are combined until:

1. either the distance is sufficient to achieve the programmed speed (see Fig. 4/1 6), or

2. the linking of the jobs is completed (see Fig. 4/17).

In both cases, when the program continues, the M-function of the last of the combined statements is output, If this statement does not contain an M-function, the last output M-function is valid.

Example:

```
N1 G1O X130 F1OOO Ml O
N2G1OXl80F1000  M20
N3 G1O X230 F1 000 M30
N4 G1O X280 F1 000 M40
N5G1OX330F1000  M50
N6......
```

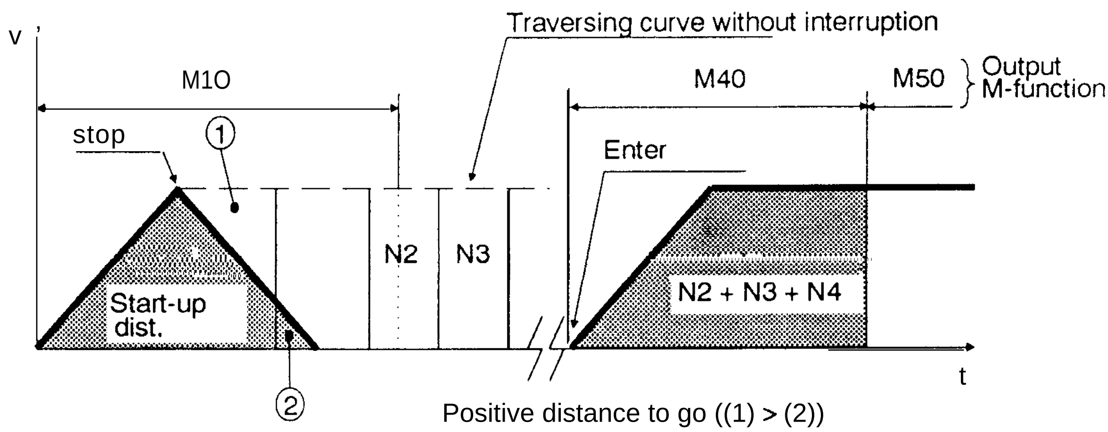
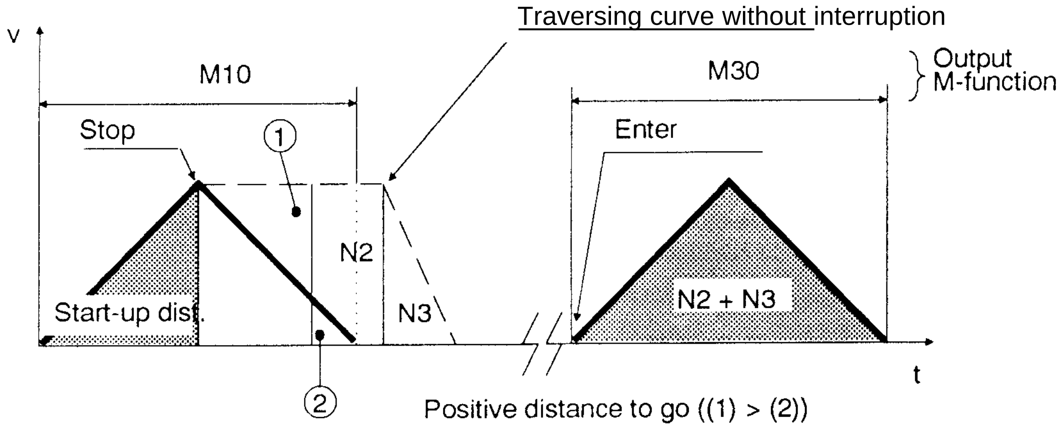


Fig. 4/1 6 Linking jobs until the distance is sufficient to achieve the programmed speed

Example: N1 G1 O X130 F1 000 Ml O
N2G1OXl80F1000 M20
N3 X230 F1000 M30
N4 M02



Fig, 4/1 7 Combining jobs until the linking is completed

4.3.4 Teach-in On/Off (Modes 10/1 1)

In the “teach-in” mode, only the JOG modes and modes “incremental approach absolute or relative” are permitted, other entries are not executed. If no position was stored in “teach-in”, the positioning module will have generated an empty machining program.

The following procedure must be followed to generate machining programs in “teach-in”:

- the mode BA1 O must be activated and the required program number specified,
- the required target points must be approached in “JOG” or in the mode “incremental approach absolute or relative”,
- with the axis at a standstill (axis status = “finished”), the current actual position must be stored in the selected program with the enter command. When you save the statement it is signalled with the message “statement saved”. It is possible to transfer the position several times. In this case, several identical machining program statements with consecutive N functions are stored in the machining program,
- when you have saved all target positions, switch off the “teach-in” mode with BA11 ("teach-in off”). The program is then completed with MO2 and entered in the program directory of the module.

A program generated in this way can be used in both automatic modes by both axes.

In the “teach-in” mode, all statements

- are stored one after the other,
- begin with NO1 and have consecutive N functions,
- are assigned the “incremental speed” and
- are stored without M functions.

When “teach-in” is switched on

- the machine data record must be valid,
- the reference point must exist,
- there must be sufficient space in the program memory of the IP247,
- a machining program number must be specified which has not yet been used on the IP247,
- no other axis of the IP247 must be in the “teach- in” mode and
- machining program input must not be active on the data channel.

> **Note**
>
> If the power supply to the positioning module is switched off during “teach-in” and if statements have already been recorded, this machining program is lost. The "teach-in” mode is no longer active. If the limit of maximum 6000 machining program characters is exceeded in “teach-in”, the mode is automatically terminated, the last stored statement is taken as the final statement.

The following command sequence is for example possible: (in the example, the incremental speed in the machine data is 2500 mm/min)

BA 10 (teach-in on), program 7, start

“Teach-in” is switched on and machining program 7 is set up.

%7

BA 1 (JOG speed 1), forward

The axis moves forward at JOG speed 1.

BA 1 (JOG speed 1), stop

The axis stops (e.g. at 1258.250 mm). After the “finished” message, the enter command can be used to enter the machining program statement. (=> Section 2.7 “Axis Attributes”).

BA 1 (JOG speed 1), enter

The axis is now stopped. With the enter command, the first statement NO1 X1258.25 F2500 of machining program 7 is generated.

%7

BA 6 (incremental approach absolute), 3000 mm, start

The axis travels to the absolute position 3000 mm and stops. The “finished” message is set.

BA 6 (incremental approach absolute), enter

N02 X3000 F2500 is entered in the second statement (N2).

6A 11 (teach-in off), start

Machining program 7 is completed and the teach-in mode switched off. The statement N3 M02 is appended to machining program 7 to indicate the end of the program.

The complete example appears as follows:

```
% 7
N1   X1258.250  F2500     : approach position 1258.250 at speed 2500 mm/min
N2   X3000.000  F2500     : approach position 3000000 at speed 2500 mm/min
N3   M02                  : final statement, program end
```

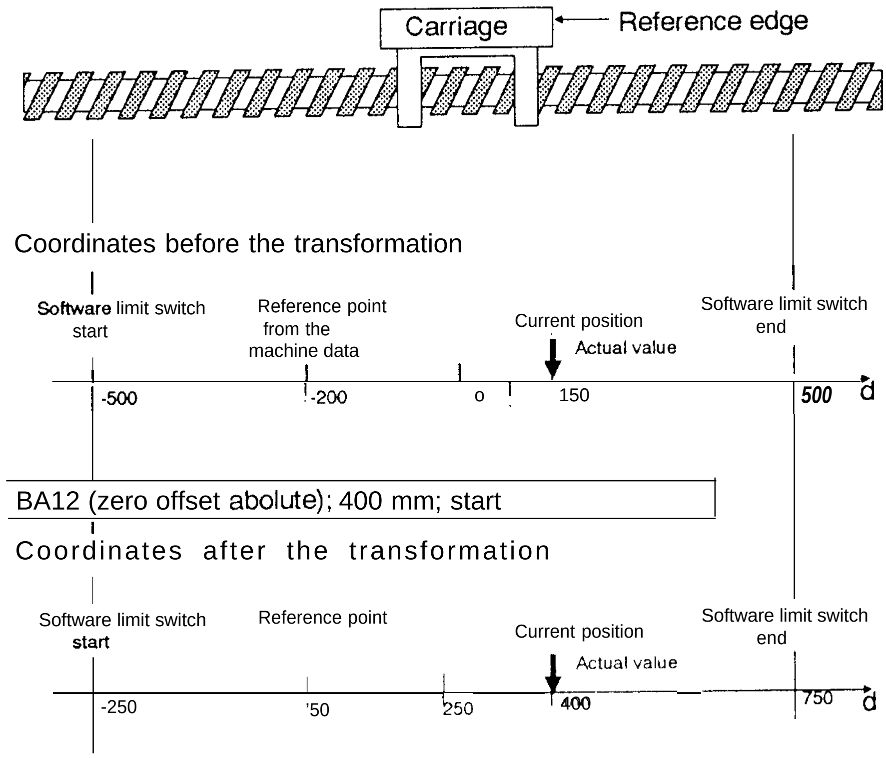
In the machining program, absolute distance specifications is the default. For a rotary axis these are approached via the shortest route when executing a machining program generated in this way, If you wish to change these presets or speeds or add further G or M functions, you can do this easily with COM247. Output the generated machining program from the module, edit the program and then store it again on the IP247.

4.3.5 Zero Offset Absolute (Mode 12)

When you start the mode “zero offset absolute”, the current actual position of the axis is assigned a new coordinate. This means that the whole coordinate system including the reference point coordinate and the software limit switches or traversing range limits are transformed. A traversing movement does not take place. The new coordinate for the current position must be transferred with the start of the mode. It is then displayed as the actual value.

The coordinate required as the transfer parameter for the mode must not exceed the maximum range (+/-100 m; +/-10000 inches; +/-100000 degrees). Greater values cause the mode to be terminated and the error message “illegal dist. spec." is displayed. The displacement of the coordinate system must be such that all the new coordinates still lie within the permitted range, This is checked by the module, and if the coordinates are outside the range, the mode is terminated and the error message “traversing range exceeded” is displayed.

Zero offsets can be stored in the machine data (zero offset 1...4) and called in the machining programs (=> Section 2.5.5.2 “Zero Point Offset”). These bring about an additional displacement of the coordinates. At the end of a machining program, the offsets executed during the program are cancelled, not however those generated by the modes “zero offset absolute” and “zero offset relative”,



Fig, ***4/1 8*** Zero offset absolute with a linear axis

The coordinate of the current position is transformed from 150 mm to 400 mm. All other position coordinates (software limit switches, reference point) are 250 mm more positive.

Effect of zero offset for a rotary axis

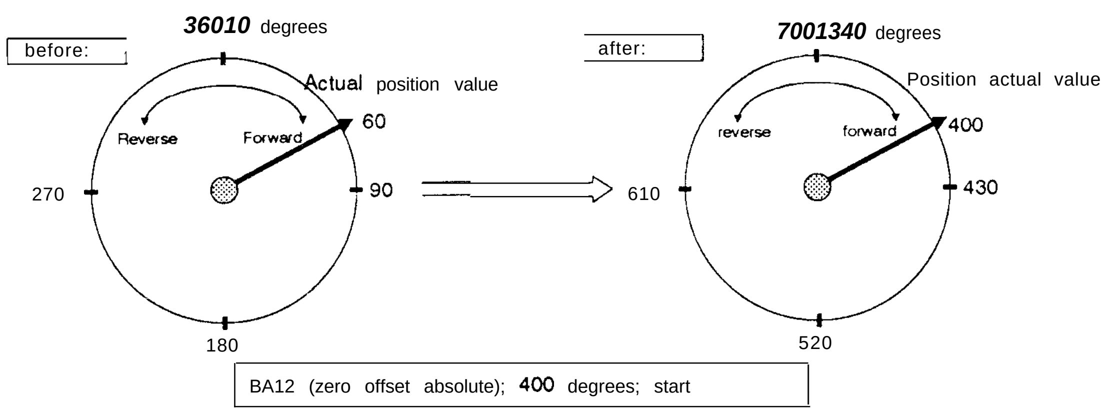


Fig. 4/1 9 Zero offset absolute with a rotary axis

The current position of *60* degrees is assigned the coordinate 400 degrees. All other position coordinates become more positive by the difference of 400 degrees -60 degrees i.e. 340 degrees, The range limits are then no longer at 0/360 degrees but at 340/700 degrees. Absolute target specifications must have values within this range following the coordinate transformation. The coordinate O degrees, for example, no longer exists after the zero offset.

You can execute any number of zero offsets one after the other. A zero offset is only possible with valid machine data but can be executed without a reference point. When the axis is calibrated (BA5) a zero offset is taken into account. (See Section 4.2.4 “Example of Reference Point Approach or Set Reference Point”.)

Note

After switching on the IP247 again, the reference point is lost, however, not the zero offset.

If new machine data are entered to the IP247 or if existing machine data are modified, zero offsets are cleared.

4.3.6 Zero Offset Relative (Mode 13)

In this mode, the coordinate system is displaced by a value specified in the input parameter. An offset “forward” means that the coordinates of the software limit switches or range limits and the reference point coordinate as well as the current actual value become more negative by the value specified. A “reverse” offset has the opposite effect. A sign entered with the parameter is also taken into account. The command zero offset relative -50 mm reverse causes a zero offset of 50 mm forwards.

The same requirements, conditions and limits apply as for mode 12.

Relative zero offsets are added to zero point offsets set with “zero offset absolute”.

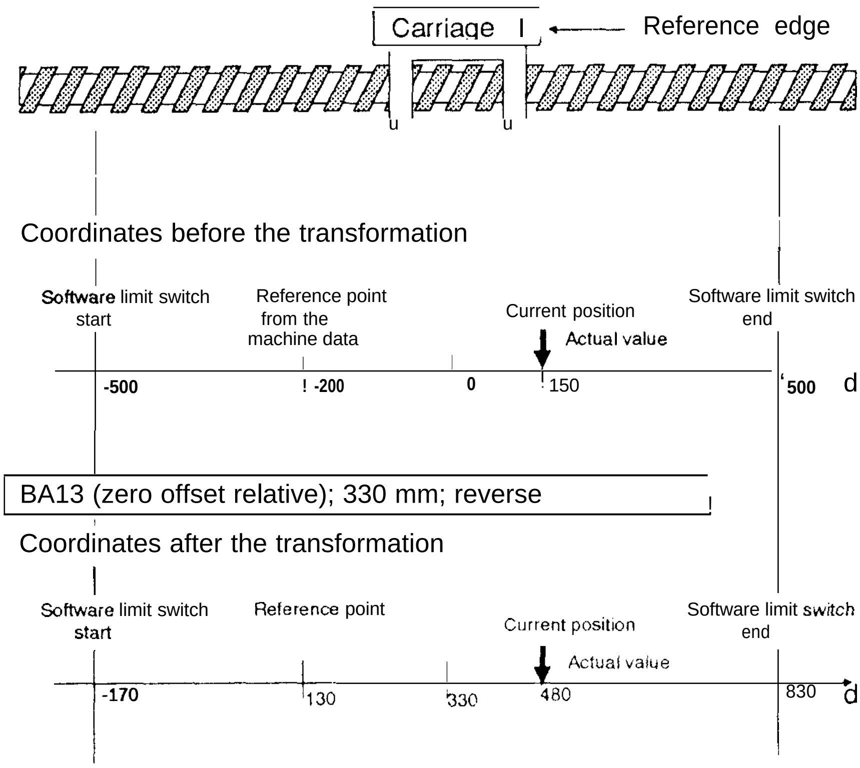


Fig. 4/20 Zero offset relative

4.3.7 **Clear Zero Offset (Mode 14)**

When this mode is started, all zero offsets, established by

- "zero offset absolute", BA12 *or*
- "zero offset relative", BA 13

or zero offsets which were not cancelled because a machining program was aborted are cleared. (=> Section **4.3.1** "Automatic").

If several zero offsets have been performed, it is not possible to clear individual offsets, only the total zero offset can be cleared. The coordinate system is then once again as it was with mode **5.**

4.3.8 **Tool Length Offset (Mode 15)**

The "tool offset" mode allows user programs and machining programs to be used when the tool length changes without having to change the program.
In the operating instruction of mode 15, you enter the value and direction of the tool length offset, The direction is stipulated with the commands "forward" or "reverse", Remember that the sign before the offset value is taken into account. The command . . .

BA 15 (tool offset); *-80* mm, reverse

. . .*corresponds* to a tool length offset of 80 mm forwards.

When any positioning movement is carried out, the **new** tool tip is brought to the specified target position. This also applies to the execution of machining programs. If mode 15 *is* called again, the offset is replaced by a new value.

The mode “tool length offset” requires valid machine data, A reference point is not necessary, A value within the limits of +/-100 m (+/- 10000 inches or +/-100000 degrees) can be entered for the tool length. Values outside these limits lead to termination of the program and to the error message “illegal dist. spec. ”.

The value displayed as the actual position following a tool offset is the position at which the tip of the tool is located. Following a tool offset, even if a software limit switch is tripped, the actual value (tool tip) must not exceed the limits of+ /-100 m (+ /-10000 inches, + /-100000 degrees). This is checked when the tool offset is executed. If this condition is not met, the tool offset is not accepted and the error message “traversing range exceeded” is set.

Example
A drilling program was written assuming that the drill is 100 mm long. The reference edge of the chuck is position 50 when retracted. Enabling conditions were generated at this position for the equipment, i.e. the chuck must always return to this retracted position. The target position of the drill tip when operating is 200.

Implementation

BA6; 50 mm ;	start	;"incremental approach absolute” to 50 mm
BA15; 100 mm;	forward	;"tool length offset”
BA6; 200 mm;	start	;"incremental approach absolute” to 200 mm
BA16;	start	;"tool length offset” off
BA6; 50 mm ;	start	;"incremental approach absolute” to 50 mm

In the parameter for mode 15, only the actual length of the drill is entered when a drill is changed, This means that the required depth is always achieved.

Before the drill is retracted, the offset is cancelled so that the chuck returns to the basic position.

In the program, you must, of course, make sure that the individual jobs are only sent when the axis status is “finished”.

Note

The detection of the software limit switch depends on the internal position setpoint. The axis begins to brake when the reference edge of the tool holder passes a software limit switch. Depending on the tool length offset, the tip of the tool may be well outside the selected traversing range.

You can also specify a tool offset in the machine data which can be switched on within machining programs by G43 or G44. (=> Section 2.6.6 “The G-Functions”). This offset is added to the offset specified with mode 15 and can be called repeatedly. This allows, for example, the estimated wear on a tool to be taken into account within the machining program by setting the tool length with mode 15 and then calling the tool length offset from the machine data in the machining program to make up for the tool wear.

The tool length offsets executed in a machining program are not cancelled at the end of the program. They are retained just as the tool offset set with mode 15, even after the IP247 is switched off, If however new machine data are entered or the existing machine data are modified, the existing tool offset is cleared.

Each new tool offset set with BA15 overwrites the previously effective tool offset. This applies to offsets set with BA15 or switched on during a machining program.

Special features for a rotary axis

If a tool offset of 30 degrees forward (clockwise) is executed at position 60 degrees for a rotary axis, the actual position is then signalled as 90 degrees (tool tip), The actual traversing range is, however, still between O degrees and 360 degrees. The coordinate system is therefore turned. On the other hand, with a zero offset the coordinate system is transformed to a different numerical range.

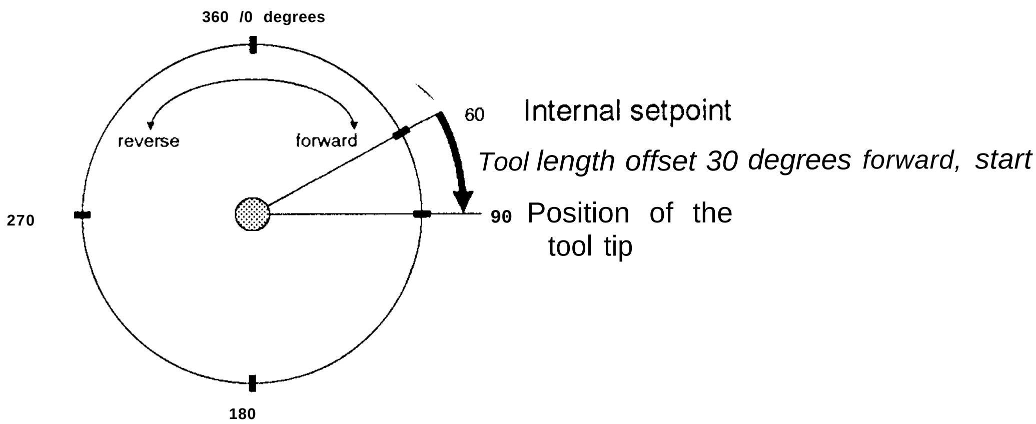


Fig. 4/21 Tool length offset with a rotary axis

The calculation of the shortest route in the incremental approach mode is always calculated from the tool tip when a tool offset is set with a rotary axis.

Restrictions on tool offsets with a rotary axis
The value of the tool offset or value of the offset resulting from mode 15 and G43/G44 in a machining program must be less than the traversing range, i.e. less than the difference between

range end - range start.

If this condition is not met, either the tool offset set with mode 15 is not accepted and the error message "illegal tool offset" is signalled or the machining program is aborted with this error message.

4.3.9 **Tool Offset Off** (Mode 16)

When this mode is started, all tool offsets are cancelled. If you cancel tool offsets set with mode 15 using "tool offset off", you also cancel the tool offsets set in a machining program (=> Section 2.6,6.6 "Tool Length Offset").

4.3.10 Clear Error **(Mode 17)**

Error messages resulting from incorrect operation or an external event (e.g. tripping a limit switch or receiving an external stop command) remain active until they are acknowledged by entering a new operating instruction on one of the interfaces. While the error message is active, any number of monitoring commands can be sent to the module and the error will be output along with the response from the module. In the axis status "finished" you can reset the error message on all axes and the data channel with the command "clear error" "start" (=> Section 7.2 "Troubleshooting").

4.3.11 Machine Data Processing (Modes 20,21,64,67 and 66)

The concept of the IP247 is that machine data records are generated initially using the communications software COM247 at the PG. These data records are then stored as required on the IP247, in the programmer memory or on diskette/hard disk. COM247 automatically uses the modes for machine data processing and ensures that the data records are correctly structured and transferred. Working with the COM247 software is described in detail in Part5“COM247 Communications Software”.

Via the PC interface, you can exchange this data between the CPU and I P247 and process it in the CPU using FBI 65. The procedure is as follows:

- generate the machine data records with COM247, transfer them to the I P247, test and optimize the machine data
- save the data on diskette or hard disk using COM247
- if necessary, save the data with FBI 65 in a data block in the CPU
- if required, store the data block from the CPU in an EPROM
- if required, modify individual data for particular applications in the CPU and transfer the data record again to the IP247.

Remember that the same machine data record which you fetched from the CPU to diskette with STEP 5 is not identical to the machine data record which you transfer to diskette using the software package COM247.

The modes for machine data processing are as follows:

- BA 20 “enter machine data”
- BA21 “delete machine data”
- BA ***64*** “read machine data directory”
- BA ***67*** “read machine data”
- BA 68 “read machine data overview”

4.3.12 Enter Machine Data (Mode 20)

Using this mode a complete machine data record is transferred via the PG interface or PC interface to the\ P247. COM247 uses mode 20 indirectly if you press the appropriate function key. If you transfer a machine data record from the CPU to the IP247, you must assign parameters to FB165 as described in Section 6.2 “Standard Function Block FBI 65”.

You can only enter machine data when

- the machine data record to be entered has the same module number as the SYSID (module identifier), Otherwise you must first run through mode 24 “enter SYSID" (=> Section 4,3.22 "EnterSYSI D”)
- the axis for which data is to be transferred is in the “finished” status,

A machine data record can only be transferred via the corresponding interface (page) to the IP247.

After the transfer, the IP247 checks the consistency of the machine data. If the machine data is correct, it is indicated as “existing” in the checkback signals.

If an error is detected in the machine data, the data record is marked by the I P247 by entering an error number,

The error number and corresponding error text are displayed in the error message line of COM247 on the PG. Via the PC interface the error number can be evaluated by reading the machine data (BA67) or by reading the machine data overview (BA68).

Note

If the positioning module is removed from the programmable controller, the machine data stored on it is lost (battery back-up via the programmable controller). A reference point is also lost.

Each time machine data are entered, zero offsets and tool length offsets are reset.

If a machine data record is deleted, the reference point remains set in the checkback signals. After entering a machine data record, the reference point is not lost if the machine data listed below have not changed from those of the old data record. The following data are relevant:

- coordinate of the reference point,
- pulses per revolution,
- distance per revolution,
- reference direction and
- synchronization.

4.3.13 Delete Machine Data (Mode 21)

This mode is used to delete a machine data record on the IP247. The axis involved must be in the “finished” status. COM247 uses this mode when you press the appropriate function key. Using FB1 65 you can also delete a machine data record via the PC interface. To do this, the mode must be triggered by FBI 65 via the axis page.

For more detailed information, refer to the description of FBI *65* in Section *6.3* “Standard Function Block FB 165”.

4.3,14 Read Machine Data Directory (Mode 64)

In this mode, you obtain information from the I P247 about

- the machine data records stored on the IP247 and
- the axis for which the records are valid.

The information for all three axes is made available simultaneously. COM247 uses this mode with the “information function”, If a destination data block is set up in the PC in which the information can be entered, you can read the machine data directory using FBI 65 via the PC interface.

For more detailed information, refer to the description of FBI 65 in Section 6.3 “Standard Function Block FBI 65”.

4.3.15 Read Machine Data (Mode *67)*

With the “read machine data” mode, a machine data record is transferred from the IP247 either to the CPU or using COM247 to the PG or to diskette/hard disk, COM247 uses this mode when you press the appropriate function key. **With FB1** 65, you can transfer a machine data record via the PC interface to the CPU. When doing this, remember that the machine data record

- can be transferred via any page if the DB numbers are different,
- must be transferred via the axis page if the DB numbers are the same,
- must be entered in an adequately long destination data block set up in the CPU.

You should use this mode to save machine data in the CPU. Only in this way can you exchange a module without using the PG.

For more detailed information about mode 67 refer to Section 6.3 “Standard Function Block FBI 65”.

4.3.16 Machine Data Overview (Mode 66)

Using this mode, you can read the following information about the machine data on the I P247 from the PC side:

- the machine data number of the data record,
- the module number for which the data record is intended,

- **the axis number** for which the data record is intended,
- **the** length of the data record **in** words and
- the machine data errors.

The overview is transferred simultaneously for all three module axes. You must first set up an adequately long destination data block in the CPU.

For more detailed information, refer to Section 6.3 “Standard Function Block FBI *65”.*

4.3.17 Executing Machining Programs (Modes 22,23,65 **and 69)**

The concept of the I P247 allows machining programs to be created easily using the communications software COM247 on the PG. These data records are then stored as required

- on the IP247,
- in the programmer memory or
- on diskette/hard disk.

COM247 automatically uses the modes for executing machining programs and makes sure that the data records are correctly structured and transferred. Working with COM247 is described in Part 5 “Communications Software COM247”.

You can also exchange and process data with FB165 between the CPU and IP247 via the PC interface. To do this, proceed as follows:

- create a machining program with COM247, transfer it to the I P247 and test it,
- save the data on diskette or hard disk using COM247,
- if necessary, save the data in a data block in the CPU using FBI 65,
- if required, store the data block from the CPU in an EPROM,
- if necessary, change the individual data in the CPU for the particular application and transfer the data record to the I P247 again.

Remember that a machining program written to diskette from the CPU using STEP **5 is** not identical to the machining program transferred from the IP247 to diskette using COM247.

The modes for executing machining programs:

- BA 22 “enter machining program”
- 6A 23 “delete machining program”

- BA 65 “read machining program directory”
- BA 69 “read machining program”

are explained below.

4.3.18 Enter Machining Program (Mode 22)

In this mode, a complete machining program is transferred to the IP247 via the PG or PC interface. COM247 uses mode 22 indirectly if you press the corresponding function key. If you transfer a machining program from the CPU to the IP247, FBI 65 must be assigned parameters as described in Section 6.3 “Standard Function Block FBI 64”.

The structure of machining programs is described in Section 2.6 “Machining Programs and their Structure”.

Since machining programs are not axis-related, they are transferred to the module via the data channel (4th page of the module).

Machining programs are stored one after the other in the memory of the IP247 in the order in which they are entered, A maximum of 255 machining programs can be stored on the module. The total number of characters is limited to 6000.

Requirements for the transfer of machining programs areas follows:

- the machining program number must not exist on the IP247. It is not possible to overwrite a program from the PC side. The old program must be deleted first with BA23 (“delete machining program”),
- there must be adequate space in the machining program memory of the IP247. (See notes in Section 4,3.19, “Delete Machining Program”).

If you wish to execute a machining program after it has been created in the CPU, proceed as follows:

- create a machining program in DIN representation with CC) M247. Fill in the input values with blanks up to the maximum number of input characters (token characters). Having done this, it is easy to change the machining program in the CPU at a later time,
- transfer the machining program to the IP247,
- read the machining program with mode 69 using FB1 65.

The program structure is now set in the CPU data block. You can modify individual parameters in the ASCII format.

- delete the machining program on the IP247 with FBI 65,
- enter the machining program to the IP247 using FBI 65 with mode 22.

Note

If the program is extended during the modification, you must update the program length in the header information, otherwise the entry of the program is aborted with an error!

If a machining program is created with COM247, the machining program can only be transferred to the module if it is syntactically correct. The syntax check is made by COM247. If, however, the machining program is transferred from the CPU to the IP247, the program is checked for syntactical errors by the firmware on the IP247. If an error occurs, the module error “machining program error” is set. The machining program error itself is written in the machining program header in DW n+3 (=> Section 6.3.8.2 “Structure of the Machining Program DB in the PC Memory”). A machining program marked as containing errors is not listed in the machining program directory. It cannot be edited using the software package COM247.

If an error is detected, you must proceed as follows:

- read the machining program with mode 69 (“read machining program”) from the IP247 to the PC memory,
- locate the machining program *error* in the program and correct it,
- delete the program on the IP247 with mode 23 (“delete machining program”),
- transfer the program to the IP247 again using mode 22 (“enter machining program”).

When a machining program is created using the software package COM247, a syntax check is carried out. Whether or not the program is consistent with the machine data can, however, only be determined when the program is executed.

4.3.19 Delete Machining Program (Mode 23)

Using this mode, a machining program is deleted from the IP247 memory. The job is either triggered using FBI 65 via the data channel *or* by the software package COM247 when you press the delete key <F5> (=> Section 5.10 “Delete”). The following requirement must be met:

- the program to be deleted is not currently being executed.

If a program is deleted from the program memory of the IP247 while the IP247 is running another program (“automatic” or “automatic single statement”) a gap may result in the memory area of the I P247. You will then receive the error message “machining program only cleared from directory”. This gap can be closed both with COM247 or from the PC as follows:

- stop all machining programs on the axes,
- read the machining program directory (BA65) from the IP247,

- save the first program on diskette, hard disk or using mode 69 (“read machining program”) in the CPU,
- delete this program on the IP247 with mode 23. The memory of the IP247 is then compressed,
- transfer the saved program to the IP247 again using mode 22 (“enter machining program”), The program now appears last in the machining program directory.

While the memory is being compressed make sure that no axis is executing a machining program (“automatic”, “automatic single statement” or “teach-in”). If one of these modes is being executed, only the memory area after the program currently being executed will be compressed and no new memory space will be made available.

4.3.20 Machining Program Information (Mode 65)

With this mode, using FBI 65, you obtain a listing of all the machining programs contained on the IP247. The machining program number and **length of the individual programs in words is output.** COM247 also uses this mode in the “information function” (=> Section 5.11 “Information”). Here, **however, the length of the machining programs is shown in bytes.**

Since machining programs are not axis related, the machining program directory can be read out to the PC interface via any interface. The only condition is that the destination data block is sufficiently long.

The programs are

- listed in the order in which they are entered and
- entered without gaps in the destination data block.

For more detailed information, refer to the description of FB165 in Section **6.3.8.5** “Structure of the Machining Program Directory”.

4.3.21 Read Machining Program (Mode 69)

With the mode “read machining program”, a complete machining program is transferred from the I P247 to the **CPU.** COM247 uses this mode indirectly if you press the appropriate function key. Via the PC interface, the transfer is made with FBI 65. The transfer can be made via any page. The only condition is that the destination data block in the **CPU is of adequate length.**

When it outputs a machining program, the IP247 adds information in the header. This includes the following:

- the length of the program in words,
- the data block number of the machining program on the IP247 (identical to the machining program number),

- if applicable, the machining program error number and
- if applicable, the number of the statement in which the error was recognized.

Following this, the machining program is output in ASCII characters.

4.3.22 Enter **SYSID (Mode 24)**

With mode 24 (SYSID input), a module identifier (SYSID) is entered on the IP247. **This is necessary when a module has been exchanged before you transfer machine data.**

When operating the IP247 using COM247, the SYSID input does not appear directly. It is triggered when you press <F1> (begin) in the presets display (=> Section 5.4 “Start COM247”). The data entered and displayed in the presets display is then written to the module.

From the PC side you can execute this mode with FBI 65 via the data channel.

The module identifier consists of the following elements:

- Module type:

The type consists of the characters “IP247”. The module type cannot be changed.

- Version:

This indicates the firmware version. For example, A02.1 for firmware version 2,1. It consists of five characters and cannot be changed.

- Module number:

This is a number between O and 99 which you assign to differentiate between positioning modules. The same number must also be entered in the machine data of the three axes.

Note

The module number can no longer be changed once a correct machine data record (MD) exists on the module. If you attempt to change this number, the error “correct MD - module number cannot be changed” is output.

- Slot number:

This number can be selected between O and 255. It is only used for documentation

- Page number:

The page number can be selected between O and 252 and is simply used for documentation. The page address set on the module can be entered here (=> Section 3.3.2 "Setting the Module Address"), The number can then be read at the programmer without having to remove the positioning module from the PC. No check is made as to whether the page address on the module is the same,

4.3.23 Read SYSID (Mode 70)

Using mode 70 (read SYSID), you can read the module identifier of the IP247 using FBI 65. This is possible via all pages. The parameters are explained in the previous section,

For more detailed information about calling mode 70 refer to the description of FBI 65 in Section 6.3.8.3 "Structure of the IP247 SYSID in the PC Memory".

COM247 starts "read SYSID" when you press <F2> (ONLINE-OFFLINE) in the presets display, The data read are then displayed in the appropriate fields (=> Section 5.4 "Start COM247").

4.4 Description of the Individual Monitoring Modes

Using the monitoring modes, you can call current information from the module. The monitoring modes are as follows:

Mode number	Type	Execute with
66	Read actual values	FB165
71	Actual position value	FBI 64
73	Distance to go	FBI 64
74	Monitoring off	FB164

The monitoring modes do not appear directly on the PG interface. They are used internally by the COM247 software package.

Monitoring modes do not influence the operating mode and can be executed at any time regardless of the axis status.

On the PC interface, the monitoring modes 71 and 73 are started with FB1 64, the monitoring mode 66 is started with FBI 65.

Function block FBI 64 continues an activated monitoring function (71, 73) periodically. You can stop these monitoring functions with mode 74.

In contrast, mode 66 (“read actual values”) supplies both actual values simultaneously using FB1 65. These are stored in the destination data block (=> Section 6.3.8.6 “Occupation of the Data Area when reading Actual Values”),

For more detailed information about the monitoring modes, refer to the description of function blocks FBI 64 and FBI 65.

5 COM247 Communications Software

5.1 Introduction

The programming package COM247 which runs on the PG, provides you with user-friendly support for programming and starting up the IP247. All the functions are executed by means of menu displays (input fields) and function keys.

If you create machine data or machining programs for the IP247, you can store the data on the programmer (PG), on the module (IP247) or on a floppy or hard disk (FD).

You start the COM247 software package by selecting the package from the Komi (command interpreter). At the Komi level, a brief description of COM247 is displayed if you press function key <F3>. By pressing <F1>, COM247 is loaded and the first display, the configuration display, appears. This displays the logo of COM247. The COM247 version and the serial number are displayed. From this display you branch to the presets display with <F1> (START).

in the presets display, you must select the drive on which the data blocks are to be read and saved. If you move the cursor to the appropriate input field, you can select the drive using the HELP key <F7>. Again using <F7>, you can select files on the previously selected drive. If there is no file on the selected drive, you must enter the name of a new file here. In addition to the file name, the fields “plant designation” and “generated by” must be completed. After entering this data, you can change to the next display with function key <F1> (BEGIN). A file with the required file name is then generated on the selected drive. The documentary information “plant designation” and “generated by” are saved in the file. If the specified file already exists however, the fields are completed with the stored information. Using <F2> (ONLINE-OFFLINE) you can set the operating mode, The two possible modes are online and offline. Online, the mode "SYSID output” (BA 70) is executed and the fields are completed with the module data of the IP247. The date and time are read from the hardward clock in the PG and entered in the fields “PG date-time”. The date and time can still be manipulated. Changes are, however, not transferred to the hardware clock of the PG. If the hardware clock is wrong, it must be set with the PCP/M-86 program “date” at the operating system level.

By pressing <F1> (BEGIN) you branch to the basic display. In the online mode, the operating mode "SYSID input” (BA 24) is executed when you change to the basic display, The module data, module number, slot number and page address are transferred to the IP247. The slot number and page address are only used for documentation. You can only change the module number, on the other hand, when there are no machine data on the IP247. If there is machine data on the IP247, whose module number is not identical with the entry “module no.”, the error message “correct MD - module number cannot be changed” is displayed in the error line.

Using the function keys, you can now enter, output, modify, delete or transfer machine data and machining programs, Test functions can be executed with the function key <F3> (TEST).

Remember that the following limits apply to files:

maximum number of machining programs per file: 250
maximum number of machine data per axis and file: 16
maximum number of files which can be selected in the presetting display with <F7>: 32

The interactive menu displays of COM247 include the following elements:

- fixed texts,
- input fields and
- output fields,

The displays are largely self-explanatory. It is, however, advisable to have the User's Guide at hand until you are completely familiar with them, The User's Guide contains descriptions of the displays and explains the significance of the input and output fields. The function keys for each display are also explained,

You make entries in the input fields of the displays at the alphanumeric keyboard or using the HELP key <F7>. These fields are displayed on the screen inversely, in this description they have a grey shaded background. The menu line of each display is also on a grey shaded background, however, there are no input fields in this line.

Output fields in the displays are used to display COM247 statuses and parameters. Outputs appear on the screen as fixed text, In the following description, they are shown in boxes with a broken line,

In all the displays explained in the following sections, the input fields are already completed with typical selections. The output fields also have values entered. These are in some cases fixed for the particular display (operating status) or may change according to the previous entries (parameters).

By pressing the RETURN key, you jump to the next input field. To edit within the input fields, use the arrow keys to move the cursor,

Error messages from the IP247 and from COM247 are always displayed in the last line (error message line), In column 1, the identifier for the current axis is displayed, followed by the delimiter character | and a blank. A O in column 1 stands for a general message. The error message is then displayed preceded by the error code. From column 60 onwards the error codes of the three axes are once again displayed (axis 1...3).

Example:

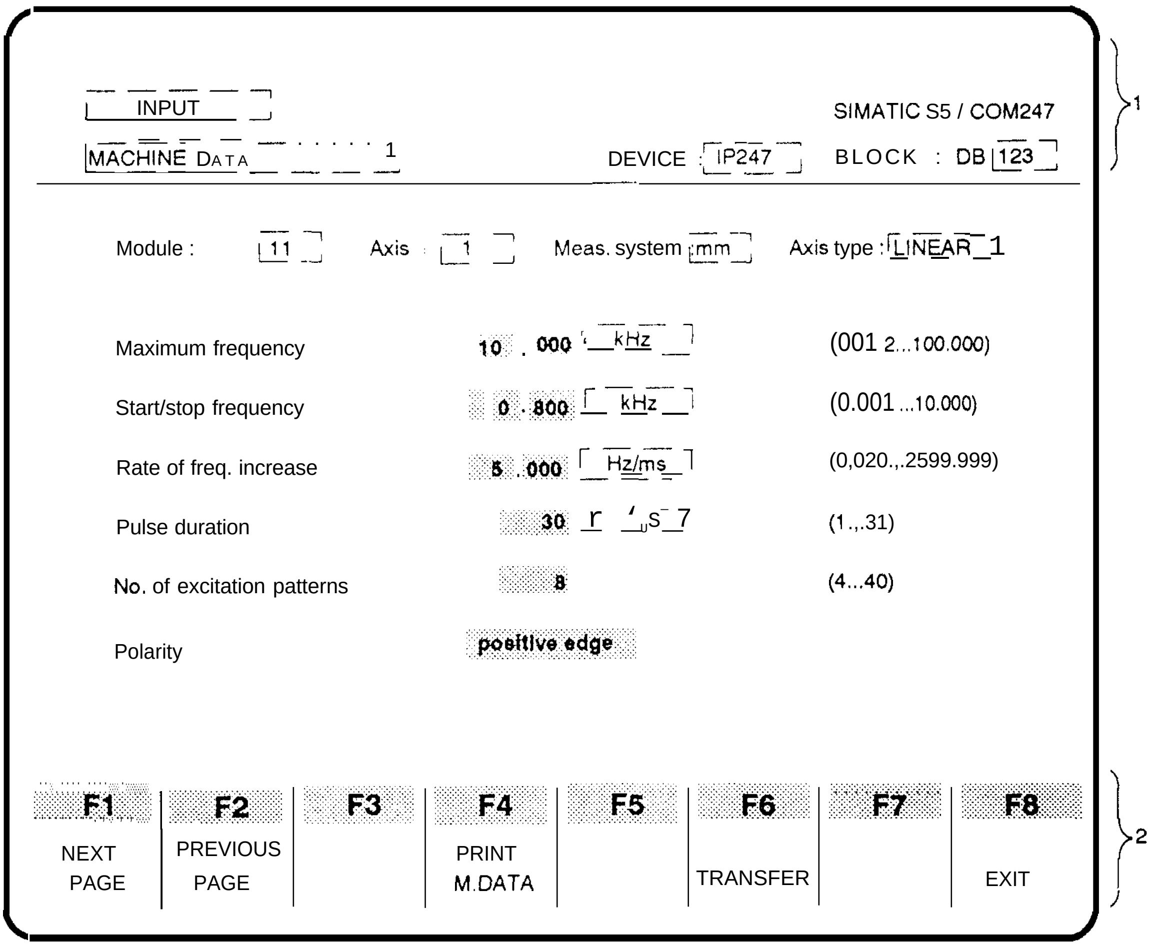
1 | F8A reference point missing F8AFO0 F00 F00

The message has the following significance

1 | : the current axis is axis 1,

F8A reference point missing: axis 1 is not calibrated,

F8A F00 F00 F00: axis 1 is signaling the module error 8A, axis 2, axis 3 and the data channel are not signaling a module error,



1 = header
2 = softkey menu

= input field = output field

Fig. 5/1 Display for entering machine data

The displays are structured so that you can always recognize the current operating status. At the top left INPUT and below this M A C H I N E D A TA may be displayed. This shows that the current function involves input of machine data. The DEVICE output field displays the target device and the BLOCK field displays the DB no. for these machine data. The output fields module, axis, meas. system and axis type indicate the values selected in a previous display.

You can now enter the actual machine data in the nine input fields (shown on a grey background).

Travel data and speeds always refer to the measuring system selected in the machine data record. The appropriate dimension is therefore always displayed following input fields and output fields involving dimensions.

Using the function keys <F1> and <F2> you can call further machine data pages. With function key <F4> you can print out the machine data on a connected printer. By pressing function key <F6> you can transfer all the machine data to the selected device. <F8> brings you back to the basic menu without transferring the machine data.

5.2 Definition of Terms

PC (or PLC) : programmable controller for SIMATIC S5

Operating system: COM247 runs under the operating system S5-DOS. Remember that S5-DOS itself consists of the operating system PC P/M-86 and the additional functions provided by the ZEFU diskettes. These functions are activated with “S5”. The operating system is not supplied with COM247 and must be ordered separately if not already available,

COM247: programming package for user-friendly operation of the intelligent I/O module IP247 from a programmer.

Function key: in the programming package COM247, function keys are the eight keys indicated by <F1>...<F8> on th PG keyboard.

IP247: intelligent 1/0 module of the SIMATIC S5 range, With this module, you can operate three independent stepper motor axes.

Display: the display or screen form used for input and output of data on the monitor.

Menu: inverse display of function keys <F1>... <F8> and a text to indicate the function currently assigned to this key.

PG: programmer for SIMATIC S5 (e.g. PG635, PG675, PG685, PG695, PG 730 and PG750).

In this User’s Guide, the following conventions have been used for all commands entered at the programmer:

- The equality sign (=) at the start of a line indicates the beginning of a new activity.
- The greater than character (>) at the start of a line indicates a keyboard input.
- Keyboard inputs start with the character displayed by the currently active program as a system prompt. This is followed by the characters to be input shown in bold upper case characters,
- <CR> stands for carriage return (the return key).
- <F1>. . <F8> stand for the function keys F1. . F8.

5.3 Getting Started

5.3.1 Consignment

Under the order number 6ES5895-5SB22—, the manual includes, among other things, this User's Guide, **a 5** 1/4 inch diskette anda31/2 inch diskette each with the file:

S5PEC1OX.CMD

The software package **COM247** runs under the operating system S5- DOS, which is not part of the consignment.

5.3.2 Setting the Configuration Register

If the operating system **S5-DOS** has never run on your PG, you must set the configuration register of the PG using the test diskette. You must enter the memory capacity, the drive configuration and other important PG characteristics in the configuration register to be able to inform various programs (e.g. **S5-DOS**) of the hardware configuration. To set the configuration register, you must **insert** the test diskette supplied with the PG in drive A: and start the PG either by turning on the power or by using the key switch. Answer the prompt "CHANGE CONFIGURATION?" with "Y" and then **mark** the appropriate information with **"+"** and the incorrect information with "-". Once you have answered all the questions, you can remove the test diskette and continue with PC P/M after a cold restart. The content of the configuration register is retained even when power is switched off.

5.3.3 Working Copy of the COM247 Diskette

Before you use the **COM247 diskette,** you **should** make a working copy and put the original away for safekeeping. To make a copy, use the **PCP/M** utility **"DSKMAINT",** with which you can check, format and copy diskettes. (In more **recent** S5 versions, **'DSKMAINT'** has been replaced by 'DISK'.)

5.3.3.1 Programmers with one Floppy Disk Drive (PG685)

=	**PCP/M** system diskette"1 of n" in drive A:
=	**Start the** PG by turning on the power or using the keyswitch
>	A> **DSKMAINT <CR>**
=	**New** diskette in drive A:
>	**<F5> <F1> Y <F8>**
=	**COM247 diskette in drive A:**
>	**<F3> <F1>**
=	**Formatted diskette in drive A:**
>	**<F1> Y**
=	**insert the diskettes required by DSKMAINT in drive A:, the COM247** diskette is the source diskette and the newiy formatted diskette is the target diskette.
>	**<F8> <F8>**

5.3.3.2 Programmers with two Floppy Disk Drives (PG675, PG635)

= PCP/M system **diskette"1** of n" in drive A:
= Start the PG by turning on the power or using the keyswitch
> A > **DSKMAINT <CR>**
= **New** diskette in drive A:
> **<F5><F1> Y <F8>**
= COM247 diskette in drive B:
> **<F3> <F3><F1 > Y**
> **<F8> <F8>**

5.3.4 System Configuration

5,3.4.1 Programmers without a Hard Disk (PG675)

To work with COM247 effectively, it is advisable to create a system diskette on which all the required programs are available, i.e. programs from the software package "PCP/M" should be copied to one diskette:

= PCP/M system diskette"1 of n" in drive A:
= Start the PG by turning on the power or using the keyswitch
> A> **DSKMAINT** <CR>
= New diskette in drive B:
> **< F 5 > <F3> Y <F8> <F8>**
> **A> PIP <CR>**
> ***B:= PCPM.SYS[RV] <CR>**
> ***B:= CCP.CMD[RV] <CR>**
= **ZEFU diskette"3 of n" in drive A:**
> ***B:= S5WX200X.CMD[RV] <CR>**
> ***B:= S5WX201X.CMD[RV] <CR>**
> ***B:= S5WX202X.CMD [RV] <CR>**
> ***B:=S5WX204X.CMD[RV] <CR>**
> ***B:= S5WX000H. CMD[RV] <CR>**
> ***B:= S5WX100X.CMD[RV] <CR>**
> ***B:= S5KXS02X.CMD [RV] <CR>**
> ***B:= S5KES02X. DAT[RV] <CR>**
> ***B:= S5KES01X. DAT[RV]<CR >**
> ***B:= S5.CMD[RV] <CR>**
> ***<CR>**

This diskette now contains the operating system "PCP/M" and all the S5-DOS programs you require to work with COM247. If this system diskette is correct, you should make it read-only by placing a protective tab over the notch, since you should only read from this diskette when working with COM247. To avoid having to repeat this operation if the diskette is damaged or lost, you should make a further copy of this diskette and keep it in a safe place.

Apart from the system diskette, you also require the COM diskette with the COM package which will later also contain the machine data and machining programs in the form of data blocks. You create this diskette by formatting a new diskette (DSKMAINT). Following this, the COM package can be copied to this diskette.

= PCP/M system diskette"1 of n" in drive A:
\> A> PIP <CR>
= Newly formatted diskette in drive B:
= COM247 diskette in drive A:
\> ***B:=S5PEC10X.CMD** <CR>
\> *<CR>

With the PG635, you do not need to create your own system diskette, The PG635 is started with the PC P/M-86 system diskette (booting). Following this, you take the system diskette from drive A: and insert the ZEFU diskette. This diskette contains all the files mentioned above and "S5 DOS", The COM247 diskette is inserted in drive B:. When you have loaded COM247 (the logo "COM247" appears on the screen), the COM247 diskette can also be removed from drive B: and a data diskette inserted in its place,

5.3.4.2 Programmers with a Hard Disk (e.g. PG685)

Programmers with a hard disk have the advantage that almost all programs and data can be accessed directly owing to the high capacity of the hard disk, This means that several SIMATIC program packages can be located simultaneously on the same data medium.

Installing PC P/M

If your programmer is new and PC P/M is not yet installed, you must first format the hard disk with the PC P/M utility "HDFORM6" (see PC P/M User' sGuide, page 6-29).

Remember
From release 1.0/5 (1 .0/6 for PG695) HDFORM6 has been replaced by HDPARTY.Please refer to the manual for further information. On the PG 750, HDFORM6 has been replaced by HDMAINT.

Caution

When formatting the hard disk, all programs and data already on the disk are lost!!

= PCP/M system diskette"1 of n" in drive A:
= Start the PG by turning on the power or using the keyswitch
\> A> **HDFORM6** <CR>
= Enter the disk capacity, e.g. 12 MBytes
\> 12
\> Y

Remember
If your programmer has one floppy disk drive, the hard disk has the logical name “B”. The operating system displays the prompt “B>”.

Next, the programs on the PCP/M diskette must be copied to the hard disk, as follows:

= PCP/M system diskette"1 of n" in drive A:
. Start the PG by turning on the power or using the keyswitch
> A> **PIP <CR>**
> ***B: =*.* <CR>**

If PC P/M was supplied on more than one diskette, the programs from the other diskette should also be copied to the hard disk:

= Next PCP/M diskette in drive A:
> *B: **=*.* <CR>**

After copying the last diskette, press <CR>, to complete the copying program “PIP”. Now that all the PCP/M system programs are on the hard disk, this can be selected as the default drive:

> A> **B: <CR>**

The operating system now searches for all programs on the hard disk, unless a floppy disk is specified explicitly.

To avoid the programs being deleted accidentally, and to make them accessible from all user areas, assign the “read-only” (RO) and “system” (SYS) attributes using the “SET’ utility as follows:

B> **SET B:*.*V[RO SYS] <CR>**

Installing COM246

The following description assumes that S5-DOS is installed on your PG. If this is not the case, please read the section “installing PC P/M”.

To install COM247 on the hard disk, you simply copy the file S5PDC10X.CMD on the supplied COM247 diskette to the hard disk and assign the attributes “RO” and "SYS" to it.

= Start the PG by turning on the power or using the keyswitch without a diskette in the drive
= COM247 diskette in drive A:
> **B> PIP B:= A: S5PEC10X.CMD <CR>**
> **B> SET B: S5PEC10X.CMD IRO SYS] <CR>**

5.4 Starting COM247

The following description assumes that you have made the preparations described in “System configuration” (you have created a system diskette or installed COM247 on the hard disk).

With PGs without a hard disk, the prepared system diskette is inserted in drive A and the data diskette in drive B
With PGs with a hard disk, drive A: must not have a diskette inserted
Start the PG by turning on the power or using the keyswitch

The S5 call loads the KOMI in the user memory of the PG. While this is being loaded, the KOMI mask appears as shown below:

SIMATICS5	S5 - Komi	
Serial-No.:	Xxxx-yyyy-zzzzzz	All rights reserved
Copyright (c) 1986		SIEMENS AG

In the “SELECT PACKAGE” menu, you can now select the required program, in this case COM247, by moving the cursor with the arrow keys. If you then press <F1> (PACKAGE) this program is loaded from mass memory. Once the package is loaded, the first COM247 display, the configuration display, appears.

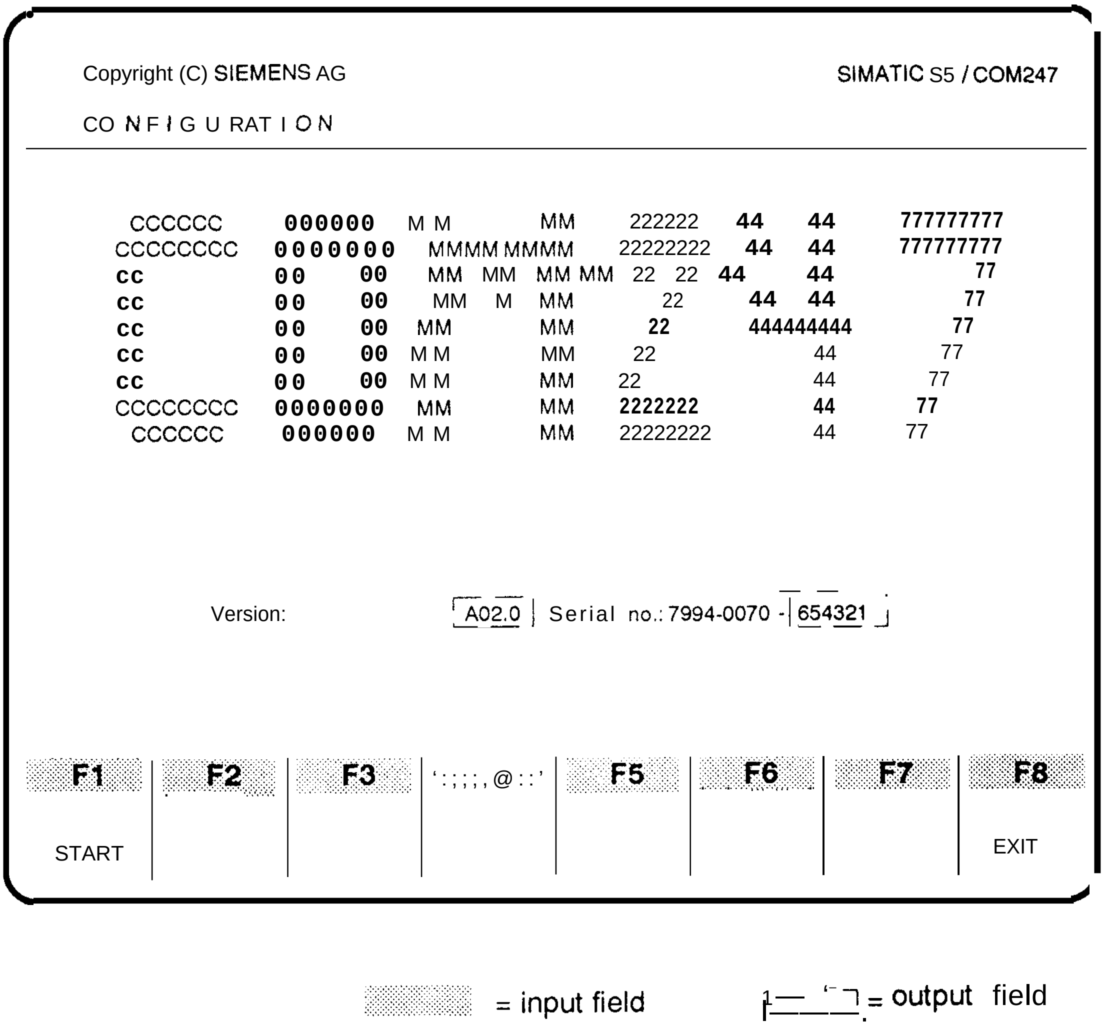


Fig. 5/2 Configuration display

This displays the COM247 logo and the version and serial number of COM247.

Description of the output fields

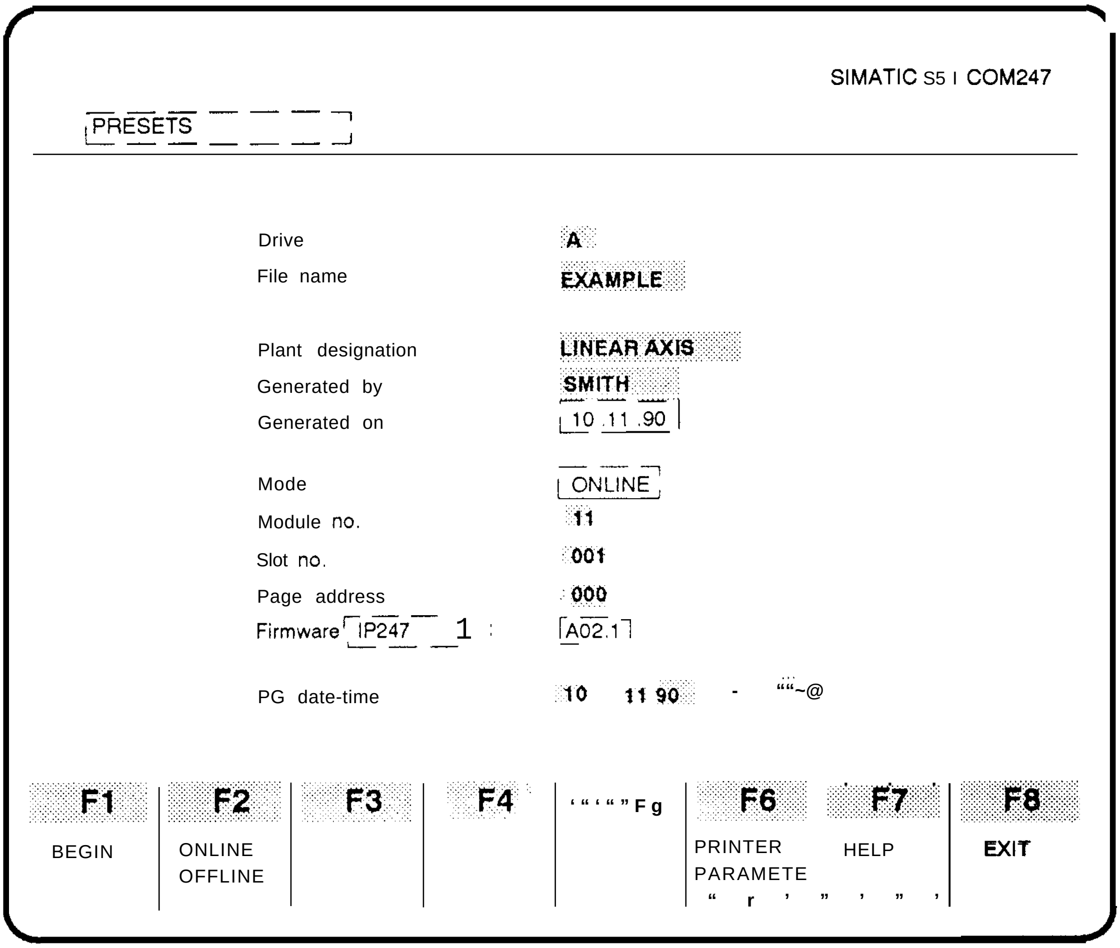
Version: this field displays the version of COM247

Serial number: each diskette has a serial number which is displayed in this field.

Significance of the function keys

<Fl >: With function key <F1> (START) you branch to the next display, the presets display.

<F8>: With this key you exit COM. You will be prompted to confirm that this is your real intention.



= input field = output field

Fig. 5/3 Presets display

Description of the output fields

P R E S E T S is displayed in the header.

Generated on:

This field displays the date on which the selected file was created, if it already exists. If you create a new file, the current date from the internal PG clock will be used as the generation date.

Mode:

The mode selected with <F2> ONLINE or OFFLINE is displayed here,

Firmware:

I mediately after “Firmware”, the module type "IP247" is displayed. The next output field displays the release of the firmware.

Description of the input fields

Each module has several characteristics (SYSI D), some of which cannot be changed and some of which can be selected. There are other characteristics which must beset, such as the module number, and some which can be set. The latter are mainly of a documentary nature and are not checked.

Drive:

Here, you specify the drive in which the user data diskette is to be inserted. With the PG685, the data can, of course, be stored on the hard disk. In this case “B” must then be selected.

Filename:

The file name identifies the file in which the data blocks are stored (here, EXAMPLE). This allows you to assign different files to different projects or plant sections. Using the HELP key, you can display all the files on a current drive with the file type .247. **As you** page through the file names, the “plant designation”, “generated by” and “generated on” fields are updated.

Plant designation:

In this field you can select a brief designation for the plant for which the data blocks are intended (here, LINEAR AXIS). This designation is written into the file header. This field must be completed, otherwise the error message “illegal input” is displayed.

Generated by:

As in the plant designation field, the name of the operator (here, SMITH) can be saved in the file, Once again this field must be completed.

Module no.:

This is a number between O and 99 which you can specify to distinguish between various positioning modules. A module number is also entered in the machine data records. If there is already machine data on the module, the module number entered here must be identical to that contained in the machine data records. This means that the module number can no longer be changed if one correct machine data record already exists on the module. (Only possible online.)

Slot no,:

You can also assign this number as required, between O and 255. The number is simply for documentation. (Only possible online.)

Page address:

The conditions for this are the same as for the slot number. The page address set on the module can be read more easily at the PG than by reading the switch on the positioning module itself (only possible online). Differences between the switch setting and the entry made here are not checked.

PG date - time:

The internal PG date and time are displayed here, If you modify anything in these fields, this is taken as a date and time change and the software clock of the PG is set to these values.

Note

After switching off the PG, this setting is lost. The hardware clock can only be set at the system level.

Significance of the function keys

<F1>: With BEGIN, you branch to SELECT FUNCTION and providing ONLINE is set, the presets (module number, slot number, page address) are written to the module. These values are, however, only accepted by the module when either no correct machine data is stored on the module or the module number is identical to the module number in the machine data (=> Section 4.3.18 "SYSID Input"),

<F2>: This key switches from OFFLINE to ONLINE and vice- versa. If you switch to ONLINE, the values on the module “module number”, “slot number”, “page number” and “firmware release” are read (=> Section 4.3.19 “Read SYSID") and output in the display. OFFLINE, these fields are deleted.

<F6>: Branch to the printer parameter display. Here, you can stipulate control character sequences,

<F7>: HELP key to select possible drives and file names contained in them.

<F8>: With the EXIT key, you return to the configuration display.

5.5 Function Selection

By pressing cF1 > (BEGIN) in the presets display, you branch to the “SELECTF UNCTION” display.
The presets are displayed once again. The fields are, **however,** no longer input fields, i.e. you cannot change the displayed values. From this display you branch *to* the individual functions. If you terminate a function with the EXIT key you always return to this display.

SIMATIC S5 / COM247

SELECT FUNCTION

Drive	A
File name	EXAMPLE1
Plant designation :	LINEAR AXIS
Generated by	SMITH
Generated on	15 09 89
Mode	ONLINE
Module no.	11
Slot no.	001
Page address	000
Firmware IP247 :	A02.1
PG date-time	15.09.89 - 12:23

F1	F2	F3	F4	F5	F6	F7	F8
INPUT	OUTPUT	TEST	TRANSFER	DELETE		INFO	EXIT

= **input field** = output field

Fig. 5/4 Basic display

Significance of the function keys

<Fl >: Branch to input of machine data/machining programs.
<F2>: Branch to output of machine data/machining Programs.
<F3>: Branch to test mode.
<F4>: Branch to transfer of machine data/machining programs to the individual media.

<F5>: Branch to deleting machine data/machining programs on the individual media.

<F7>: Branch to information (overview) about machine data/machining programs as they exist on the individual media.

<F8>: Return to the presets display.

5.6 Input

If you press <F1> (INPUT) in the basic display (select function) you branch to data input. Here, you can generate machine data or machining programs and store them on the module, the PG or a data drive. You stipulate the destination device in this mask (function keys <F1>...<F3>).

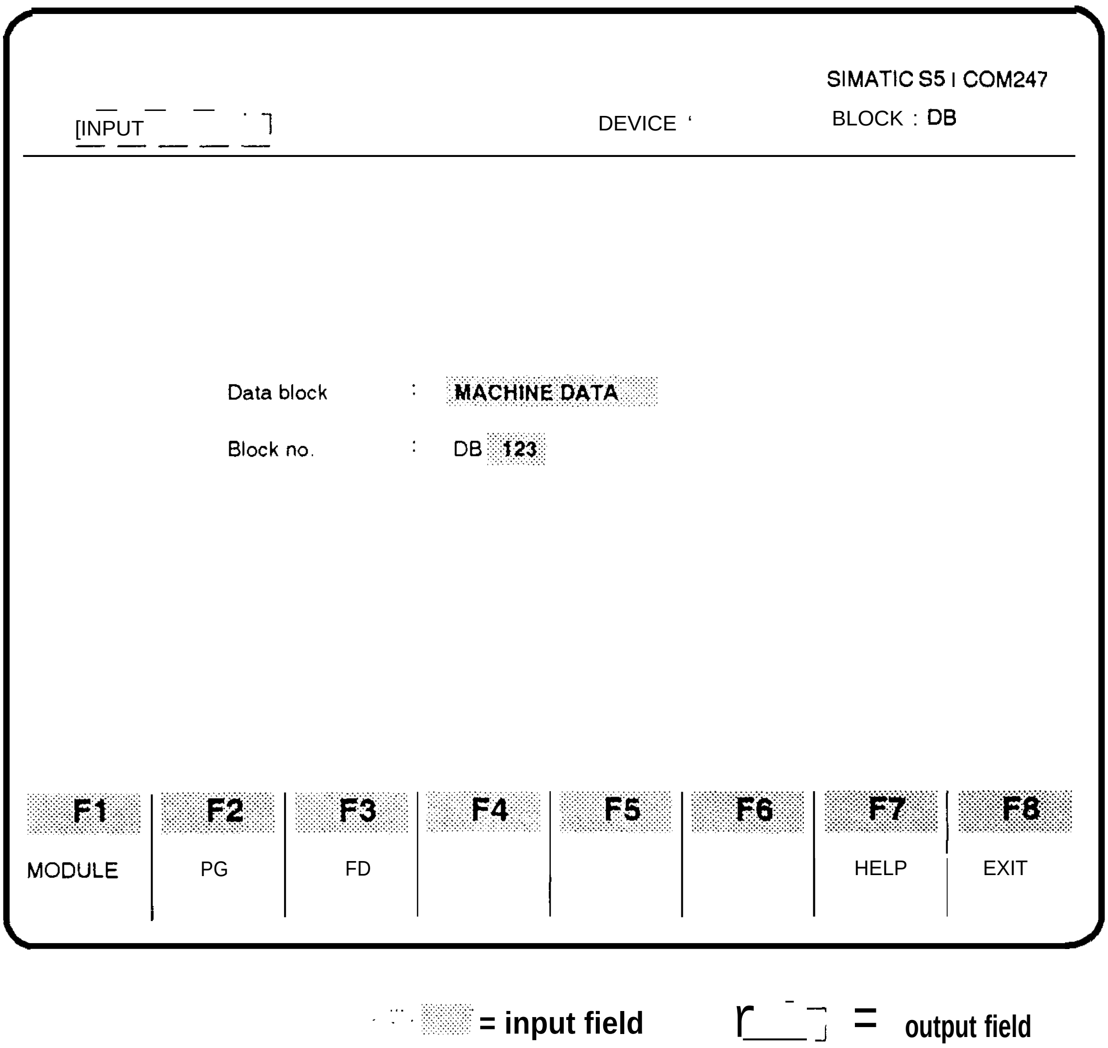


Fig. 5/5 Block selection

Description of the output fields

I N P U T is displayed in the header. The other fields are still blank.

Description of the input fields

Data block:

> **Select machine data or machining program. This field can be manipulated with the function key <F7> (HELP).**

Block no.:

You select the data block number under which the generated data is to be stored. The DB number can be a value between O and 255.

Description of the function keys

<Fl >: The destination device is the IP247 module. You branch immediately to the next display, either to the first machine data or the first machining program display.

<F2>: The destination device is the programmer. You branch immediately to the next display, either to the first machine data or the first machining program display.

<F3>: The destination device is the data drive entered in the configuration display. You branch immediately to the next display, either to the first machine data or the first machining program display.

<F7>: With the help key you can select the text “machine data” or “machining program” in the “data block” input field.

<F8>: If you press the EXIT key, you return to the basic menu (function selection).

5.6.1 Entering Machine Data

5.6.1,1 General Information about Machine Data

Before traversing movements can be executed, each axis requires the technical data of the drive. This information is known as machine data (=> Section 2.5 “Machine Data and their Structure”).

When you input the machine data, the module checks that the data is consistent. If you make an error, an error message is output and the program branches to the display in which the incorrect value might be located. After correcting the value, you can once again transfer the data record to the module.

The data block numbers of the machine data for the three axes can be identical.

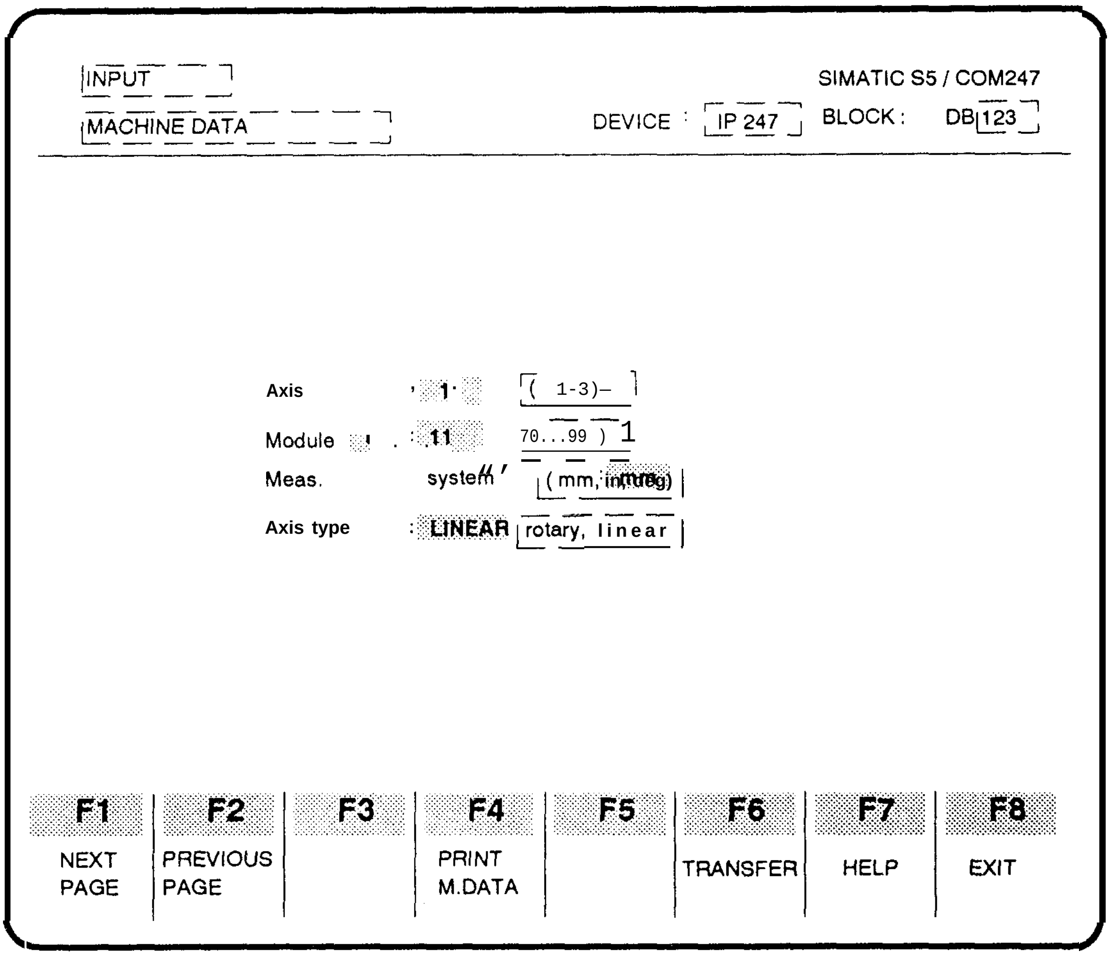
When you delete and re-enter a machine data record, the reference point need not necessarily be lost; this depends on the machine data in the new data record which have changed compared with the old record (=> Section 4.3.21 “Machine Data Processing”). Zero offsets and tool length offsets are, however, reset each time new machine data are input to the module.

The individual machine data are only explained briefly here. For more detailed information about machine data, refer to Section 2.5 “Machine Data and their Structure”.

5.6.1.2 Compiling Machine Data

If you selected “machine data” in the data block selection display, specified the block number and pressed one of the function keys <F1>...<F3>, the axis selection display appears. The destination device (I P247, PG, FD) is now fixed and can no longer be changed for this input.

As an example in this section, a machine data record (data block number 123) will be generated, The destination device on which the data will be stored is the IP247 module, module number 11. The data record refers to a linear axis, axis 1, with metric dimensions.



= input field = output field

Fig. 5/6 Axis selection

Description of the output fields

INPUT and MACHINE DATA are displayed in the header. The previously selected destination device is displayed in the DEVICE field and the previously selected DB no. in the BLOCK field.

Description of the input fields

Axis:

In this field, you enter the number of the axis for which the machine data record is to be created. The number can be either 1, 2 or 3.

Module:

In this field, you can enter the number of the module for which the machine data record is to be created. This is necessary, since several I P247 modules can be installed in a system.

Meas. system:

Here, you enter the required measuring system. mm stands for millimetres (basic unit pm), in for inches (basic unit 0,0001 in) and deg for degrees (basic unit 0.001 degrees).

Axis type:

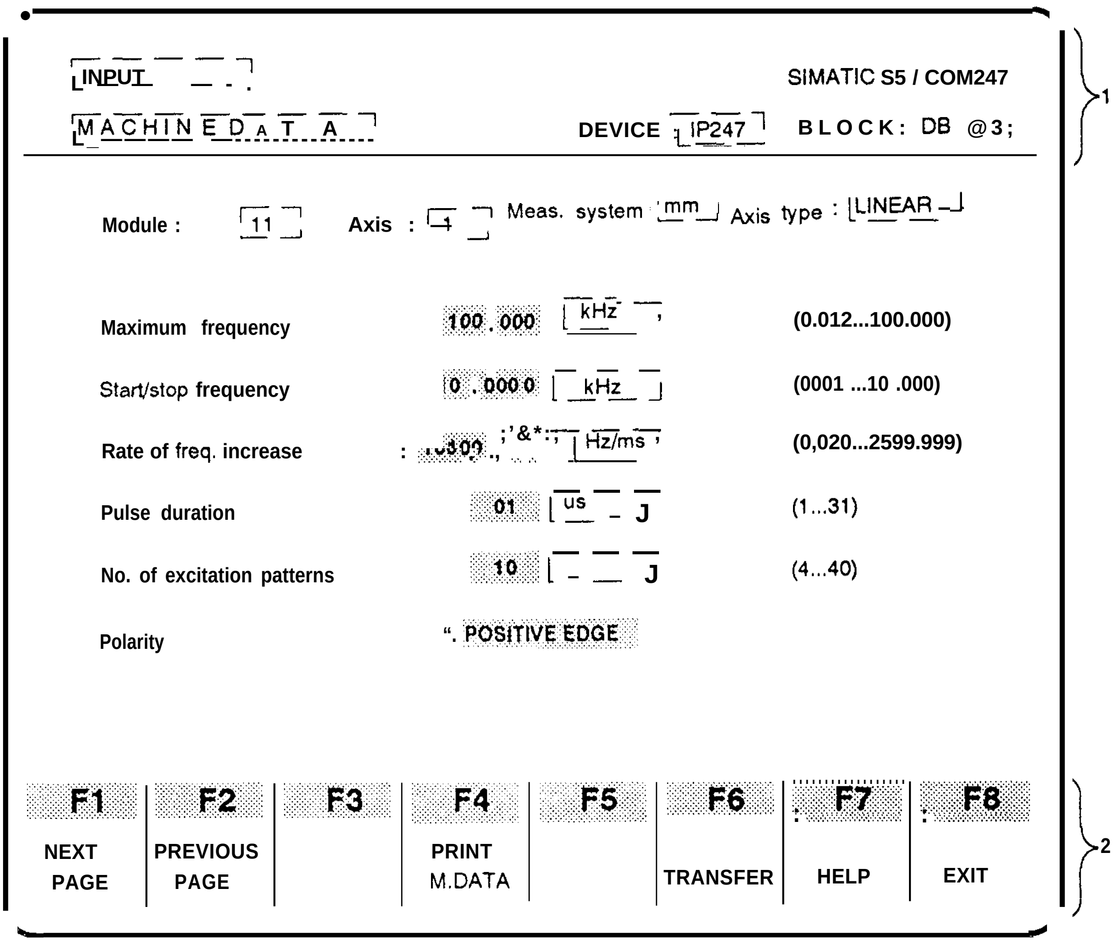
Using the help key <F7> you can select either a rotary axis “rotary” or linear axis “linear”,

These values are used as output values for all machine data pages.

Significance of the function keys

<F1>: With function key <F1> you branch to the first of four displays for machine data compilation.
<F2>: From this display, you branch to the last machine data page.
<F4>: Output all the machine data on the printer.
<F6>: Store all the machine data on the selected destination device. This is, however, only possible when all the input fields of all the pages have had values entered.
<F7>: Switchover the measuring system or axis type, providing the cursor is located in the appropriate input field.
<F8>: Return to the basic mask (function selection) after confirmation.

Machine Data Page 1



1 = header
2. softkey menu

= input field

= output field

Fig. 5/7 Machine data page 1

In this display, you transfer the machine data required to generate the acceleration and deceleration ramp. The acceleration up to the maximum frequency (speed) is calculated according to the following formula:

$$\boxed{f = F(1 - e^{-t/\tau}) + f_{ss}}$$

Where:

- f: **frequency at given time**
- f_{ss}: **start-stop frequency**
- t: **acceleration time** (0...3τ)
- τ: **time constant**
- f_{max}: **maximum frequency**
- F: $(f_{max} - f_{ss})/0.95$

Description of the output fields

The header remains the same as described in the section “Machine Data Compilation”.

Module:
The previously entered module number is displayed here,

Axis:
The previously selected axis number is displayed here.

Meas. system:
The previously selected dimension is displayed here.

Axis type:
The previously selected axis type (“LINEAR” or “ROTARY”) is displayed here.

Description of the input fields

Maximum frequency (f_{max}):
This is the highest frequency to be output to achieve the maximum speed in the selected half or full step mode.

Start/stop frequency (f..):
The maximum frequency at which the stepper motor can startup from a standstill without losing a step or can brake to a standstill immediately taking into account the load and half or full step mode.

Rate of freq. increase (a):
a is the slope of the function

f(t) = F(I -e-VT) + F.. => a = F/1

Pulse duration:
This is the width of the pulse per period in microseconds. The pulse duration must always be less than the period of the maximum frequency.

No. of excitation patterns:
For the stepper motor to move on, it must be excited differently from step to step until it returns to a position corresponding to the initial position. From this position, it can once again be moved with the same excitation pattern. In this field, you enter the number of steps that must be output between two equivalent positions. In the half step mode, this number is twice as high as in the full step mode.

Polarity:
You select either “positive edge” or “negative edge” as the active edge of the pulse, to which the power unit reacts. At the same time, the outputs also change their inactive levels. (=> Part 4 “Functions”).

Significance of the function keys

<Fl >: Select the next machine data page.
<F2>: Select the previous machine data page.
<F4>: print out all machine data,
<F6>: Store all machine data on the destination device.
<F8>: Return to the basic display without saving the data

Machine Data Page 2

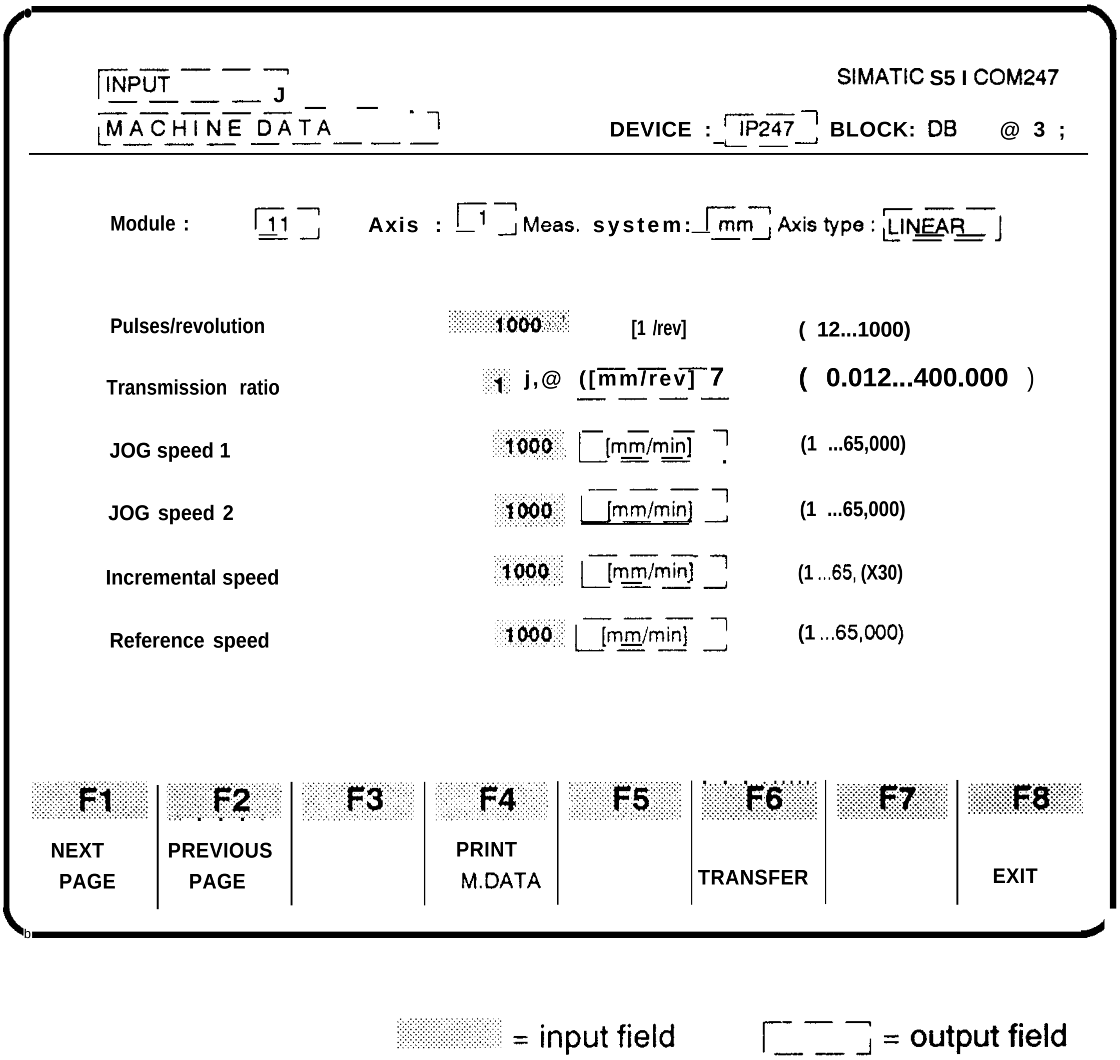


Fig. 5/8 Machine data page 2

Description of the output fields

The header remains as described in the section “Machine Data Compilation”.

Module, axis, meas. system, axis type: see machine data page 1.

Description of the input fields

Pulses/revolution:

Steps of the stepper motor per revolution in full or half step mode. Half step mode means twice the pulse count of full step mode.

Transmission ratio:

The transmission ratio indicates the distance travelled for one motor revolution.

Resolution = transmission ratio/(pulses **per revolution)**

Example:

A motor with **200** steps in full step mode connected directly to a leadscrew with **4** mm pitch/revolution, is to be operated in the half step mode.

⇒ Pulses/revolution: **200.2 = 400** pulses/revolution

⇒ Transmission ratio: 4.00 mm/rev

⇒ Resolution: $\frac{4mm}{400pul.} = \frac{1mm}{100pul.}$

The maximum resolution is 1 pm/pulse.

JOG speed 1 (V_{JOG1}):

This is the speed of travel in the mode “JOG speed 1“. This speed must correspond to a frequency less than or equal to f_{max}.

The following must apply:

$V_{JOG1} \leq V_{max} \Rightarrow f_{JOG1} < f_{max}$

JOG speed 2 (V_{JOG2}):

This is the speed of travel in the mode “JOG speed **2“.** This speed must correspond to a frequency less than or equal to f_{max}.

The following must apply:

$V_{JOG2} < V_{max} \Rightarrow f_{JOG2} < f_{max}$

Incremental speed (V_{inc}):

This speed is used for the operating mode “incremental speed absolute” and “incremental speed relative”. This speed must correspond to a frequency less than **or equal to** f_{max}.

The following must apply:

Vine **C** $V_{max} \Rightarrow$ **fine** $< f_{max}$

Reference speed (V_{ref}):

This speed is used in the “reference point approach” until the reference point marker (BERO) is found for the first time. This speed must correspond to a frequency greater than the start-stop frequency and less than or equal to the maximum frequency.

The relationship between frequency and speed is calculated from the machine data “pulses/revolution” and “transmission ratio”.

The frequencies corresponding to the following speeds must be within the range of values of f_{max} ,

Example

JOG speed 1:	3600 mm/min
Pulses/revolution:	500 l/rev.
Transmission ratio:	1 OOOmm/rev.

f_{JOG1} = V_{JOG1} [mm/sec] , pul./rev. [l/rev.]/transmission ratio [mm/rev.]

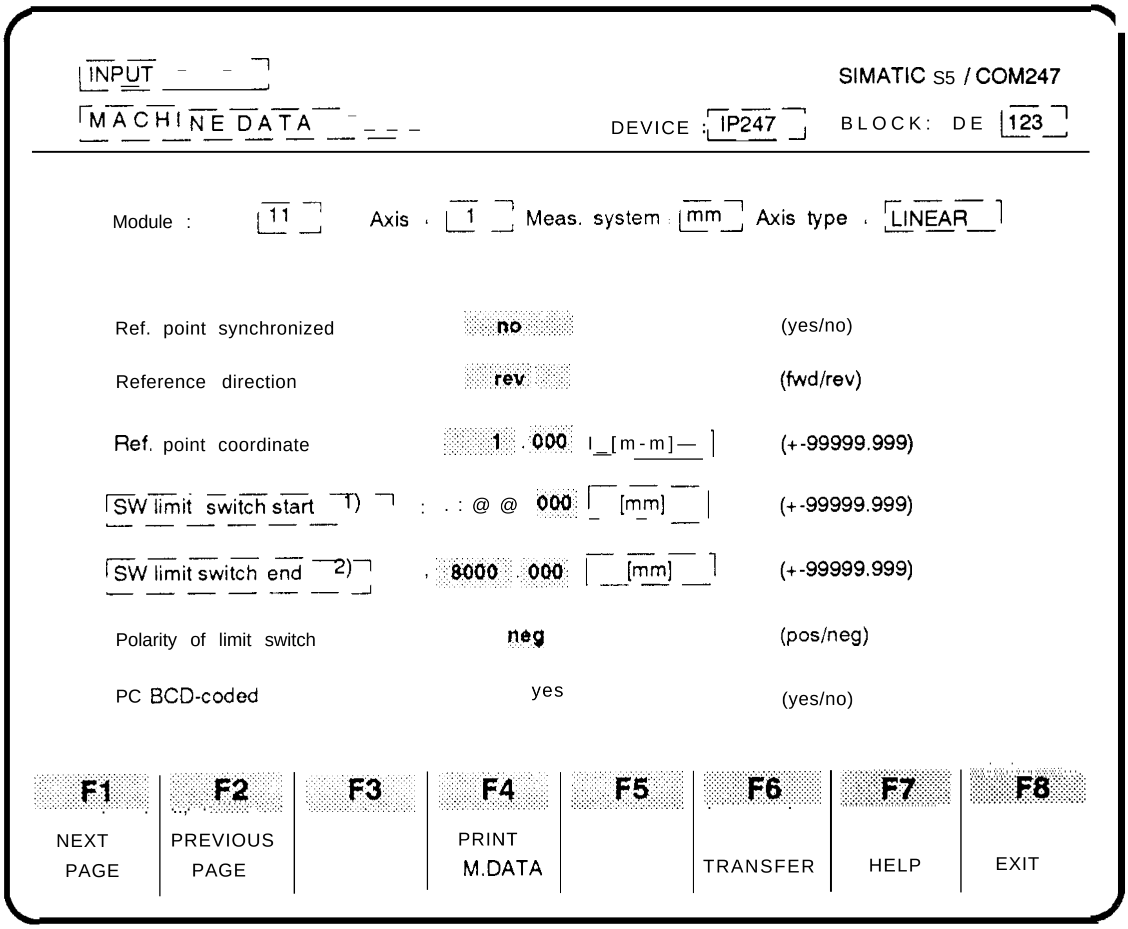
f_{JOG1} = 3600/60x 5000/10001/seC
= 30000 l/see
= 30 kHz

$f_{JOG1} \leq f_{max}$ (from the machine data)

Significance of the function keys:

<Fl >: Select the next machine data page.
<F2>: Select the previous machine data page.
<F4>: Print out all machine data.
<F6>: Store all machine data on the destination device.
<F8>: Return to the basic display without savng the data.

Machine Data Page 3



1) Rotary axis: traversing range start

2) Rotary axis: traversing range end

- input field [ ] = output field

Fig. 5/9 Machine data page 3

Description of the output fields

The header remains as described in section “Machine Data Compilation”.
Module, axis, meas. system, axis type: see Machine Data Page 1.

Description of the input fields

Ref. point synchronized

No: the reference point is set with the negative edge of the reference signal.

Yes: after the negative edge of the reference signal, the motor continues in single step operation until the counter of the excitation patterns (software counter in the firmware) reads zero.

Reference direction:
Here, you specify the direction in which the reference point is approached.

Ref. point coordinate (x_{ref}):

$x_{ref} < X_E$

SW limit switch start (X_A):

This value specifies the coordinate of the software start limit switch.

$X_A < X_{ref}$

Note

The value of the software start limit switch must be less than the value of the reference point coordinate and less than the value of the software end limit switch. All these coordinates must be within the hardware limit switches.

Traversing range start (X_A):

(For a rotary axis), This value indicates the start of the traversing range.

SW limit switch end (X_E):

This value indicates the coordinate of the software end limit switch. It must be greater than the software start limit switch, and greater than the reference point coordinate and must be within the hardware limit switches.

Traversing range end (XE):

(For rotary axis), This value indicates the end of the traversing range of the rotary axis. This is physically the same point as the start point. The actual position display jumps automatically from the end value to the start value.

Polarity of limit switch:

Here, you can decide whether the hardware limit switches are detected as having been activated on a positive or negative edge.

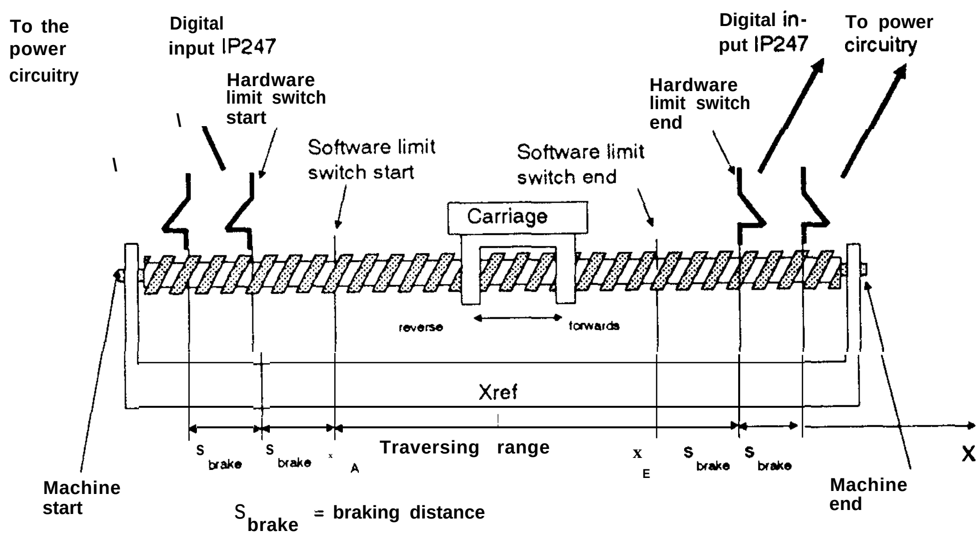


Fig. 5/1 O Position of limit switches

PC BCD-coded:

If you enter "yes" in this field (HELP key <F7>), all coordinates (targets, travel increments, zero point offsets and tool length offsets) transferred from the PC are interpreted by the IP247 in BCD format. The range of values in BCD format is limited to +/- 9999999 µm.

Significance of the function keys

<F1>: Select the next machine data page.
<F2>: Select the previous machine data page.
<F4>: Print out all machine data.
<F6>: Store all machine data on the destination device.
<F8>: Return to the basic mask without storing the data.

Machine Data Page 4

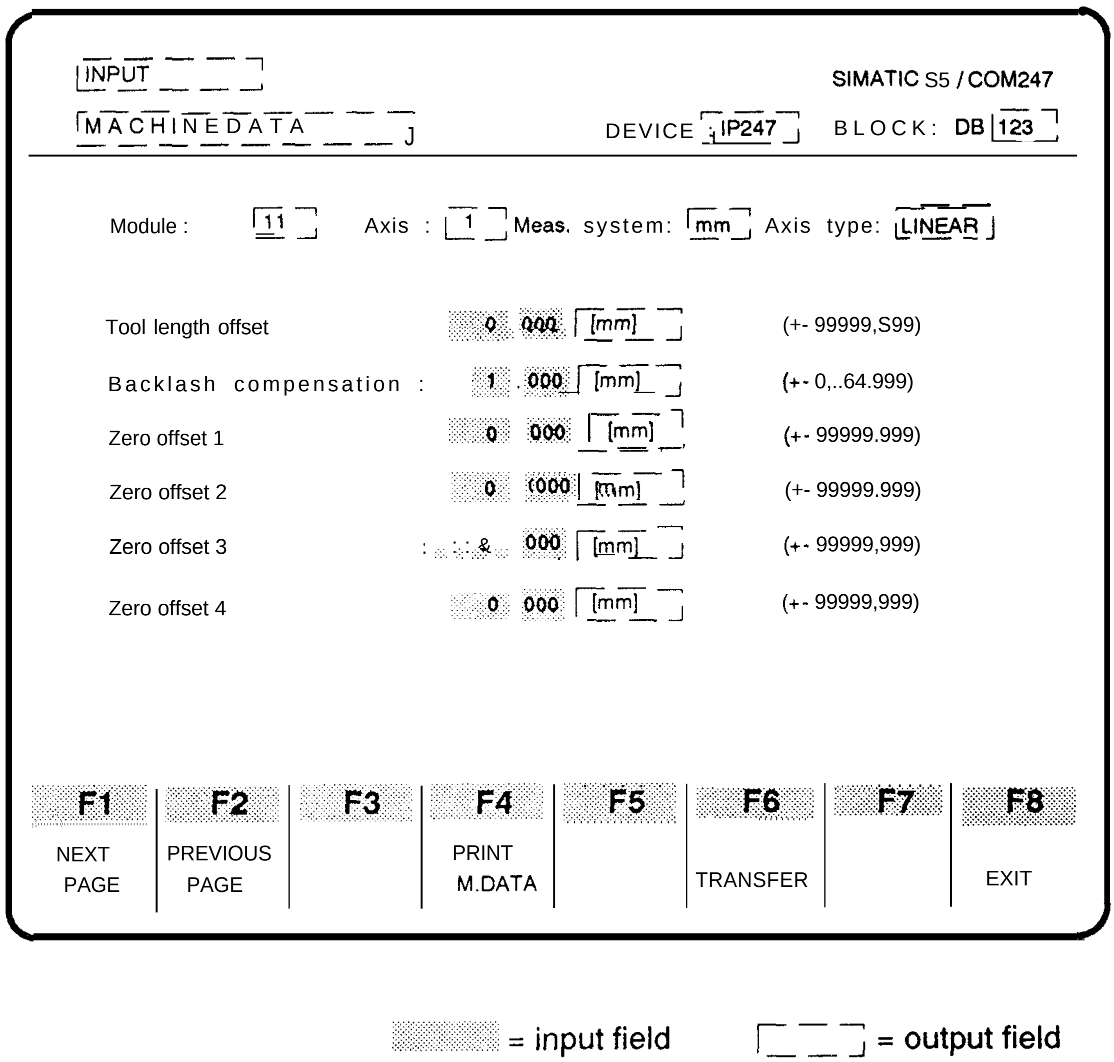


Fig. 5/1 1 Machine data page 4

Description of the output fields

The header remains as described in section “Machine Data Compilation”,

Module, axis, meas. system, axis type: see Machine Data Page 1.

Description of the input fields

Tool length offset;

> The value specified here can be selected in machining programs. Each specified target is corrected by this value. The length must be less than the range allowed by the software limit switches. This correction is added to a tool length offset previously selected and can be called repeatedly in machining programs.

Zero offset 1 **...4:**

> The values of the four zero offsets are independent of each other and can be called individually in machining programs. The range of values of the four offsets is ±99999.999 mm and must not be exceeded.

Backlash compensation:
This value is added to the distance to be travelled whenever the axis changes direction. This allows any backlash in the drive to be compensated.

Significance of the function keys

<Fl >: Select the next machine data page.
<F2>: Select the previous machine data page.
<F4>: print out all machine data.
<F6>: Store all machine data on the destination device.
<F8>: Return to the basic display without saving the data.

5.6.1.3 Print Machine Data

The data (DB number) selected using the functions “input” or “output” of machine data can be printed out using function key <F4> (PRINT M. DATA). Thedataareformatted in a fixed frame-work.

A header is printed out at the start of each machine data page and a footer at the end.

MACHINE DATA	Source device	DE no.	Axis	Module	Meas. sys.
LINEAR	IP247	123	1	11	mm

Printout header

Maximum frequency	100,000	[kHz]	(0,012...100.000)
Start/stop frequency :	10.000	[kHz]	(0,001...10.000)
Rate of freq. increase :	100.000	[Hz/ms]	(0.020...2599.999)
Pulse duration	01	[us]	(1...31)
No. of excitation patterns :	10		(4...40)
Polarity	POSITIVE EDGE		
Zero offset 4	0.000	[mm]	(+- 99999,999)

SIEMENS AG SIMATIC S5 COM247 - IP247	PRINTOUT MACHINE DATA AXIS 1 LINEAR AXIS SMITH	Date: 16,09.89 Page: 1

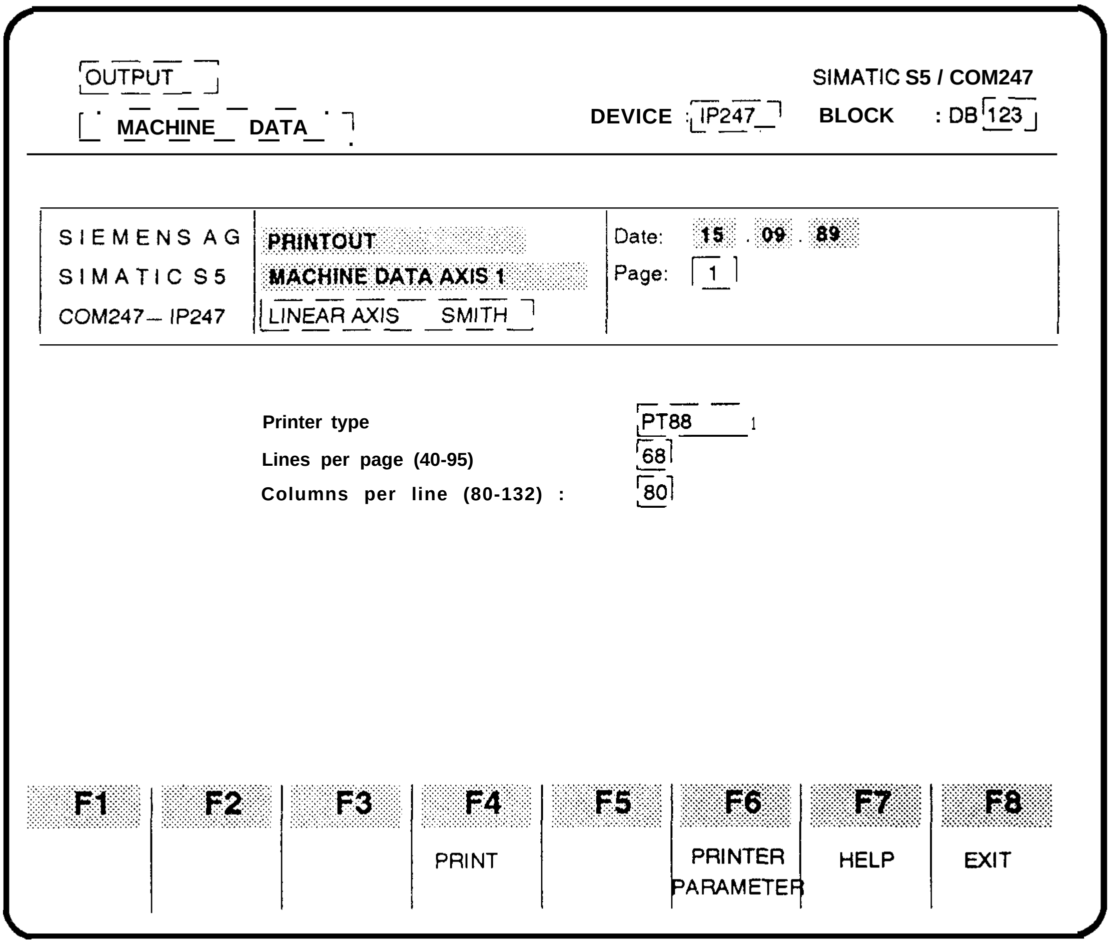
Footer

Fig. 5/1 2 Machine data printout

The following information is supplied in the header:

- the machine data are for a linear axis,
- the source device (FD, PG or IP247) from which they were read,
- the DB number under which they are stored,
- the axis and module for which they are intended and
- the measuring system (mm, in or deg.) of the machine data.

The footer is explained in the machine data printout display.



= input field = output field

Fig. 5/13 Machine data printout display

In this display there are two lines available to enter information about the machine data, e.g. the plant for which they are intended etc. This information is only used for documentation (here, PRINTOUT MACHINE DATA AXIS 1), The fields in the third line, “plant designation” and “generated by” are completed automatically from the information in the presets display. The date can also be entered. The page number is incremented automatically following each formfeed.

Description of the output fields

In output field 1 in the header, either INPUT or OUTPUT is displayed. M A C H I N E D A TA is displayed in output field 3. DEVICE indicates the previously selected destination or source device and BLOCK shows the DB number.

Page:
The numbers of the pages are displayed here during the printing. The page number is incremented by 1 at each form feed.

Third comment line:
The fields “plant designation” and “generated by” from the presets display are entered in this line (here, LINEAR AXIS, SMITH).

Printer type:
This field displays the printer selected in the printer parameter display. The default is the Siemens PT88 printer.

Lines per page:
The number of lines per page selected in the printer parameter display is shown. The default is 68 lines per page.

Columns per line:
The number of columns per line selected in the printer parameter display is shown here. The default is 80 columns per line.

If you have not yet printed out or not yet set the printer parameters, branch to the “printer parameter display” with <F6> (PRINTER PARAMETER). Refer also to Section 5.6.1.4 "Assigning PrinterParameters”.

Description of the input fields

Comment:
In two lines of the footer you can enter a comment about the machine **data to be printed out, This** comment is then printed out as a footer on each page.

Date:
In these three input fields you can enter the date on which the machine data were created. This date is also printed out on each page.

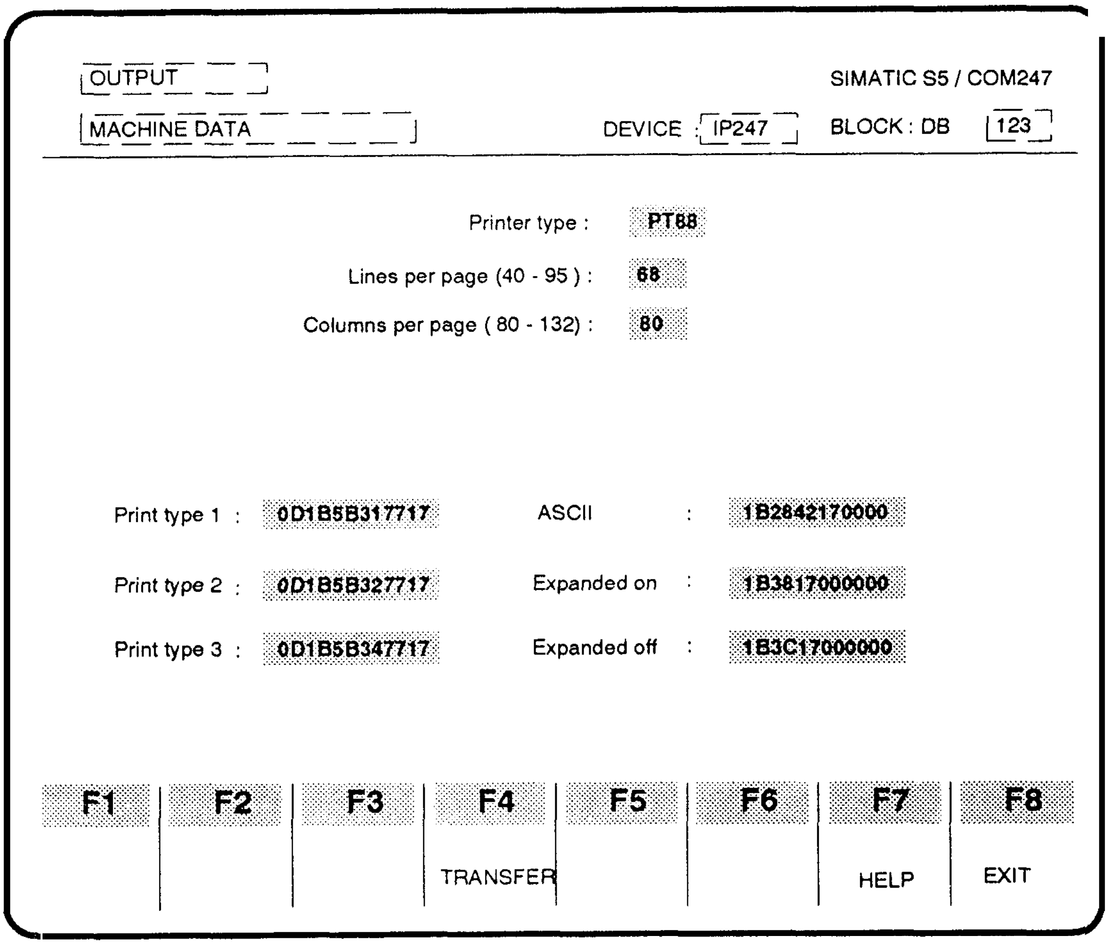
Significance of the function keys

<F4>: The printout is started with this function key.
<F6>: This function key branches to the printer parameter display.
<F7>: Help key: no function.
<F8>: Exit the print option without printing out.

5.6.1.4 Assigning Printer Parameters

From the presets display and from the display for printing machine data, you can branch to the printer parameter display by pressing <F6> (PRINTER PARAMETER) . Here, you can select the Siemens printers PT80 and PT88 or other printers using the IBM or EPSON mode. The values for the number of lines per page (default 68) and the number of columns **per line (default 80) can be** changed. You can also adapt the control characters for print type and character set to any printer. The control characters must be entered in ASCII code without gaps or separators. A maximum of 5 ASCII characters can be entered. If a control character sequence is less than 5 characters, you must complete the sequence with ASCII NIL characters

At present, only the parameters for print type 2 can be used.



= input field = Output field

Fig. 5/1 4 Printer parameter display

Description of theoutput fields

The header is the same as described in Section “Machine Data Compilation”.

Description of the input fields

Printer type:

Using the HELP key, you can select one of the four printers PT80, PT88, IBM and EPSON. The fields print type 1 to 3, “ASCII”, “expanded on” and “expanded off” are then completed accordingly; they can, however, still be adapted to your particular requirements.

Lines per page:
Here, you specify the number of lines per page.

Columns per page:
In this field you specify the number of columns per page.

Print type 1:
For the PT88 printer, the control characters (0DH), ESC (1 BH), '[1 w' (5BH, 31 H, 77H) and the string end character 17H are defaults. This printer then prints normal print with 17 characters per inch.

Print type 2:
For the PT88 printer, the control characters (0DH), ESC (1 BH), '[2w' (5BH, 32H, 77H) and the string end character 17H are defaults. This printer then prints condensed print with 12 characters per inch.

Print type 3:
For the PT88 printer, the control characters (0DH), ESC (1 BH), '[4w' (5BH, 34H, 77H) and the string end character 17H are defaults. This printer then prints super-condensed print with 10 characters per inch.

ASCII :
For the PT88 printer, the control characters ESC (1 BH), '(B' (28H, 42H) and the string end character 17H are defaults, This printer then prints with the ASCII character set.

Expanded on:
For the PT88 printer, the control characters ESC (1 BH), '8' (38H) and the string end character 17H are defaults. Expanded print is then set on this printer. Each character is then printed in double width.

Expanded off:
For the PT88 printer, the control characters ESC (1 BH), '<' (3CH) and the string end character 17H are defaults. This switches off expanded print on this printer. Each character is then once again printed in normal width.

For IBM or EPSON printers please refer to the control characters in the appropriate manual.

5.6.2 Entering Machining Programs

5.6.2.1 General Information about Machining Programs

The structure of the machining program generally corresponds to a subset of the representation described inDIN66025, The programs consist of a sequence of ASCI I characters with a maximum length of 1023 characters.

Machining programs are packaged by COM247 in data blocks in keeping with the STEP 5 representation, The blocks are distinguished by their data block number. A data block generated by COM247 contains exactly one machining program. The data block number is **entered in the machining program header by COM247 as the machining program number. Numbers 0...255 are possible.**

For more information about machining programs, refer to Section 2.6 "Machining Programs and their Structure".

Machining programs can be generated in two methods of representation:

- representation according to DIN 66025
- representation in text mode

It is also possible to generate machining programs using TEACH-1 N. The test mode of COM247 provides the necessary support. (See also Section 5.8 "Test" or Section 4.3.4 "Teach-in On/Off").

5.6.2.2 Generating Machining Programs

If you select "MACHINING PROGRAM" in the data block selection display, specify the block number and press one of the function keys <F1>...<F3>, the first display for machining programs is output. The destination device (I P247, PG, FD) is now fixed and cannot be changed for this input.

In the following machining program displays examples of data for machining programs have been entered, The destination device is the drive (FD) selected in the presets display, The data is stored in the set file as DB1 55.

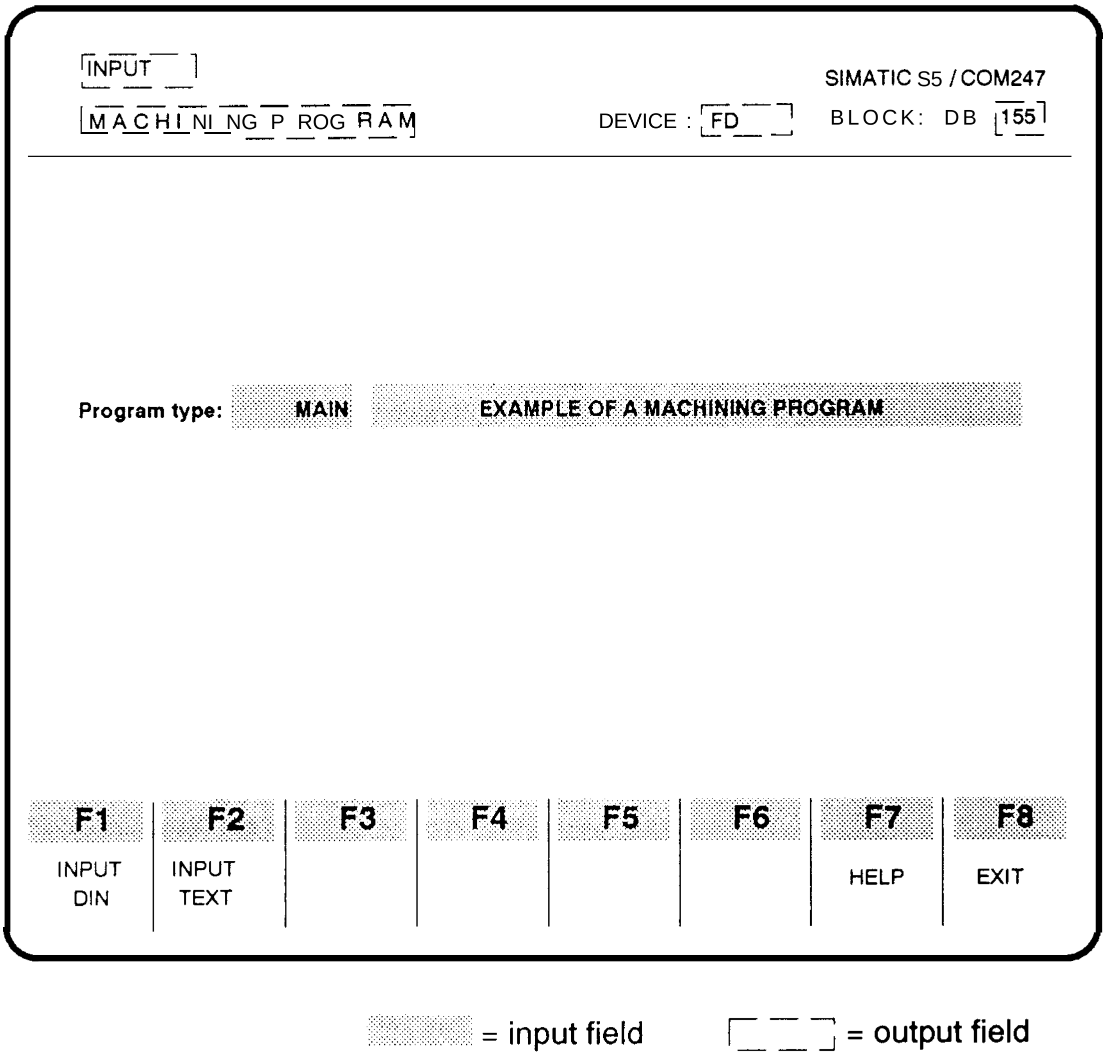


Fig. 5/1 5 Machining program display

Description of the output fields

INPUT is displayed in the header and M A C H I N I N G P R O G R A M is displayed in output field 2. DEVICE displays the previously selected target device and BLOCK the previously selected DB no.

Description of the input fields

In the first input field “program type” you can select between a main program (“?4.”) and a sub-routine (“L”). The permitted entries are MAIN and SUB.

In the next input field, you can enter a comment, e.g. to provide information about the machining program.

Significance of the function keys

<Fl >: With this function key you branch to input of machining programs according to DIN.
<F2>: With this function key you branch to input of machining programs in the TEXT mode.
<F7>: With the HELP key you can select the type of program (MAIN or SUB).

<F8>: If you press the EXIT key, you will be prompted to confirm abandoning the machining program, if you answer with YES you return to the basic display (function selection) and if you answer NO you continue with machining program input.

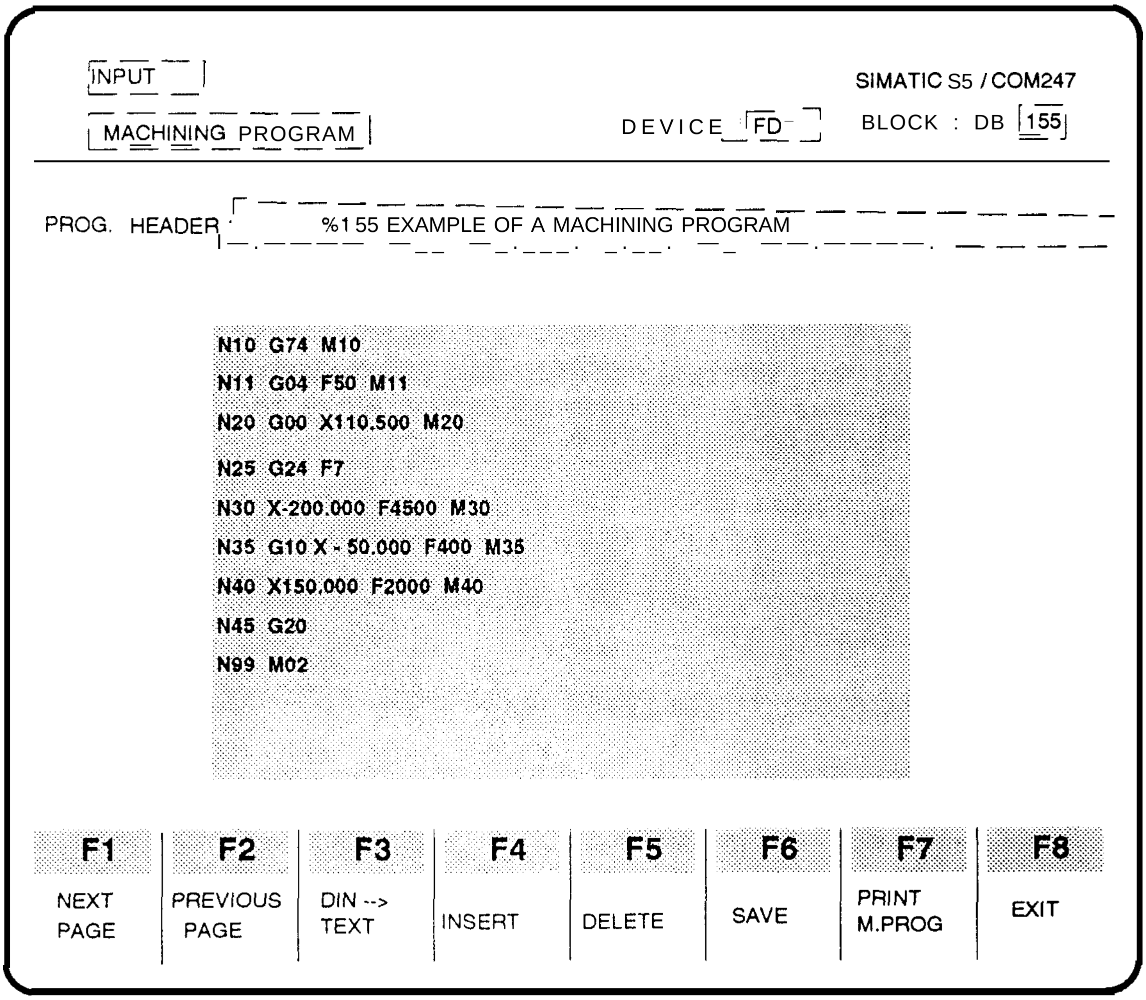
5.6.2.3 Entering Machining Programs according to DIN

In the DIN representation, only one statement of a traversing program can be written per line. Each statement must begin with the statement type and statement number.

The IP247 processes all statements as “normal statements”. Normal statements are identified by “N”, The statement identifiers “/N” for “suppressable statements” and ":N" for “main statement” are permitted, however, they have no significance.

The statement number comprises a maximum three digit number. The range of values is 0,..999. Apart from the N-function (statement type and statement number) the following functions are permitted:

- L-function (subroutine call)
- G-function (preparatory positioning condition)
- X-function (target function)
- F-function (speed, time, loop repetitions)
- M-function (auxiliary function)



.. = input field [ ] = output field

Fig. 5/1 6 Machining program display according to DIN

Description of the output fields

The header is as described in the section “Generating Machining Programs,

The program type (“%" for main program or “L” for subroutine), the program number and any comments are output in the output field PROG HEADER.

Description of the input fields

You enter the statements of the machining program in the individual lines. Once you reach the last line, the machining program is scrolled one line upwards, i.e. a new page is begun. The previous page is displayed again if you press <F2>.

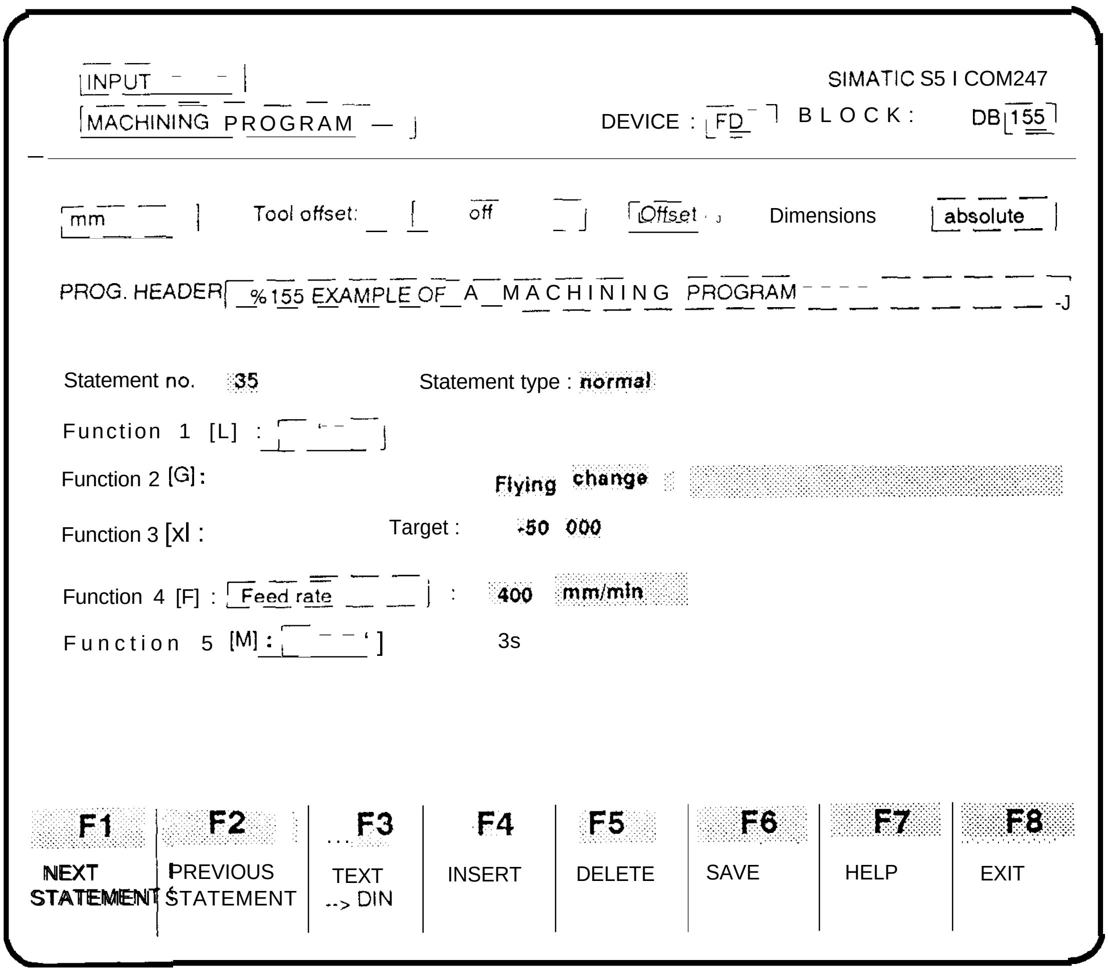
Significance of the function keys

<Fl >: This function key is used to page forwards when the machining program is longer than one page and when you are not on the last page.

<F2>: Analogous to <Fl >, this function key is used to page backwards.
<F3>: With this key, you can switch to text representation.
<F4>: This function key inserts a line in front of the current cursor position.
<F5>: This function key deletes the line marked by the cursor.
<F6>: If the machining program is syntactically correct, you save the program on the previously selected device under the specified DB number with this key. If the program already exists on this device and with this DB number, you will be asked whether or not to overwrite the data block.
<F7>: The machining program is printed out if you press this key. The displays are the same as those for printing machine data (cf. Section 5.6.1.3 “Print Machine Data”), When machining programs are being printed out, “MACHINING PROGRAM DIN” is displayed in the third output field of the header.
<F8>: With the exit key, you exit the input function without saving the data.

5.6.2.4 Entering Machining Programs in the Text Mode

In the text mode, only one statement is displayed on the screen. The tYPe of statement and the G function can be selected using the HELP key <F7>. With the other functions, you must enter the appropriate numerical values.



= input field = output field

Fig, 5/1 7 Machining program display in the text mode

Description of the output fields

The header is as described in Section 5.6.2.2 “Generating Machining Programs”.

Meas. system:
The measuring system in which the displayed statement is to be interpreted can be seen in the first output field after the header, The default unit is mm. Alternatively, “0.1 in” will be displayed (G70 or G71).

Tool offset:
The current tool length offset is displayed here. Possible displays are “off” (G40), “negative” (G44) and “positive” (G43). “Off” is the default. Only the last selected G-function (G40, G43 or G44) is displayed, not the resulting tool length offset. A change of sign, for example, G44 with -10 mm is not taken into account.

Offset:
The last selected offset is displayed. Possible displays are "undef" (G53), "1 on” (G54), "2 on” (G55), "3 on” (G56) and”4 on” (G57). "Undef" is the default.

Dimensions:
The numerical values of the target functions (X functions) can be specified in “absolute” (G90) form or in “incremental” (G91) form. “Absolute” is the default.

Description of the input fields

Statement no.:
The statement number is entered here as a numerical value. The statement number can be up to three digits long. It is not necessary to enter statements in ascending order. The individual statements are processed in the order in which they are entered, regardless of the statement number.

Statement type:
Using the help key, you can select one of the three possible statement types “main” (’:N’), “normal” (’N’) and “’suppressible” (’/N’).

L-function:
Here you enter the subroutine number to be called by the current program. Once you have entered a subroutine number, the text “subroutine no.” is displayed before the input field.

G-function:
Using the help key, you can display the possible G-functions. Entry of other values is rejected as an error.

X-function:
Here, you enter the target. The maximum range is +/- 99999.999. The value is either interpreted as a distance (with G91) or as an absolute coordinate (with G90).

F-function:
Depending on the previous functions, either the feed rate (following an X-function), a dwell time (following G04) or the number of repetitions (following G24) is entered. The appropriate text (“feed rate“, “loop run through” or “dwell time”) is displayed before this input field and the appropriate dimension after the input field.

M-function:

The M-function is output at the beginning of the statement. The M-function MOO means “programmed halt”, the M-function M02 means program end. After M02, no further statements can be appended. After entering M02, the text “program end” is displayed before this input field and the text “program halt” is displayed after entering MOO.

Significance of the function keys

<F1>: With this function key you can display the next statement if the machining program is longer than one statement and the last statement is not currently displayed.

<F2>: Analogous to <F1>, you can page back one statement.

<F3>: With this key you can switch over to text representation.

<F4>: This function key inserts a new statement before the statement currently displayed.

<F5>: This function key deletes the displayed statement.

<F6>: If the machining program is syntactically correct, you save the program on the previously selected device under the specified DB number with this key. If the program already exists on this device under this DB number, *you* will be asked whether or not to overwrite the data block.

<F7>: With the HELP key you can select the alternatives for the fields “statement type” and “G function”,

<F8>: If you press the EXIT key you exit the input function without saving the data.

5.7 output

By pressing <F2> (OUTPUT) in the basic display ("function selection") you branch to the "OUTPUT" function. Here, machine data or machining programs can be output from the module, the PG or from a floppy/hard disk drive. It is then possible to change the data and write it back to the source. The function keys <F1>...<F8> have the same assignment as in input. **The displays are also identical with the exception of the header.** Here, OUTPUT is displayed instead of INPUT.

The first display is the block selection display. Here, you must select the data block you wish to display. You can select machine data or machining programs using the function key <F7> (HELP). After specifying

- the block number,
- the axis number (only for machine data output) and
- the source device (with <F1>... <F3>) from which the data block is to be read,

the first machine data/machining program display will appear.

5.7.1 Output Machine Data

Description of the output fields

OUTPUT and M A C H I N E D A T A are displayed in the header. The previously selected source device is displayed in DEVICE and the DB no. in BLOCK.

Description of the input fields

The input fields for machine data are completed with the stored values. You can modify the data and write it back to the source device with the function key <F6> (TRANSFER). Otherwise the display is the same as for the input of machine data.

5.7.2 Output Machining Program

Description of the output fields

OUTPUT and MACHINING PROGRAM are displayed in the header. The previously selected source device is displayed in DEVICE and the DB no. in BLOCK.

Description of the input fields

The machining program of the selected data block is displayed. You can change individual statements and to write them back to the source device with the function key <F6> (SAVE). Otherwise, the display is the same as for the input of machining programs.

5.8 Test

In this branch of the program you can test the IP247 module and the drive in all operating modes. Machining programs can be started manually and already existing machining programs can be tested. Actual values are displayed at the PG online. The test mode also allows machining programs to be generated in TEACH-IN

> **Note**
>
> Owing to a hardware feature of the programmers, the keys have a repeat function. If you press a key for a longer time, its key code will be written to the key buffer until it is processed. This means that commands will be processed until the buffer is empty. A STOP command may, therefore, not be executed immediately. This is not an error; the stop command will still be executed. Commands entered accidentally after the stop command will, however, also be executed.
> In the test mode, an emergency stop switch must be easily accessible from the PG.

5.8.1 Starting the Test Mode

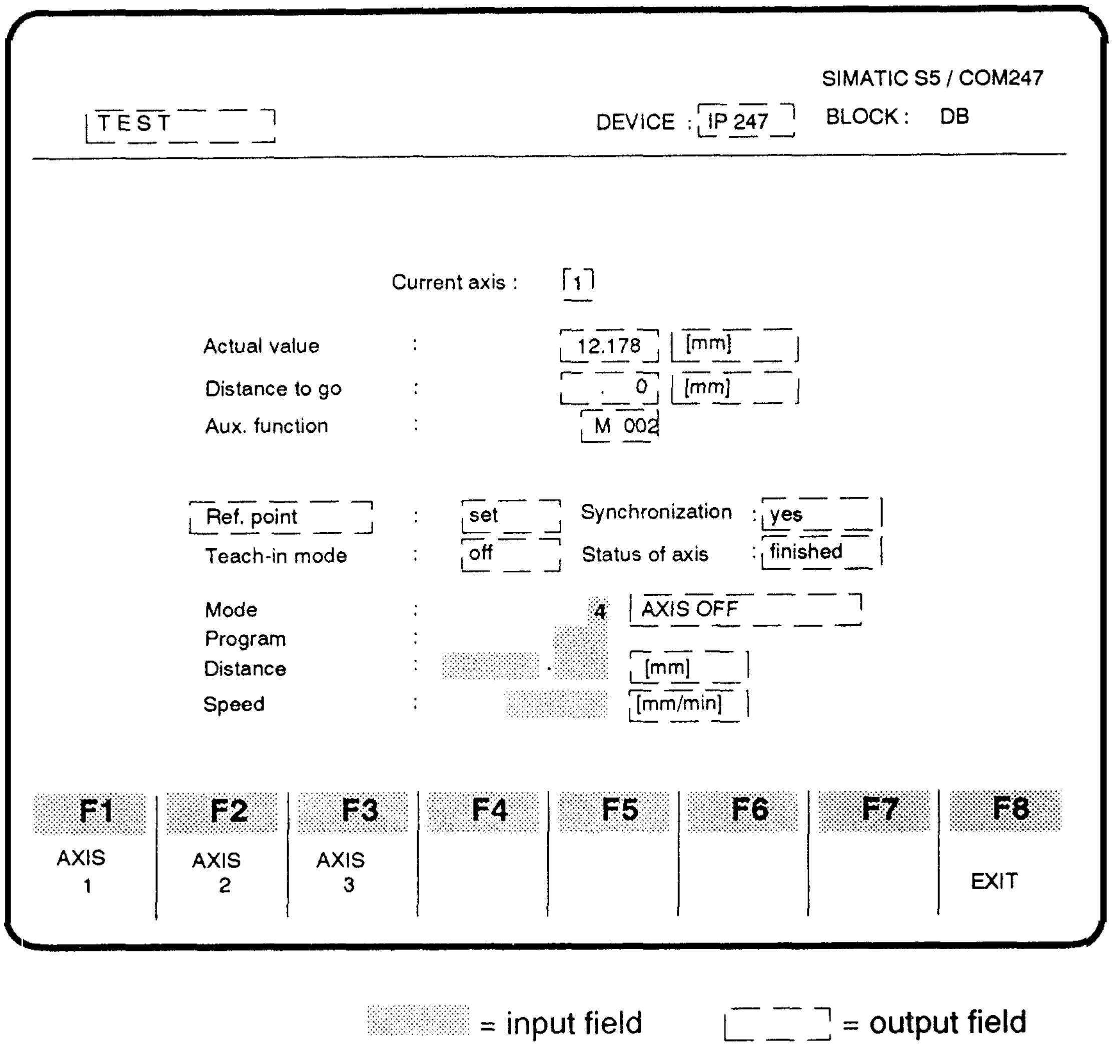
By pressing <F3> (TEST) in the basic display (function selection) you branch to the “test” function.

Conditions:

- The mode is online. You can change the mode in the presets display using the function key <F2> (ONLINE-OFFLINE).
- The link from the PG to the IP247 is established and
- the I P247 is operational (green LED lit).

In this section, each display has values entered in the input and output fields. The dimensional unit is in mm.

The first display in “test” is the test axis selection display,



Fig, 5/1 8 Test axis selection

Description of the output fields

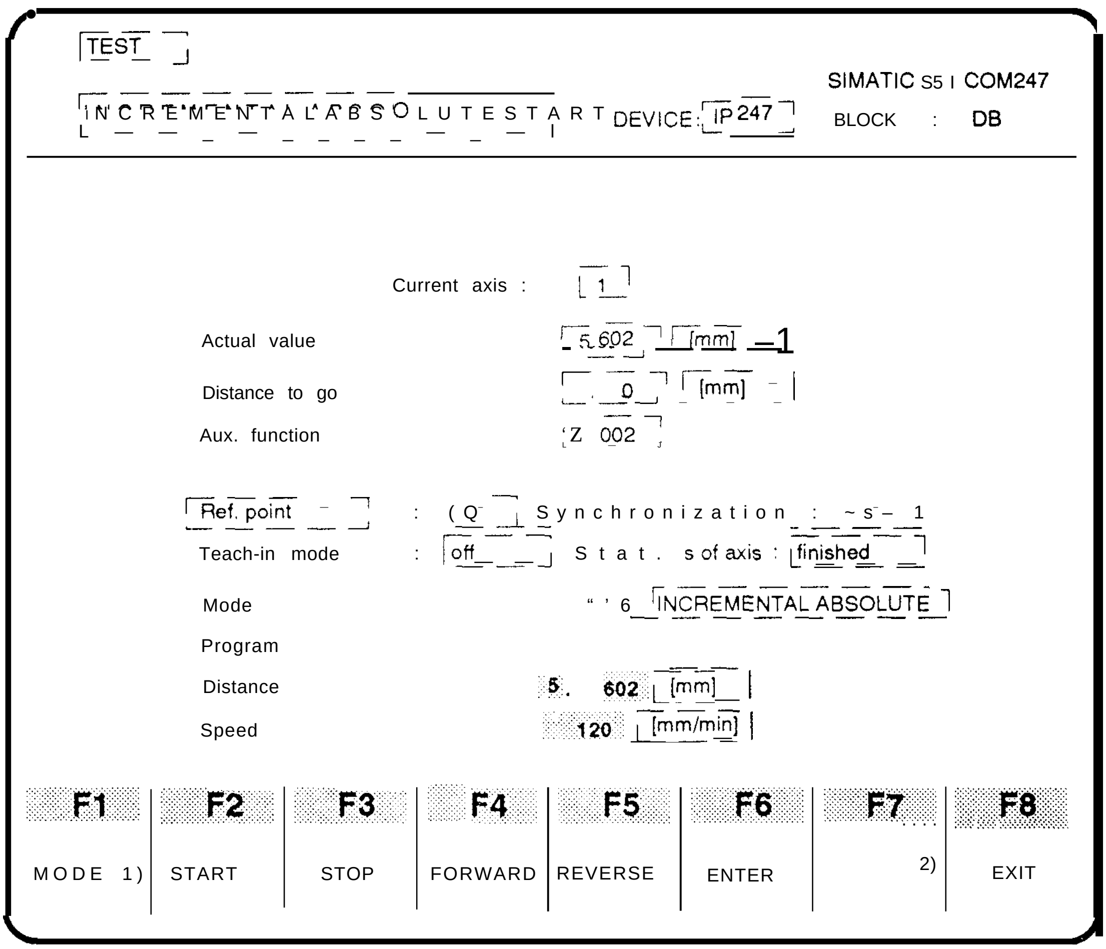
T EST is entered in the header.”1 P247° is displayed in the DEVICE output field.

Significance of the function keys

<Fl >:	Test axis 1. After pressing this key, the program branches to the mode display,
<F2>:	Test axis 2. After pressing this key, the program branches to the **mode display.**
<F3>:	**Test axis 3.** After pressing this key, the program branches to the mode display.
<F8>:	The EXIT key returns you to the basic display (“function selection”),

5.8.2 Modes

If you select the axis with <F1> (axis 1), <F2> (axis 2) or <F3> (axis 3) in the test axis selection display, you branch to the mode display, to the actual value display mode.



1) ACTUAL VALUES appears if you press this key

2) HELP appears here in the operating mode display

= input field = output field

Fig. 5/1 9 Mode display

Description of the output fields

"IP247" is displayed in the output field DEVICE. The remaining text in the header depends on previous entries. Either simply "TEST" is entered or "TEST" and the current operating mode of the axis and the last command to be executed are displayed.

Note If you return to the basic display ("function selection") by pressing function key <F8> (EXIT) in the test axis selection display, COM247 starts mode 17 ("clear error"). This is then displayed in the mode display in output field 3 in the header,

Actual value:
The current position coordinate (actual position value) of the selected axis is displayed. The value is displayed in the appropriate dimension.

Distance to go:
This displays the difference between the actual position value and the target coordinate. This only applies to modes “AUTOMATIC” and “AUTOMATIC SINGLE STATEMENT”,”1 N-CREMENTALABSOLUTE"and" I NCREMENTALRELATIVE".

Aux. function:
Inthemodes “AUTOMATIC” and “AUTOMATIC SINGLE STATEMENT” the programmed M function is displayed as a numerical value. The auxiliary function (M function) M02 is the default.

Reference point:
This field shows whether the reference point is “set” or “cleared” (not set).

Synchronization:
The possible displays here are “yes” or “no”. Synchronization yes means that the reference point is determined by the negative edge of the reference signal and the excitation pattern counter equals O. Otherwise, the reference point is only determined by the negative edge.

Teach-in mode:
This displays whether the selected axis is in teach-in (“on”) or not (“off”).

Status of axis:
Here, the status of the axis is displayed. Possible displays are “finished” and “running”.

If there are no correct machine data for the axis on the module, then no actual value and no distance to go will be displayed. The statuses reference point, synchronization, teach-in mode and axis status remain unchanged. Without machine data, only mode 17 (clear error) can be executed apart from input and output of data.

Description **of the input** fields **for changing modes**

The inversely displayed input fields are only completed after pressing function key **<F1>** (MODE). If you press <F1>, the display changes to the change mode function.

Note

The displayed values, actual value (actual position value), distance to go, auxiliary function and the displayed axis attributes (=> Section 2.7 “Axis Attributes”), are then no longer updated.

The function key **<F1>** changes to “ACTUAL VALUES”, the function key **<F7>** changes to “HELP”. By pressing <F1> **(ACTUAL VALUES)** you return to the actual value display mode.

Mode:

Here, the required mode and selected axis are entered. You can select the mode from the mode table with the HELP key <F7>. After you have entered the mode number (right-justified), the corresponding text is displayed to the right of the mode number. Modes 1,.,17 are permitted. If a different number is specified, you branch automatically to the mode table, Depending on the selected mode, the function keys <F2>...<F6> have different functions,

Program:

You can only write to this input field in modes 8 (AUTOMATIC), 9 (AUTOMATIC SINGLE STATEMENT) and 10 (TEACH-1 N ON). In these cases, you must enter the machining program number,

Distance:

You can only write to the distance input field in modes 6 (INCREMENTAL ABSOLUTE), 7 (INCREMENTAL RELATIVE), 12 (ZERO OFFSET ABSOLUTE), 13 (ZERO OFFSET RELATIVE) and 15 (TOOL LENGTH OFFSET). In each case you must enter the distance or coordinate in the selected dimension.

Speed:

In this field, you can enter the start speed within the range 1,,.65000, Value O selects the value stored in the machine data. If you enter a value greater than the maximum speed, the value of the maximum speed is set.

Significance of the function keys

<F1>: With <F1>, you can switch between the actual value display mode and the mode change function. In the actual value display mode the values actual value, distance to go, aux. function and the displayed axis attributes of the selected axis are displayed and continuously updated. If you select the mode change function the mode and corresponding parameters can be changed.

<F2>: "START" is only permitted in modes 4..,6,8...12 and 14...17.

<F3>: "STOP" is only permitted in modes 1,2 and 6...9.

<F4>: The command "FORWARD" is permitted in modes 12,7, 13 and 15. If a rotary axis is selected as the axis type, the "FORWARD" command is also permitted in mode 6.

<F5>: The command "REVERSE" is permitted in modes 12,7, 13 and 15. If a rotary axis is selected as the axis type, the "REVERSE" command is also permitted in mode 6.

<F6>: The function of the "ENTER" key depends on the mode.

In the "AUTOMATIC SINGLE STATEMENT" mode, this key executes the next statement of an automatic program.

In the "AUTOMATIC" mode, the "ENTER" key is used to acknowledge a programmed halt (MOO).

I nthe "AUTOMATIC" and "AUTOMATIC SINGLE STATEMENT" modes, this key continues an interrupted machining program.

If the teach-in mode is active and the axis status is "finished", the "ENTER" key is used to save a statement.

<F7>: Providing the cursor is located in the input field "mode", the HELP key <F7> can be used to branch to the mode table. From this table you can select a mode and return to the mode display with <F6> (ENTER).

<F8>: The EXIT key returns you to the test axis selection display. The header remains unchanged.

5.8.3 Mode Table

By pressing the HELP key in the mode display (mode change function), you can branch to the mode table display. This contains all the possible modes and mode numbers.

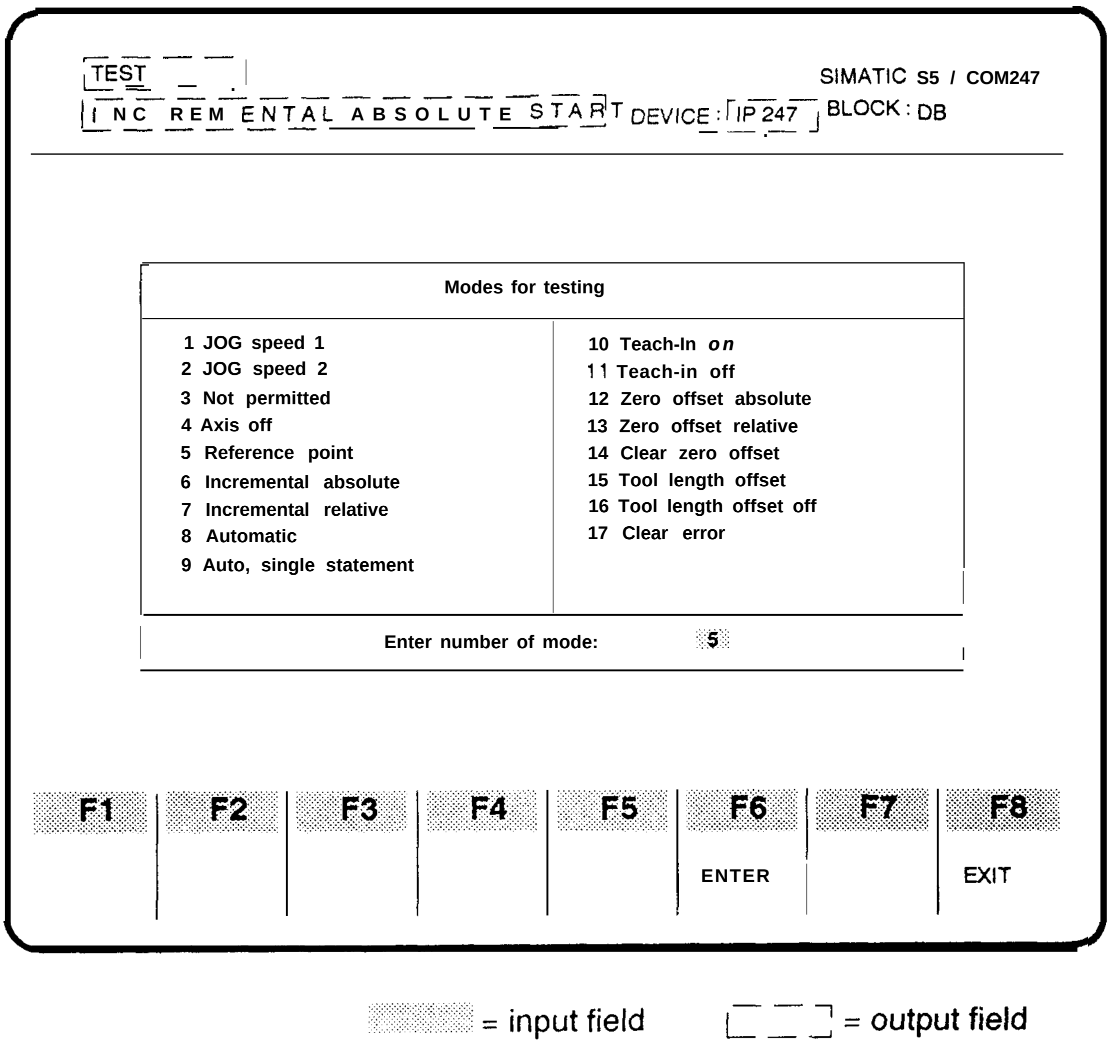


Fig. 5/20 Mode table

Description of the output fields

The header remains unchanged.

Description of the input field

The display has only one input field in which you can enter the number of the required mode. All other values apart from those listed lead to the error message “FFF illegal input”.

Significance of the function keys

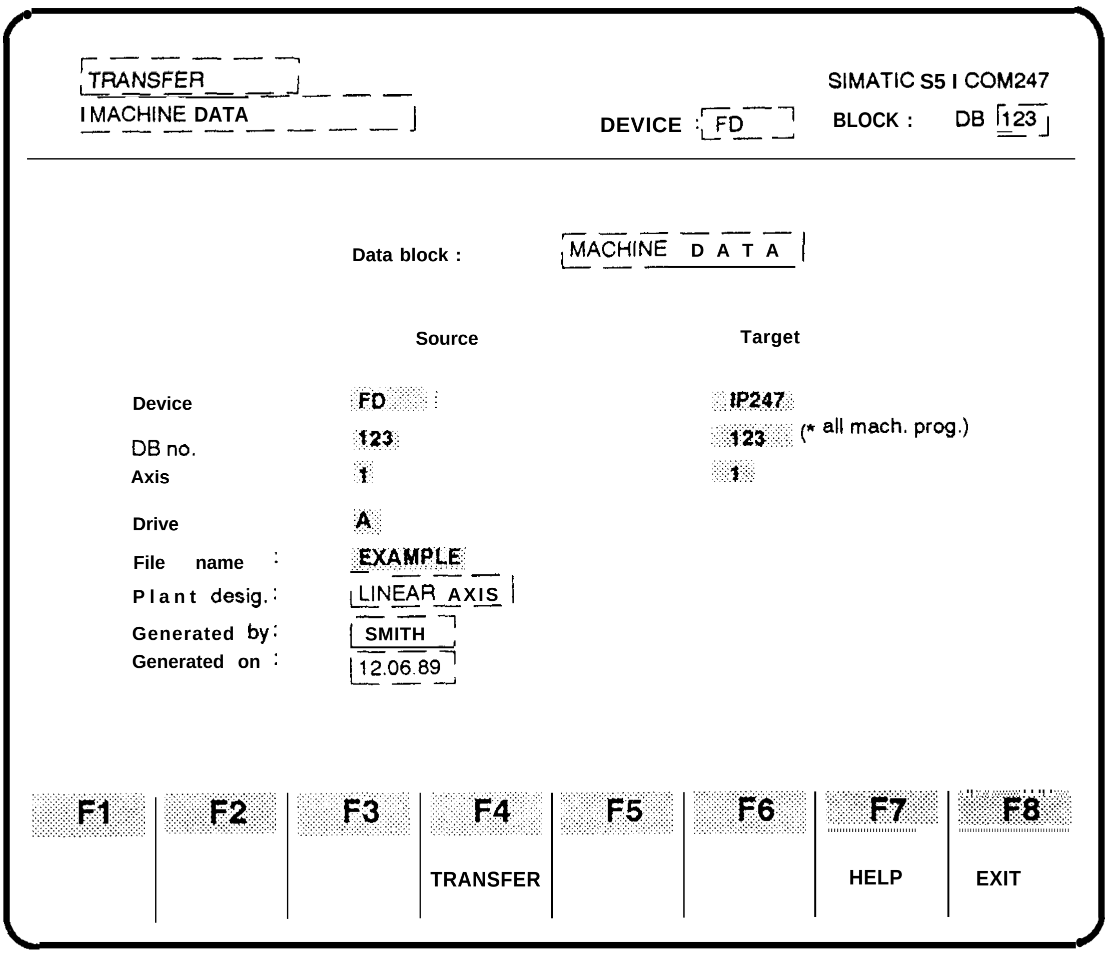
<F6>: Using the ENTER key the mode number is entered in the “mode” field of the mode display. The corresponding text is displayed at the same time.

<F8>: The EXIT key returns you to the mode display (mode change function). AXIS OFF is then entered as the mode.

5.9 Transfer

By pressing <F4> (TRANSFER) in the basic display you branch to the transfer display.

In this branch of the program you can transfer machine data or machining programs from one device to another.



= input field = output field

Fig. 5/21 Transfer display

Description of the output fields

Following the start of the transfer, TRANSFER and either M A C H I N I N G P R O GRAM or M A C H I N E DATA are entered in the header. The output fields BLOCK and DEVICE are completed with the appropriate values. The DB number of the data record to be transferred is displayed in BLOCK and the source device is entered in DEVICE.

Description **of the input fields**

Data block:
With <F7> (HELP) you can select between machine data and machining programs.

Device:
With <F7> (HELP) you can select the source or destination device. The possible entries are module (I P247), the programmer (PG) or the data drive (FD).

DB no.:
As the source, you must specify the number of the block to be transferred. The same number will be entered as the target. This can, however, be changed. The range of values is from 0,.,255. If a “*” is entered for the DB no. of the source, all DBs (either machine data or machining programs) on the selected source device will be transferred to the destination device. The DB number of the destination device is then meaningless.

> **Note**
>
> The PG can only store one machine data record and one machining program.

Axis:
When transferring machine data, you must enter the axis number in the machine data record to be transferred under source. If there is no machine data record with this axis number on the selected device, the error message “data block does not exist” is displayed. When transferring the machine data, it is possible to change the axis number of the target DB.

Drive:
If the selected *source* device is a floppy or hard disk the name of the drive is also required. You can use the HELP key <F7> to make the entry.

File name:
If the source is a drive (floppy or hard disk) the file name must be selected using the help key. If there is no file on the selected drive with the extension .247 no DB can be transferred, In this case the error message “data block does not exist” is displayed. If the target device is a drive, the file name selected in the presets display will be used and the corresponding data for “plant designation”, “generated on” and “generated by” will be displayed in the appropriate fields. You can only transfer to the drive selected in the presets display.

Significance of the function keys

<F4>: This function key starts the transfer.

<F7>: With the HELP key, you make selections in the “block number”, source device and destination device fields. Remember **that no blocks can be transferred if the source and destination device are identical.** Otherwise, possible source drives and the files contained can be selected.

<F8>: Pressing this function key abandons the “TRANSFER” function and you return to the basic display (“function selection”).

5.10 Delete

Pressing <F5> (DELETE) in the basic display ("function selection") branches to the delete display.

In **this program branch you can delete machine data or machining programs stored on a device (IP247, FD, PG).**

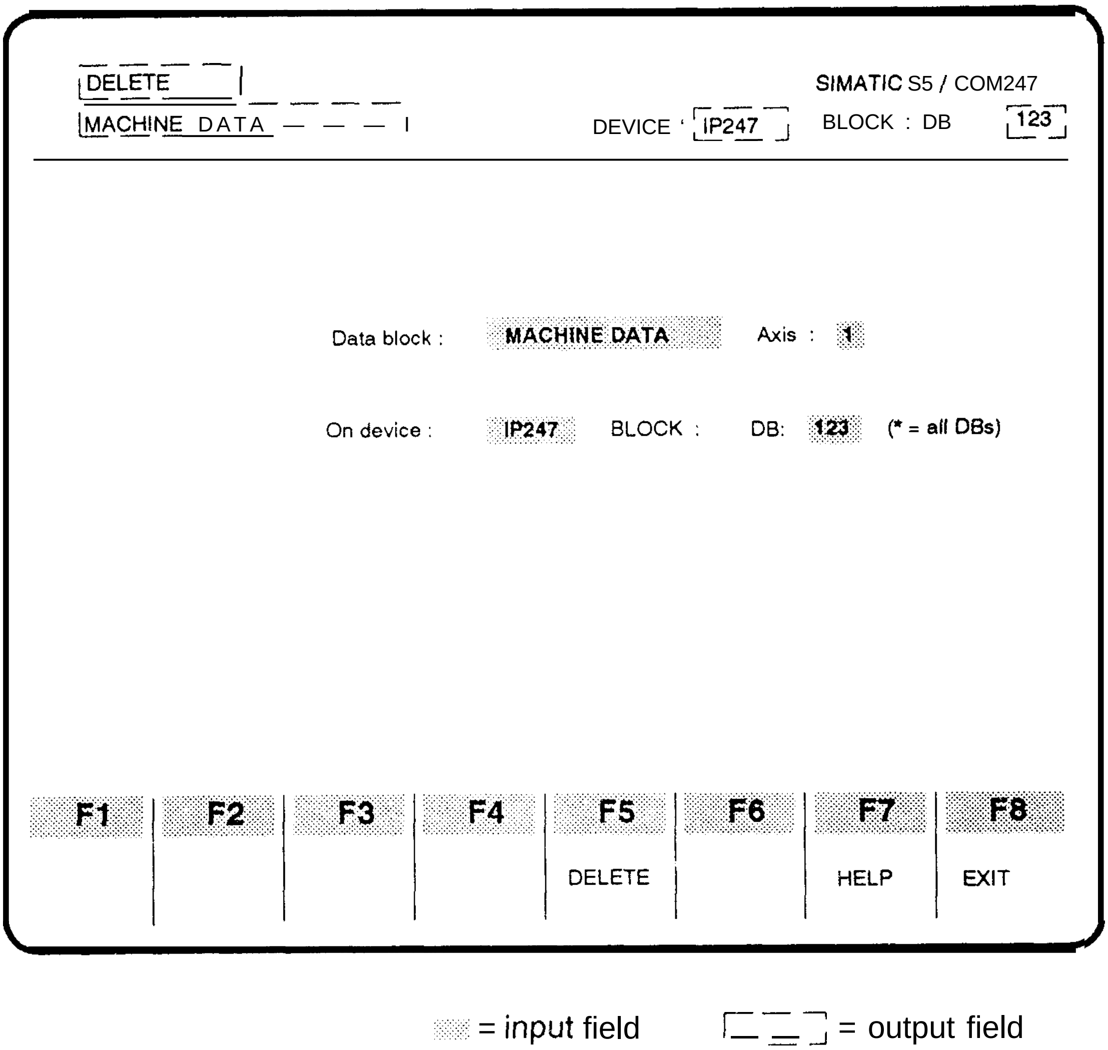


Fig. 5/22 Delete display

Description of the output fields

After starting the delete function, DELETE and either MACHINING PROGRAM or MACHINE DATA are displayed in the header. The output fields BLOCK and DEVICE are then completed with the appropriate entries. The DB number of the data record to be deleted is displayed in BLOCK and the device containing the data record to be deleted is displayed in DEVICE.

Description of the input fields

Data block:
With <F7> **(HELP),** you select between machine data and machining programs.

Axis:
If machine data are to be deleted, you must enter the axis number of the machine data record to be deleted here. This number is stored in the machine data record.

On device:
With <F7> (HELP) you can select the device on which the data block is to be deleted. Possible selections are module (1 P247), the programmer (PG) or the data drive (FD).

Block:
The number of the block to be deleted is entered here. The range of values is 0...255. If you enter "*" all DBs (machine data records or machining programs) on the selected device will be deleted.

Significance of the function keys

<F5>: This function key starts the delete function.
<F7>: Using the help key you can select alternatives in the "data block" and "source device" fields.
<F8>: With this function key you abandon the delete function and return to the basic menu.

5.11 Information

If you press <F7> **(INFORMATION)** in the basic display ("function selection") you branch to the information display,

In this program branch, you can obtain an overview of all the machine data or machining programs stored on a device (1 P247, **PG,** FD). One screen page can list a maximum of 48 entries. If there are more than 48 machine data records or more than 48 machining programs stored on a device, you can page forwards or backwards. The assignment of the function keys is automatically changed <F1> (NEXT PAGE) and <F2> (PREVIOUS PAGE). These two keys can be used to page forwards or backwards until there are no further entries to be listed. In this case, the message "no more entries" is displayed.

After selecting the source device with the keys <F1>···<F3>, the data blocks are displayed with their DB number and length. With machine data, the axis number for which the machine data record is intended is also displayed. By pressing <F8> (EXIT) you return to the basic display (function selection).

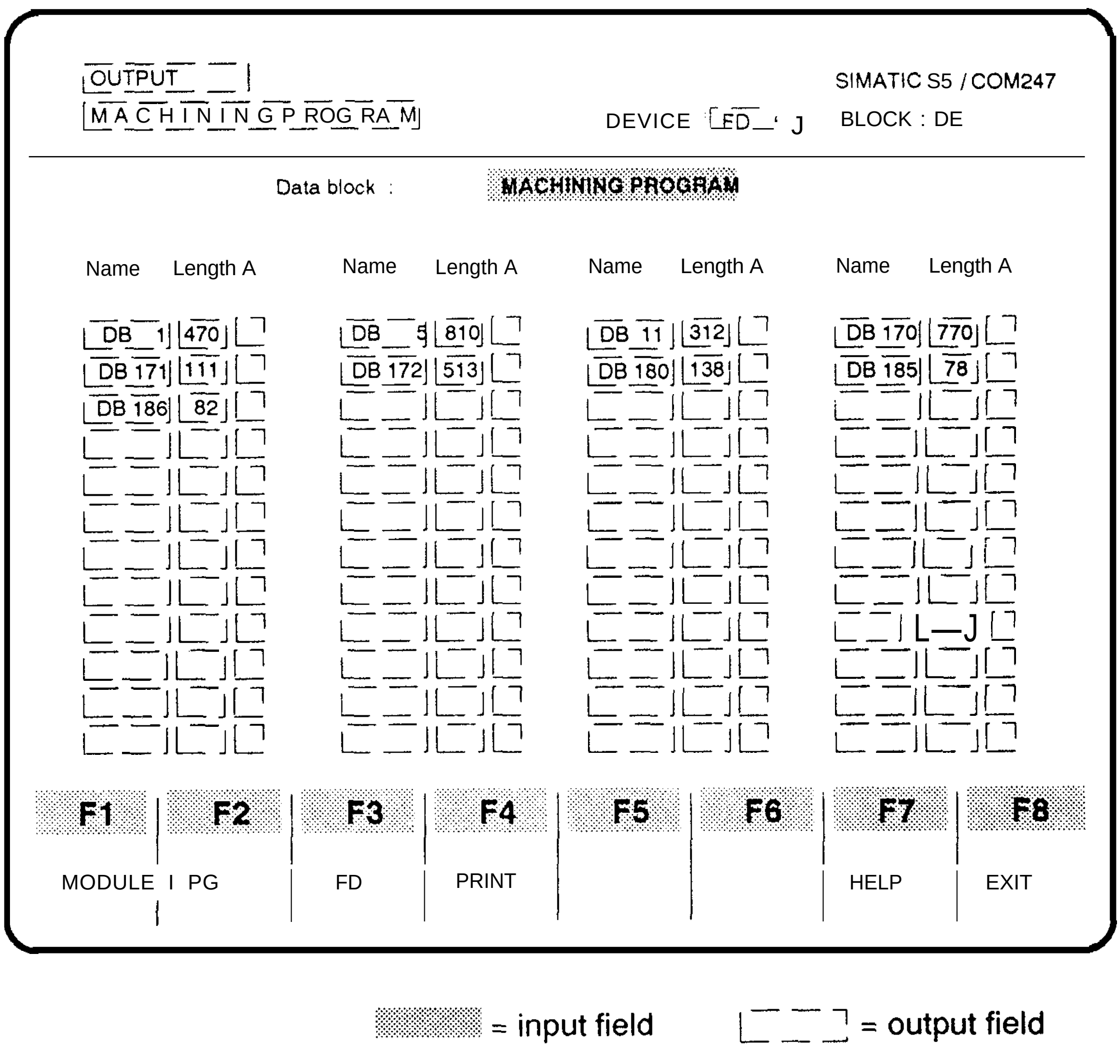


Fig. 5/23 Information display

Description of the output fields

After starting the information function, INFORMATION and either MACHINING PROGRAM or MACHI NE DATA is entered in the header. The source of the data is displayed in the DEVICE output field, The BLOCK output field remains unchanged.

Description of the input fields

Data block:
Using **<F7> (HELP),** you can select between information about machine data or information about machining programs.

Significance of the function keys

<F1> ..<F3>:
With these function keys you select the devices (1 **P247, FD** or PG) from which the data blocks are to be read. If there are more than 48 entries <F1> and <F2> are assigned the paging function.

<F4>: This starts the printout of the data blocks.

<F7>: With the HELP key you select the type of data block (machine data or machining programs).

<F8>: With the EXIT key you can exit the information function.

6 Standard Function Blocks FB164 and FB165

6.1 General Notes

6.1.1 Overview

This part describes the two standard function blocks

FBI 64 (PER:POS) “operating and monitoring the positioning module” and
FBI 65 (PER:PDAT) “positioning module parameter assignment”

FBI 64 is used to operate and monitor the IP247 positioning module. With FBI 64 you can start the IP247 operating modes BA1 - BA17from the user program. FB164 also supplies constantly updated information about the current status of an axis (errors, M-functions, ...). As soon as one of the monitoring modes is started, the selected value is read and output cyclically by FBI 64.

FBI 65 is used to assign parameters to the IP247. It is responsible for the data exchange between your user program and the IP247. By calling FB165, you can execute the following functions via the PC interface:

- read machine data and machining programs from the I P247, delete them and transfer them to the IP247
- read the system identification from the IP247 and transfer it to the IP247,
- request an overview of the machine data or machining programs stored on the IP247 and
- read actual values (actual position value, distance to go).

The function blocks FBI 64 andFB165 are used in the following programmable controllers

S5-115U (CPU 941 to CPU 944)
S5-135U (CPU 922 and CPU 928)
S5-150U
S5-155U

in conjunction with the following IP247 positioning modules

6ES5247-4UA31 (for ventilated operation)
6ES5 247-4UA41 (for non-ventilated operation)

This User’s Guide assumes that you are familiar with the IP247 and the programmable controller.

The function blocks FBI 64 and FBI 65 are supplied on the diskette with one example under one of the following file names:

S5-115U all CPUs	:	S5TA50ST.S5D
S5-135U CPU922/928	:	S5TB22ST.S5D
S5-150U	:	S5TA40ST.S5D
S5-155U	:	S5TA60ST.S5D

6.1.2 Notes

The IP247 positioning module is addressed by means of pages. It has three positioning axes and a data channel and therefore requires four page addresses.

Function **block FBI 64 must be called once for each axis.**
Function block FB165 can be called conditionally.
Calls in the processor time interrupt OBS are not permitted.

The function blocks FBI 64 and FBI 65 operate with the data handling blocks SEND and RECEIVE, FB165 also requires the FETCH data handling block.

The handling blocks are (automatically) assigned parameters and called by the FBs. The pages must beset up in the start-up OB (OB20, OB21 and OB22 or OB21 and OB22 for the S5-1 15U) with the SYNCHRON data handling block (FB1 25, FB185 and FB249).

6.1.2.1 **Overview of the Data Handling Blocks**

	S5-11 5U	S5-135U	S5-150U	S5-155U	
SYNCHRON	FB249	FB125	FB185	FB125	
SEND	FB244	FB120	FB180	FB120	
RECEIVE	FB245	FB121	FB181	FB121	
FETCH	FB246	FB122	FB182	FB122	Only **for** FB165

6.1.2.2 Installing an Interface in OB20, OB21 or OB22 with the S5-135U

```
         : JU FB125
NAME     :SYNCHRON
SSNR           KY0,2   Interface 2
BLGR           KY0,0   Block size
PAFE           FY1     SYNCH RON call - parameter assignment error
```

The SYNCHRON call must be programmed for each interface to be addressed in the cyclic program section (cf. Section 6.4 “Examples”).

In the BLGR parameter, you can select the length of the blocks of data to be transferred by FB165.

BLGR	S5-115	S5-135/155
0,0	64	128
0,1	32	32
0,2	32	32
0,3	64	64
0,4	128	128
...	...	...
...	...	...
0,255	128	128

6.1.2.3 Use of FB164/165 in the Various Programmable Controllers

When using the FB164/165 in programmable controllers, please note the points in the following table, when interrupting the user program and when starting the program.

	115U	135U	150U	155U
User program can be interrupted				
at:	Command boundaries	Block boundaries or command boundaries	Block boundaries	Block boundaries or command boundaries
When using the interrupt OBS the scratchpad flags must be saved and loaded again before exiting the interrupt OB	FY200 to FY255	FY200 to FY255	FY200 to FY255	work with FB38, 394)
		RS 60 to RS 63		
Calling handling blocks in interrupt branches	not permitted 1)	not permitted if interrupts are at command boundaries 1)	permitted	see S5-135U 1)
Start-up types				
Cold restart beginning of the cyclic processing 2)	OB21 at start of OB 1	OB20 at start of OB 1		
Automatic warm restart beginning of the cyclic processing	OB22 at start of OB 1	OB22 at interrupt point		
Manual warm restart		OB 21, at interrupt point		
FB164 call in start-up OB20 - OB22	not permitted	not permitted	not permitted in OB20 in OB21 and OB22 see note 3)	not permitted
Saving scratchpad flags and operating system data in OB21 and OB22		save FYB200 to FY255 4)		FB38, **39**
		RS60 to RS63		

1) If this is necessary, you must ensure that FB164 is not interrupted in the cyclic program.
2) FB164 should be run through once for each axis before the first operation job is sent, to allow the binary identifiers in the **DB** for each specific axis to be updated.
3) See S5-150U
4) See note on following page

Note

To save and load the scratchpad flag area you must use the standard function blocks FB38 and FB39. The function blocks operate with a data block (see example in Section 6.4, DB255). This must be created up to and including data word DW820. The function blocks must be used in pairs, i.e. the interrupt OBS must not be exited prematurely with the statement BEC.

6.1.3 Using the Positioning Module in Multiprocessor Operation (applies to the S5-135U and S5-155U)

If the positioning module is used in a programmable controller with more than one processor, you must ensure that an axis is only ever addressed by one CPU module.

Note

Access by several CPUS to the same axis is not permitted and leads to program errors,

6.2 The Standard Function Block FB164

6.2.1 Functional Description

The function block FBI 64 "operating positioning module" allows the following functions to be executed:

- Starting a job (modes 1 ...17) on the IP247 from the user program.
- Cyclic reading of the actual position value, or distance to go from the IP247. These values are output as BCD or binary numbers depending on the assignment of the BCD parameter,
- Constant reading of the set mode, the current M function, the checkback signals (=> Section 2.6 "Axis Attributes") and the module error from the assigned interface. These are available at the parameter outputs of the function block or in the axis data block.

You can assign parameters to the function block FBI 64 directly or indirectly. With direct parameter assignment the user data required to start a mode (BA 1...17) are at the inputs of the function block, With indirect parameter assignment, FB164 supplies the parameter values from the data block valid before its call.

For certain modes, specific job parameters are required. These must be stored in the axis data block as byte, word and double word parameters before the mode is started.

Before calling FB164, the axis data block must be set up and contain valid values.

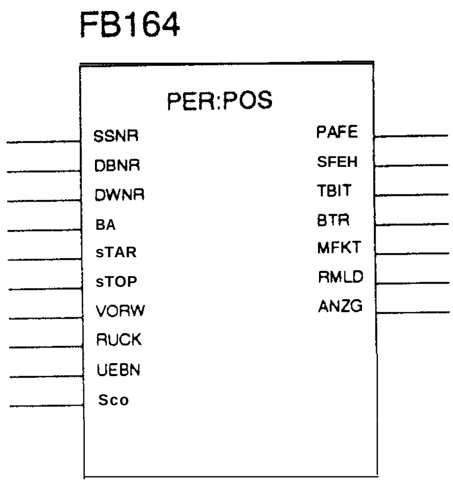
6.2.2 Calling Function Block FBI 64

6.2.2.1 S5-135U,S5-150U,S5-155U

In STL (Statement List):

```
            : JU FBI 64
NAME        : PER:POS
SSNR
DBNR
DWNR
BA
STAR
STOP
VORW
RUCK
UEBN
BCD
PAFE
BFEH
TBIT
BTR
MFKT
RMLD
ANZG
```

In LAD/CSF (Ladder Diagram or Control System Flowchart)

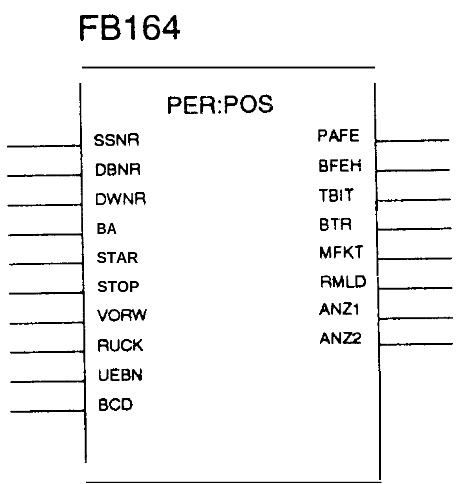


6.2.2.2 S5-115U

In STL (Statement List):

```
            :JUFB164
NAME        :PER:POS
SSNR        :
DBNR        :
DWNR        :
BA          .
STAR        :
STOP        :
VORW        :
RUCK        :
UEBN        :
BCD
PAFE        :
BFEH        :
TBIT        :
BTR
MFKT        :
RMLD        :
ANZ1        :
ANZ2        :
```

In LAD/CSF (Ladder Diagram or Control System Flowchart)



6.2.3 Overview of the Parameters

NAME	PARA TYPE	DATA TYPE	SIGNIFICANCE
SSNR	D	KF	Interface number
DBNR	D	KY	DB type, DB number (of axis data block)
DWNR	D	KF	First data word in axis data block
BA	D	KF	Mode (mode number)
STSR	I	BI	START command for the axis
STOP	I	BI	STOP command for the axis
VORW	I	BI	FORWARD command for the axis
RUCK	I	BI	REVERSE command for the axis
UEBN	I	BI	ENTER command for the axis
BCD	I	BI	Parameter ANZG in BCD ('1 or binary ('O')
PAFE	Q	BI	Parameter assignment error
BFEH	Q	BI	Module error
TBIT	Q	BI	Active bit
BTR	Q	BY	Output of the mode set for the axis
MFKT	Q	BY	Output of the M function of the axis
RMLD	Q	BY	Output of checkback signals (axis attributes) from the axis
ANZG	Q	D	Output of the value of the selected monitoring job

FortheS5-115U, the parameters ANZ1 and ANZ2 correspond to the parameter ANZG.

ANZ1	Q		Output of the value of the selected monitoring job
ANZ2	Q		

6.2.4 Explanation of the Parameters

SSNR : D,KF x

Specification of the page number (cf. switch setting J64, Section 3.2 “Setting the Module Address”) of the corresponding axis.

x = interface (page number)
OS- x ≤ 255

DBNR : D, KY x,y

Specification of the data block type and the data block number of the axis data block. With the S5-1 15U andS5-150U programmable controllers, data block type DX cannot be programmed.

x = data block type
x = O : data block type DB
x >< O: data block type DX

Y = data block number
5 ≤ = y ≤ = 255 where x = O
1 ≤ = y ≤ = 255 where x O

Direct parameter assignment via the block parameters (axis data block)
y = 0
Indirect parameter assignment via the data block opened before the FB164 call

DWNR : D,KF x

Specification of the first data word in the axis data block.

x = first word
0< x ≤ 236
where: 5 ≤ parameter DBNR 183 and
166 <parameter DBNR 255

16< x ≤ 236
where: parameter DBNR = 164
(DB164 = working DB for
standard function block FB164)

48 ≤ x ≤ 236
where: parameter DBNR = 165
(DB165 = working DB for
standard function-block FB165)

BA : D,KF x

Specification of the mode or monitoring function to be started on the IP247.

x = mode (mode number) or number of the monitoring function
$1 \leq x \leq 17$ and
71 + 73 and
x =74 switch off monitoring

Job number	Mode
1	JOG speed 1
2	JOG speed 2
4	Axis off
5	Reference point (approach/set)
6	Incremental absolute
7	Incremental relative
8	Automatic
9	Automatic single statement
10	Teach-in on
11	Teach-in off
12	Zero offset absolute
13	Zero offset relative
14	Clear zero offset
15	Tool length offset
16	Tool length offset off
17	Clear error (module error)
	Monitoring functions
71	Actual position value
73	Distance to go
74	Interrupt the cyclic execution of the last monitoring job

STAR :	I,BI	These parameters represent the possible commands
STOP :	I,BI	and start the data transfer to the positioning
VORW :	I,BI	module on the positive-going edge O --> 1, recognized by FB164 by comparing the binary identifiers in the axis DB
RUCK :	I,BI	The following data from the axis data block
UEBN :	I,BI	aretransferred:

- byte parameters (DR n),
- word parameters (DW n+1) and
- double word parameters (DD n+3)

The modes 18 and 19 permitted by FB 164 are acknowledged negatively by the IP247 on all four pages, since these modes do not exist on the IP247.

On the data channel, the modes 1-16 are also negatively acknowledged.

Following a negative acknowledgement, the parameter PAFE is set in FBI 84. The SEND block writes "C1H" in its PAFE.

The monitoring function 72 permitted by FBI 64 is acknowledged negatively on all pages. The monitoring functions 71 and 73 are also negatively acknowledged on the data channel.

Here, the PAFE parameter is set following a negative acknowledgement. The RECEIVE block writes "C1H" in its PAFE.

BCD : I,BI

If the BCD parameter has the signal state"1", the variables actual position value and distance to go are converted to a seven digit BCD number with sign. If the signal state is "O", these values are output in binary.

In the BCD format a maximum +/- 9999999 (pm, 0.0001 in, 0.001 degrees) can be represented. If one of these limits is violated, the output value (parameter output ANZG or the corresponding data words) is output as a binary number.

PAFE : Q,BI

If an error is made in the parameter assignment, the PAFE parameter has the signal state"1".The error can be identified by the settings in

flag byte FY255

(=> Section 7.2 "Troubleshooting").

BFEH : Q,BI

The parameter BFEH (module error) has signal state"1" when the IP247 positioning module signals an error. The type of error can be identified from

flag byte FY254

(=> Section 7.2 "Troubleshooting").

TBIT : Q,BI

Active bit: The module is executing the transferred job (BA 1...BA 17).

The "active bit" is set by function block FBI 64 when a job (BA 1...BA 17) is transferred to the positioning module. After executing or abandoning the job, the active bit is reset by the IP247 (=> Section 6.2.6 "Relationship between the Parameter TBIT and the current Checkback Signals").

BTR : Q,BY

Output of the mode currently set on the IP247 module.

MFKT : Q,BY

During automatic operation (BA 8 and 9) the M functions programmed in the automatic program are output by the IP247 (=> Section 2.6.9 "The M Function"). In all other modes M02 is output.

RMLD : Q,BY

Output of the checkback signals (axis attributes) of the IP247 positioning module (=> Section 2.7 “AxisAttributes”).

Bit			
0	0 metric	0 inches	1 degrees
1	1	0	0
2	1: in position		0: not in position
3	1 : reference point synchronized		0: ref. point not synchronized
4	1: axis in teach-in		0: axis not in teach-in
5	1: reference point missing		0: set
6	1: machine data missing		0: machine data exist
7	1: job finished		0: job running

ANZG : Q, D with S5115U: **ANZ1** : Q,W
ANZ2 : Q,W

The parameter contains the values: actual position value (BA 71 set) or the distance to go (BA 73 set). If cyclic monitoring is switched off with mode 74, the value zero is output.

The output is either in BCD format (BCD = signal state”1”) or binary (BCD = signal state "0") according to the BCD parameter.

If parameters are assigned indirectly the following output parameters of FBI 64 are no longer updated:

BTR – mode type
MFKT – M-functions
RMLD – checkback signal (axis attributes)
ANZG – display of the monitoring job

The updated values must then be taken from the axis data block.

In the axis data block (parameter DBNR) only the value selected with modes 71...73 is updated. The other values are deleted (KH0000).

6.2.5 Notes on using Actual Operands

The designations STAR (1 ,BI), STOP (1 ,BI), VORW (1 ,BI), RUCK (1 ,BI), UEBN (1 ,BI) and PAFE (Q,BI), BFEH (Q,BI) and TBIT (Q,BI) must not be occupied by the “scratchpad flags”.

The designations BTR (Q,BY), MFKT (Q,BY), RMLD (Q,BY) and ANZG (Q,D) orANZI (Q,W) and ANZ2 (Q,W) must also not be occupied by the scratchpad flags used by function block FBI 64 (=> Section 6.2.8 “Technical Data of FB164").

When specifying data bytes, data words or a data double word, the information is stored in the axis data block. Make sure that the axis data area for a particular axis is not overwritten.

6.2.6 Relationship between the Parameter TBIT and the current Checkback Signals

6.2.6.1 General

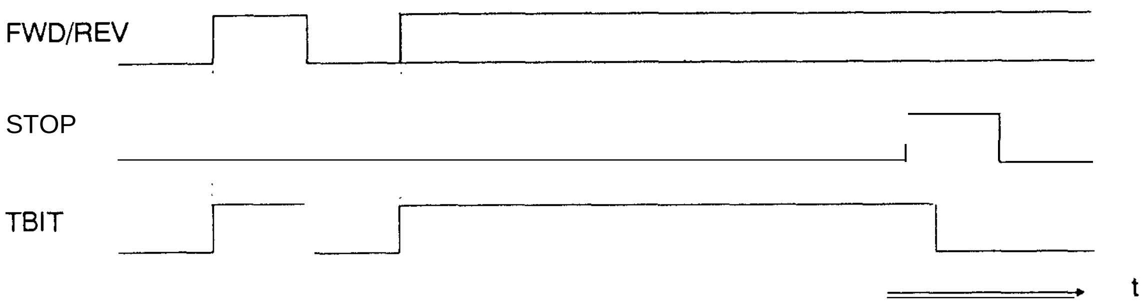
FB 164 signals a currently active job it has started on the IP247 (BA 1...17) at parameter output TBIT. With this, you are informed within the same cycle in which FBI 64 was exited that the I P247 is processing a job. The job finished bit is still set at this point. If the IP247 terminates the job it-self, it sets the job finished bit, and possibly also the position reached bit, but it does not directly affect the TBIT. The TBIT is then reset by FBI *64,* when the IP247 informs it that the job is finished. This means that you receive the message that the job has been completed in the same cycle after exiting FB1*64.* The job finished bit or the position reached bit has already been set at this time.

If a job is so short that the job finished bit is reset and set again during a PC cycle, it is not possible to detect the acceptance and completion of a job from this bit. Even in this case, the TBIT supplies a reliable signal edge change.

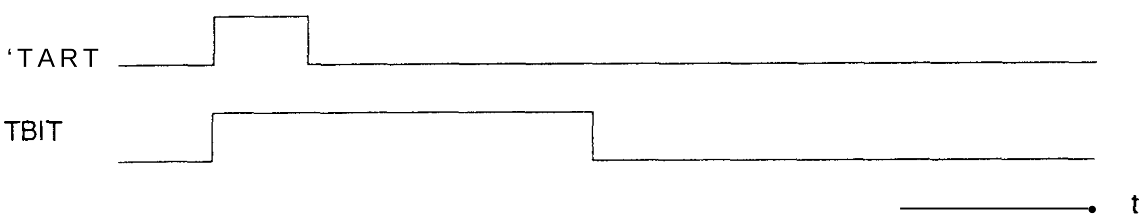
6.2.6.2 The Parameter TBIT with the Individual Modes

The following diagrams are not to scale and do not take into account the cycle time of the user program and the IP247.

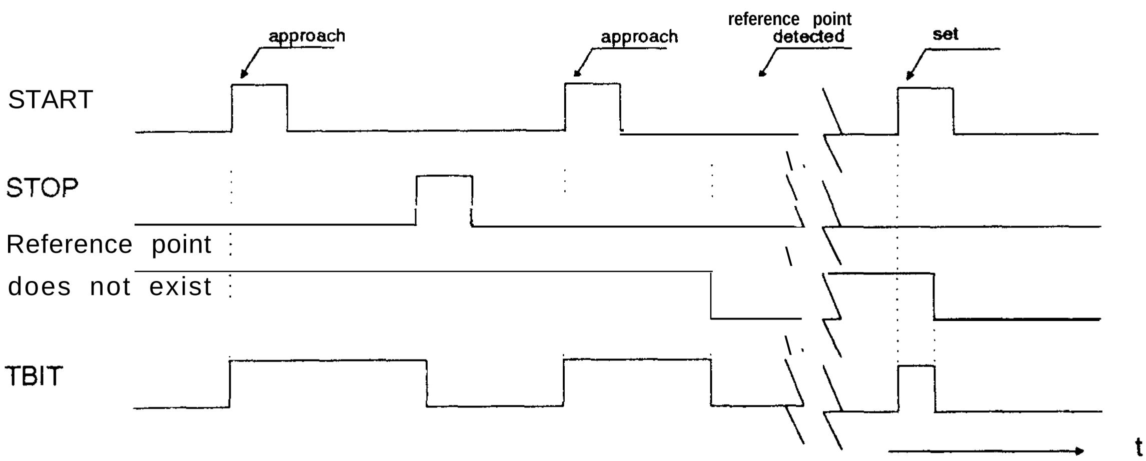
Mode – JOG speed 1 (BA 1)
– JOG speed 2 (BA 2)



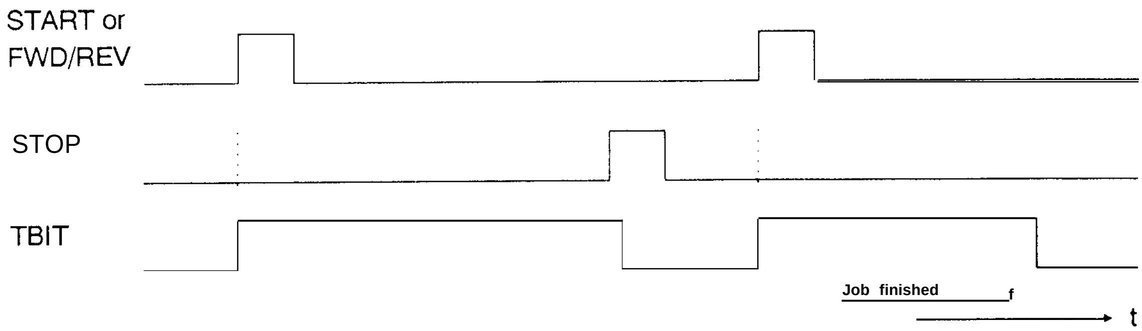
Mode – axis off (BA4)



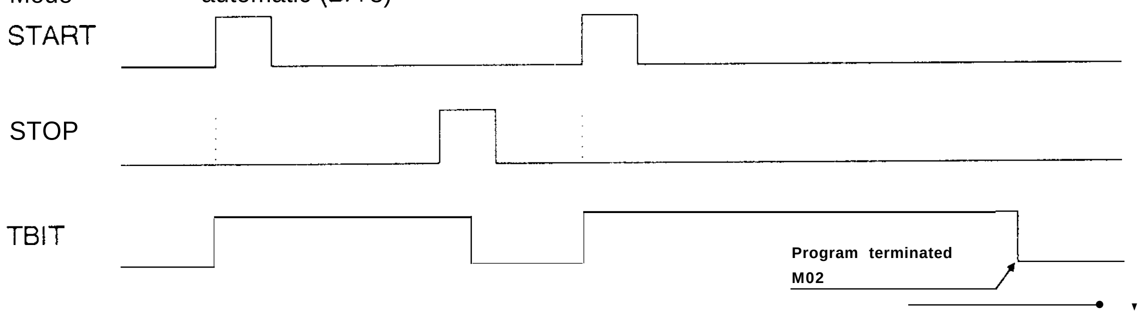
Mode – reference point (BA 5)



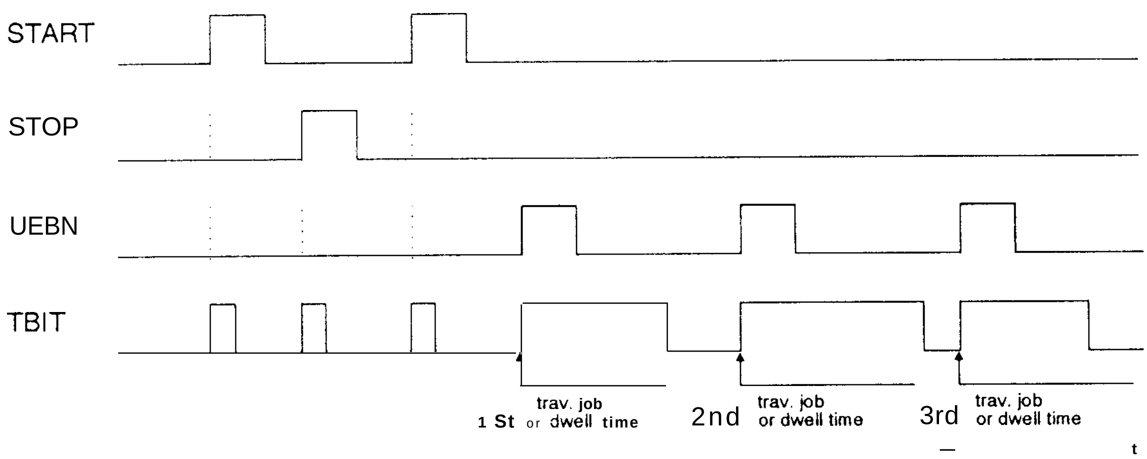
Mode – incremental approach absolute (BA 6) and
– incremental approach relative (BA 7)



Mode – automatic (BA 8)



Mode – automatic single statement (BA 9)

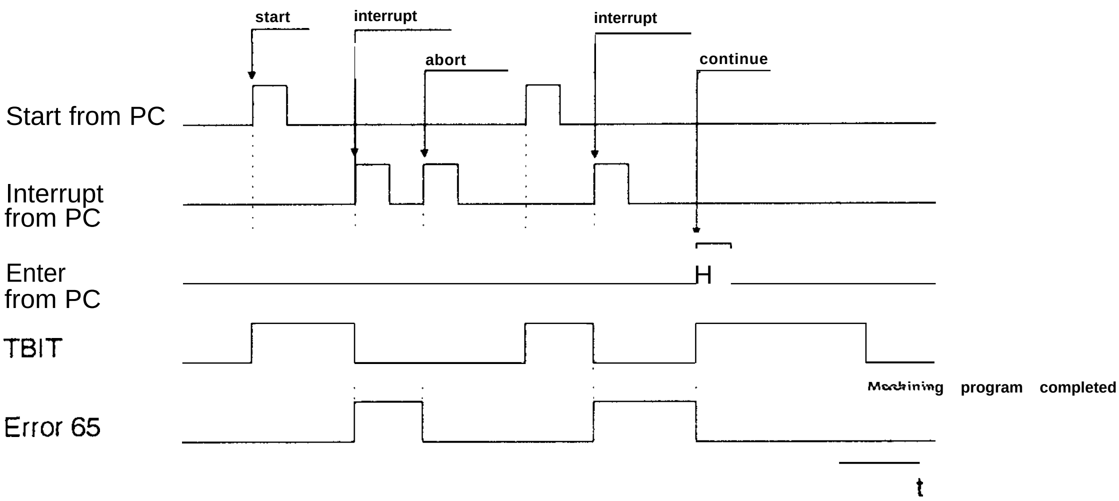


With the enter command (U EBN), the next statement of the automatic program (traversing job or dwell time) is started. If there is a stop between two statements in "automatic single statement" operation, the parameter TBIT is also set by FBI 64 and reset by the IP247.

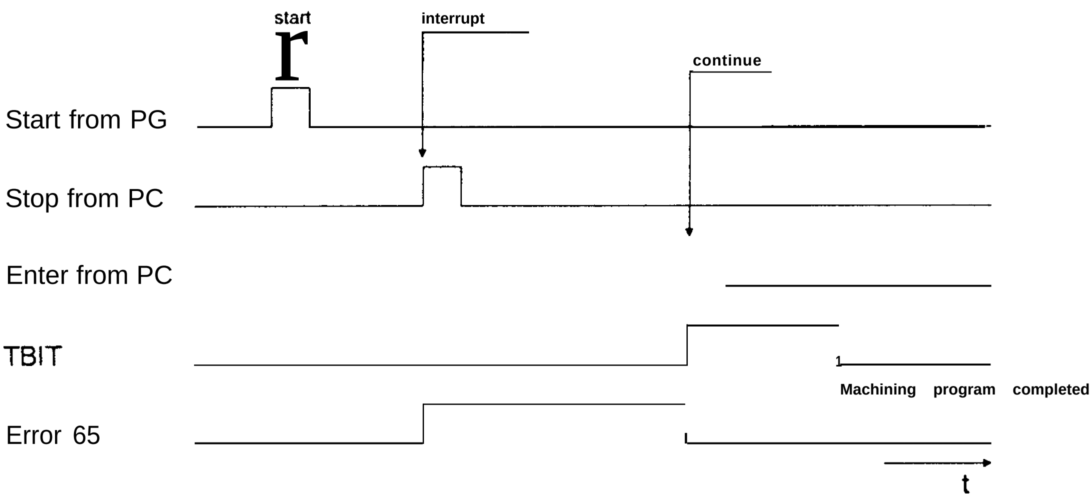
The parameter TBIT of FBI 64 when a machining program is interrupted

The parameter TBIT of standard function block FBI 64 is set whenever a job is triggered by FBI 64. After the job has been executed, the I P247 instructs FBI 64 to reset this bit.

If the machining program is started from the PC, the parameter TBIT is set by FBI *64,* If the machining program is interrupted, the parameter TBIT is reset again by FB1 64. The same conditions apply for resetting the bit as for changing the axis status from running to finished. If the machining program is continued by the PC with an enter command, the parameter TBIT is set again by FBI 64 and reset after the next interruption or on completion of the machining program.



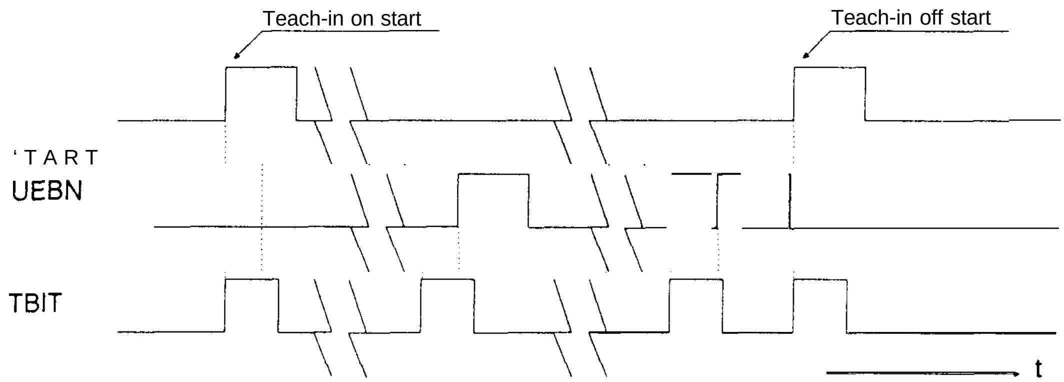
If a machining program which was started by the PG is interrupted by the PC, the parameter TBIT is reset again by FBI 64 after the interruption.



Note

If an interrupted machining program is continued again, it is treated from this point onwards as if it had been started via the interface from which the enter command was sent to the module. This means that the TBIT parameter is not set if an interrupted machining program is continued again from the PG.

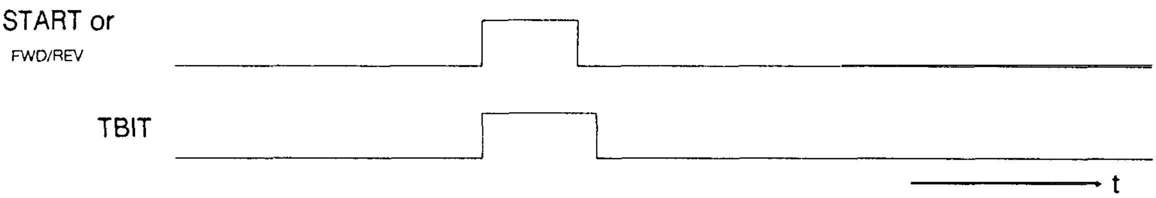
Mode - teach-in on (BA 10) and
teach-in off (BA 11)



The parameter TBIT is reset by the IP247 after storing the position (UEBN) or on termination (STOP) of the teach-in from FB164. In teach-in, various positions can be approached, e.g. in incremental approach or in the JOG mode. The conditions explained for these modes also apply to the parameter TBIT.

Mode - zero offset absolute (BA 12),
zero offset relative (BA 13),
clear zero offset (BA 14),
tool length offset (BA 15),
tool length offset off (BA 16) and
clear error (BA 17)

The parameter TBIT remains set until the mode is complete. These modes cannot be aborted.



6.2.7 Data Area Requirements

The standard function block FB164 works with data block DB164. It requires data words DW8 to DW15 as its working area.

An axis data block must be specified using the parameter DBNR. This data block is used for the following:

to store the monitoring values, checkback signals, M functions etc. read from the IP247,
for indirect assignment of parameters (DW1 to DW7) to the function block FB164 and
to store the data required for the mode to be started.

6.2.7.1 Indirect Assignment of ParameterstoFB184

You can assign parameters to the function block FBI 64 indirectly. You must preset the value

KY 0,0

as the actual operand for parameter DBNR. With this assignment, FBI **64** takes the values for its parameters from **the data block valid before its call**.

You can use all permitted data blocks. Even data block DB164 or the axis data block would be possible.

Indirect parameter assignment requires data words DW1 to DW7 of the data block inclusive; these data words have a fixed assignment. When using DB1 84, this does not lead to conflicts, since FB164 uses DW8toDW15 inclusive as its working area. If the axis data block is open before FB164 is called, you must enter at least the value 8 in DW5 (parameter DWNR) as the first data word to make sure that the data for indirect parameter assignment are not overwritten.

When using indirect parameter assignment, the same conditions apply to the individual parameters (DW1...DW7) of the open data block as for direct parameter assignment (=> Section 6.2,4 "Explanation of the Parameters").

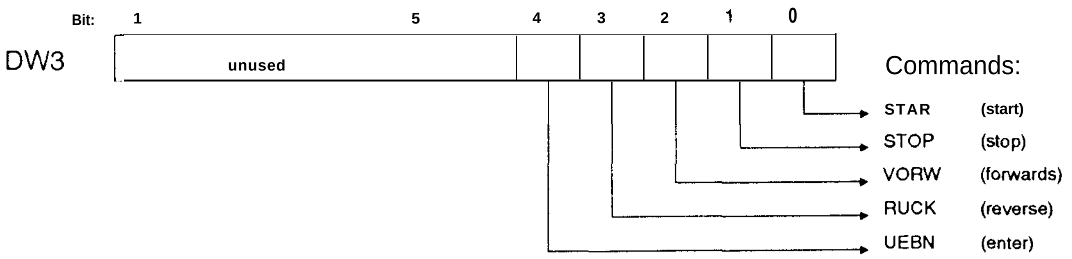
With indirect parameter assignment, the formal operands PAFE, BEFE, TBIT are updated in the actual operand of theFB164 call as in direct parameter assignment.

			Recommended data format
DW0	Free		KH
DW1	Parameter BA, mode (mode number)		KF
Dw2	Free		KH
DW3	Commands: STAR, STOP, VORW, RUCK, UEBN		KM
DW4	Parameter DBNR KY DB type, number of the axis data block		KY
DW5	Parameter DWNR (first data word)		KF
DW6	Parameter SSNR interface or page number		KF
DW7	Parameter BCD identifier output	KY 0,0= binary KY 0,1= BCD	KY

You must assign values to data words DW1 to DW7 before the function block FB164 is called.

Note on programmable controllers S5-1 15U andS5-150U
Data block type DX cannot be programmed with these units.

Structure of data word DW3 (commands)



6.2.7.2 Structure of the Axis Data Block

The data words from parameter DWNR to DWNR + 19 are required for an axis in the axis data block assigned with the parameter DBNR. The same data block can be used for several axes. The next axis can use the area from DWNR + 20 in the same data block.

The data block is structured as follows:

Axis 1 **(parameter DWNR = n)**

			Recommended data format
DW n	Used byFB164	BYTE parameter	KY
DW n+1	WORD parameter		KF
DW n+2	high — DOUBLE WORD parameter		KH
DW n+3	low		KH
DW n+4	occupied		KH
DW n+5	high — actual position value of the axis		KH
DW n+6	low		KH
DW n+7	high — free		KH
DW n+8	low		KH
DW n+9	high — distance to go		KH
DW n+10	low		KH
DW n+11	operating mode	M function	KY
DW n+12	checkback signals from the axis	error messages	KM
DW n+13	occupied		KH
DW n+14	binary identifiers		KH
DW n+15	occupied		KH
DW n+16	high — condition code bits from the SEND block		KM
DW n+17	low		KF
DW n+18	occupied		KM
DW n+19	occupied		KF

Axis 2 **(parameter DWNR = k)**

DW k	Used by FB164	BYTE parameter	KY
DW k+1	WORD parameter		KF
DW k+2	etc. (further structure analogous to axis 1)		

You must supply the following data words in the axis data block for each axis:

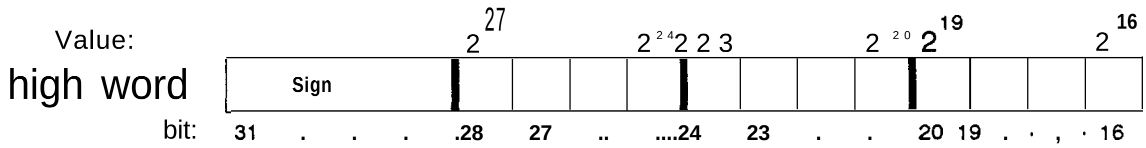
DR n : byte parameter
DW n + 1 : word parameter
DW n +2 : double word parameter

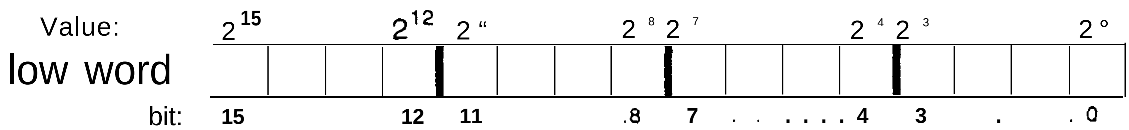
Depending on the mode (BA 1...BA 17; cf. parameter BA), the following convention applies:

	Mode	Permitted command	Byte parameter	Word parameter	Double word parameter
	JOG speed 1	Forward Reverse stop	Binary : x BCD : 0 : 1	Speed Speed Speed/10	---
	JOG speed 2	See JOG speed 1		Speed	---
	Axis off	Start	---	---	---
	Reference point	Start stop	Approach : ="0" Set : < > "o"	---	---
	Incremental absolute	Start stop Forward Reverse Enter	Binary : x BCD : 0 : 1	Speed Speed Speed/10	Absolute target
	Incremental relative	See JOG speed 1		Speed	Relative target
	Automatic	Start Stop Enter	Program number	---	---
	Automatic single statement	Start stop Enter	Program number	---	---
	Teach-in on	Start	Program number	---	---
	Teach-in off	Start	---	---	---
	Zero offset absolute	Start	---	---	Absolute coordinate
	Zero offset relative	Forward Reverse	---	---	Relative value
	Clear zero offset	Start	---	---	---
	Tool offset	Forward Reverse	---	---	Offset value
	Tool offset off	Start	---	---	---
17	Clear error	Start	---	---	---

Parameters without an entry in the table are not evaluated by the IP247 positioning module.

In the parameter “PC BCD coded” in the machine data record of the axis, you can decide whether all distances (double word parameter in the axis data block) and speeds (word and byte parameters in the axis data block) supplied by the PC to the IP247 are to be interpreted as BCD or binary. (=> Section 2.5.6 “Other Parameters”). Bits 28...31 represent the sign. In binary representation, negative distances must be specified in two’s complement.





Example:
incremental absolute (mode 6) to 120000 µm and -120000 µm.
(1) at 1000 mm/min
(2) at 10000 mm/min

120000_{10} = = > $0001D4C0_{16}$

=.> 0000000000000001 1101 010011000000 (binary)
DWORD PARAMETER
= = > 00000000000100100000 000000000000 (BCD)

-120000_{10} = = > FFFE2B4016

= = > 11111111111111100010 101101000000 (binary)
DWORD PARAMETER
==> 1111 00000001 00100000000000000000 (BCD)

1000_{10} = = = > $03E8_{16}$

(1)	=>	000000111110 1000 (binary)	WORD PARAMETER
		xxxx xxxx (any)	BYTE PARAMETER
	=>	00010000000000002	WORD PARAMETER
		000000002	BYTE PARAMETER

10000_{10} = > 2710_{16}

(2)	=>	0010011100010000 (binary)	WORD PARAMETER
		xxxx xxxx (any)	BYTE PARAMETER
	=>	0001000000000000 (BCD)	WORD PARAMETER
		000000012	BYTE PARAMETER

A”1” in the byte parameter means that the IP247 multiplies the word parameter by 10.

For more detailed information about the significance of the parameters in the individual modes, refer to Part *4* “Functions”.

The actual position, and the distance to go are updated in the axis DB regardless of how parameters are assigned to FBI *64* (direct or indirect parameter assignment).

Only when indirect parameter assignment is selected and byte DR7 (corresponds to the parameter BCD) is not zero, are these values in BCD format in the axis DB. When direct parameter assignment is selected and the parameter BCD is a”1” signal, one of these values (BA 71 ...73) is available at the output parameter ANZG (ANZ1/ANZ2 for the S5-1 15U) ofFB164 in BCD representation, **however, the value is stored in binary in the axis DB.**

Data words DW n+1 1 (axis mode, current M function) and DW n+12 (checkback signals from the axis, error message from the axis) of the axis DB are only updated with indirect parameter assignment, With direct parameter assignment they have the value KHOOOO.

The error message byte DR n+12 of the axis DB is identical to flag byte FY251 (=> Section 7.2 “Troubleshooting”),

6.2.8 Technical Data of FB164

	S5-115U	S5-135U	S5-150U	S5-155U
Block number	164	164	164	**164**
Block name	PER:POS	PER:POS	PER:POS	PER:POS
Library no.	71200-S5164-D-2	P71200-S9164-D-2	P72100-S4164-D-2	P71200-S6164-D-2
Call length	**20** words	19 words	19 words	19 words
Block length	**1012** words	618 words	646 words	681 words
Nesting depth	1	1	1	1
Secondary blocks	ntegrated handling blocks	Handling blocks	Handling blocks	Handling blocks
Occupation of data area	**-19** data words from parameter DWNR of the axis data block DBNR - DB164 occupied from data word DW8 to DW15 - In indirect parameter assignment via DBx: data word DW1 to DW7			
Occupation in flag area	FY206 to FY255 scratchpad flags	FY206 to FY255 scratchpad flags	FY200 to FY255 scratchpad flags	FY202 to FY255 scratchpad flags
Occupation in system data area	none	RS60 and RS61	yes	**yes**
System statements	**yes**	yes	yes	yes
Miscellaneous	nterrupts blocked at times n the FB by commands IA/ RA. An IA command is cancelled by this (also S5-155)	Handling blocks FB120 SEND, FB121 RECEIVE and FB125 SYNCHRON must be loaded. Special functions called.	Handling blocks FB180 SEND, FB181 RECEIVE and FB185 **SYNCHRON must be loaded.** .	Handling blocks FB120 SEND, FB121 RECEIVE and FB125 **SYNCHRON must be loaded. Special functions called.**

Maximum run-times in ms () with binary-BCD conversion	CPU 941	CPU 942	CPU 943	CPU 944	CPU 922	CPU 928	CPU 928/2		
Idling, monitoring off	2.6.7	8.7	5.2	1.1	8.8	3.3	2.7	0.8	1.0
Monitoring on mode = **71, 73**	34.4 (81.0)	13.8 (18.0)	12.4 (14.2)	6.0 (6.8)	12.1	5.9	5.7	5.6	4.4
For command transfer (STAR, STOP, VORW, RUCK, UEBN)	40.6	17.8	11.0	5.6	11.8	7.6	6.0	5.2	4.1
Extra runtime required for FB with direct param. assignment	2.2	1.9	1.7	**0.07**	**0.3**	**0.1**	**0.1**	**0.04**	**0.03**

6.2.9 Using Function Block FB164

In cyclic operation it is not possible to address a module both with indirect and direct parameter assignment.

Function block FBI 64 works with data block DB1 64. This must be installed up to and including data word DW15. A particular assignment of the data words is not necessary.

Data block DB164 is divided into two areas, in which data words DW1 to DW7 are reserved for indirect assignment of parameters to the function block. Data wordsDW8toDW15 are the working area for FB1 84. You must **not** change the working area.

Before calling FB1 64, the axis parameters (byte, word and double word parameters) must be written to the axis data block (parameter DBNR) as required for the mode to be started. The data block must be a minimum of x words long, where

x . parameter DWNR + 19

e.g.: axis 1 : parameter DWNR = 1 -> x = **20**
axis 2 : parameter DWNR . **21 ->** x . 40
axis 3 : parameter DWNR = 41 -> x **= 60**

If only one axis is required, the data block must be available up to and including data word DW20.

If all three axes are used and if the parameters for the axes are contained in one data block, this must be available up to and including data word DW60. The data block number (parameter DBNR) and the data word number (parameter DWNR; start address in the data block) can be selected as required.

The data block is setup with a programmer, e.g. with the PG685 STEP5 under S5-DOS using the following commands (see programming instructions for STEP 5):

<F1>	(input)	
<F1>	(block)	
Pc	(input device)	
DB160	(block)	enter key <|>
DWO: KY = 000,000		
DW 1 : KF = +00000		
DW 2: KH = 0000		
DW3: KH = 0000		
DW 19: **KF = +00000**		enter key <|>

The function block FB164 must be called unconditionally once per cycle for each axis. This is necessary in order to update the edge flags (binary identifiers in the axis DB) of the parameters STAR, STOP, VORW, RUCKand UEBN.

To ensure that the signal edge evaluation is effective, the selected mode must remain active in the function block until the traversing movement is complete. The command bits should, however, be reset as quickly as possible.

If there is a power failure while a command bit is set and if, after the return of power *the* same command must be sent with a cold restart in the first PC cycle, this is not possible because the edge flag in the binary identifiers in the axis DB is still set to /1/, The FB therefore considers that the job has already been started.

Once a job has been triggered, it is sent to the positioning module immediately when the function block FBI 64 is next called.

A job is only automatically repeated when a parameter assignment error in the SEND data handling block is signalled.

As soon as a valid monitoring job (BA = 71, 73) is recognized, it is executed at each JU FBI 64 call, providing there is no operating job in the call.

Mode 74 interrupts the cyclic monitoring. The monitoring function is resumed when one of the modes BA 71, or 73 is transferred.

The information read is written to the axis data block as follows: (with parameter DWNR = n):

DW n+5 and DW n+6 :	actual position value, binary or BCD
DW n+7 and DW n+8 :	free
DW n+9 and DW n+10 :	distance to go, binary or BCD

The output at parameter ANZG (ANZ1 and ANZ2 with theS5-115U) or in the axis data block is in binary in fixed point double word format (32 bits).

- with direct parameter assignment: when the parameter BCD has the signal state “O”,
- with indirect parameter assignment: when data byte DR7 of the open DB has the value KBOO.

The output is as a seven digit BCD number with sign

- with direct parameter assignment: **when the parameter** BCD has the signal state”1”,
- with indirect parameter assignment: when data byte DR7 of the open DB is not equal to KB00.

If a conversion from binary to BCD is not possible (representable BCD numerical range exceeded), the content of the parameter ANZG (ANZ1 and ANZ2 with the S5-1 15U) is unchanged with **direct parameter assignment.** If the representable BCD numerical range is exceeded with **indirect parameter assignment, the monitoring** value is stored as a 32-bit fixed point number (2’s complement) in the axis data block.

The positioning module IP247 does not service interrupts.

With indirect parameter assignment, the current data block (DB or DX) must be open and must have values supplied before function block FBI 64 is called.

6.2.9.1 Special Feature of the Parameter STOP

The STOP command has the highest priority and can be transferred during any mode. If mode 71 <= BA <= 73 is selected, the module is not read for one cycle and the stop command is transferred to the positioning module with mode 1 (JOG 1). In the following PC cycle, the module is read once again.

If the STOP signal is constantly set (static), no start, forward or reverse or enter job will be sent to the module.

6,2.9.2 Special Features of the Parameters VORW and RUCK

If modes 1 and 2 (JOG 1, and 2) are selected, these commands result *in* a jogging operation, On a signal change from O to 1, the axis is started in the selected direction, on the signal change from 1 to O, the axis is stopped. H is also possible to transfer the STOP command. If a signal change O to 1 of the commands VORW and RUCK is recognized simultaneously, the STOP command is sent to the axis,

To ensure that the signal edge evaluation is effective, the mode **must** remain valid in the function block until the traversing movement is complete.

6.2.9.3 BCD Output

With the S5-135U, -150U, -155U:

	Sign	Decades						
ANZG	V V V V	10^6	10^5	10^4	10^3	10^2	10^1	10^0
bit:	31 28	27 . 24	23 . . 20	19 . 16	15. 12	11 . . 8	7 4	3 0

FD60	FY60	FY61	FY62	FY63

In the axis DB

e.g. actual value	DW n+5	DW n+6
	DD n+5	

6.2.9.4 BCD Output with theS5-115U

		Sian	Decades		
Parameter ANZ1		VVVV	10^6	10^5	10^4
	bit	15.. 12	11 . . . 8	7... 4	3.. , 0

		Decades			
Parameter ANZ2		10^3	10^2	10^1	10^0
	bit	15.. 12	11... 8	7... 4	3.. 0

6.3 Standard Function Block FB165

6.3.1 Functional Description

The function block “assigning parameters to the positioning module” handles the data exchange between the user program and the IP247 positioning module. Each valid job number causes a data transfer IP247 <===> PC.

Data exchange PC ===> IP247:

The data to be transferred is located in a data block of your choice (source DB). With direct parameter assignment, the data block must be planned at the block parameters of FBI 65, with indirect parameter assignment, in the axis data block.

Data exchange IP247 ===> PC:

The data to be read from the I P247 positioning module is stored in a data block in the PC memory (destination DB). With direct parameter assignment, this data block must be planned at the block parameters of FBI 65 and with indirect parameter assignment in the axis data block.

The following functions are possible via the PC interface using FBI 65:

- read machine data and machining programs from the IP247, delete and transfer them,
- read the SYSID from the IP247 (BA 70) and transfer them to the IP247 (BA 24),
- request an overview of machine data or machining programs stored on the IP247 and
- read actual values (actual position value, distance to go) simultaneously.

The function block FBI 65 can have parameters assigned directly or indirectly. With direct parameter assignment, the data and parameters for a job are applied to the parameter inputs of FBI 65. With indirect parameter assignment, the axis data block is planned in the data block valid before its call. The remaining parameters are taken by FBI 65 from the axis data block.

Before FBI 65 is called, the axis data block must be set up and with indirect parameter assignment must be supplied with the values required to start the mode.

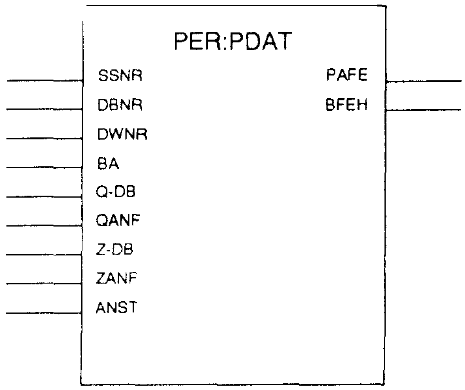
6.3.2 Calling Function Block FBI 65

In STL (Statement List):

```
      .     JU FBI 65
NAME  :     PER:PDAT
SSNR  :
DBNR  :
DWNR  :
BA    :
Q-DB  :
QANF  :
Z-DB  :
ZANF  :
ANST  :
PAFE  :
BFEH  :
```

In LAD/CSF (Ladder Diagram or Control System Flowchart)

FBI 65



6.3.3 Overview of the Parameters

NAME	PARA TYPE	DATA TYPE	SIGNIFICANCE
SSNR	D	KF	Interface number
DBNR	D	KY	DB type, DB number (of the axis data block)
DWNR	D	KF	First data word in the axis DB
BA	D	KF	Binary/BCD conversion, mode
Q-DB	D	KY	DB type, DB number (of source DB)
QANF	D	KF	Start address DW in source DB
Z-DB	D	KY	DB type, DB number (of destination DB)
ZANF	D	KF	Start address DW in destination DB
ANST	I	BI	Trigger data transfer with direct parameter assignment
PAFE	Q	BI	Parameter assignment error
BFEH	Q	BI	Module error

6.3.4 Explanation of the Parameters

SSNR : D,KF x

Specification of the page number (cf. switch setting J64, Section 3.3.2 “Setting the Module Address”) of the appropriate axis.

x = interface (page number)
O ≤ x <255

DBNR : D, KY x,y

Specification of the data block type and data block number of the axis data block. With the programmable controllers S5-1 15U and S5-150U, *it is* not possible to program the data block type DX.

x = data block type

x = O: data block type DB
x >< 0 : data block type DX

Y = data block number

5 ≤ y ≤ 255 where x = O
1 ≤ y ≤ 255 where x >< 0
direct parameter assignment via the block parameters

y = 0

indirect parameter assignment via the data block open before the FB165 call

DWNR : D,KF x

Specification of the first data word in the axis data block.

x = first data word
OS- x ≤ 241
where: 5&parameter DBNR ≤ 163 and
166 ≤ parameter DBNR ≤ 255

4&— x ≤ 241
where: parameter DBNR = 165 (DB165 = working DB for FB165)

16≤ X ≤ 241
where: parameter DBNR = 164 (DB164 = working DB for standard function block FBI 64)

BA : D, KY x,y

Specification of the mode to be executed, selection binary/BCD conversion.

x = 0 : no binary/BCD conversion
x > < 0 : binary/BCD conversion of actual position value, and distance to go.
Evaluation only in mode BA *66.*

y = Operating mode (job **number)**

20< y $\leq$ 24 write and delete jobs

64 $\leq$ y $\leq$ *70* read jobs

Job number	Operating mode	Possible on
20	input machine data	one axis
21	delete machine data	one axis
22	input machining program	data channel
23	delete machining program	data channel
24	input SYSID	data channel
64	read machine data directory	all axes + data channel
65	read machining program directory	all axes + data channel
66	read actual values	one axis
67	read machine data	one axis + data channel*
68	read machine data overview	all axes + data channel
69	read machining program	all axes + data channel
70	read SYSID	all axes + data channel

If you attempt to send a mode via an illegal axis (data channel), the I P247 sends a negative acknowledgement (see FB 164).
* Can only be read out via the data channel, when all DB numbers are different.

Q-DB : D, KY x,y

Specification of the source data block. For the programmable controllers S5-1 15U andS5-150U it is not possible to program data block type DX.

x = data block type

x = O : data block type DB
x> <0: data block type DX

Y = source data block number

DB: 5< y $\leq$ 255
DX: 1 $\leq$ y $\leq$ 255
In modes 20, 22 and 24 (write jobs) **the specified data block (source DB) is in the** PC **memory.**

O <= y <= 255
In modes 67 and 69 (read jobs) **the specified data block (source DB) is in the** RAM **of the positioning module.**

QANF : D,KF x

Specification of the first data word from which the data is to be read out of the specified source DB.

x = source first data word

0 ≤ x ≤ n
DB164: 16 ≤ x ≤ n;
DB165 :48 ≤ y ≤
A source first data word is only required for modes 20, 22 and 24 (write jobs)

(n= max. 2047 words: max. data block length, the advisable range is between O and 255)

Z-DB : D, KY x,y

Specification of the destination data block. For the programmable controllers S5-1 15U and S5-150U it is not possible to program the data block type DX.

x = data block type

x = O : data block type DB
x >< O: data block type DX

y = destination block number

DB : 5< y < 255
DX : 1< y < 255
For modes 64 to 70 (read jobs) **the specified data block (destination DB) is in the PC memory.**

O ≤ y &255
For modes 20 to 23 (write and delete jobs) **the specified data block (destination DB) is in the RAM of the positioning module.**

ZANF : D,KF x

Specification of the first data word from which data will be written to the specified destination DB.

x = destination first data word

0 ≤= x ≤= n
DB164: 16 ≤ x ≤ n;
DB165 : 48&y ≤
a destination first data word is only required for modes 64 to 70 (read jobs)

(n= max. 2047 words: max. data block length, the advisable range is between O and 255)

ANST : I,BI

When assigning parameters via the block parameters (direct parameter assignment) the pending job is executed on a signal change from "0" to"1" at the ANST parameter, You must set the parameter, If the job (mode) has been completed, the parameter is reset by FB165 (acknowledged).

PAFE : Q,BI

The parameter PAFE has the signal state"1" if the parameter assignment is incorrect. The error can then be identified in

flag byte FY255

(=> Section 7.2 "Troubleshooting").

BFEH : Q,BI

The parameter BFEH (module error) has signal state"1" when an error is signalled by the positioning module, The error can be identified in

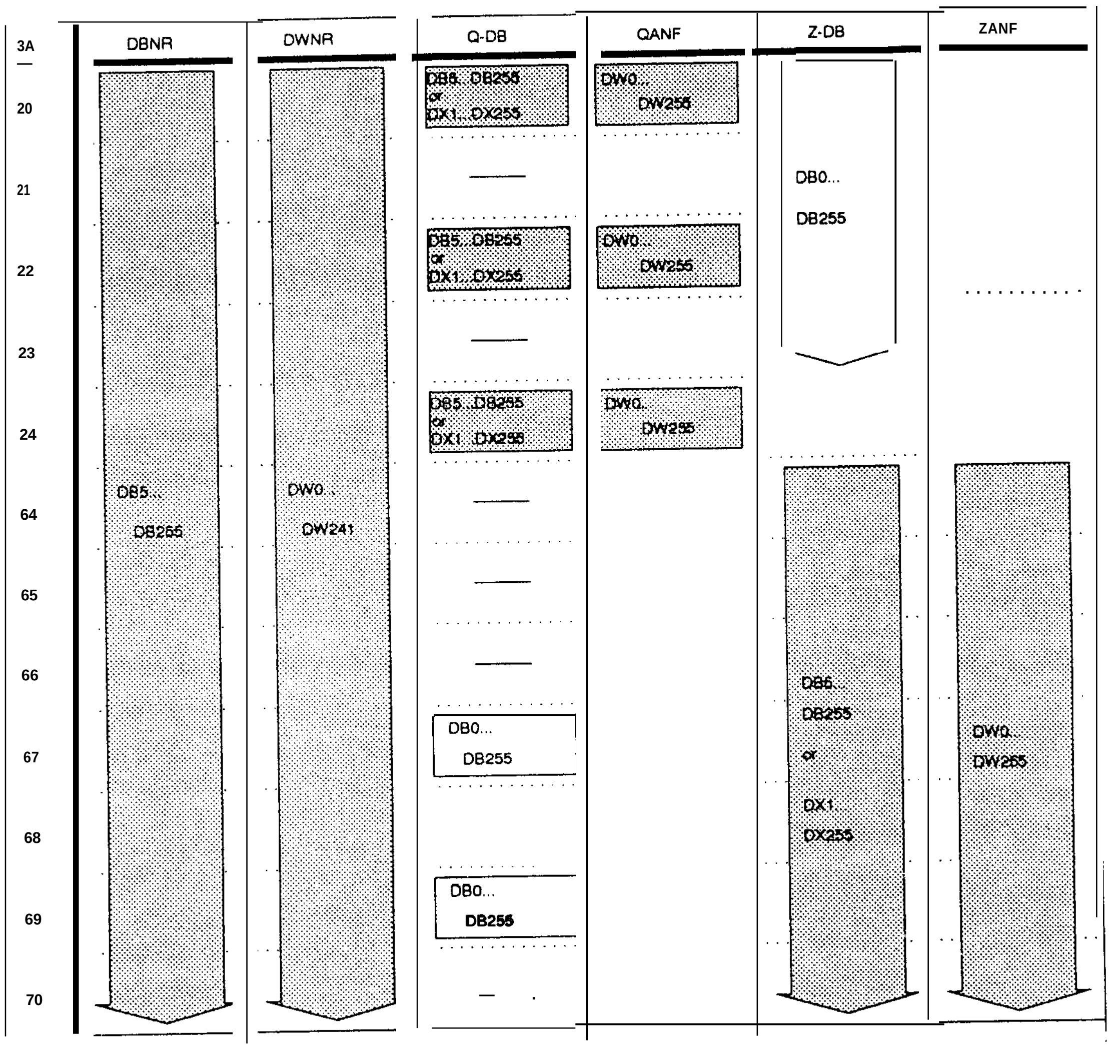
flag byte FY254

(=> Section 7.2 "TroubleShooting").

6.3.5 Notes on using Actual Operands

The parameters ANST (I, BI),PAFE (Q, BI) and BFEH must not be occupied by the "scratchpad flags" of function block FB1 65 (see technical data).

6.3.6 Overview of the Permitted and Advisable Parameter Area for the Standard Function Block FB165



Exceptions in the grey fields are DB164 and DB165:

DB165	DW48... DW241	DB165	DW48 DW241	DB165	OW48 DW241
DB164	DW16 DW241	DB164	DW16... DW241	DB164	DW16 DW241

Data block type DX can only be selected in the programmable controllers S5-135U and **S5-155U**.

The data blocks without a grey background are located in the **RAM of the positioning module.**

When assigning data blocks, remember that if you use DB165 as the axis data block for the parameters DBNR, Q-DB or Z-DB (indirect parameter assignment), data words DW3 to DW47 are required by function block FBI 65 (working area of FB165), **You must not use these data words for any other purpose.**

6.3.7 Data Area Requirements

The standard function block FBI 65 works with data block DB165. It requires data words DW3 up to and including DW47 for its working area.

An axis data block must be specified using the parameter DBNR. A job field with a length of 15 data words must be available in this axis data block for each axis.

6.3.7.1 Indirect Assignment ofParameterstoFB165

It is possible to assign parameters to the function blockFB165 indirectly. The parameter DBNR must have the

value KY 0,0

set as the actual operand. FB165 then obtains the parameters DBNR and DWNR from the data block open before its call. The remaining input parameters are supplied from the specified axis data block.

Any permitted data block can be used. Even data block DB165 or the axis data block are possible.

Indirect parameter assignment requires data words DW1 and DW2 of the open data block.

When using DB1 65, conflicts do not arise, sinceFB165 uses DW3 to DW47 as its working area. If the axis data block is open before FBI **65 is called,** you must enter at least the value 3 in DW2 (parameter DWNR) as the first data word, to ensure that the data are not overwritten when using indirect parameter assignment.

Structure of the data block with indirect parameter assignment.

		Recommended data format
DW0	Free	KH
DW1	Parameter DBNR KY DB type, DB number of the axis data block	KY
DW2	Parameter DWNR (first data word)	KF

You must supply values for DW1 and DW2 when using indirect parameter assignment before function block FB165 **is** called.

The DB type (DL1) and DB number (DR1) define the axis data block. The DW number (DW2) indicates the start of the job field of the job to be executed in the axis data block.

6.3.7.2 Structure of the Axis Data Block for an Axis

An axis requires the data words from parameterDWNRtoDWNR+14 inclusive from the axis data block selected with the parameter DBNR. The same data block can be used for several axes, the next axis then occupies the area from DWNR +15.

The data block is structured as follows:

Axis 1 **Parameter DWNR = n**

		Recommended data format
DW n	Parameter BA; binary/BCD conversion (only BA 66), mode	KY
DW n+1	Parameter Q-DB; DB type, source data block	KY
DW n+2	Parameter QANF; source first data word	KF
DW n+3	Parameter Z-DB; DB type, destination data block	KY
DW n+4	Parameter ZANF; destination first data word	KF
DW n+5	Parameter SSNR; interface number	KF
DW n+6	binary identifiers	KY
DW n+7	high – bits from the SEND block	KM
DW n+8	low	KF
DW n+9	occupied	KM
DW n+1O	occupied	KF
DW n+11	high – bits from the FETCH block	KM
DW n+12	low	KF
DW n+13	occupied	KM
DW n+14	occupied	KF

There must be a "job field" with the structure above for each axis addressed.

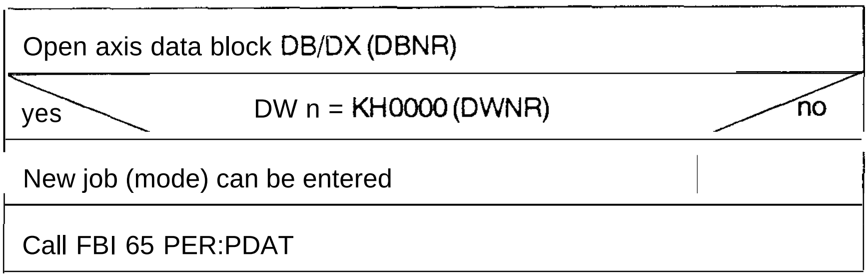
The data words DWn to DWn+5 inclusive must only be completed if the function block is to have parameters assigned indirectly.

The data words DWn+6 to DWn+14 are used by function block FBI 65 and you can only read them.
E.g. evaluation of the interface error in the high byte of the condition code bytes:

DL n+7 and DL n+11: High byte of the condition codeword. Corresponds to flag byte FY250.

With indirect parameter assignment, you enter the required mode in data word DWn of the axis data block. The function block FBI 65 executes the entered mode and acknowledges by entering the value KH0000 in data word DWn. You can now enter a new mode.

Schematic diagram of indirect parameter assignment:



6.3.8 Structure *of* the Source or Destination Data Blocks in the PC Memory for the Individual Modes

6.3.8.1 Structure of a Machine Data DB in the PC Memory

BA = 20: machine data transferred from PC to IP247,

BA = 21: machine data deleted on the IP247,

BA = 67: machine data read from the IP247.

The individual machine data are explained in Section 2.5 “Machine Data and their Structure”.

Structure of the data block DBx from data word DWn: typical values have been entered.

		Recommended data format	
DW n	+00070	KF	Length in words
DW n+1	0044	KH	00, 'D'
DW n+2	066,001	KY	"B', DB number of the data record
DW n+3	000,001	KY	Module number, axis number
DW n+4	001,000	KY	Meas. system (mm), machine data error
DW n+5	0000	KH	Maximum frequency [Hz] (5000 Hz)
DW n+6	1388	KH	
DW n+7	0000	KH	Start-stop frequency [Hz] (200 Hz)
DW n+8	00C8	KH	
DW n+9	0000	KH	Frequency increase [mHz/ms] (50000 mHz/ms)
DW n+10	C350	KH	
DW n+11	+00030	KF	Pulse duration [us]
DW n+12	0040	KH	Polarity 00H: negative edge 40H: positive edge
DW n+13	+00004	KF	Number of excitation patterns
DW n+14	+00200	KF	Pulses/revolution [l/rev.]
DW n+15	0000	KH	Transmission ratio [urn/rev.] (2000um/rev.)
DW n+16	0700	KH	
DW n+17	0000	KH	JOG speed 1 [mm/min] (3000 mm/min)
DW n+18	0BB8	KH	
DW n+19	0000	KH	JOG speed 2 [mm/min] (3000 mm/min)
DW n+20	0BB8	KH	

(Continued on the following page)

		Recommended data format	
DW n+21	0000	KH	Incremental speed [mm/m in] (3000 mm/min)
DW n+22	0BB8	KH	
DW n+23	0000	KH	Reference speed [mm /rein] (3000 mm/m in)
DW n+24	0BB8	KH	
DW n+25	0000	KH	Reference point synchronized O: yes/1: no
DW n+26	0020	KH	Reference direction 00H: right/ 20H: left
DW n+27	0000	KF	Reference point coordinate [urn] (O urn)
DW n+28	0000	KF	
DW n+29	FFF9	KF	Start of range or software limit switch Start [urn] (-400000um)
DW n+30	E580	KF	
DW n+31	0007	KH	End of range or software limit switch End [urn] (500000um)
DW n+32	A120	KF	
DW n+33	0001	KH	Polarity of limit switches O: positive edge active/ 1: negative edge active
DW n+34	0000	KH	Tool length offset [urn] (2300um)
DW n+35	08FC	KH	
DW n+36	0000	KH	Backlash compensation [urn] (0um)
DW n+37	0000	KH	
DW n+38	0000	KH	Zero offset 1 [urn] (10000um)
DW n+39	2710	KH	
DW n+40	0000	KH	Zero offset 2 [urn] (25000um)
DW n+41	61A8	KH	
DW n+42	0000	KH	Zero offset 3 [urn] (3250um)
DW n+43	OCB2	KH	
DW n+44	FFFE	KH	Zero offset 4 [urn] (-75000um)
DW n+45	DB08	KH	
DW n+46	0000	KH	Axis type 00H: linear/ 80H: rotary
DWn+47	0000	KH	PC BCD-coded 00H: binary/01 H: BCD
DWn+48	0000		Reserve
DWn+69	0000		

Parameters requiring two words, e.g. zero offset, are 32-bit fixed point numbers. Negative values are stored as 32-bit fixed point numbers in 2’s complement. When the value is input or interpreted, remember that the programmer does not make any suitable format available for this.

The machine data can *only* be transferred, read or deleted via the axis interface (parameter SSNR).

6.3.8.2 Structure of a Machining Program DB in the PC Memory

BA = 22	machining programs transferred from PC to IP247,
BA = 23	machining programs deleted on the IP247,
BA = 69	machining programs read from the IP247.

The structure and syntax of machining programs is explained in Section 2.6 “Machining Programs and their Structure”.

Structure of data block DBx from data word DWn as an example:

```
%     1 EXAMPLE        ; main program DB1, comment
N1    X1 OOF1OOOM1O ; 1st statement
N2    M02              ; 2nd statement
```

	Recommended data format		Header information	
DW n	+00025	KF	Length of the machining program, number in words	
DW n+1	0044	KH	00, ‘D’	
DW n+2	066,001	KY	‘B’, DB number of the machining program	
DW n+3	+00000	KF	Machining program error	
DW n+4	+00000	KF	Number of the incorrect statement	
DW n+5	2520	KH	% = main program; blank hundreds	
DW n+6	2031	KF	Blank (tens), ‘1 (ones) of the machining program number in ASCII	
DW n+7	2045	KH	Blank ‘E’	
DW n+8	5841	KH	‘X’, ‘A’	
DW n+9	4D50	KH	‘M’, ‘P’	
DW n+1O	4C45	KH	‘L’, ‘E’	
DW n+11	0A00	KH	<LF>	
DW n+12	4E31	KH	‘N’, ‘1‘	
DW n+13	2058	KH	Blank, ‘X’	
DW n+14	3130	KH	‘1‘, ‘o’	
DW n+15	3020	KH	‘O’, Blank	
DW n+16	4631	KH	‘F’, ‘1‘	1st statement
DW n+17	3030	KH	‘o’, ‘o’	
DW n+18	3020		‘O’, Blank	
DW n+19	4D31	KH	‘M’, ‘1‘	
DW n+20	300A	KH	‘O’, <LF>	
DW n+21	4E32	KH	‘N’, ‘2’	
DW n+22	204D	KH	Blank, ‘M’	2nd statement
DW n+23	3032	KH	‘o’, ‘2’	(without DR n+24)
DW n+24	0A00	KH	<LF>	

The length of the machining program depends on the number of programmed statements. The machining program DB can have a maximum length of512 words.

If a machining program DB is generated or modified in the PC, the length in words must be updated in DWn. The length of the machining program is the area from data word DWn up to and including the data word in which <LF> follows M02 (DWn+y).

Machining programs are not restricted to a specific axis, They can be transferred and deleted only via the data channel of the IP247 (parameter SSNR for the 4th page of the IP) and read via all pages.

6.3.8.3 Structure of the SYSID of the IP247 in the PC Memory

The system identification SYSID (module identifier) can be

transferred in part to the IP247 with **BA = 24**

and

read completely from the IP247 with **BA = 70.**

Read SYSID (BA = 70)

The system identification SYSID stored in data block DBx from data word DWn. The system identification occupies nine words when read from the IP247.

		Recommended data format	
DW n	'IP'	KS	Module version here IP247
DW n+1	'24'	KS	
DW n+2	'7 ␣'	KS	
DW n+3	'A0'	KS	Firmware release here A02.1
DW n+4	'2.'	KS	
DW n+5	'1'	KS	
DW n+6	000,000	KY	DRn + 6: module number
DW n+7	000,000	KY	DRn + 7: slot number
DW n+8	000,000	KY	DRn + 8: page number

Enter SYSID (BA = 24)

The system identification SYSI D is stored in data block DBx from data word DWn. The system identification is limited to three data words when writing to the IP247.

	Recommended data format	
DW n	000,000	KY DR n: module number [0...99]
DW n+1	000,000	KY DR n+1: slot number [0...255]
DW n+2	000,000	KY DR n+2: page number[0...252]

If the same data block is used for reading the SYSID from the module and for writing the SYSID to the module, then the value in the parameter QANF must be increased by six (writing to the I P247) compared with the value in the parameter ZANF (reading from the IP247).

6.3.8.4 Structure **of the Machine Data Directory**

The machine data directory can be read from the IP247 with

BA = **64**

The machine data directory has a constant length of six data words. if the machine data record is missing for an axis, the data words have the value zero.
The machine data directory in data block DBx from data word DWn has the following structure:

	Recommended data format			
DW n	**000,001**	KY	DB no. of the machine data record	Axis 1
DW n+1	**+00070**	KF	Length in words	
DW n+2	**000,005**	KY	DB no. of the machine data record	Axis 2
DW n+3	**+00070**	KF	Length in words	
DW n+4	**000,007**	KY	DB no. of the machine data record	Axis 3
DW n+5	**+00070**	KF	Length in words	

6.3.8.5 Structure of the Machining Program Directory

You can read the machining program directory from the IP247 with

BA = 65

The length of the directory is variable and depends on the number of machining programs on the positioning module. A maximum of **255** machining programs can be stored on the IP247 (DBO to DB255). In the directory, two data words are required for each machining program. The directory can therefore be a maximum of 510 words long.

The entries in the directory are not sorted according to the DB number but are stored in the order in which they are entered on the IP247.

If the data block (DBx) selected for entry of the machining programs is not long enough, the remaining data are stored in the next data block (DBx+1) from data word DWO.

You can only select a destination start address (parameter ZANF) for DBx. Entries are made only up to data word DW255 in the data blocks DBx and DBx+l.

If two DBs are required, data block DBx must be installed up to and including DW255. Otherwise the program will be aborted with an error message.

The following rule applies to the length of data block DBx:

length = (possible entries x 2) + destination start address ZANF

for DBx+1 the following applies:

length = remaining entries x 2.

A further switch to a data block DBx+2 is not possible. If the directory cannot be stored completely in the PC memory, the job is aborted with an error message.

Example 1

The positioning module has the maximum number of machining programs (255). Data word DW0 must be specified as the destination start address. The data blocks DBx and DBx+1 must be installed up to and including data word DW255. The machining program directory is stored in data blocks DBx and DBx+1 from data word DW0 onwards.

DB x :

		Recommended data format		
DW 0	000,001	KY	Machining program DB number on the IP247	Entry 1
DW 1	+00025	KF	Length in words	
DW 2	000,078	KY	Machining program DB number on the IP247	Entry 2
DW 3	+00044	KF	Length in words	
DW 254	000,165	KY	Machining program DB number on the IP247	Entry 128
DW 255	+00473	KF	Length in words	

DB x+1:

		Recommended data format		
DW 0	000,050	KY	Machining program DB number on the IP247	Entry 129
DW 1	+00126	KF	Length in words	
DW 2	000,092	KY	Machining program DB number on the IP247	Entry 130
DW 3	+00145	KF	Length in words	
DW 252	000,187	KY	Machining program DB number on the IP247	Entry 252
DW 253	+00035	KF	Length in words	

Example 2

There are three machining programs on the positioning module, the destination start address in DBx is data word DW253. The directory is then stored as follows:

DB x:

		Recommended data format		
DW 0				
DW 253	000,001	KY	Machining program DB number on the IP247	Entry 1
DW 254	+00025	KF	Length in words	
DW 255			Not written to, must however exist!	

DB x+1:

		Recommended data format		
D	000,078	KY	Machining program DB number on the IP247	Entry 2
DW 1	+00044	KF	Length in words	
DW 2	000,165	KY	Machining program DB number on the IP247	Entry 3
DW 3	+00473	KF	Length in words	

6.3.8.6 Occupation of the Data Word when Reading Actual Values

The actual values (actual position value and distance to go) can be read from the IP247 with FBI 65 using mode

6A = 66.

They require a constant length of six data words.

The actual values are stored in data block DBx from data word DWn as follows:

DW n	0000	**KH**	Actual position [urn]: **47.287mm**
DW n+1	B8B7	**KH**	
DW n+2	0	KH	Not used
DW n+3	0	KH	
DW n+4	0000	KH	Distance to go [urn]: **12.602mm**
DW n+5	313A	KH	

The actual position value and the distance to go are interpreted as 32-bit fixed point numbers. Negative values are stored as 32-bit fixed point numbers in 2's complement.

It is, however, possible to output the actual values as BCD numbers. This is achieved with direct parameter assignment by means of the block parameter

BA = KY 255,66

or with indirect parameter assignment using the job field of the axis data block in data word

DWn = KY 255,86

The actual values are then stored as seven decade BCD numbers with sign in the destination DB (parameter Z-DB).

In the BCD format, the maximum value which can be represented with a 32-bit number is +/- 9999999 urn (0.0001 in, 0.001 degrees). If a conversion from binary to BCD is not possible (representable BCD range exceeded), the value is entered in the data block as a 32-bit fixed point number (2's complement). The monitoring values which could not be converted can be read from flag byte FY249 (=> Section 7,2 "Troubleshooting").

If you assign parameters so that the monitoring values reed are output as a BCD number when FB165 is called, remember the following points:

Since the actual values are always first entered as a binary number in the data block and later converted to a BCD number, if the cyclic reading of the actual values is not coordinated, the value might be read once as a BCD number or once as a binary number.

To prevent this, actual values to be output as BCD values should only be read when the appropriate "trigger flag" (BA 66) has signal state "O".

With indirect parameter assignment, the actual values must only be evaluated when data word DWn = KH0000.

Structure of the BCD number:

	Sign	Decades						
	V V V V	10^6	10^5	10^4	10^3	10^2	10^1	10^0
bit:	31 28	27 . . 24	23 . . 20	19 . . 16	15 . . 12	11 . . 8	7 . . 4	3 . . 0

DW n	DW n+1
DD n	

e.g.
actual position value

6.3.8.7 Structure of the Machine Data Overview

The machine data overview is an extended machine data directory. You can read the overview from the IP247 with

BA = 66

Fifteen data words are required. If there is no machine data record on an axis, the data words are assigned the value zero.

The machine data overview stored in data block DBx from data word DWn is as follows:

DW n	+00001	KF	Machine data DB number on the IP247	Axis 1
DW n+1	+00000	KF	Module number	
DW n+2	+00001	KF	Axis number	
DW n+3	+00070	KF	Length of machine data DB in words	
DW n+4	+00023	KF	Machine data error	
DW n+5	+00002	KF	Machine data DB number on the IP247	Axis 2
DW n+6	+00000	KF	Module number	
DW n+7	+00002	KF	Axis number	
DW n+8	+00070	KF	Length of machine data DB in words	
DW n+9	+00000	KF	Machine data error	
DW n+10	+00003	KF	Machine data DB number on the IP247	Axis 3
DW n+11	+00000	KF	Module number	
DW n+12	+00003	KF	Axis number	
DW n+13	+00070	KF	Length of machine data DB in words	
DW n+14	+00000	KF	Machine data error	

6.3.9 Technical Data

	S5-115U	S5-135U	S5-150U	S5-155U
Block number	165	165	165	165
Block name	PER:PDAT	PER:PDAT	PER:PDAT	PER:PDAT
Library number	P71200-S5 165-D-2	P71200-S9165-D-2	P71200-S4165-!3-2	P71200-S6165-D-2
Call length	13 words	13 words	13 words	13 words
Block length	706 words	573 words	614 words	659words
Nesting depth	1	1	1	1
Secondary blocks	Integrated handling blocks	Handling blocks	Handling blocks	Handling blocks
Occupation of data area	-15 data words from parameter DWNR of the axis data block DBNR - DB165 occupied from data word DW0 to DW47 - In indirect parameter assignment via DBx: data word DW1 and DW2			
Occupation of flag area	FY206 to FY255 scratchpad flags	FY218 to FY255 scratchpad flags	FY200 to FY255 scratchpad flags	FY200 to FY255 scratchpad flags
Occupation in system data area	none	RS60 and RS61	yes	yes
System statements	yes	yes	yes	yes
Miscellaneous	Interrupts blocked at times in the FB by commands IA/ RA. An 1A command is cancelled by this (also S5-1 55)	Handling blocks FB120 SEND, FB121 RECEIVE FB122 FETCH and FB125 SYNCHRON must be loaded.	Handling blocks FB180 SEND, FB181 RECEIVE FB182 FETCH and FB185 SYNCHRON must be loaded.	Handling blocks FB120 SEND, FB121 RECEIVE FB122 FETCH and FB125 SYNCHRON must be loaded.

Maximum runtimes in ms: Dependent on the BA and the selected field size (standard value = O)

	S5-115U CPU 941	CPU 942	CPU 943	CPU 944	S5-135U CPU 922	CPU 928	CPU 928/2	S5-150U	S5-155U
Write and delete jobs BA20 to BA24	25.7 to 32.4	9.6 to 17.4	5.5 to 13.6	3.8 to 9.1	[illegible].0 to 3.0	1.0 to 6.6	0.5 to 6.3	1.7 to 5.8	2.0 to 5.9
S5 cycles (min.)	2-6				2-4			2-4	2-4
Read jobs BA64 to BA70	25.8 to 34.5	88.0 to 15.0	6.5 to 13.0	3.2 to 17.5	[illegible].0 to 3.0	1.0 to 6.6	0.5 to 6.3	2.5 to 9.2	1.8 to 6.1
With BA66 max.	81.0	21.2	14.8	8.3					
S5 cycles (min.)	2-4				2-4			2-3	2-3
Extra runtime of FB with direct parameter assignment	2.2	1.3	1.7	0,07					

6.3.10 Notes on Starting Up the IP247 Positioning Module via the PC Interface

if you startup the positioning module via the PC interface, the system identification (SYSID) must be transferred to the module before the machine data are transferred. After power up, the following defaults are set:

- module number = **0,**
- slot number = O and
- page number = **0.**

The machine data can then be transferred to the module, Only machine data with a module number (DLn+3 of the machine data DB) identical to the module number in the SYSID (DRn+6 or DRn) are permitted.

If the machine data are valid, the axis can be moved in the JOG mode or incremental relative mode. Absolute targets can only be approached after calibration of the axis (mode 5).

There is no "overwrite mode" for machine data. If an axis requires new machine data, the following operations must be carried out:

- delete the "old" machine data record on the axis (BA21)
- transfer the "new" machine data record to the axis via the page assigned to it (BA20).

You can assign any permitted DB number to a machine data record on the I P247 (DB0 to DB255). There is, however, only ever one machine data record for an axis. The assignment of the machine data records to the axes is made using the axis number in the machine data record (DRn+3). The axes can therefore be assigned a machine data record with the same DB number, however, with a different axis number (=> Section 4.3.12 "Enter Machine Data" or Section 4,3.13 "Delete Machine Data"), For more information about machine data, refer to Section 2.5 "Machine Data and their Structure".

A maximum of 255 machining programs (DB0 to DB255) can be stored on the positioning module. An existing machining program cannot be overwritten. If you wish to modify **a** machining program stored on the IP247, then a certain procedure must be adhered to, just as with the machine data:

- output the machining program (machining program DB) from the IP247 to the PC memory, unless it already exists there (BA 69),

- delete the machining program (machining program DB) on the IP247 (BA 23).

- transfer the modified machining program to the IP247 (BA 22).

A machining program is not restricted to an axis. All three axes can execute the same machining program simultaneously. Machining programs can only be transferred and deleted via the data channel (4th page). A machining program can, however, only be deleted when no other axis is using this machining program (=> Section 4.3.17 "Executing Machining Programs"). For more information about machining programs, refer to Section 2.6 "Machining Programs and their Structure".

6.3.11 Using the Function Block

[n cyclic operation it is not possible to address a module both with indirect and direct parameter assignment.

Function block FBI **65** works with data block DB1**65.** This must be installed up to and including data word DW47. No particular assignment of the data words is necessary.

Data block DB165 is divided into two areas, in which data words DW1 and DW2 are reserved for indirect assignment of parameters to the function block. Data words DW3 to DW47 are the working area for FBI 65. You must **not** change the working area.

When assigning parameters to FB1 65, remember that the specified data blocks of the source and destination parameters must exist and must be adequately long.

The axis data block (parameter DBNR) must have the following length:

length = parameter DWNR + 14

The DB/DX number and the DW number can be selected as required.

The data block is setup with a programmer, e.g. with the PG 685, STEP 5 under S5-DOS with the following commands (see corresponding documentation):

<F1>	(input)	
<F1>	(block)	
Pc	(input device)	
DB160	(block)	enter key <|>

DWO: KY = 000,000
DW 1 : KY= 000,000
DW2: KY = 000,000
DW3: KY = 000,000

DW14: KY = 000,000 enter key <|>

FBI 65 can be called conditionally. The call must be made cyclically until the assigned mode is completely executed. The mode (job) runs

– with direct parameter assignment as long as the parameter ANST has the signal state”1”,
– with indirect parameter assignment as long as data word DWn is not equal to KH0000 in the axis data block.

You must ensure that the parameter assignment is not overwritten while a mode is being executed.

With indirect parameter assignment, the current data block must be open and supplied with the parameters DBNR (DW1) and DWNR (DW2) before the function block FB1 *65 is* called.

The positioning module IP247 does not service interrupts.

6.4 Examples

Note

You can use the example program without modifications only on the **IP246** positioning module.

Since the data transfer with the IP247 uses page numbers n to n+3 (data channel), where n is the selected page number (base address, switch *S2)* you must change the example as follows:

In the start-up OBS:
synchronize page numbers n to n+3.

Modification in the example of direct parameter assignment:
in FB51, in segment 4, the parameter SSNR of FBI 65 must be changed as follows: for modes 20, 21 and 66 a selected page number between n and n+2 should be entered. For the remaining FBI 65 modes the page number n+3 (data channel) should be entered.

Modification in the example of indirect parameter assignment:
in data block DB1 66, data word DW6 must be overwritten with a selected page number between n and n+2 when the modes 20, 21 and 66 are called. For the remaining FB165 modes the page number n+3 (data channel) should be entered.

6.4.1 General Notes on the Examples

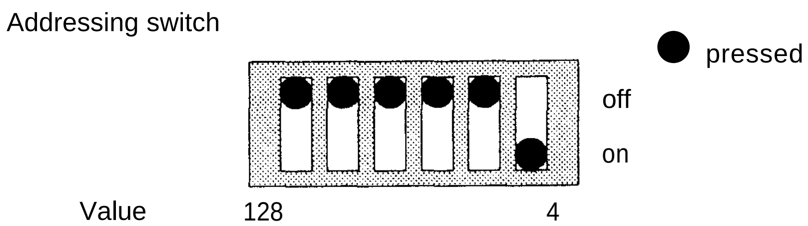
The following examples of the use of FBI 64 and FBI 65 are on the diskette supplied. The examples can be loaded completely in the PC memory to test the module. They illustrate a possible parameter assignment for an axis,

All the required blocks with the exception of the handling blocks are available. The diskette also provides a complete “program framework” which you can use.

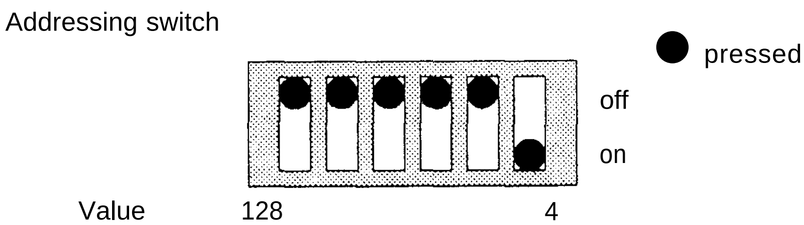
6.4.2 HardwareRequirements

The following hardware is required to implement the examples:

one digital input module 6ES5420-.... coded as IB4 *)



one digital output module 6ES5441. . . . . coded as QB4

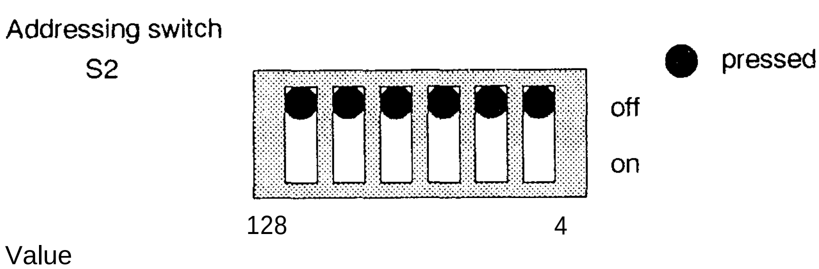


*) The following applies for the S5-1 15U:

one digital input module 6ES5420-.... (fixed slot addressing) inserted in slot number 1 in the central controller (IB4 to IB7).

one digital output module 6ES5441-.... (fixed slot addressing) inserted in slot number 2 in the central controller (QB8toQB11).

one IP247 positioning module coded as page number O (=> Section 3.3.2 “Setting the Module Address”) inserted in a CP slot in the central controller of the programmable controller.



The remaining jumpers on the IP247 must be set for the specific equipment (=> Section 3.3.2 “Setting the Module Address”).

6.4.3 Assignments for the Examples

6.4.3.1 Digital Inputs: (valid for all Programmable Controllers)

IB4 **BA** Mode with indirect parameter assignment in format KF

Mode with direct parameter assignment:

		FBI 64	FBI 65
I 4.0	REF	Reference point	ReadSYSID
I 4.1	TIPP1	JOG 1	Read machine data directory
I 4,2	TIPP2	JOG 2	Read actual values
I 4.3	SMR	Incremental relative	Read machine data
I 4.4	IW	Read actual value	Read machine data overview
I 4.5	SA	Not used with IP247	WriteSYSID
I 4.6	RW	Read distance to go	Write machine data
I 4.7	LLOE	Clear cyclic monitoring	Delete machine data
I 5.0	STAR	STARTcommand	
I 5.1	STOP	STOP command	
I 5.2	VORW	FORWARDcommand	
I 5.3	RUCK	REVERSEcommand	
I 5.4	UEBN	Enter data command	
I 5.5	BCD	Output in BCD code	
I 5.6	INDI.AUF	Indirect param. ass. enter job FBI 65	
I 5.7	DIR. AUF	Direct param. ass, trigger job FBI 65	
I 6.0	POS/PDAT	0 = execute FB164 / 1 = execute FB165	
I 6.1	IN/DI	O)= indirect / 1 = direct parameter assignment	
I 6.2		Overwrite DB withKHFFFF(FB191 and FB1 92/FB165)	
I 6.3	RFEH	Clear latching error	
I 6.4			
I 6.5			
I 6.6			
16.7			

If none of the inputs I **4.0** to I **4.7** has signal state"1" in FB1 64, then the set mode is "axis off" (mode **4)** with direct parameter assignment.

6.4.3.2 Digital Outputs: (valid forS5-135U,S5-150U and S5-155U)

Q 4.0	**PAFE**	Parameter assignment error FBI 64 andFB165
Q 4.1	**BFEH**	Module error FB164 and FB165
Q 4.2	TBIT	Active bit FB164
Q 4.3		
Q **4.4**		
Q 4.5		
Q 4.6		
Q 4.7		

QB5	PAFE	Image of the PAFE byte FY255, latching
Q 6.0 Q 6.1 Q 6.2 Q 6.3	PAFES	PAFE (latching) FBI 64and FBI 65

6.4.3.3 Digital Outputs: (valid for S5-115U)

Q 8.0	PAFE	Parameter assignment error FB164 and FB165
Q 8.1	BFEH	Module error FBI 64 andFB165
Q 8.2	TBIT	Active bit FBI 64
Q 8.3		
Q 8.4		
Q 8.5		
Q 8.6		
Q 8.7		
QB9	PAFE	Image of the PAFE byte FY255, latching
Q 10.0 Q 10.1 Q 10.2 Q 10.3	PAFES	PAFE latchingFB164and FBI 65

6.4.3.4 Occupation of the Data Area

The data blocks DB150, DB151 and DB152 are occupied from DW0 to DW32. These data blocks are used to save the scratchpad area and the free system data area.

In theS5-155U, the data block DB255 must be specified with a length of 826 words.

6.4.3.5 Occupation of the Flag Area

F 0.0	NULL	"RLO 0" flag
F 0.1	EINS	"RLO 1" flag
FY4		Corresponds to IB4
FY5		Corresponds to IB5
FY6		Corresponds to IB6
FY7		Corresponds to IB7
FYI 4		Corresponds toQB4orQB8withtheS5-115U
FYI 5		Corresponds toQB5orQB9withtheS5-115U
FY16		Corresponds to QB6 or QB1 O with the S5-1 15U
FYI 7		Corresponds toQB7orQB11 with the S5-1 15U

FY50	RBTR	Mode checkback signal
FY51	RM-FKT	M function checkback signal
FY52	RPOS	Module checkback signals
FD60	ANZ	Condition code bits of the monitoring job
FY99	PAFE	SYNCH RON PAFEbyte
FYI 00	BTR	Mode selection
FYI 01	BEF	Command selection
FYI 02	TBIT	Image job active
FY105	FLM/I MP	Signal edge and pulse flags

Scratchpad flags from FY200 to FY255

6.4.3.6 BlockAssignments

OB1	ZYK	Cyclic program execution
OB2	IRA	Process interrupt servicing IR-A or I 0.0
OB13	WECK	Time interrupt servicing
OB20	NEUSTAR	Cold restart at programmable controller (not with S5-1 15U)
OB21	MANWIED	Manual warm restart/cold restart with S5-1 15U
OB22	AUTWIED	Automatic warm restart

FB50	INDX.164	Example indir.param. ass. FBI 64 via DX block
FB51	IP247DIR	Example dir. param. ass.FB165
FB52	IP247IND	Example indir.param. ass, FB165
FB53	IP247DI	Example dir. param. ass.FB164
FB54	IP247IN	Example indir.param. ass. FB164
FB120	SEND	Handling blockS5-135U/1 55U
FB121	RECEIVE	Handling block S5-135U/1 55U
FB122	FETCH	Handling block S5-135U/1 55U
FB123	CONTROL	Handling block S5-135U/1 55U
FB124	RESET	Handling block S5-135U/1 55U
FB125	SYNCHRON	Handling block S5-135U/1 55U
FB151	BS-RETT	Save RS60to RS63
FB152	BS-LAD	Load RS60to RS63
FB164	PER:POS	Standard FB for control of the positioning module
FB165	PER:PDAT	Standard FB for data transfer
FB180	SEND	Handling block S5-150U
FB181	RECEIVE	Handling block S5-150U
FBI 82	FETCH	Handling block S5-150U
FB183	CONTROL	Handling block S5-150U
FB184	RESET	Handling block S5-150U
FB185	SYNCHRON	Handling block S5-150U

FB244	SEND	Handling block S5-115U
FB245	RECEIVE	Handling block S5-115U
FB246	FETCH	Handling block S5-115U
FB247	CONTROL	Handling block S5-115U
FB248	RESET	Handling block S5-115U
FB249	SYNCHRON	Handling block S5-115U
DB104	SMDAT	Write machine data
DB106	SPRG	Write machining program
DB107	SSYS-ID	WriteSYSID
DB150	RETOB2	Save flags OB2
DB151	RETOB13	Save flags OB 13
DB152	RETANL	Save flags OB21/OB22 (not required with S5-115U)
DB160	IP246AN1	User DB
DB161	IP246AN2	User DB (not used in example)
DB164	IP-FB164	Fixed working DB forFB164
DB165	IP-FB165	Fixed working DB forFB165
DB166	IP246AN3	User DB FB1 65, indirect parameter assignment
DB167	IP246AN4	User DB FB1 65, direct parameter assignment
DB200	LMDIR	Read machine data directory
DB201	LPRGDIR	Read machining program directory
DB203	LIW	Read actual values
DB204	LMDAT	Read machine data
DB205	LMDATÜB	Read machine data overview
DB206	LPRG	Read machining program
DB207	LSYS-ID	ReadSYSID
DX160	IP246AN3	User DX (only with S5-135U and S5-155U)
DX161	IP246AN4	User DX (only withS5-135U andS5-155U)

6.4.4 Schematic Diagrams of the Organization Blocks (Program Framework)

6.4.4.1 OB1

copy ID4 to FD4

yes — direct param. ass. and execute FB164 — no
F 6.1 = 1 and F 6.0= O

Call FB53 | ———

yes — indirect param. ass. and execute FB164 — no
F 6.1 = 0 and F 6.0 = 0

Call FB54 or FB50 (DX) only withS5-135U andS5-155U | ———

yes — direct param. ass. and execute FBI 65 — no
F 6.1 = 1 and F 6.0 = 1

Call FB51 | ———

yes — indirect param. ass, and execute FB165 — no
F 6.1 = O and F 6.0 = 1

Call FB52 | ———

Call FBI 92: copy DB

yes — Clear error — no

FY15 = O, reset F 16.0
Copy FD14 to QD4 (QD8 with S5-1 15U) | ———

6.4.4.2 The Interrupt OBs

Process interrupt 06s and time interrupt OBs
Save flags -> FY200 to FY255 Save operating system data (S5-135U)
User program if interrupt
Load operating system data (S5-135U) Load flags -> FY200 to FY255
END

6.4.4.3 OB21 and OB22 with S5-115U
OB20 and OB22 with S5-135U
OB20 withS5-150U andS5-155U

F 0.0 = RLO "O" F 0,1 = RLO"1"
Synchronize interface
User program
END

6.4.4.4 OB21 withS5-135U,S5-150U andS5-155U
OB22 withS5-150U

STP (direct change to stop state)

6.4.5 Example of Function Block FB164

In the example, function block FBI **64** PER:POS works with the function blocks FB53 and FB54 and with data blocks DB160 and DB1 **64.** The following requirements must be met:

– input of 16.0 must have signal state “O”

– the type of parameter assignment can be selected via input 16.1:
signal state “O” = indirect parameter assignment via FB54
signal state”1” = direct parameter assignment via FB53

The function block FB53 shows direct parameter assignment to FBI 64; FB54 shows indirect parameter assignment. With indirect parameter assignment, the actual operands are stored in DB160 from data word DW1 to DW7.

The example of indirect parameter assignment covers all possible modes, whereas the example for direct parameter assignment is restricted to the following modes:

reference point approach

- JOG 1 and 2
- incremental relative
- read actual value and distance to go
- disable monitoring
- axis off

6.4.5.1 Function Block FB53 (Schematic Diagrams)

The function block FB53 shows the use of the function block FBI 64 with direct parameter assignment via the block parameters. FBI **64** must be called once for each required mode.

Segment 1:

```
Parameter list: STRT 1, W
```

Segment 2:

Load actual operand in the scratchpad flag area
Parameter STRT to FW200

Is a job still active?

yes	no
———	Set FYI 00= O

Segment 3:

Condition	Function	Step
F100.0 or F102.0	Reference point	X=5
F100,1 or F102.1	JOG 1	x=1
F100.2 or F102,2	JOG 2	x=2
F100,3 or F102,3	Incremental relative	X=7
F100,4	Read actual position value	X=71
F100.5	Free	x=72
F100.6	Read distance to go	x=73
F100.7	Disable	x=74
F102.4 else	Axis off	x=4

Call function block FB164 depending on command

```
              FB164
NAME  :       PER : POS
SSNR  :             KF+0
DBNR  :             KY0, 160
DWNR  :             KF+8
BA    :             KF+x
STAR  :             F 101.0
STOP  :             F 101.1
VORW  :             F 101.2
RUCK  :             F 101.3
URBN  :             F 101.4
BCD   :             F 101.5
PAFE  :             F 14.0
BFEH  :             F 14.1
TBIT  :             F 14.2
BTR   :             FY50
MFKT  :             FY51
RMLD  :             FY52
ANZG  :             FD60          or    ANZ1 : FW60 } with the S5-115U
                                        ANZ2:  FW62 }
```

Assign auxiliary flags F 102,0 to F 102,4,
as long as output F 14,2 “TBIT” still has signal 1,

Segment 4:

yes	**Parameter** PAFE O -> 1 edge? (FY 105)	no
	Set PAFE latching F 16.0 Store PAFE byte (FY255) in FY15	———

Segment 5:

BE

6.4.5.2 Function Block FB54 (Schematic Diagrams)

Function block FB54 shows the use of function block FBI 64 with indirect parameter assignment via the data block DB1 60. The assignment of the data words is fixed!

Segment 1:

Parameter list: STRT I,W

Segment 2:

Load actual operand in the scratchpad flag area: STRT -> FW200

Segment 3:

Call user DB (DB160)

Formulate "job" for axis:

FY200	->	DW1	BA
FY201	->	DW2	COMMANDS
KYO, 160	->	DW4	DBNR
KF+8	->	DW5	DWNR
KB0	->	DW6	SSNR

Cond. code BCD? F 201.5

yes	no
L KB1	L KB0

T DR7

Call FBI 64

```
NAME  : PER : POS
SSNR  :       KF+0
DBNR  :       KY0,0
DWNR  :       KF+0
BA    :       KF+0
STAR  :       F 0.0
STOP  :       F 0.0
VORW  :       F 0.0
RUCK  :       F 0.0
UEBN  :       F 0.0
BCD   :       F 0.0
PAFE  :       F 14.0
BFEH  :       F 14.1
TBIT  :       F 14.2
BTR   :       FYO
MFKT  :       FYO
RMLD  :       FYO
ANZG  :       FDO     or   ANZI : FWO
                           ANZ2: FWO    }  with the S5-1 15U
```

Segment 4:

yes	Parameter PAFE 0 -> 1 edge? (FY 105)	no
	Set PAFE latching **F 16,0** **Store** PAFE **byte** (FY255 **) in FYI 5**	———

Segment 5:

BE

6.4.6 Example of Function Block FB165

In the example, function block FB165 works with the function blocks FB51 and FB52 and with the data blocks DB1*04,* 106 and 107 (for write data), DB165 (working DB), DB166, 167 (axis DBs) and DB200 to 207 (for read data). The following requirements must be met:

– input I 6.0 must have signal state"1"

– the type of parameter assignment can be selected via input I 6.1:
  signal state "O" = indirect parameter assignment **via** FB52
  signal state"1" = direct parameter assignment via FB51.

Function block FB51 shows the direct parameter assignment of FB165, FB52 shows indirect parameter assignment. With indirect parameter assignment, the actual operands (job field) are stored in data block DB166 from data word DW1 to DW6.

The example of indirect parameter assignment covers all possible modes, whereas the example of direct parameter assignment is restricted to the following modes:

- read SYSID
- read machine data directory
- read actual values
- read machine data
- read machine data overview
- write SYSID
- write machine data
- delete machine data

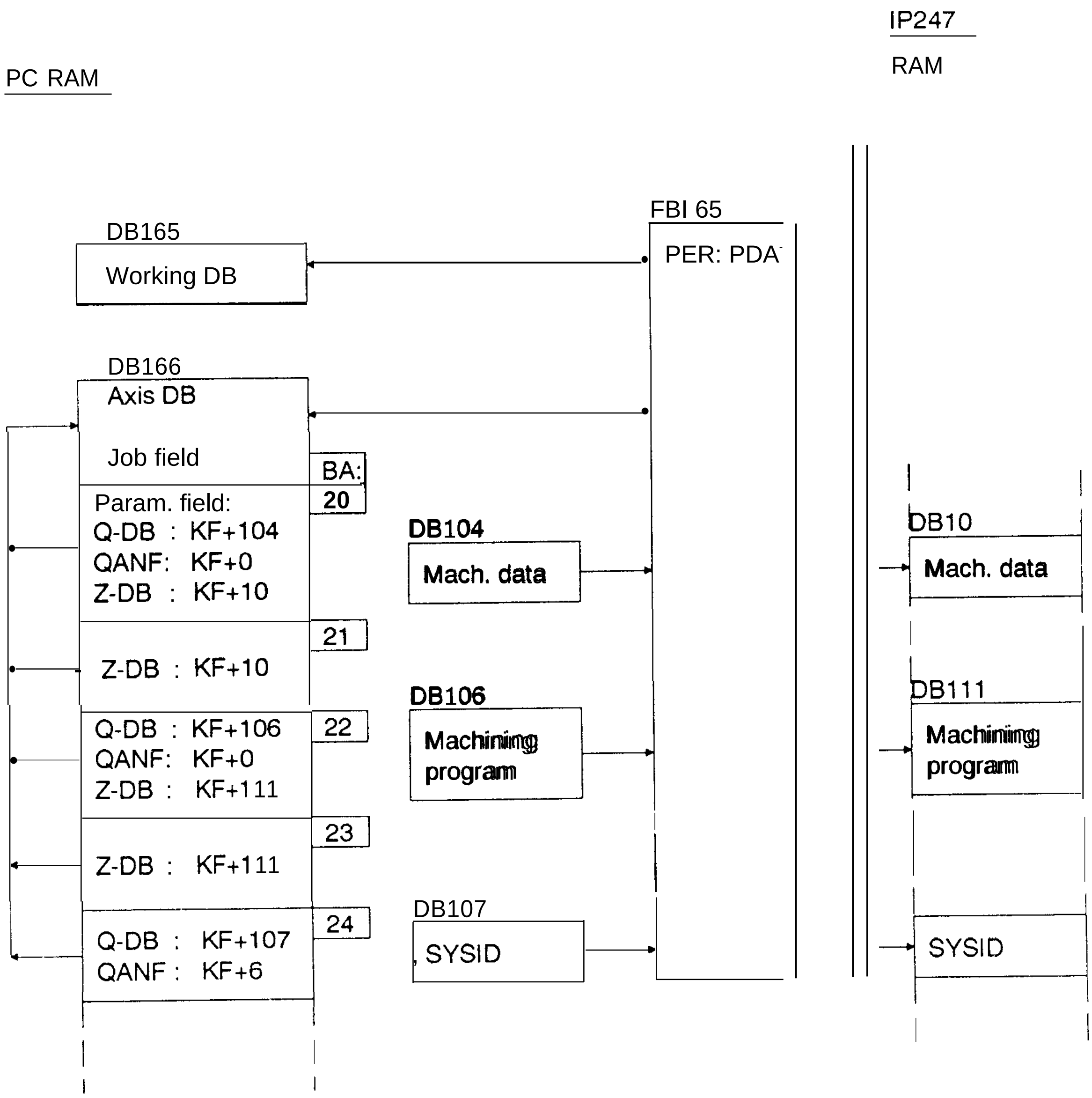
6.4.6.1 Overview of the Relationship between the Mode and the Data Blocks in the RAM of the CPU and the Positioning Module

Machine data and machining programs are stored on the positioning module as data blocks.

The absolute DB and DW numbers refer to the example.

Writing data to the IP247 and deleting data on the IP247

Modes BA: **20** to **24** (=> Part **4** “Functions”)



To be able to transfer a data record to the positioning module, you must supply the following parameters to the function block:

Mode (BA), source (Q-DB, QANF) and destination parameter (Z-DB)

Parameters not required are assigned KF+0.

Example:

Parameter assignment to transfer machine data (PC->I P247):

- Mode BA :	KF+20	
- Source DB Q-DB :	KF+104	= DB104 PC memory
- Source start QANF :	KF+0	= DW0 from DW0
- Dest. DB Z-DB :	KF+1O	= DB10 IP247 memory
- Dest. start ZANF :	KF+0	= irrelevant

The machine data DB (DB104) in the PC memory is transferred as machine data DB (DB1O) to the IP247 memory.

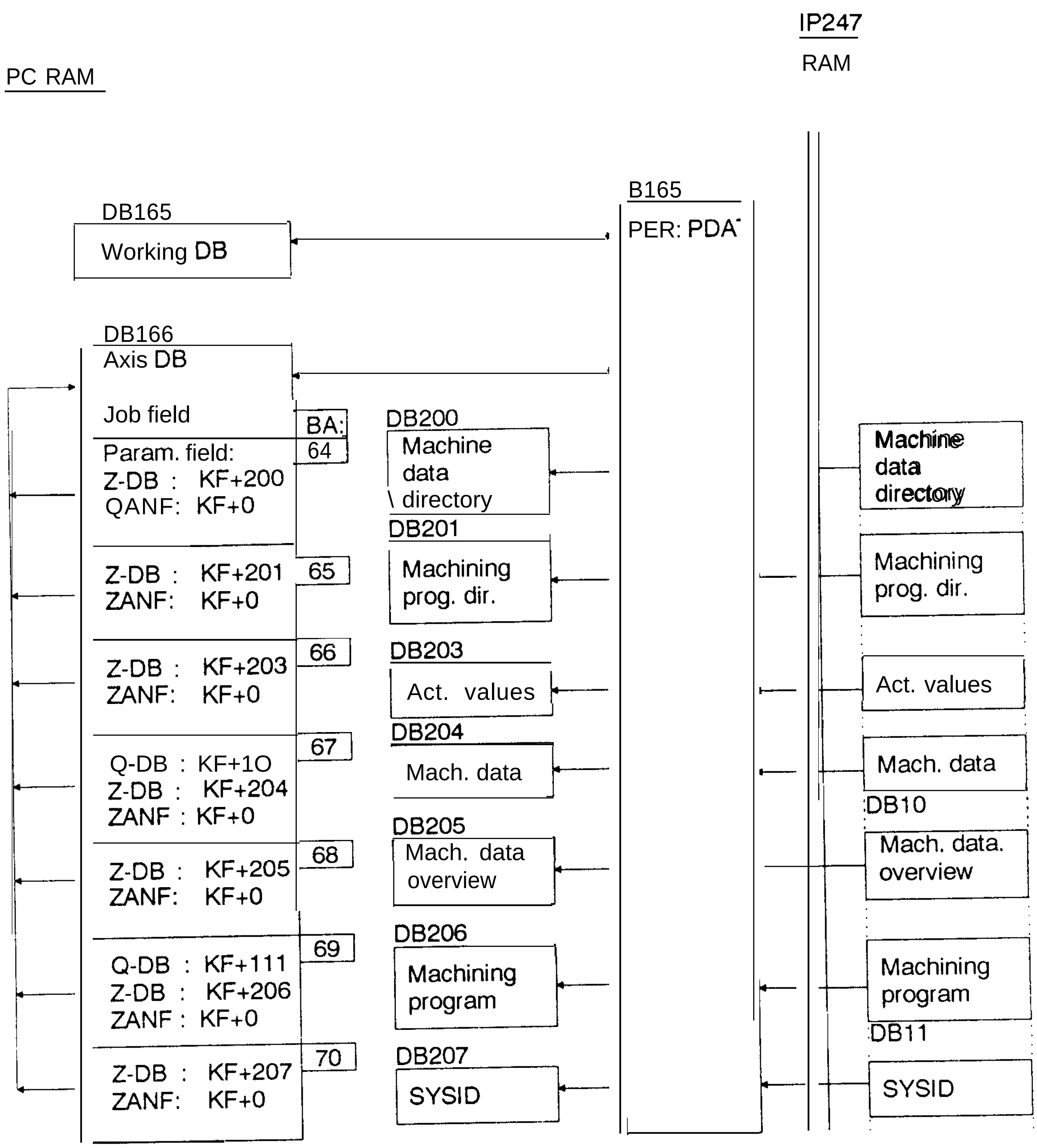
If the machine data record is to be deleted, the following parameter assignment must be made:

- Mode BA :	KF+21	
- Source **DB Q-DB:**	KF+0	= irrelevant
- Source start QANF :	KF+0	= irrelevant
- Dest. **DB** Z-DB :	KF+1O	= DB10 IP247 memory
- Dest. start ZANF :	KF+0	= irrelevant

The machine data DB (DB1O) on the positioning module is deleted.

Reading data from the IP247

Modes BA 64 to 70 (=> Part 4 "Functions")



To be able to read a data record from the positioning module, the following parameters must be specified for the function block:

Mode (BA), source (Q-DB) and destination parameters (Z-DB, ZANF)

Parameters not required are assigned KF+0.

Example:
Parameter assignment to read machine data (I P247->PC):

- Mode BA:	KF+67	
- Source DB Q-DB:	KF+1O	= DB1O IP247 memory
- Source start QANF :	KF+0	= irrelevant
- Dest. DB Z-DB:	KF+204	= DB204 PC memory
- Dest. start ZANF :	KF+0	= DW0 from DW0

The machine data DB (DB1O) on the positioning module IP247 is stored as machine data DB (DB204) from data word DW0 in the PC memory.

6.4.7 Function Block FB51 (Schematic Diagrams)

The function block FB51 shows the use of the function block FB165 with direct parameter assignment via the block parameters. FBl 65 must be called once for all required modes.

DB167 from DWOtoDW14 is used as the axis data block.

Segment 1:

Parameter list: STRT 1, W

Segment 2:

Load parameters in -> FW200

Signal edge evaluation (rising) F 201.7 { F 105.1 EDG, F 202.1 PUL }

Segment 3:

Trigger byte = zero? (FY 100)
yes | no

Trigger pulse job?
yes | no

FY200 --> FY1OO (PAR: ANST) | BEC | ———

Segment 4:

Job execution :

A F 100. x
JC FB165

Call according to priority!

ANST:

F1 00.0	F1 00.1	F100.2	F1 00.3	F1 00.4	F100.5	F100.6	F1 00.7	else
BA=70 Z-DB= 207 ZANF=0	BA=64 Z-DB= 200 ZANF=0	BA=66 Z-DB= 203 ZANF=0	BA=67 Q-DB= 0,10 Z-DB= 204 ZANF=0	BA=68 Z-DB= 205 ZANF=0	BA=24 Q-DB= 107 Q-ANF=6	BA=20 Q-DB= 104 Q-ANF=0 Z-DB= 10	BA=21 Z-DB= 10	
SYSID	Mach. data directory	Actual values	Mach. data	Mach. data overview	SYSID	Mach. data	Mach. data	B E u
read					write		delete	

FB 165 call depending on command

FB165

NAME : PER : PDAT
SSNR : KF+0
DBNR : KYO, 167 (DBn = DB167)
DWNR : KF+0 (DWn = DWO)
BA : KYx,Y
Q-DB :
QANF :
Z-DB :
ZANF :
PAFE : F 14.0
BEFEH : F 14.1

Parameters not required are assigned zero!

yes PAR:ANST = O ? no

yes Mode BA66? no

Copy monitored values

Set FY1OO = O

Segment *5:*

yes	Parameter PAFE 0 -> 1 edge?	no
	Set PAFE latching F 16.0 Store PAFE byte (FY255) in FY15	———

Segment 6:

BE

6.4.8 Function Block FB52 (Schematic Diagrams)

The function block FB52 shows the use of the function block FBI 65 with indirect parameter assignment via data block DB1 66. The assignment of data words (DWn to DWn+6) is fixed!

The pointer to the "job field" is entered in data block DB1 65. The data block number must be stored in data word DW1 and the data word number in data word DW2.

DB165	DW1	KY0,1 66	--> DB166 from DW1 to DW15
	DW2	KF+1	

Segment 1:

Parameter list: STRT 1, W

Segment 2:

Load parameters in --> FW200;

Edge evaluation *(rising)* F 201.6 { F 105.0 EDG
F 202.0 PUL

Segment 3:

C DB166 User DB for FB165
Coordination word >< Zero? (DWn = DW1)
= F 202,1 (aux. flag for condition call)

no | yes

Pulse "enter job" present?

yes | no

Write or delete job, 20 <=BA <=24 ?

yes | no

Assign calculated values:

DW job=
b+((BA-a)x4)
where a=20 and b=16
BA --> FY200
a --> FY203
b --> FY204

Read job?
64 <= BA <= 70?

yes | no

BA --> FY200
a --> FY203
b --> FY204

Calculate the address of the job field
of the source/destination parameters

(DWn --> FY205)
(DWn+2 --> FY206)

Formulate job: (DB166)

6A --> DW1 BA
DDn --> DD2 Q-DB, QANF
DDn+2 --> DD4 Z-DB, ZANF

Assign job: C DB165

KY0,166 --> DW1 (DBn =DB166)
KF+1 - - > DW2 (DWn=DW1)

Assign job C DB165
KY0,166 --> DW1 (DBn-DB166)
KF+1 --> DW2 (DWn-DW1)

Segment 4 :

Call DB165

yes	Auxiliary flag for conditional call = “O”? (A F 202.1)	no
	Call function block FBI 65 FB165 NAME : PER : PDAT SSNR : KF+0 DBNR : KY0,0 DWNR : KF+0 BA : KY0,0 Q-DB KY0,0 QANF KF+0 Z-DB KY0,0 ZANF KF+0 PAFE F 14.0 BFEH F 14.1	——

Segment 5:

yes	Parameter PAFE 0->1 edge? (FY165)	no
	Set PAFE latching F 16.0 Store PAFE byte (FY255) in FYI 5	——

Segment 6:

BE

7 Planning, Installation and Service

7.1 Planning

7.1.1 BasicConsiderations

- Which torque characteristic and which maximum torque are required?
- Can a stepper motor achieve the required torque?
- Will large fluctuations in load occur which can lead to loss of steps? (Load torque briefly greater than motor torque.)
- Will feedback (monitoring) of the actual axis position via additional position detectors be necessary? (Possibly stepper motor with integrated position encoder.)
- Is it advisable to use a drive unit which can detect and correct loss of steps?

7.1.2 Selection Criteria for the Stepper Motor

Mechanical dimensions and designs are not dealt with here.

- What is the maximum torque?
- Up to what pulse frequency can the motor achieve the required torque?
- How high must the step number of the motor be to achieve the required position resolution?

7.1.3 Determining the MotorCharacteristics

Plant data

required positioning resolution

$k=$ ____________ [μm/pul]

required traversing speed

$v_{max}=$ ____________ [mm/min]

maximum load torque of the shaft

$M_{max}=$ ____________ [Ncm]

The transmission ratio r on the spindle and the step number S of the motor must be selected so that their quotient produces the required resolution.

$$r = S \times k \quad => \quad k = \frac{r}{S}$$

S= [pul/rev]

r= [mm/rev]

The maximum pulse frequency f_{max} is obtained as follows:

$$f_{max} = \frac{v_{max}[mm/min]}{k \times 60\ [\mu m/pul]}$$

f_{max} = [kHz]

From the characteristics of the motors, you must now select a type capable of the required load torque at the calculated frequency f_{max} without loss of steps. You must also select a suitable power unit for the motor.

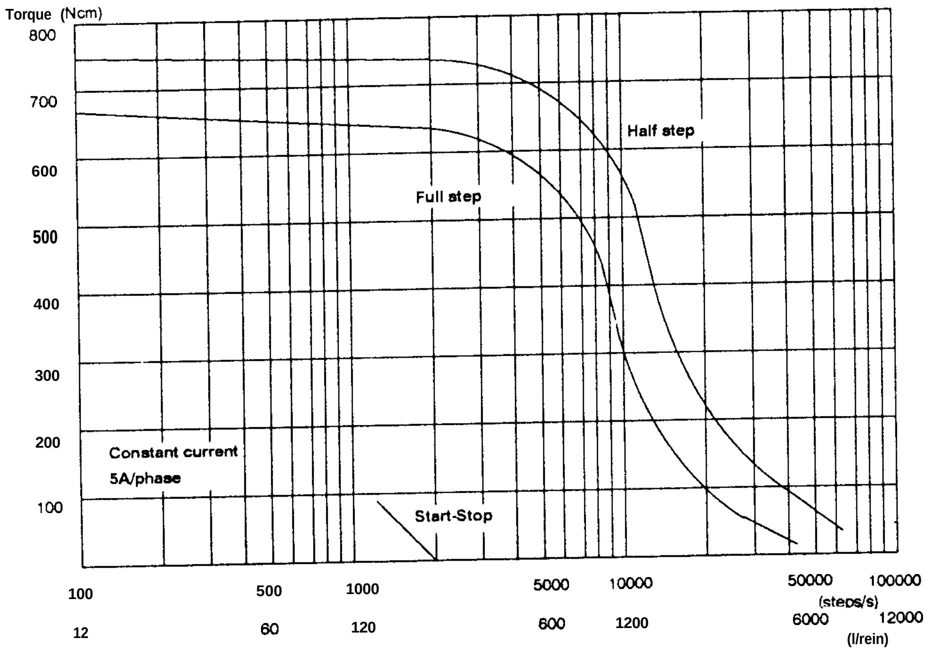


Fig. 7/1 Typical torque characteristics of a stepper motor

Note:

If the required torque characteristics can only be achieved in the half step mode with this motor, then for a given resolution, select a motor with half the step number. If you remain by the previously selected step number, twice the frequency will be required to reach the selected speed, since the resolution is halved.

Selecting the power unit

Signal from IP	Type	Pulse length	Signal at power unit	Required level [V]	Req. duration [ins]	Active level [high/low]
Clock pulse T	T T	1-31µs				
Direction level RP	RP RP	Voltage level				
Reset pulse RS	RS RS	100ms				

The signals listed above are available as 5 V differential signals and as 24 V signals. It is also possible to use a special voltage between 5 V and 24 V which must be applied externally. The voltage is set for all axes of the module. The active level (high or low) can be selected separately in the machine data for each channel.

When selecting the power unit, make sure that the maximum pulse frequency fmex can be processed without errors. To check that each power unit is ready for operation, there is a binary input per axis,

Ready message (BB) from power unit

The IP247 requires 24 V active high at its input or a floating contact which can be supplied with power by the IP247.

Voltage supply for a floating BB contact of the power unit (see above)

24 V/l 20 mA short-circuit proof

If the power unit does not output a 24 V ready signal and does not have a floating contact, then the 24 V output and the BB input must be jumpered at the cable end. The latter should be inside the power unit to monitor that the cable is connected.

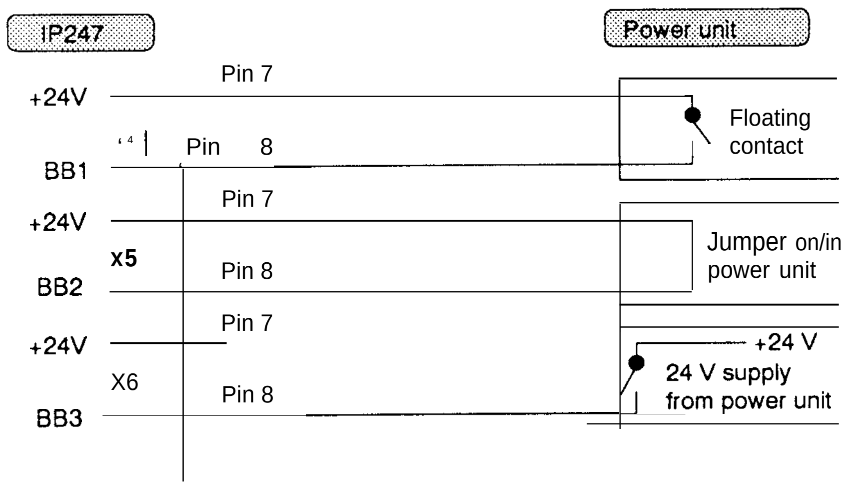


Fig. 7/2 Three possible ways of implementing the ready signal.

The I P247 does not evaluate any other signals from the power unit (in some cases, other signals can be evaluated by the CPU).

Signals which can be exchanged between the IP247 and the plant

Binaryoutputs:

Position reached:	24 V high active 120mA

Binary inputs:

Reference point:	24 V high active	
Limit switch 2x:	24 V high or low active (selectable)	
Ext. start/stop:	24V	falling edge = start
		rising edge = stop

The following information can also be evaluated by the CPU:

selected measurement system		[mm/inch/deg]
Axis is position (also as output)		[yes/no]
Reference point synchronized		[yes/no]
Axis in teach-in **mode**		[yes/no]
Reference point exists		[yes/no]
Machine data exists		[yes/no]
Job completed		[yes/no]
Actual position		
Distance to go		
Auxiliary functions (M-function)		

Current mode

This information can be processed in the user program and, if required, can be displayed via binary outputs or communications processors.

7.1.4 Planning the Machine Data

Axis number (plant-specific)

[1, 2,3]

Module number (must be the same for all three axes)

[0-999]

Measurement system

[mm,inch,deg]

Axis type (rotary/linear)

[rotary, linear]

Maximum frequnecy f_{max} (according to planning data)

f_{max} =

[40Hz-100kHz]

Start/stop frequency f_{ss} (from torque characteristics)

f_{ss} =

[1 Hz-1 OkHz]

Rate of frequency increase a
(should be selected as high as possible, see Manual)

a =

[0.020-2599Hz/ms]

Pulse duration t_p (according to planning data)

t_p =

1 -31µs

Number of excitation patterns
Number of phases x 2 (full step)
Number of phases x 4 (half step)

[4-40]

Polarity (of the clock pulse output TN)
Normal level high - negative edge is
Normal level low - positive edge evaluated

[pos./neg. edge]

Number S of steps per revolution
(Number of steps of the motor in the full step/half step mode set at the power unit)

S = [12-1000 'rev]

Transmission ratio r
(Distance travelled by the drive per motor revolution)

r = [0,012 - 400,000]

JOG speed 1 v_1

v_1 = [64999 mm/min]

JOG speed 2 v_2

v_2 = [64999 mm/min]

Incremental speed v_s

v_s = [64999 mm/min]

These speeds must be less than or equal to the maximum speed v_{max} from the planning data.

Reference speed v_{ref}

v_{ref} = [64999 mm/min]

$v_{ss} \leq v_{ref} \leq v_{max}$

Start/stop speed $v_{ss} = \dfrac{f_{ss} \times r \times 60}{S}$

Reference point synchronized
(=> Part 4 **"Functions"**) [yes/no]

Reference direction
(dependent on the plant or as required) [fwd./rev.]

Reference point coordinate x_{ref}
(dependent on the plant)

x_{ref} = [± 99999.999mm]

Software limit switch start x_A

x_A . [± 99999.999mm]

$x_A < x_{ref}$

Software limit switch end x_E

x_E = [± 99999.999mm]

$x_{ref} < x_E$

Polarity of limit switches (positive/negative)

BERO or normally open → positive
Normally closed → negative [positive/negative]

PC BCD-coded (yes/no)

[yes/no]

Tool length offset
(this can only be activated or deactivated in the machining program, cumulative=> Part 4 "Functions")

[± 99999.999mm]

Backlash compensation
(in multiples of the resolution) [0 - 64.999mm]

Zero offset 1 [± 99999.999mm]

Zero offset 2 [± 99999.999mm]

Zero offset *3*

	[± 99999.999mm]

Zero offset *4*

	[± 99999.999mm]

You can **only activate** these zero offsets in the machining program (**=>** Part *4* “Functions”).

7.1.5 Installation

7.1.5.1 PreliminaryRequirements

The programmable controller is correctly configured. The power supply has been connected according to the regulations (=> manual of the programmable controller).

> **Note**
>
> If a spindle or similar device is to be driven by the motor, **all the limit switches must be connected.**

- There must bean emergency stop switch to switch off the whole equipment. **ok** O
- Two limit switches signaling directly to the module. Here you can use either normally open or normally closed contacts. **ok** O
- Two normally closed contacts as limit switches, outside the limit switches mentioned above which either switch off the power unit directly or suppress the input pulse train of the power unit. **ok** O

You also require the following:

- The IP247 module
- A programmer, **PG 635, PG 675,** PG685, PG695, **PG730 or PG750 with the** S5-DOS **operating system**
- The COM247 communications software for your **PG**

7.1.5.2 Preparing the Module

Set the signal level required by your power unit on the module.

5 V differential inputs or optocoupler input	Connector X30 Jumper 2-4 inserted Connector X31 Jumper 2-3 inserted	☐ ☐
24 V optocoupler input	Connector X30 Jumper 3-4 inserted Connector X31 Jumper 1-2 inserted	☐ ☐
5 V -24 V optocoupler inputs (Special voltages)	Connector X30 Jumper 4-6 inserted Connector X31 Jumper 1-2 inserted	☐ ☐

Set the module for the required BASP response.

There is no CPU in operation or the module should not react to the BASP signal

Connector X21 Jumper 1-2 inserted ☐

If the BASP signal is received, the signal output must be blocked

Connector X21 Jumper 2-3 inserted

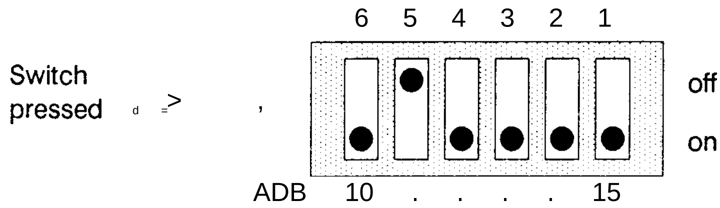
Set the addresses.

If the module is to be controlled by the CPU, you must set the appropriate page address (in whole multiples of four, e.g. 0,4, 8...252). These page addresses must only be assigned once in the programmable controller.

Address set at switch S2 on the module,

Page address							

The address selected and the three following addresses are not used for any other purpose, The following switch setting must always be made at switch S1:



Jumpers XI 4 / 2-3
X15 / 1-2
X16 / 2-3
XI 7/2-3
XI 8/2-3

Fig, 7/3 Setting at switch S1

The following jumpers must always be inserted:

x 14	x 15	X 16	x 17	X 18	X 10	X 11	x 12	x 13
1 2 3	1 2 3	1 2 3	1 2 3	1 2 3	1 2 3	1 2 3	1 2 3	1 2 3

Fig. 7/4 Jumper settings

If all the jumpers are correctly set, the IP247 can be inserted in the programmable controller. Make sure that the power supply to the programmable controller is switched off.

7.1.5.3 Preparing the Power Units

The signal lines must be connected to the inputs of the power units as explained in the manual and according to the instructions of the power unit manufacturer.

Module	Color code	Power unit 1	2	3
Connector X4, X5, X6				
Pin 1 Reset signal	blue 1 ring			
Pin 2 Inverse reset signal	red 1 ring			
Pin 3 Clock pulse	grey 1 ring			
Pin 4 Inverse clock pulse	yellow 1 ring			
Pin 5 Direction signal	green 1 ring			
Pin 6 Inverse direction signal	brown 1 ring			
Pin 7 24 V for BB contact	white			
Pin 8 Input for ready signal	black 1 ring			
Pin 9 Ground	blue 2 rings			

Note

If the power units do not output a ready signal (24 V), jumpers in connectors X4, X5 and X6 must be inserted between pins 7 and 8. In the specially made connecting cables the white and black wires at the open end must be connected together.

Select the required mode, full step or half step on the power unit. ok ◯

Wire any required enable signals for the power unit (current drop, burst) externally, ok ◯

Set the motor current according to the instructions of the power unit manufacturer. ok ◯

Connect the cables to the IP247:

- 24 V for digital outputs (FASTON terminal) ok ◯
- signals to power unit channel 1 ok ◯
- signals to power unit channel 2 ok ◯
- signals to power unit channel 3 ok ◯
- connections to the switches and "position reached" indicator ok ◯
- programmer (can be connected or disconnected at any time). ok ◯

Note

All the plug-in connections should be screwed tight,

ok O

If the 24 V at the FASTON terminal is supplied by an external power supply unit, the ground of this voltage source must be connected to the chassis of the programmable controller.

ok O

If you use a special voltage for the signals to the power units, and the positive pole is connected to X7, pins 23, 24 and 25, you must connect the negative pole (ground) with the chassis of the programmable controller.

ok O

Before you switch on the plant, the carriages (or similar) must be within the limit switches which send signals to the IP247. If necessary, you must move the axes to within the permitted range manually.

ok O

Check all the connecting cables and switch on the voltage sources in the following order:

Switch on the PC voltage
(after power up, the LEDs must flash alternately, following this the green LED must be lit steadily, if this not the case, there is a hardware problem). ok O

If applicable, switch on the 24 V ok O

If applicable, switch on the special voltage ok O

Switch on the power units ok O

Connect the programmer to connector X8 and load the COM 247 communications software (=> Part 5, "COM 247 Communications Software"). After switching to the online mode, the SYSID with the firmware version must appear on the PG, otherwise there is a module fault.

Release: IP247 . . .

ok O

Enter the machine data on the module or transfer a complete data record from a prepared diskette to the IP247 (=> Section 7.1.4 "Planning the Machine Data"). After the transmission of a data record, the appropriate power unit automatically receives an initialization pulse.

ok O

Switch the module to test mode (function key 3 on the PG) and select the required axis (F1 - 3).

ok O

Note

Thenextoperations muStbe performed at lowspeeds (machine data: JOG speed 1). You must also make sure that you can switch off the motors at any time (emergency stop or external limit switch accessible).

Select the mode “JOG 1“ and press the “forward” or “reverse” key. The drive must now move at a uniform speed.

ok O

(If the drive is running at a uniform speed, you can continue and check the limit switches.)

Problem	Possible cause of problem
Motor “howls”, but does not move	Start/stop frequency too high
Motor jerks and stops	Rate of frequency increase too high
Motor accelerates and then stops and “howls”	fmax too high or load torque too high
Error message: Axis waiting for external start	The signal at the start/stop input is high (=> Part 4 “Functions”)

If the axis switches to the “running” status and if the actual value is being incremented or decremented, but the drive is not moving, check whether pulses are being output at the TN or TN-N output. Measure the pulse frequency and pulse width with an oscillograph.

The signals are correct at the output of the IP247, but the motor is not moving.	The signal lines to the power unit have been incorrectly connected, The reset signal is permanently active. You may have to change over RS and RS-N The power unit may require a separate enable signal.
No signals can be measured at the output, although an actual value is being counted.	The transmitter power is not correctly connected or there is a hardware fault on the module.

Check the function of the limit switches which send signals directly to the power unit.

Check whether the travel direction is the direction you require in the “JOG” mode.

ok O

If the direction is not as required, change the setting of the power unit direction level or change over the RP and RP-N connections in connector X4-X6.

ok O

Test whether the *two* limit switches which send signals to the IP247 actually respond.

In a forward direction, the end limit switch must respond. ok O

In the reverse direction, the start limit switch must respond. ok O

If necessary, change over the limit switches at connector X7.

At maximum speed, test whether there is sufficient braking distance after the hardware limit switches which send signals to the IP247.

Approach the limit switches at maximum speed. After the IP247 switches off automatically, the limit switch connected to the power unit must not be tripped, otherwise a loss of pulses occurs and the reference point is incorrect. ok O

Make a reference point approach or set a reference point using the software (=> Part 4 “Functions”), ok O

Test the position and function of the software limit switches when traversing at maximum speed (f_{max}). The axis should only begin to brake when it reaches the software limit switch. The axis must not continue to the hardware limit switch, if it does, you must change the machine data. ok O

Once you have tested these basic functions, you can try out the other modes.

1 JOG 1 ok O

2 JOG 2 ok O

3- (no significance) ok O

4 Axis off (an active mode is terminated) ok O

5 Reference point approach/set ok O

6 Incremental absolute ok O

7 Incremental relative ok O

8 Automatic (later) ok O

9 Automatic single statement (later) ok O

10 Teach-in on (do not forget program number) ok O

11 Teach-in off ok O

12 Zero offset absolute (set actual value) ok O

13 Zero offset relative ok O

14 Clear zero offset ok O

15 Set tool length offset ok O

16 Clear tool length offset ok O

17 Clear error ok O

Enter an automatic program on the IP247 (=> Part 5 "COM 247 Communications Software").

ok O

Test the automatic program in mode 8.

ok O

Test the automatic program in mode 9.

ok O

Generate a machining program in the teach-in mode and test the program in modes 8 and 9.

ok O

Test the external start/stop function (=> Part 4 "Functions").

ok O

Link the **IP247 into the user program of the** CPU

Load the handling blocks for the appropriate CPU

	Send	Receive	Synchron
S5-11 5	FB244	FB245	FB249
S5-135/CPW2z928	FB120	FB121	FB125
S5-150	FB180	FB181	FB185
S5-155	FB120	FB121	FB125

Load the standard function block FB164 for the appropriate CPU. (If required, use the supplied example program.)

ok O

Call the function block “SYNC CHRON" in the start-up OBS 20-22 once for each axis you wish to operate (parameter assignment: => Part 6 “Standard Function Blocks FB164 and FBI 65”).

ok O

Assign parameters inFB164 and call it **unconditionally once in** each cycle.

ok O

7.1.6 Controlling the IP247 by Means of the Programmable Controller

Once you have tested the combination of drive and IP247, you must make sure that your positioning application is linked into the STEP*5* program.

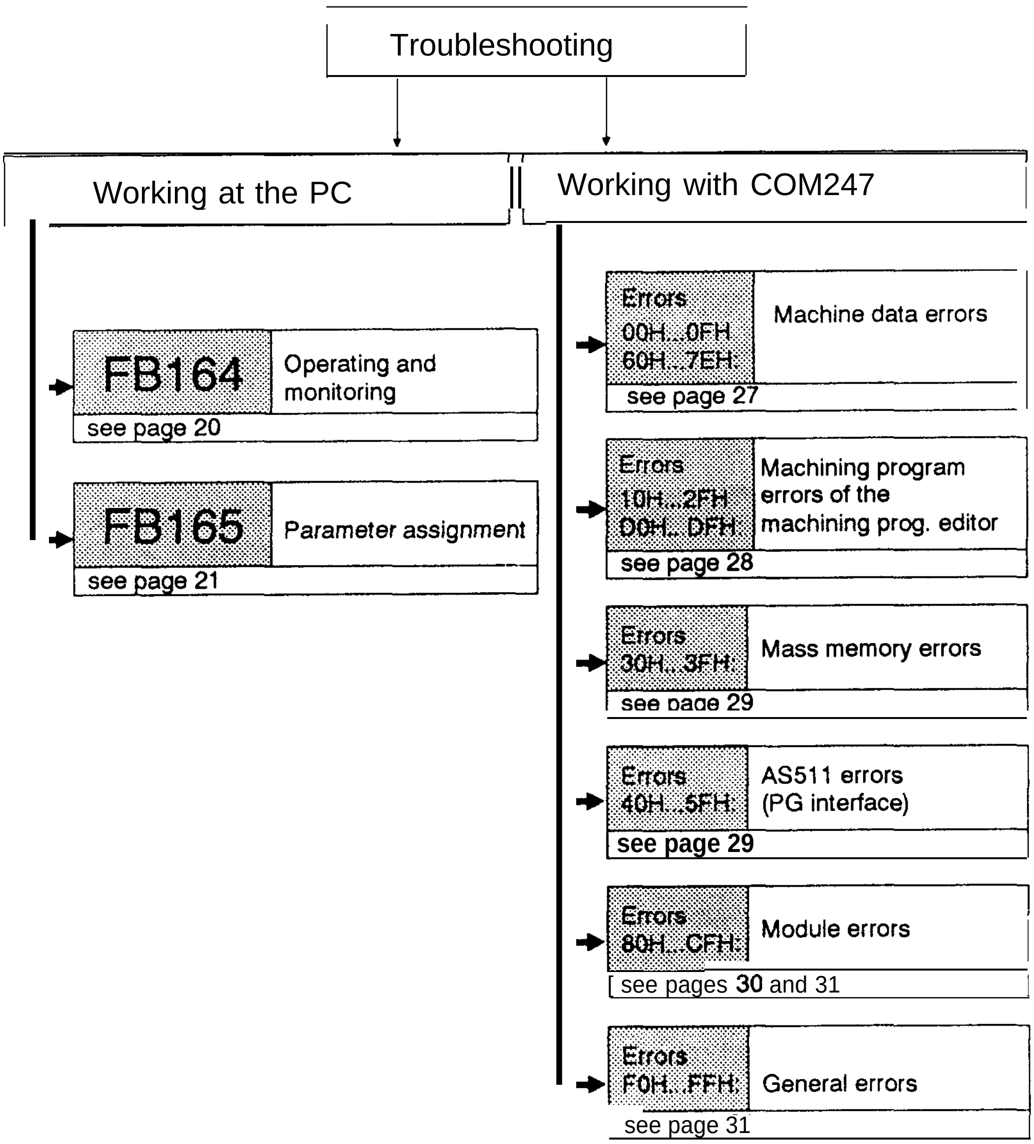
If you have not yet written your own program, you can start by using the example program. Remember that this program was written for page address “O” and is intended for axis 1 and the data channel. For the first trials, the module should beset accordingly.

For further information, refer to the description of the example (=> Section 6.4 “Examples”). If you wish to base your program on the example, it is advisable to print out the whole program.

The example program of FBI 65 must be modified for the functions executed via the data channel.

7.2 Troubleshooting

The following diagrams provide you with a routine which you can use for troubleshooting. The machine data errors and messages are explained in detail in Sections 7.2.1 ,7.2.2 and 7.2.3.



FB164 from page 19 → **PAFE** = signal state '1'

FY 255

F 255. O: simultaneous signal change at the command inputs. Only the highest priority command is transferred, the others are lost.

Commands in decreasing priority:

STOP --> STAR --> VORW --> RUCK --> UEBN

F 255.1 : binary/BCD conversion not possible (in BA71 ... BA73)

F 255,2: user data block number not permitted.

F 255.3: the specified data block does not exist, is too short or the parameter DWNR is greater than 236.
If DBNR = 164; DWNR is within the working area of FBI 64.

F 255.4: the specified interface does not exist.

F 255. 5: the specified parameter BA is not permitted or the command is not allowed for the mode to be executed.

F 255.6:parameter assignment error SEND (FBI 20, FB180, FB244)

→ **FY 253** Parameter PAFE of SEND
see page 22

F 255. 7: Parameter assignment error RECEIVE (FB121,FB181,FB245)

→ **FY 252** Parameter PAFE of RECEIVE
see page 22

FB164 from page 19 → **BFEH** = signal state '1'

FY 254

F 254. O: error message of the interface with operation

→ **FY 250** or DR (DWNR+16) of the axis DB
see pages 24 and 25

F 254.1 : module error caused by modes 1...17

→ **FY 251**
see page 25

F 254. 2: not used

F 254. 3: not used

F 254. 4: not used

F 254. 5: not used

F 254. 6: not used

F 254. 7: not used

FB165 → **PAFE** signal state '1'

from page 19

FY 255

F 255. 0: not used

F 255. 1 : binary/BCD conversion not possible (with BA66)

→ **FY 249**

continuation on page 25

F 255.2: user data block number not allowed

F 255. 3: the specified data block does not exist, it is too short
or the parameter DWNR is greater than 241
if DWNR = 165: DWNR is within the working area of FB165

F 255. 4: the specified interface does not exist

F 255. 5: the specified parameter BA is not permitted

F 255. 6: parameter assignment error SEND-ALL or SEND-DIR

→ **FY 253** parameter PAFE of SEND-ALL

→ **FY 252** parameter PAFE of SEND-DIR

see page 22

F 255. 7: parameter assignment error of RECEIVE-ALL or FETCH

→ **FY 253** parameter pAFEof RECEIVE-ALL

→ **FY 252** parameter PAFE of FETCH

see page 22

FB165 → **BFEH** = signal state '1'

from page 19

FY 254

F 254. 0: not used

F 254. 1 : error message from the interface when entering data

→ **FY 250** or DL (DWNR+7) of the axis DB

see page 25

F 254. 2: error message of the interface when outputting data

→ **FY 250** or DL (DWNR+1 1) of the axis DB

see page 25

F 254. 3: not used

F 254. 4: not used

F 254. 5: not used

F 254. 6: not used

F 254. 7: not used

FY 253	parameter PAFE of SEND

continuation from page 20 (FB164)

FY252	parameter PAFE of RECEIVE

continuation from page 20 (FB164)

FY 253	parameter PAFE of SEND-ALL
FY 252	parameter PAFE of SEND-DIR

continuation from page 21 (FB165)

FY 253	parameter PAFE of RECEIVE-ALL
FY 252	parameter PAFE of FETCH

continuation from page 21 (FB165)

PAFE byte of the HDBs

bit:	7	6	5	4	3	2	1	0

Bit 0:
1 : error
0: no error

Bits 7–4:
0: no error
1: QTYP/ZTYP wrong
2: memory area does not exist
3: memory area too small
4: timeout QVZ (area does not exist)
5: wrong condition codeword
6: no source or destination parameter exists
7: interface does not exist
8: interface not ready
9: interface overloaded
A: interface occupied by a different CPU (multiprocessor operation)
B: illegal job number
C: error in handshaking (neg. acknowledgement)
D: other interface arrors (e.g. field length illegal)
E: other errors (e.g. no DB open when using indirect parameter assignment)
F: HDB call illegal (double call by interrupts)

FY 251	or DR (DWNR+16) of the axis DB
continuation from page 20 (FB164)	

FY 251	module error triggered by BA1...BA19

0: no error
1: PG job list full *)
2: job not permitted *)
3: statement saved
4: axis active ==> entry not possible
5: PC job list is full *)
6: motor waiting for external start ●)
7: speed range exceeded ●)
8: status after power down on module
9: free
10: reference point does not exist
11: free
12: correct MD - module number cannot be changed *)
13: data block does not exist
14: wrong or no machine data
15: error in machine data

Mach. data record	machine data error
page	

16:free
17: overwrite machine data?
18: max. number of programs reached
19: data block does not exist
20: overwrite machining program?
21: free
22: processing more than one reach. prog. not permitted ●)
23: traversing range exceeded
24: not enough space for machining program
25: start limit switch tripped
26: end limit switch tripped
27: external STOP received
28: software start limit switch tripped
29: software end limit switch tripped
30: mode not permitted in teach-in *)
31: free
32: free
33: cycle time exceeded
34: pulse generator defect ●)
35: error at start of statement
36: subroutine DB no. too high
37: G function not permitted
38: closed loop only as outer loop
39: nesting depth exceeded
40: X function wrong
41: F function wrong
42: traversing distance too long

continued on page 24

possible causes of module errore marked
with *)
are listed from page 35 onwards.

continued from page 23

FY 251 module errors tripped by BA1...BA19

43: traversing speed too high
44: error at end of statement
45: program end before loop end
46: illegal mode on this axis *)
47: change of direction illegal with flying change
48: machining program error

→ **Machining program DB** syntax errors
see page 27

49: machining program already exists! Change prog. no.
50: free
51: machining program is active ●)
52: flying change could not be executed ●)
53: switch on power unit ●)
54: error in ramp table generation ●)
55: PC failure ●)
56: error accessing ramp table
57: statement not yet fully interpreted *)
58: machining program speed too low
59: reference cam switch defective ●)
60: free
61: machining program only cleared from directory *)
62: illegal dist. spec.
63: illegal tool length offset
64: free
65: machining program waiting to continue
66: distance not in BCD
67: speed not in BCD

possible causes of module errors marked
with ●)
are listed from page 35 onwards

FY 250	error messages from the interface
continuation of page 20 (FBI 64) & page 21 (FBI 65)	

FY 250	error message from the interface

bit: 7 6 5 4 3 2 1 0

0: no error

1: error entered in PAFE byte of the HDBs (FY252/FY253)

2: error in module error byte (FY251)

3: free

4: free

5: offset too great

6: execution not possible at present

7: DB elready exists

8: DB does not exist

9: wrong ORG identifier

A: destination DB too small

B: source DB too small

C: source DB too large

D: area in PC too small

E: area blocked for output/input

F: wrong checkback signal

FY 249	binary/BCD conversion not possible
continuation from page 21 (FB165)	

FY 249	binary/BCD conversion not possible

F 249. O: binary/BCD conversion of actual value (actual position) not possible

F 249. 1: binary/BCD conversion of dist. to go not possible

F 249, 2:

F 249. 3: FB164 does not exist (only S5-115U)

F 249. 4: not used

F 249. 5: not used

F 249. 6: not used

F 249. 7: not used

Machine data record	machine data errors
continuation from page 23	

Mach. data record	in machine data DB DRn+4

0: no error in machine data
1: machine data not yet checked
2: wrong pulse duration *)
3: wrong maximum or start/stop frequency *)
4: wrong JOG or incremental speed *)
5: wrong pulse count/revolution *)
6: wrong rate of frequency increase *)
7: software limit switch wrong *)
8: reference point wrong *)
9: wrong transmission ratio *)
10: wrong number of excitation patterns*)
11: wrong dimensional unit *)
12: wrong axis/module number *)
13: zero offset too large *)
14: wrong tool length offset *)
15: wrong value for ref. direction and synchronization●)
16: ramp table incorrectly generated *)
17: wrong reference speed *)
18: wrong value for edge evaluation *)
19: wrong polarity for limit switches *)
20: backlash distance too great *)
21: end of range wrong *)
22: wrong start/stop frequency

Machining prog. DB	syntax error
continuation from page 24	

Machining prog. DB	in machining prog. DW n+3

0: no error
1: text too long or <LF> missing
2: statement type wrong or N function missing
3: statement number too high
4: subroutine number too high
5: function not allowed after G function
6: function not allowed after X function
7: free
8: <LF> missing after final statement
9: statement end: <LF> missing or wrong function
10: traversing distance too great
11: value of F function too high
12: too many decimal places
13: end of loop missing
14: start of loop missing
15: end of program missing
16: function not known
17: value of M function too high
18: new statement after final statement
19: the statement is too long
37: G function not implemented

possible causes of machine data errors marked
"with *)
are listed on pages 32, 33 and 34

Errors 00H..0FH 60H..7EH:	machine data errors
continuation from page 19 (COM247)	

Error message line	machine data errors

0:
1:
2: wrong pulse duration *)
3: wrong maximum or start/stop frequency*)
4: wrong JOG or incremental spaed ●)
5: wrong pulse count/revolution *)
6: wrong rate of frequency increase*)
7: software limit switch wrong *)
8: reference point wrong ●)
9: wrong transmission ration *)
A: wrong number of excitation patterns *)
B: wrong dimensional unit *)
C: wrong axis/module number ●)
D: zero offset too large *)
E: wrong tool length offset *)
F: wrong value for ref. direction and synchronization *)

60: ramp table incorrectly generated *)
61: wrong reference speed ●)
62: wrong value for edge evaluation *)
63: wrong polarity for limit switches *)
64: backlash distance too great *)
65: end of range wrong*)
66: wrong start/stop frequency

possible causes of machine data errors marked with ●)
are listed on pages 32, 33 and 34

Errors 10H...2FH D0H...DFH	machining program errors of the reach. program editor
continuation from page 19 (COM247)	

Error message line — machining program errors of the machining program editor

10: illegal input
11: memory overflow
12: separate functions with blanks
13: program exists already
14: statement syntax incorrect
15: field cannot be exited
16: terminate processing?
17: final function already exists
18: entry *not* permitted after L function
19: X function does not exist
1A: entry not permitted after last function
1B: value outside permitted range
1C: error in X function -> correct
1D: insertion not permitted
1E: cannot save -> machining program incomplete
1F: output impossible -> DB no, not identical
20: statement type not permitted
21: function key blocked -> statement incomplete
22: G function -> illegal input
23: no further functions allowed with L function. Delete?
24: error in F function
25: statement type does not exist
26: statement number does not exist
27: statement complete -> function key
28: current G function requires an entry
29: X function must be followed by F function
2A: no X function -> entry illegal

2B: final statement exists -> function key blocked
2C: error in L function
2D: error in M function
2E: statement number wrong
2F: error in G function

D0: only closed loop allowed
D1: loop end missing

Errors 30H..3FH:	mass memory errors
continuation from page 19 (COM247)	

Error message line	mass memory errors

30:
31: drive not defined
32: external storage defect
33: element directory does not exist
34: data block does not exist
35: DB or file exists already
36: file type not defined
37: identification headers not identical
38: external storage read-only
39: file read-only
3A: buffer not long enough
3B: number of allowed elements too large
3C: file does not exist
30: directory full
3E: diskette full
3F: file cannot be interpreted

Errors 40H..5FH:	AS511 errors (PG interface)
continuation from page 19 (COM247)	

Error message line	AS 511 errors

40: syntax error/name wrong
41: not allowed
42: data block does not exist
43: overwrite data block?
44: data block does not exist
45: delete DB?

48: illegal value

4C: cable not connected

50: data block does not exist
51: cable not plugged in at PG
52: not enough memory on module
53: timeout on module
54: transfer error
55: error in data transfer
56: error in data transfer
57: BREAK received
58: mod. not answering
59: transfer error
5A: wrong baud rate

5D: parity error
5E: overflow error
5F: frame error

Errors 80H..CFH:	Module errors

continuation from page 19 (COM 247)

Error message line	module errors

80:
81: PG job list is full*)
82: job not permitted *)
83: statement saved
84: axis active ==> entry not possible
85: PC job list is full
86: motor waiting for external start *)
87: speed range exceeded
88: status after power down on module
89:
8A: reference point does not exist ●)
8B:
8C: correct MD - module number cannot be changed *)
8D: data block does not exist
8E: wrong or no machine data
8F: error in machine data
90: PG is offline
91: overwrite machine data?
92: max. number of programs reached
93: data block does not exist
94: overwrite machining program?
95: automatic not permitted
96: processing more than one reach. prog. not permitted *)
97: traversing range exceeded
98: not enough space for machining program
99 start limit switch tripped
9A: end limit switch tripped
96: external STOP received
9C: software start limit switch tripped
9D: software end limit switch tripped
9E: mode not permitted in teach-in *)
9F:
AO:
Al: cycle time exceeded
A2: pulse generator defect *)
A3: error at start of statement
A4: subroutine DB no. too large
A5: G function not permitted
A6: closed loop only as outer loop
A7: nesting depth exceeded
A8: X function wrong
A9: F function wrong
AA: traversing distance too long
AB: traversing speed too high
AC: error at end of statement
AD: program end before loop end
AE: illegal mode on this axis*)
AF: change of direction illegal after flying change
BO: machining program error
B1: machining program already exists! Change prog. no

continued on page 31

possible causes of module errors marked with ●)
are listed from page 35 onwards.

Error message line continued from page 30

B2:
B3: machining program is active*)
B4 flying change could not be executed*)
B5: switch on power unit*)
B6: error in ramp table generation*)
B7: PC failure*)
B8: error accessing ramp table
B9: statement not yet fully interpreted *)
BA: machining program speed too low
BB: reference cam switch defective *)
BC: free
BD: machining program only cleared from directory *)
BE: illegal dist. spec.
BF: illegal tool length offset
C0: free
C1: machining program waiting to continue
C2: distance not in BCD
C3: speed not in BCD

possible causes of module errors marked with *)
are listed from page 35 onwards

Errors F0H.. FFH: general errors
continuation from page 19 (COM247)

Error message line general errors

F0:
F1: error mess. does not match this COM
F2: printer not assigned parameters
F3: delete everything?
F4: only machining programs
F5: abort printing
F6: mode not permitted
F7: wrong time entered
F8: no plant designation entered
F9: no file name entered
FA: DB transferred
FB: last page reached
FC: illegal key
FD: HELP key not permitted here
FE: exit COM247?
FF: input prohibited

7.2.1 Machine Data Errors and their Causes

When machine data are transferred to the module, they are checked on the module. If a machine data error is recognized, the error "error in machine data" is set and the machine data record is marked as containing errors by entering the number of the machine data error in the data record. The COM247 software package evaluates this error number and displays the error in plain text in the error message line on the PG. When transferring data with FB165 (= > Part 6 "Standard Function Blocks FB164 and FB165"), the number must be read out of the data record if an error has been detected. This is achieved using modes 67 and 68.

- Error 2 (COM247: F02H; in DB: 2) **Wrong pulse duration"**
  The pulse duration must be within the limits 1...31 μs and be less than half the period of the maximum frequency.

- Error 3 (COM247: F03H; in DB: 3) **Wrong maximum frequency"**
  The maximum frequency must be within the limits 0.012...100.000 kHz.

- Error 4 (COM247: F04H; in DB: 4) **"wrong JOG or incremental speed"**
  The frequency for the JOG or incremental speed must be within the limits, 1 Hz... maximum frequency.

- Error 5 (COM247: F05H; in DB: 5) **"wrong pulse count/revolution"**
  The ratio pulse count/revolution must be within the limits 12...1000.

- Error 6 (COM247: F06H; in DB: 6) **"wrong rate of frequency increase"**
  The rate of frequency increase must be within the limits 0.020...2599.999 Hz/ins so that the corresponding τ is in the limits of 5...2600 ms.

- Error 7 (COM247: F07H; in DB: 7) **"software limit switch wrong"**
  This error occurs when the software start limit switch has a higher value than the software end limit switch.

- Error 8 (COM247: F08H; in DB: 8) **"reference point wrong"**
  The reference point must be between the software limit switches or the range limits.

- Error 9 (COM247: F09H; in DB: 9) **Wrong transmission ratio"**
  The transmission ratio must be within the limits 0.012...400.000 and the quotient of the transmission ratio and pulses per revolution must produce a resolution $\geq 1 \mu m$.

- Error 10 (COM247: FOAH; in DB: 10) **"wrong number of excitation patterns"**
  The number of excitation patterns must be between 4 and 40 and the relationship pulses/revolution must be a whole multiple of this number.

- Error 11 (COM247: FOBH; in DB: 11) **"wrong dimensional unit"**
  The following coding for the dimensional unit must be adhered to:
  mm = 1
  inches = 2
  degrees = 3

- Error 12 (COM247: F0CH; in DB: 12) **"wrong axis/module number"**
  The axis/module number in the machine data does not match the number in SYSID.

- Error 13 (COM247: F0DH; in DB: 13) “zero **offset too large”**
  The zero offset must be within the limits* 100 m and a zero offset must not displace the software limit switches out of the traversing range of ± 100 m.

- Error 14 (COM247: FOEH; in DB: 14) **“wrong tool length offset”**
  The tool offset must be within the limits of ± 100 m and after the offset has been executed, the actual value of the tip of the tool must not be outside the traversing range of* 100 m.

- Error 15 (COM247: F0FH; in DB: 15) **‘Wrong value for reference direction and synchronization”**
  The following coding must be adhered to for the reference direction:
  fwd = 0H
  rev = 20H
  For synchronization: yes: O; no: 1

- Error 17 (COM247: F61H; in DB: 17) **‘Wrong reference speed”**
  The frequency of the reference speed must be greater than the start-stop frequency and less than the maximum frequency.

- Error 18 (COM247: F62H; in DB: 18) Wrong **value for edge evaluation”**
  The following coding must be adhered to for the edge evaluation:
  negative edge = O
  positive edge = 40H

- Error 19 (COM247: F63H; in DB: 19) ‘Wrong **polarity for limit switches”**
  When the machine data are entered, the system checks whether the selected limit switches actually exist.

- Error 20 (COM247: F64H; in DB: 20) **‘backlash value too high”**
  The backlash compensation value must be within the limits 0...64999 pm.

- Error 21 (COM247: F65H; in DB: 21) **“end of range wrong”**
  This error message appears when the start of the range is higher than the end of the range for a rotary axis.

- Error 22 (COM247: F66H; in DB: 22) **‘Wrong start-stop frequency”**
  The start-stop frequency must be in the range of 0.001...10.000 kHz.

7.2.2 Module Errors and Possible Causes

This section deals with the module errors or errors for a specific axis which occur on the IP247 positioning module and are output both by the PC and PG interface. With COM247, the error numbers have an offset so that the error numbers output at the PG differ from those at the PC by 80H.

COM247 provides an additional set of error messages which occur

- when there is an operator error with COM247,
- when the COM247 software accesses floppy disk or hard disk drives,
- when COM247 is communicating with the IP247 and
- when machine data and machining programs are input.

The software of the IP247 generates two types of error messages for module errors:

- the actual axis errors which lead to a traversing movement being aborted and
- warnings or indications which are simply to inform the user (errors 1 to *9).*

If an axis error is displayed in COM247, not only the error number but also a message is displayed, so that the cause of the error can normally be recognized immediately.

There are, however, some error messages which require further explanation to allow you to find the cause of the error and to remedy it more quickly. Some error messages are therefore explained in more detail.

- Error 1 (COM247: F81 H, PC: 1) "**PG** job list is full"
  Owing to mechanical inertia, the module cannot execute the jobs as quickly as they are being entered. The last job entered from the PG has been lost and must be repeated.
- Error 2 (COM247: F82H, PC: 2) "job **not** permitted"
  The last job sent from the PC or PG either has no defined mode or is not feasible at this point. Example: starting an axis which is already running, The active mode is terminated and an error message output.
- Error 5 (COM247: F85H, PC: 5) 'PC **job list is full"**
  Corresponds to error 81 from the point of view of the PC. The last job must be repeated.
- Error 6 (COM247: F86H, PC: 6) **"motor waiting** for external **start"**
  The execution of the selected mode is blocked by a signal"1" at the digital input "external start-stop" (=> Section 2.8.4 "External Start-Stop").
- Error 7 (COM247: F87H,PC: 7) **"speed range exceeded"**
  If, in the JOG or incremental modes, a speed is specified in the speed parameter which corresponds to a frequency outside the limits 1 Hz... maximum frequency, the traversing frequency is set to the limit of the frequency range and this message is output.
- Error 10 (COM247: F8AH, PC: 10) **"reference point does not exist"**
  After switching on the power and loading machine data for the first time and when changing certain machine data, the reference point is missing. Execute mode 5 reference point approach/set reference point.

- Error 12 (COM247: F8CH, PC: 12,) "correct MD - **module** number cannot be **changed**"
  As soon as the positioning module has at least one correct machine data record, the module number can no longer be changed. The number already stored can be read in the presets display of COM247 (=> Section 4.3.22 "Enter SYSID" and Section 2.5.6 "Other Parameters").

- Error22 (COM247: F96H, PC: 22) **"processing more than one reach. prog. not permitted"**
  This error occurs when an axis is creating a machining program in the teach-in mode and you attempt to transfer, modify or delete a second machining program via the data channel. It is also not possible to have more than one axis in the teach-in mode.

- Error 30 (COM247: F9EH, PC: 30) **"mode not permitted in teach-in"**
  When the teach-in mode is active, only the JOG modes and incremental modes are permitted. Input and transfer of machining programs is also not permitted.

- Error 34 (COM247: FA2H, PC: 34) **"pulse generator defect"**
  If this error message appears, there is a hardware fault.

- Error46 (COM247: FAEH, PC: 46) **"illegal mode on this axis"**
  This message appears if you attempt to enter or delete a machining program on axes 1...3. The message also appears if you attempt to enter or delete machine data on the data channel (axis 4) or to execute one of the operational modes 1...17 with the exception of mode 17.

- Error 51 (COM247: FB3H, PC: 51) **"machining program is active"**
  A machining program cannot be modified or deleted while it is being executed.

- Error 52 (COM247: FB4H, PC: 52) **"flying change could not be executed"**
  If this error occurs, one of the conditions of the flying change has not be met (=> Section 2.6.6.3 "G1 O: Flying Change").

- Error 53 (COM247: FB5H, PC: 53) **"switch on power unit"**
  Each axis has a digital input with which the power unit can be monitored by the module. This input must have a high signal when the power unit is switched on. If the power unit does not have a "ready" contact, the ready signal must be simulated by jumpering wires BBxL and BBx on the power unit.
  (x = axis number).

- Error 54 (COM247: FB6H, PC: 54) **"error in ramp table generation"**
  This error occurs if it is not possible to generate an acceleration ramp with the corresponding machine data. The combination of maximum frequency, start-stop frequency and rate of frequency increase is not feasible.

- Error 55 (COM247: FB7H, PC: 55) **"PC failure"**
  If jumper X21 is connecting pins 2 and 3, the IP247 recognizes when the CPU outputs the BASP signal (block command output). This error message terminates the traversing movements on all three axes (=> Part 3 "Hardware").

- Error57 (COM247: $FB9_H$, PC: 57) 'Statement **not yet fully interpreted"**
  If this error occurs sporadically, there is an execution time problem. When a flying change is programmed, the next statement is interpreted while the last statement is being executed. if the time required to execute the current statement is less than the time required to interpret the following statement, this message is output and the machining program terminated.

- Error 59 (COM247: FBBH, PC: 59) "reference **cam switch defective"**
  This error appears when the axis is to leave the reference cam in single steps during the reference point approach, but the switch does not output a negative edge within 2500 steps.

- Error61 (COM247: FBDH, PC: 61) "machining program **only** cleared from **directory"**
  This message indicates that a machining program with the same number as the program deleted on the module can be transferred, but that no space has become free in the machining program memory area on the module.

7.2.3 PG Interface Errors

- Error $F53_H$ **'timeout on module"**
  This error only appears on the PG interface, i.e. in COM247 and indicates situations in which there is no connection established to the positioning module. The possible causes are as follows:

  the connecting cable is not plugged in or there is a wire break
  the positioning module has no power supply.

7.3 Supplementary Notes

When using the IP247 positioning module there are several characteristics of the SIMATIC S5 system which must be taken into account. The following sections deal with these characteristics.

7.3.1 Keyboard Character Buffer

The keyboards of the programmers have a buffer in which characters entered at the keyboard are temporarily stored when characters are entered more quickly than they can be processed. This can become apparent in the test display of COM247 when, for example, a fast sequence of “forwards” and “reverse” commands is entered in the JOG mode. The execution then lags behind the input. A stop command can only be executed when all the commands previously stored in the character buffer have already been processed.

Note

If a start command is entered after the stop command the drive starts again immediately.

7.3.2 MultiprocessorO peration

The S5-135U and S5-155U programmable controllers are designed for multiprocessor operation. Several processors could access the same positioning module independently. This is not permitted with the IP247 positioning module,

7.3.3 Restarts

A reset pulse on the bus resets the I P247. A current job is then no longer active, just as if the power had been switched off and on again.

The start-up characteristics of function blocks FBI 64 and FB165 are discussed in Part 6 “Standard Function Blocks FB164 and FBI 65”. The IP247 positioning module reacts as follows when it is restarted:

Following each restart of the module, the reference point is deleted and the “axis off” mode is active. If the batttery back-up of the programmable controller was absent (the module was removed), all the machine data and machining programs on the IP247 are lost, zero offsets and tool length offsets are also deleted. If the module was backed up by the battery, the machine data remain valid. The reference point coordinate is then signalled as the actual value.

7.4 Troubleshooting Questionnaire

If, despite careful installation and programming, you still encounter problems with positioning operations and cannot localize the problem, please follow the routine outlined below:

Before calling your branch representative, please complete the questionnaire so that the necessary information is readily available. The more exact the description of the problem and the events leading up to it, the faster your representative will be able to help you.

1) Which module are you using?

IP247 MLFB: 6ES5 247 - 4UA ___________

Version marked: ______________________

Firmware release display by COM: ____________________________

2) **Which COM are you using?**

MLFB: 6ES58 _ - 5 ___

Version issue: A __ . ____

3) **Which PG are you using?**

PG: ______________________

Version: ___________________

4) System components

(please enter order numbers)

Controller: ______________________________

Power supply: ______________________________

CPU: ______________________________

Slot number:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Version

Expansion unit: ______________________________

Power supply: ______________________________

Interface module pair: -

Slot number:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Version

Have the drive specification% machine data and any machining programs ready (see machine data planning or print out this data).

Make sure you know the jumper and switch settings on the module.

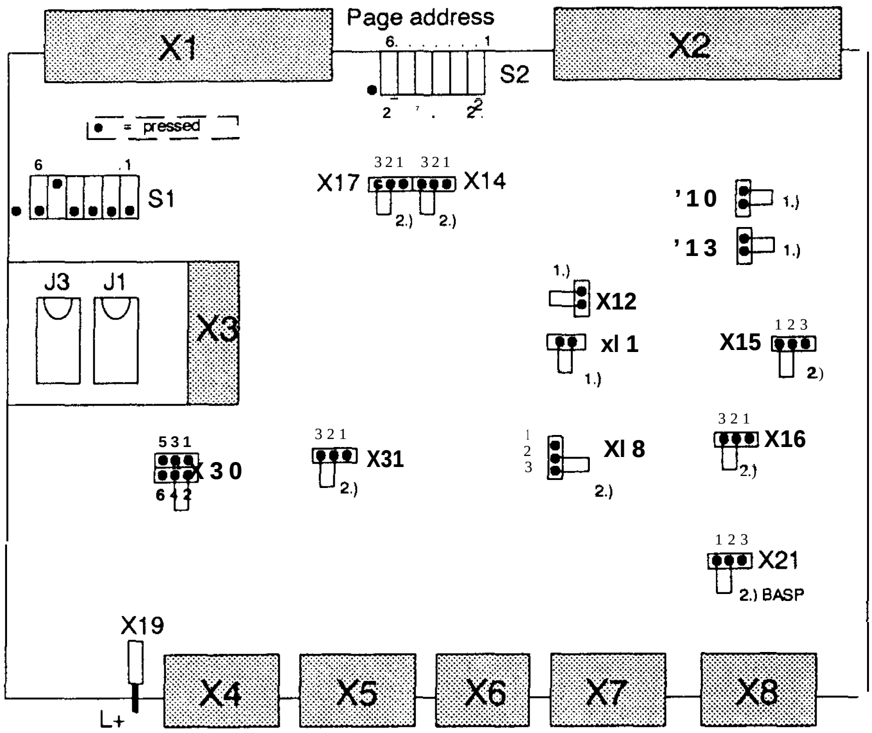


Fig. **7/5** Position of the switches, jumpers and fuses

5) What is going wrong?

---

6) Which error messages were output by the COM software?

---

7) Which error message is set at the output of the standard function block?

---

8) Which error numbers are entered in the appropriate flag bytes?

---

9) Isthe error reproducible?

____

10) Does the error occur sporadically?

____

11) Does the error occur when operating the module from the PG as well as from the PC?

____

12) Which modes are being used?

____

13) In which modes does the error occur?

____

14) When the positioning is incorrect, is the distancetravelled

- always too long ______________
- always too short______________
- always wrong by the same amount? ______________

15) Does the error only occur with a particular sequence of jobs?

____

16) What type of axis are you using?

Rotary axis: ____________ (0)

Linear axis:
Vertical axis: ____________ (0)
Horizontal axis: ____________ (0)

17) What kind of drive are you using?

Stepper motor
2-phase: ____________ (0)
4-phase: ____________ (0)
5-phase: ____________ (0)

Manufacturer: ____________
Type: ____________

18) Istheretransmission?

Type of transmission
belt ____________ (0)
gear wheel ____________ (0)
chain ____________ (0)

Transmission ratio: ____________

19) Which power unit are you using?

Manufacturer: ____________
Type: ____________
What signal inputs does the power unit have?:
5 V differential ____________ (0)
5 V optocoupler ____________ (0)
24 V optocoupler ____________ (0)
5...24 V optocoupler ____________ (0)

20) When using externally ventilated IP247 modules in the S5-115U is there an additional fan?

(Yes (0) /No (0))

21) Is FB164 called once per cycle and axis?

(Yes (0) /No (0))

Jobs triggered by momentary pulse?
(Yes (0) /No (0))

Which error messages are displayed?

____

____

22) Are the scratchpad flags being saved in the interrupting OBS?

(Yes (0) /No (0))

Page	Blank page	Release 01	Release 02
o-1 to o-3		X	
04			X
05 to 06		X	
1-1 to 1-5		X	
1-6	X		
2-1			X
2-2 to 2-59		X	
2-60	X		
3-1 to 3-19		X	
3-20	X		
4-1 to 4-18		X	
4-19			X
4-20 to 4-44		X	
5-1 to 5-23		X	
5-24			X
5-25 to 5-56		X	
6-1 to 6-2		X	
6-3 to 6-4			X
6-5 to 6-17		X	
6-18			X
6-19 to 6-21		X	
6-22			X
6-23 to 6-24		X	
6-25			X
6-26 to 6-74		X	
7-1 to 7-1o		X	
7-11			X
7-12 to 7-31		X	
7-32			X
7-33 to 7-43		X	
7-44			X
8-1 to 8-10		X	

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

V

W

Z