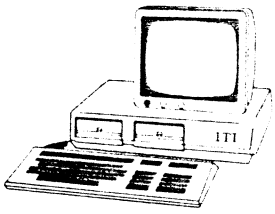

ITT 3030
Microcomputer
Hardware

User Manual
16-bit Upgrading Kit

ITT

Registered trademarks

CP/M	Digital Research Inc., USA
MS-DOS	Microsoft Inc., USA
Z80	Zilog, Inc., USA
Intel	Intel Corp., USA

Production, editing and printing:
Ingenieurbüro für Dokumentation, D-7412 Eningen

CONTENTS

1 Introduction

1.1 Contents

The information given in this user manual covers the 16-bit upgrading kit for the ITT 3Ø3Ø and completes the 8-bit version user manual. If you wish to set up the microcomputer in the 16-bit version yourself, then special note should be taken of the chapter "Installation". However this can also be performed by your dealer. If you decide to assemble the computer yourself, you will of course become accustomed with its modular construction which will enable you to extend and improve assemblies without tools.

In the chapter "Operation" you will find instructions on loading the operating system. The following chapters contain technical details about the processor, the CP/M-86 operating system, the ESC sequences and the I/O ports. A subject index and recommended reading can be found at the end.

1.2 Advantages of 16-bit technology

The following examples illustrate the advantages of the 16-bit processor.

Operating and waiting times are reduced by 5Ø to 75% of that required by an 8-bit version. New operating systems for the 16-bit-operation mean that the microcomputer is becoming easier to use and understand. This is attained through certain improvements in the hardware and software. Because a larger address range is available (up to one megabyte), the processor has less data to transfer from the memory to the permanent storage.

A larger processor throughput can be attained by using more extensive commands and through the use of a 16-bit data bus there is a considerable reduction in response time.

More highly developed operating systems are available for the 16-bit processor. These are especially easy to use and enable better management of the hardware and peripheral equipment (printer and diskette drives), as they offer a larger range of commands.

The use of the dual processor with the ITT 3Ø3Ø has also proved advantageous since both the 8-bit and 16-bit operating systems may be used (MS-DOS, CP/M-8Ø and CP/M-86) and the respective application programs written on the CP/M-8Ø can also be used on the CP/M-86.

All existing peripheral equipment and drives are also compatible.

1.3 The 16-bit version of the ITT 3030

Using the upgrading kit 3Ø3Ø/16-bit, it is possible to use the ITT 3Ø3Ø microcomputer with the CP/M-8Ø - 2.2, the CP/M-86 - 1.1 and the MS-DOS 2.11 operating systems.

All three operating systems may be used, hardware and configuration changes are not necessary.

2 Installation

2.1 General information

The modular construction of the ITT 3Ø3Ø is particularly advantageous since:

- it arranges functions clearly
- it makes upgradings easier
- it reduces manufacturer's storage and dispatch costs
- it simplifies diagnosis and the elimination of failures
- it enables outdated or defective assemblies to be exchanged and
- it allows the ITT 3Ø3Ø to adapt to new requirements and technical improvements.

A complete and consistent application of the modular concept from the initial hardware and software development, through the production and up to the user is an important prerequisite for the points mentioned. The user can insert and remove boards easily.

Please pay attention to the following points when undertaking any modifications to the ITT 3Ø3Ø:

1. Avoid damaging contents when opening package.

2. Retain any special packaging in case of a new dispatch.

3. Make sure your ITT 3Ø3Ø is disconnected from the mains supply. Observe the safety regulations.

4. Avoid electrostatic charging in the near vicinity of the microcomputer. Failure to observe this could result in components being damaged or destroyed. These electrical charges are mainly caused by static electricity which develops when the air is dry and ground insulation is high. By walking on a synthetic carpet, it is possible to charge yourself with several thousand volts. If you then touch a sensitive component, it is highly likely that it will be destroyed by the electrical discharge. You should therefore ensure that your body is grounded whenever handling such components as chips, transistors, diodes and other semiconductors. This contact should if possible be continuous. Any interruptions should be kept as short as possible. Quickly touching an earthed object, for example the central heating is usually sufficient. Do not touch two objects at the same time.

Electrostatic charging can usually be avoided by artificially raising the air humidity. A relative humidity of 5Ø to 6Ø% should be sufficient for this purpose.

5. Avoid direct contact with the components.

6. Carefully place circuit boards in the marked positions, making sure there is a firm contact.

7. Do not block the central unit air intake opening.

2.2 Scope of delivery

The upgrading kit 3Ø3Ø/16 bit comprises the following parts:

Designation	Order Number
CPU 186 board 128 KB Extension to 512 KB possible	795Ø1 12116
16-bit bus adaptor or	795Ø1 12119
16-bit bus adaptor with interface RS 232 and RS 422 or	795Ø1 12117
16-bit bus adaptor with 2 interface RS 422	795Ø1 12118
Diskette 5 1/4" with operating system and utility programs	795Ø1 25xxx
User manual, English	795Ø1 28045

The operating system CP/M-86 (or MS-DOS) is on a diskette delivered together with the equipment mentioned above. The CP/M-86 utilities can be transfered to existing CP/M-8Ø files using the PIP-function. After transfer, the CP/M-86 operating system can be manually or automatically loaded (refer to section 3).

2.3 Installation of the 16-bit version

Please observe the instructions in section 2.1.

Insert finger tips into the gap between cover and base and lift off cover with a jerking motion. If removal should be difficult, insert two thin plastic rules, one at each side, in the gap between cover and base and twist until the cover springs free. Lift the cover clear.

Remove the floppy disk drives by slightly lifting until the front retaining lugs are clear, at the same time pulling drive unit towards you until it disengages from its sockets, and the locating peg is clear of the disk drive carrier. Carefully place the disk drives on a flat surface in the same order as found in the ITT 3Ø3Ø.

Turn the small locking lever on the left underside of the floppy disk carrier. Without disturbing any circuit boards, turn the floppy disk carrier to face forward and position it so that the circuit boards can later be fitted without obstruction. It is advisable to place a sheet of cardboard between the circuit boards and the floppy disk carrier beforehand.

Fit the bus adaptor into the plastic guide socket at the back left position and plug into the plastic socket (Fig. 1).

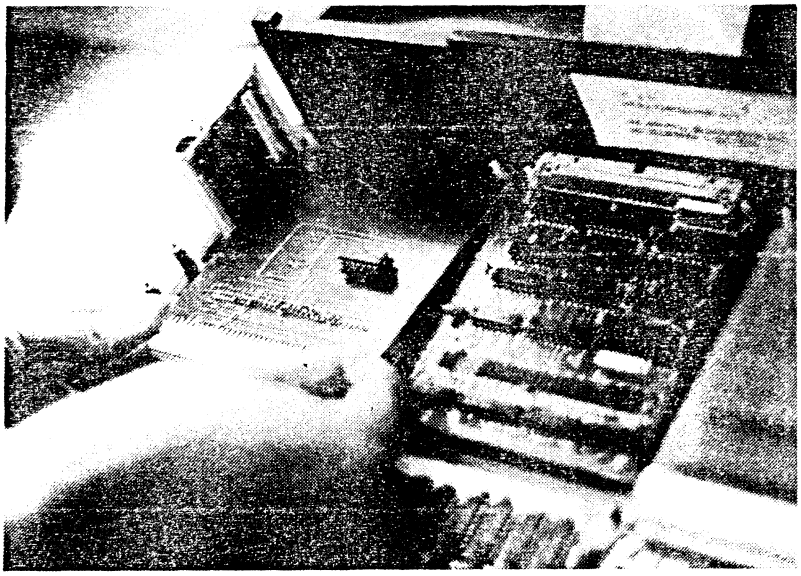

Fig. 1: Installing the bus adaptor

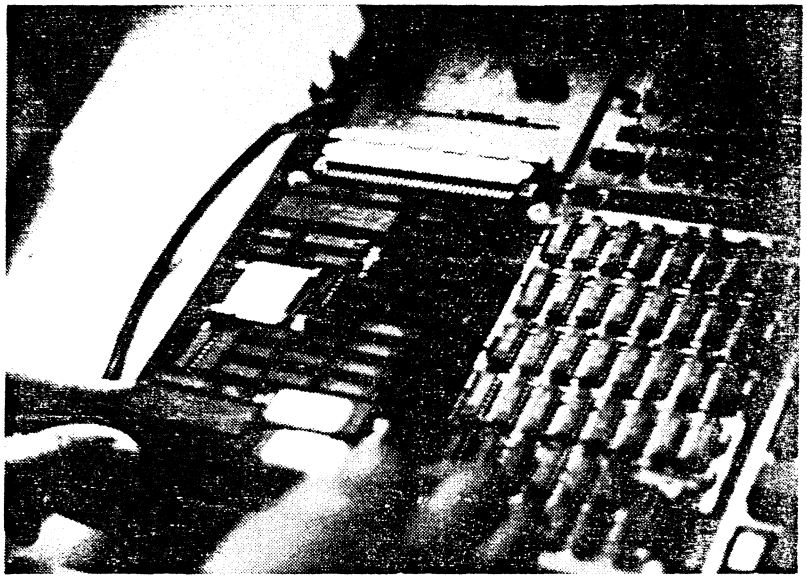

Fig. 2: Installing the CPU 186 board

Plug the CPU 186 board into the bus adaptor socket (Fig. 2).

Replace the floppy disk carrier and lock in securely.

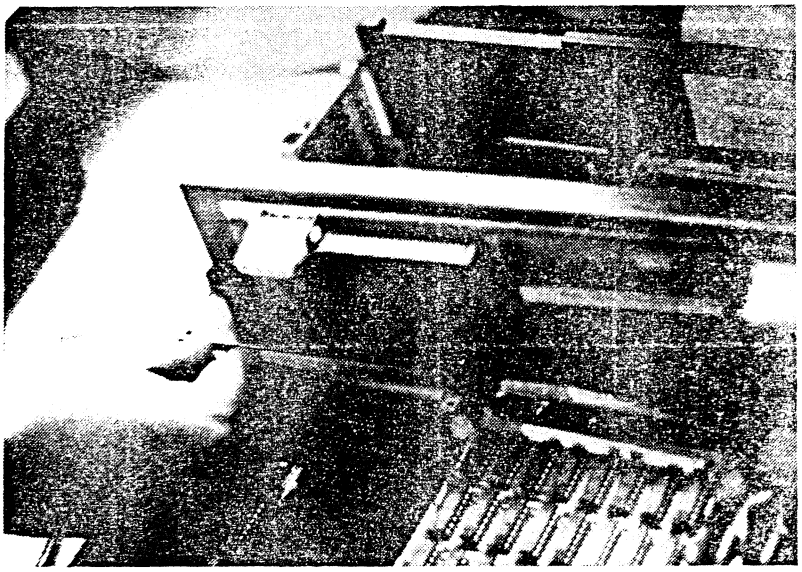

Fig. 3: Replacing the floppy disk carrier

Replace the disk drives and the cover.

The installation is now complete.

3 Operation

3.1 Preparation

Insert the diskettes following the instructions given in the microcomputer user manual.

Do not open unit while switched on. Keep cover closed to allow proper ventilation and prevent radio interference.

3.2 Installing the CP/M-80 operating system

Insert the system diskette CP/M-8Ø and boot the system by pressing the B key. The following or a similar text will appear on the screen:

```
CP/M 2.2 - 5.33 S DT
A>
```

You may now proceed as usual.

3.3 Installing the CP/M-86 operating system

3.3.1 Manual loading in 2 steps

Insert the CP/M-86 system diskette in the default drive (usually on the right) and boot the system by pressing the B key.

The same or a similar text to that found with the CP/M-8Ø will appear on the screen (see above):

```
CP/M 2.2 - 5.33 S DT
A>
```

Now enter the command:

CPM86 ↵

The CP/M-86 operating system is ready for use when the following text appears on the screen:

```
-->(CP/M-86)
A>
```

You may now proceed as usual by calling up further programs.

To return to CP/M-80, enter the command:

CPM8Ø ↵

3.3.2 Automatic loading

Insert the system diskette marked with "CP/M-86 AUTOBOOT function" in the default drive (refer to 5.1.4 and 5.1.5) and boot the system by pressing the B key.

The following or a similar text will appear on the screen:

```
CP/M 86 --->LOADING...
ITT 3030/16 (xxxKB) - CP/M 86 Standard Operating System
A>
```

xxx will be replaced by the CP/M-86 memory size, which is automatically recognized by the operating system.

You may now proceed as usual.

If you enter the command

CPM80 ↵

you can return to the CP/M-80 operating system.

If you enter the command

CPM86 ↵

CP/M-86 is loaded.

3.4 Loading the MS-DOS operating system

Insert the system diskette MS-DOS in the default drive and boot the system by pressing the B key.

The following or a similar text will appear on the screen:

```
MS-DOS VERS. 2.11
COPYRIGHT 1981,82,83 MICROSOFT CORP.
Command V.2.02
```

Now enter the date and time as shown:

```
Current date is:  Tue 01-01-84
Enter new date:  mm-dd-yy
Current time is:  11:59:20
Enter new time:  hh:mm:ss
```

You may now load the desired application or utility program.

4 The 80186 processor

Fig. 4 shows the Z8Ø processor register record and Fig. 5 that of the 8Ø186 processor.

Bit no.	15 7	14 6	13 5	12 4	11 3	10 2	9 1	8 0	7 7	6 6	5 5	4 4	3 3	2 2	1 1	0 0	
PSW				A					S	Z		AC		P		CY	Accumulator/Flags
B,C				B								C					
D,E				D								E					Register
H,L				H								L					
							SP										Stack-Pointer
							PC										Program Counter
							IX										Index Register
							IY										Index Register
							I										Interrupt Register
				I			R										Refresh register

Fig. 4: The Z8Ø processor

Bit no. (1 16-bit register)		15			0
Bit no. (2 8-bit register)		7	0	7	0
Designation	16-bit register	8-bit register			
Main register	AX BX CX DX	AH BH CH DH		AL BL CL DL	

Bit no.		15 0
Designation		
Pointer and index registers	Stack Pointer Base Pointer Source Index Destin. Index	SP BP SI DI
	Program Counter	PC
	Status Register	PSW
Segment Register	Code Segment Register Data Segment Register Stack Segment Register Extra Segment Register	CS DS SS ES

Fig. 5: The 8Ø186 processor

A Summary of the Registers

The 8Ø186 processor comprises four main registers, two pointer registers, one program counter and a status register. Each of the main registers comprises two 8-bit registers.

The contents of the main register can be changed by arithmetical, logical and input-output operations. The advantage of the division of the 16-bit register lies in the fact that those 8-bit calculations which are faster than 16-bit calculations can now be performed. Storage space can also be saved.

In the locating registers the storage spaces are addressed in the stack segment. Operands for 16-bit arithmetical and logical operations can also be stored here.

Storage spaces which contain string elements can be addressed through the index register. As with the pointer registers, operands can also be stored here.

The program counter contains the address of the next command.

The segment register works out storage addresses.

The status register displays the processor status after an operation. Bit allocation is shown in diagram 6.

Bit Nr.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					O	D	I	T	S	Z		A		P		C

Fig. 6: The 8Ø186-processor status register

Blank data fields are reserved and usually allocated zeros.

Key:

C carry
P parity
A auxilliary carry
Z zero
S sign
T trap
I interrupt
D direction
O overflow

5 The CP/M-86 operating system

5.1 Functions and commands

5.1.1 Introduction

As soon as CP/M-86 is booted, the letter "A" (meaning drive A) will appear on the screen. The system is now ready to accept commands. Commands are restricted to control characters and program selection.

The CP/M-86 comprises two sorts of programs:

- those which are loaded into the RAM as soon as the CP/M-86 system takes over from the operating system (resident programs), and
- those which are located in an external storage medium, eg. a diskette (transient programs).

5.1.2 File names and cluster designations (wild cards)

Format: d:filename.type

"d" represents the chosen drive (= drive designation)

"name" name consisting of any combination of 8 letters and figures

"." separates name and type

"type" the additional information (suffix), consisting of a maximum of 3 letters or figures may generally be chosen without restriction, although some programs require a specific type designation.

When commands differ only slightly in name, it is possible to call several files using a common command. For this purpose, wild cards may be used instead of single characters. A single character is replaced by a question mark (?) and a complete name component (ie. name or type) is replaced by a star (*). This is illustrated in section 5.1.4.

5.1.3 List of commands

Command	Meaning	Explanation
ASM86	ASSEMBLER	translates assembly language programs into machine code form.
AUTOBGB	AUTOBOOT	modifies CP/M-2 standard diskette to CP/M-86 boot diskette (specialised program for ITT 3030).
COPYDISK	COPYDISK	creates a copy of a disk that can contain CP/M-86, program files or data files.
CPM80	CPM80	installs the CP/M-80 operating system.
CPM86	CPM86	installs the CP/M-86 operating system.
DDT86	DDT	checks out programs and interactively corrects bugs and programming errors.
DIR	DIRECTORY	lists the names of files in the current user number that have the directory (DIR) suffix.
DIRS	DIRECTORY	lists the names of files that have the system (SYS) suffix.
ED	EDITOR	creates and alters character files for access by various programs.
ERA	ERASE	erases a filename from a disk directory and releases the storage occupied by the file.
GENCMD	GENCMD	uses the output of ASM-86 to produce an executable command file.
HELP	HELP	displays information on how to use each CP/M-86 command.
PIP	PIP	combines and copies files.
REN	RENAME	renames a file.
SET86GB	SETUP	sets up default values for the device drivers (specialised program for the ITT 3030).
STAT	STATUS	examines and alters file and disk status, and assigns physical I/O devices to CP/M-86 logical devices.
SUBMIT	SUBMIT	sends a file of commands to CP/M-86 for execution.
TOD	TIME OF DATE	sets and displays the system time and date.
TYPE	TYPE	lists the contents of an ASCII file on the screen.
USER	USER	changes one user number to another.

A complete review and description of all commands can be found in the Digital Research CP/M-86 User's Guide.

5.1.4 Command summary

ASM86

Purpose: The ASM-86 utility converts 8088 and 8086 assembly language source statements into machine code form (.H86).
Syntax: ASM86 filespec ($parameter-list)
Example: A>ASM86 TEXT1

AUTOBGB

Purpose: AUTOBGB modifies CP/M-2 standard diskettes to CP/M-86 diskettes. This utility program does not belong to the standard CP/M-86 system. However it simplifies the handling of programs in the ITT 3030. A full explanation is given in section 5.1.5.
Syntax: AUTOBGB
Example: A>AUTOBGB

COPYDISK

Purpose: The COPYDISK utility copies all the information on one disk to another disk, including the CP/M-86 system tracks if they are present on the source disk using interactive mode. Copying time is about 12 minutes. When using the DIDI utility on the CP/M-80, copying time is about 5 minutes.
Syntax: COPYDISK
Example: A>COPYDISK

CPM80

Purpose: Changeover from CP/M-86 to CP/M-80.
Syntax: CPM80
Example: A>CPM80

CPM86

Purpose: Changeover from CP/M-80 to CP/M-86.
Syntax: CPM86
Example: A>CPM86

DDT86

Purpose: This command installs DDT86, and can be used to test and change hex files.
Syntax: DDT86 (filespec)
Example:
```
A>DDT86 TEXTDAT.CMD
```

DIR and DIRS

Purpose: The DIR and DIRS resident commands display the names of files cataloged in the directory of an on-line disk. DIR lists the names of files in the current user number that have the Directory (DIR) suffix. DIR accepts wild cards in the file specification. The DIRS command displays the names of files in the current user number that have the System (SYS) suffix.
Syntax: DIR (filename) or DIRS (filename)
Examples:
```
A>DIR
A>DIR *.TXT
B>DIR A:*.C?D
A>DIRS *.CMD
```

ED

Purpose: The ED utility helps create and edit disk files. The ED utility is a line-orientated and context editor. This means that character files can be changed line-by-line, or by referencing individual characters within a line. The ED utility can create or alter the file named in the file specification.
More information can be found in the CP/M-86 User's Guide.
Syntax: ED input-filespec
Example:
```
A>ED TEXT1.DAT
```

ERA

Purpose: The ERA resident command erases one or more files from the directory of a disk. Wild cards may be used. Directory and data space are automatically reclaimed for later use by another file.
Syntax: ERA (filespec, type)
Examples:
```
A>ERA TEXT1.TXT
A>ERA *.*
B>ERA A:*.TXT
```

GENCMD

Purpose: The GENCMD utility uses the hex output of ASM-86 and other language processors to produce a CMD file. An optional parameter list follows the file specification.

Syntax: GENCMD filespec

Example:
```
A>GENCMD TESTDAT
```

HELP

Purpose: The HELP command provides summarized information for the most important CP/M-86 commands described in this manual. HELP with no suffix displays a list of all the available topics. Entering HELP with a filename displays information about that file. The following program functions can be displayed:
ASM86, COMMANDS, COPYDISK, DDT86, DIR, DIRS, ED, ERA, FILENAME, GENCMD, HELP, PIP, REN, STAT, SUBMIT, TOD, TYPE, USER.

Syntax: HELP CP/M-86 command

Examples:
```
A>HELP
A>HELP PIP
A>HELP STAT
```

PIP

Purpose: The PIP utility copies single files or clusters (wildcards). PIP looks for the file specified on the default (or specified) drive, copies it to the specified drive and gives it the name specified.

Syntax: PIP destination filename = source filename (options)

Examples:
```
A>PIP B:=A:*.CMD
A>PIP B:=A:PROG????.*
A>PIP B:=A:OLDNAME.DAT
A>PIP B:NEWFILE.DAT=A:OLDFILE.DAT
A>PIP NEWFILE=FILE1,FILE2,FILE3
A>PIP B:FILE.TXT=CON
A>PIP (RETURN-key)after loading:
*      A:DESTINATION.DAT=B:SOURCE.DAT
```

REN

Purpose: REN changes the name of a file that is cataloged in the directory of a disk.
The filename oldname identifies an existing file on the disk. The filename newname is not in the directory of a disk. The REN command changes the file named by oldname to the name given as newname.

Syntax: REN (drive specifier) file newname=file oldname

Examples:
```
A>REN NEWNAME.A86=OLDNAME.A86
B>REN A:NEWLIST=OLDLIST
```

SET86GB

Purpose: This utility program adjusts the device driver values to suit the 16-bit bus adaptor. The entry "8274A" should be used for dataport channel A and the entry "8274B" for dataport channel B.
This utility program does not belong to the CP/M-86 standard program. However it simplifies the handling of the ITT 3030 and is explained in more detail in section 5.1.6.

Syntax: SET86GB

Example:
```
A>SET86GB
```

STAT

Purpose: STAT gives information about the disk drives, files and devices associated with the computer. STAT can change the suffixes of files and drives. Physical devices can also be assigned to the STAT logical device names.

Syntax: STAT (drivespec)

Examples:
```
A> STAT
A> STAT *.*
A> STAT B:=RO
A> STAT MY*.* SIZE
A> STAT TEXT1.TXT RO
B> STAT A:*.COM SYS
A> STAT B:DSK:
A> STAT USR:
A> STAT VAL:
A> STAT CON: = CRT:
A> STAT LST: = LPT:
```

SUBMIT

Purpose: SUBMIT groups a set of commands together for automatic processing by CP/M-86.
Normally, commands are entered one line at a time.
However, if a sequence of commands must be entered often, it is easier to batch the commands together using the SUBMIT utility. To do this, a file, complete with commands, must be created. The file is identified by the filename, and must have a SUB suffix. When the SUBMIT command is entered, SUBMIT reads the file named and prepares it for interpretation by CP/M-86.

Syntax: SUBMIT (filename)

Example:
```
A>SUBMIT SUBFILE
```

TOD

Purpose: Using the TOD utility the day and the time can be examined and set.

Syntax: TOD mm/dd/yy hh:mm:ss

Examples:
```
A>TOD 03/31/84 08:45:00
A>TOD
```

TYPE

Purpose: TYPE displays the contents of a character file (ASCII code) on the screen.

Syntax: TYPE filename

Example:
```
A>TYPE TEXT1.DAT
```

USER

Purpose: USER displays and changes the current user number. The disk directory can be divided into distinct groups according to a user number (0 to 15).

Syntax: USER (number)

Examples:
```
A>USER
A>USER 3
```

5.1.5 AUTOBGB

The automatic loading of the CP/M-86 operating system requires the following steps:
Using the PIP utility, AUTOBGB (comprising the files AUTOBGB.COM and AB.COM) must be transfered from the CP/M-80 system onto the required diskette.
The AUTOBGB utility can be found on the CP/M-86 diskette. By entering the command AUTOBGB the following text will appear on the screen:

```
*****ITT 3030 CP/M-86 Autoboot Generator V1.1*****

This program will modify a CP/M-2.2 standard diskette
to a CP/M.86 boot diskette. Please place your diskette
into a suitable drive and enter the name of this drive
using "A", "B" or "C". If you wish to leave the program,
enter the letter "V".


Please select a letter ("A", "B", "C" or "V") :
```

If, for example, you enter "A", the program will finish the run after about 25 seconds and the following text will appear on the screen:

```
Diskette modification successfully completed.
Please select a letter ("A", "B", "C" or "V") :
```

You may now end the program by entering "V". However, if you enter "A" again, the following text will appear on the screen:

```
Diskette has already been modified!
Do you wish to remove AUTOBOOT (J/N) ?
```

If you enter "J" (YES), the AUTOBGB utility will be removed.
If you enter "N" (NO), the following text will appear on the screen:

```
Please select a letter ("A", "B", "C" or "V") :
```

If you now enter "V", the program will be terminated. The operating system prompt will appear.

If a non-existent drive is entered, the following text appears on the screen:

```
ERROR. CCP start sector not found.
Please select a letter ("A", "B", "C" or "V") :
```

After modification, the AUTOBGB utility will be permanently registered on the required disk or diskette.

5.1.6 SET86GB

The SETUP utility SET86GB may be used for setting up the device drivers to the serial interfaces of the 16-bit bus adaptor:

After entering

SET86GB

the following message appears:

```
                                        ITT 3030/16

CP/M 86 SETUP (Release 1.1)

   SETUP

The utility for simplified system generation.

A program for the ITT 3030 from the SEL AG
(c) 1984 by S&P / O.M. Bartels, Stuttgart

To continue press any key:                  ---->
```

The following main menu will now appear:

```
                                        ITT 3030/16

CP/M 86 SETUP / MAIN MENU

S  : Adjust the default values for the device driver
V  : Leave program

Please select a letter ("S" or "V"):
```

By entering "S" you will be requested to answer the following questions:

```
CP/M-86 SETUP/CHARACTER INPUT-OUTPUT - driver default values

Driver identification:                    Interface:
Name of device driver   6 spaces          Input:      Output:
                                          Device:
Baud rate (adjustable:  ) :               Protocol:

Ø - none              8 - 1200 baud       Possible protocol .. :
1 - 50    baud        9 - 1800 baud       Protocol type ..... :
2 - 75    baud       10 - 2400 baud
3 - 100   baud       11 - 3600 baud       Ø: -/1: XON-XOFF / 2:ETX-ACK
4 - 134,5 baud       12 - 4800 baud
5 - 150   baud       13 - 7200 baud
6 - 300   baud       14 - 9600 baud       Data:               Parity:
7 - 600   baud       15 - 19.2kbaud
                                          7-bit ... :         employed:

Current baud rate:                        2-stop bit:         Odd parity:
```

After each entry the cursor jumps to the next position. Default values may be assumed by pressing the RETURN key.

The "up-arrow" key allows you to return to the preceding position and with the "right-arrow" and "down-arrow" keys, you can jump to the next position.

The "ESC" key allows you to return to the main menu at any time. To leave the main menu again, simply press "V".

After all the data has been registered, the following text will appear on the screen:

```
Do you wish to store the new parameters (J/N) --->
```

If you now enter "J" (Yes), the new data will be stored in the diskette.

5.2 Display driver interface

The interface is orientated according to the characters. Special characters and escape sequences are interpreted. The character which is to be transfered is located in register C. All registers are changed.

5.2.1 Special characters

a) Ø7 --> Bell
The buzzer sounds for approx. 1 second.

b) Ø8 --> Cursor left (backspace)
The cursor moves one position to the left.

c) ØA --> Linefeed
The cursor moves to the same position on the line below. If the cursor is already on the last line, the screen will roll one line upwards. The bottom line is erased. The cursor remains in the same column.

d) ØC --> Erase screen (form feed)
The screen controller is initialised and the driver is set to its start position. The cursor appears in the bottom lefthand corner. The screen is erased.

e) ØD --> End line (carriage return)
The cursor moves to the start of the line.

f) 11 --> Enlarge mode on (only combi I)
The screen is erased. The cursor appears enlarged in the bottom lefthand corner. The background attributes are reset.

g) 12 --> Termination code
All background attributes are reset.

h) 13 --> Enlarge mode off (only combi I)
The screen is erased. The cursor appears in normal size in the bottom lefthand corner. The background attributes are reset.

i) 1C --> Inverse video on
All following characters are displayed in inverse mode.

5.2.2 CP/M-86 Escape Sequences

The monitor driver is able to interpret up to 64 escape sequences. The escape codes, complete with the order of functions, are listed in a table of 256 bytes. Some functions contain sequence parameters which can be transfered in two ways, according to their function:

a) in Hex code (hh)
eg. the code Ø8H is transfered as a Hex byte Ø8H
b) in ASCII code (xx)
eg. the code Ø8H is transfered as a 2-byte-ASCII 3ØH,42H

The following list of escape sequences corresponds with the sequence order in the table mentioned above. The code 1BH is used as a default value for the escape symbol. Other values can however be similarly substituted.

5.2.2.1 Cursor movements

1B,11 Cursor home
The cursor returns to the top lefthand corner.

1B,12 Cursor off
The cursor is turned off, but still follows operations.

1B,13 Cursor on
The cursor is turned on.

1B,1C Cursor up
The cursor moves to the line above.

1B,1A Cursor right
The cursor moves one position to the right. If the cursor is already at the end of a line, it will move to the start of the next line (default value).

1B,1F,hh,hh Positioning the cursor with offset
The cursor will appear at the given position. The first hex number defines the line and the second defines the column. The default offset value is for both line and column 2ØH.

5.2.2.2 Erase functions

1B,1B — Erase remainder of line
The line is erased from the cursor position onwards. The cursor position remains constant.

1B,19 — Erase remainder of line
The line is erased from the cursor position onwards. The cursor jumps to the start of the next row.

1B,17 — Erase remainder of screen
The screen is erased from the cursor position onwards. The cursor position remains constant.

5.2.2.3 Special functions

1B,24,xx — Transparent mode on
The associate byte is displayed on the screen without interpretation.

1B,25 — Erase keyboard buffer
The keyboard is initialised and the FIFO buffer storage is erased.

1B,2Ø — Initialisation
The screen monitor and keyboard are initialised. The screen can be erased when desired. The cursor appears in the bottom lefthand corner.

Parameter: XX (only default values):

bit Ø = Ø Cursor returns to home position after last screen position.
bit Ø = 1 Linefeed after last screen position (default values).

bit 1 = Ø Keyboard interprets the following key codes immediately (default values):
81 cursor moves to the left and Ø8 is returned
82 cursor moves to the right and the character covered by the cursor is returned.
bit 1 = 1 All key codes are transmitted.

bit 2 = Ø 7-bit ASCII returns (default values).
bit 2 = 1 8-bit ASCII returns

bit 3 = Ø No transformation (default values).
bit 3 = 1 All small letters are transformed into capital letters.

bit 4 = Ø Screen is erased (default values).
bit 4 = 1 Display remains.

1B,1Ø,hh Display blanks
The number of blanks (hh) is displayed.

1B,1Ø,hh,ii Display several symbols with one code
hh characters are displayed in the ii code.

1B,16,hh,ii Positioning the cursor without offset
The cursor is transfered to a given position without the use of offset.

1B,14 Screen rolling (only combi I)
The screen rolls up one line.
The top line appears at the bottom and the relative cursor position remains constant.

5.3 Controlling the screen: I/O-ports

Screen controlling consists of two interfaces; one for the CPU and one for the adaptor card at each output unit. The essential factor for screen controlling is a highly integrated control component of the type VTAC 5Ø27 or TMS 9927.

All the functions of this control component can be programmed through output ports. They can be found at the following addresses:

2ØH to 26H, 2AH, 2BH, 2EH and 35.

Function	Port	Z8Ø-command
Time control		
stop	2A	OUT 2AH
start	2E	OUT 2EH
Control register	2Ø..26	OUT 2ØH..OUT 26H
Scrolling	2B	OUT 2BH

Stopping and starting of the time control and scrolling are initiated as soon as the output operations have been carried out. For this purpose, it is sufficient to enter any character into the output-ports. The control register which is programmable through the ports 2ØH to 26H, must however be loaded with precisely defined values.

For every character line, the default is 12.

The functions of the control register are shown in the following table:

Programming the video control (VTAC 5Ø27/TMS 9927)

<table>
<tr><th>Register number</th><th>Bit number</th><th>Programmed function</th></tr>
<tr><td>2Ø</td><td>7..Ø</td><td>Line length, determined by

$\frac{\text{Video clock rate (=12,8112 MHz)}}{\text{Character width} \cdot \text{Scan lines} \cdot \text{Picture frequency}}$

The following values are given (European Standards):

Video clock rate = 12,8112 MHz
Character width = 8
Scanning lines = 312 (non-interlaced)
= 625 (interlaced)
Picture frequency= 5Ø (non-interlaced)
= 25 (interlaced)</td></tr>
<tr><td>21</td><td>7

6..3

2..Ø</td><td>Scanning mode
interlaced = 1
non-interlaced = Ø
Width of line synchronisation pulse in units of character width.
Distance between right edge of screen and line synchronisation pulse in units of character width</td></tr>
<tr><td>22</td><td>6..3
2..Ø</td><td>Scanning lines per character (-1)
Number of suitable characters per line:
Key:
<table><tr><th>bit 2</th><th>bit 1</th><th>bit Ø</th><th>No. of characters</th></tr><tr><td>Ø</td><td>Ø</td><td>Ø</td><td>2Ø</td></tr><tr><td>Ø</td><td>Ø</td><td>1</td><td>32</td></tr><tr><td>Ø</td><td>1</td><td>Ø</td><td>4Ø</td></tr><tr><td>Ø</td><td>1</td><td>1</td><td>64</td></tr><tr><td>1</td><td>Ø</td><td>Ø</td><td>72</td></tr><tr><td>1</td><td>Ø</td><td>1</td><td>8Ø</td></tr><tr><td>1</td><td>1</td><td>Ø</td><td>96</td></tr><tr><td>1</td><td>1</td><td>1</td><td>132</td></tr></table></td></tr>
</table>

<table>
<tr><td>23</td><td>7,6</td><td>Delay factor for picture signal and cursor in relation to normal screen position (in units of character width)
Key:
<table><tr><th>bit 7</th><th>bit 6</th><th>Display signal delayed by character:</th><th>Cursor</th></tr><tr><td>Ø
Ø
1
1</td><td>Ø
1
Ø
1</td><td>Ø
1
2
2</td><td>Ø
Ø
1
2</td></tr></table></td></tr>
<tr><td>24</td><td>7..Ø</td><td>Number of scanning lines per screen.
Key:
interlaced:
$Z_m = \frac{\text{Scanning lines} - 513}{2}$
non-interlaced:
$Z_o = \frac{\text{Scanning lines} - 256}{2}$</td></tr>
<tr><td>25</td><td>7..Ø</td><td>Number of empty scanning lines at start of screen (between start of screen synchronisation pulse and first line of data).</td></tr>
<tr><td>26</td><td>5..Ø</td><td>Number of last character line on full screen. Counting begins with zero.</td></tr>
</table>

Example:

24 * 8Ø display (non-interlaced)

The registers should be loaded as follows:

Register	Contents
2Ø	65H
21	2EH
22	5DH
23	57H
24	1DH
25	12H
26	17H

e.g.: LD A,65H
OUT 2ØH,A

Address format for access to video RAM:

A15	A14	A13	A12	A11	A1Ø A9 A8 A7	A6 A5 A4 A3 A2 A1 AØ
Ø	Ø	1	1		Line number	Character position

Using a multiplexer, this originally 12-bit address is reduced to an 11-bit address and transfered to the memory. The address allocation appears as shown:

Internal character addresses in the video RAM.

Line number	Character position	
	Ø to 63	64 to 79
Ø	ØØØ .. Ø3F	6ØØ .. 6ØF
1	Ø4Ø .. Ø7F	64Ø .. 64F
.	.	.
.	.	.
.	.	.
15	3CØ .. 3FF	7DØ .. 7DF
16	4ØØ .. 43F	62Ø .. 62F
.	.	.
.	.	.
.	.	.
31	7CØ .. 7FF	

Important: The storage areas for the characters 64 to 79 in lines Ø to 23 and the characters Ø to 63 in lines 24 to 31 are identical. Consequently, the largest possible format displays are either 32 lines for 64 characters or 24 lines for 8Ø characters.

Example:

To display a character on the screen.

```
        LD      A,17H       ;Load register 6. Positioning
        OUT     26H,A       ;initialising
        LD      C,CHARACTER ;the displayed character
        LD      HL,POSN     ;address in picture repetition
                             memory
                            ;ie. line and column

GAP:    IN      A35,H
        BIT     6,A
        JP      Z,GAP

        CALL    ACCESS      ;Call up UPR for access
                            ;to internal storage area
                            ;UPR can be found above CØØØH

        CALL    ØFE73       ;Waiting for a key to be pressed
                            ;Until a key is pressed,
                            ;the character
                            ;can be observed
        RST     38H         ;Restart ZSID

ACCESS: LD      A,ØØ
        OUT     ØF6,A
        LD      HL,C        ;Storage access, page 8
        LD      A,1D        ;Return to page Ø
        OUT     ØF6,A
        RET
```

5.4 Input-output address space

Hard disk	1ØH - 1FH
Keyboard/CRT	2ØH - 3FH
16-bit card	4ØH - 41H
Floppy controller	5ØH - 5FH
8"-adaptor	6ØH - 6FH
SIO card	8ØH - 8FH
PIO card	9ØH - 9FH
Arithmetic unit, real-time clock	AØH - AFH
Colour graphics card	BØH - B7H
CPU, timer etc.	EØH - FFH

Note: The addresses CØH to DFH are not yet allocated.

5.5 I/O-byte allocation

Log. dev.		Phys. dev.	Peripheral unit
CONSOLE: (CON:)	*	TTY: CRT: BAT: UC1:	8274-B Screen/keyboard Input = AXI: Output = LST: Screen/keyboard
READER (RDR:/AXI:)	*	TTY: RDR: UR1: UR2:	8274-B 8274-A Combo input Combo input
PUNCH (PUN:/ACO:)	*	TTY: PTP: UP1: UP2:	8274-B 8274-A Combo output Combo output
LIST (LST:)	*	TTY: CRT: LPT: UL1:	PIO output Screen Combo output 8274-A

* Default value.

5.6 Allocation of diskettes, hard disks and RAM-floppies (CP/M)

Drive	Allocation
A	5 1/4-inch floppy P 560
B	5 1/4-inch floppy P 560
D	External 8-inch floppy
E	free
H	free
I	free
J	External hard disk
K	External hard disk
P	RAM floppy

5.7 Literature

Digital Research CP/M-86	User's Guide Programmer's Guide System Guide
Russell Rector George Alexy	The 8Ø86 Book McGraw-Hill, Inc., USA
Rodnay Zaks	CP/M Handbook with MP/M Sybex Verlag Düsseldorf
Carl Townsend	How to get started with MS-DOS Dilithium Press, Beaverton, Oregon, USA
Tony Zingal	Intel's 8Ø186: A 16-Bit Computer on a Chip. BYTE, April 83, pg.132 - 146

INDEX

Notes

Notes

Order No. 79501 28045 GB

Standard Elektrik Lorenz AG
Gruppenbereich
Private Kommunikationssysteme
Produktbereich Mikrocomputer
Motorstraße 25
D-7000 Stuttgart 31

Standard Telephon & Radio AG
Brandschenkestraße 178
CH-8027 Zürich

ITT