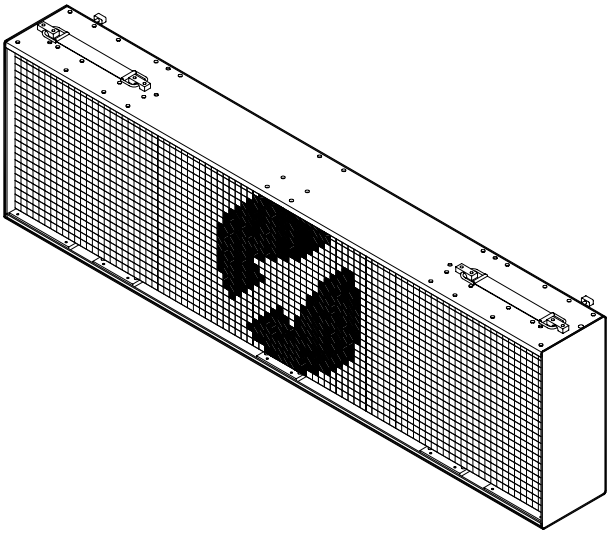
GRAPHIC DISPLAYBOARD

µGRAPH Light

User manual



Version 1.01

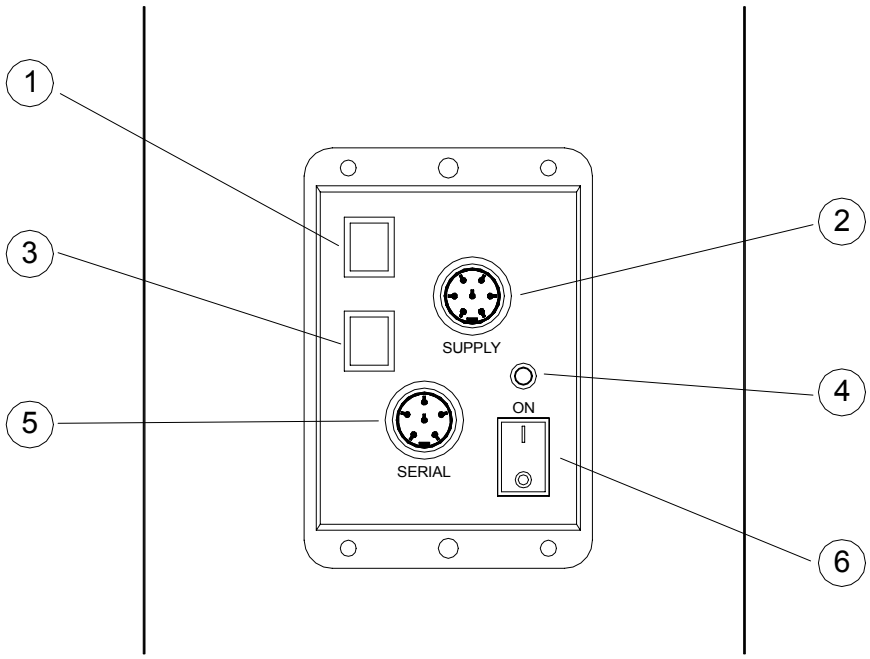
MICROGATE

Microgate S.r.l.
Via Stradivari, 4
I-39100 BOLZANO - ITALY
http://www.microgate.it

INDEX

1 GRAPHIC DISPLAYBOARD μGRAPH LIGHT (MICROGRAPH)

1.1 CONTROL PANEL

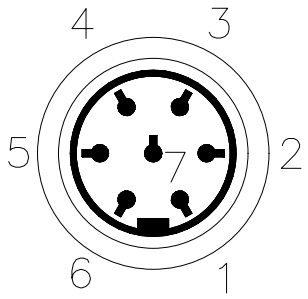


1. Green button START STOP (MODIFY DISCHARGE/CHARGE) used for:
   - modification of values in program setting (keep pressed down for fast forward)
   - selection of battery discharge and recharge
2. 7 pole Amphenol (or plug socket) connector for external power supply and battery recharge
3. Yellow button LAP RESET (SETUP DIRECT CHARGE) used for:
   - confirmation of program settings
   - selection of immediate battery recharge selection
4. Internal battery status signal Led
5. 6 pole Amphenol input/output connector for Serial
6. On/Off switch

1.1.1 Connections

- ***SUPPLY Input (7 pole Amphenol)***

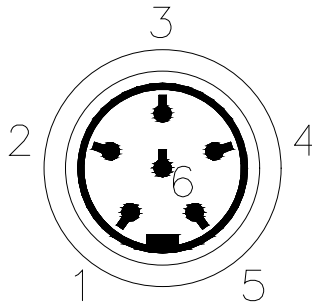
1 Ground
2 Ground
3 Ground
4 External Supply (8-25V)
5 External Supply (8-25V)
6 External Supply (8-25V)
7 Remote on/off input



7 pole Amphenol cable connector

- ***SERIAL Input (6 pole Amphenol)***

1 SERIAL output RS232 TX
2 SYNC IN
3 SERIAL input RS485 + RX
4 SERIAL input RS485 - RX
5 Ground (cable braiding)
6 SERIAL input RS232 RX



6 pole Amphenol cable connector

1.2 POWER SUPPLY

Power can be supplied in three ways:

- By connecting the µGRAPH displayboard to the MICROGATE battery charger. In this way it is possible to supply a mains graphic displayboard and to keep the batteries charged at the same time. This guarantees perfect functioning also when the mains power supply is interrupted. The ACC062 adaptor accepts an input of alternate current at 50 or 60Hz, within a range of 100 and 240 Volts
- By using the internal batteries of the displayboard. In this case autonomy is usually above 30 hours of continuous functioning (depending on the type of display used).
- By connecting the displayboard to any continuous current supply (whether steady or not) between 10 and 30 Volts which is able to supply at least 30W peak power and about 4W average power. A car battery guarantees several days' autonomy.

If 2 or more µGRAPH displayboards must be powered, a single dedicated adaptor can be requested from MICROGATE.

Important note: the adaptor *ACC062* for the µGRAPH displayboard **is not suitable for outdoor use**. Consequently Microgate does not accept any responsibility for damage to persons or things due to incorrect use of the adaptor.

1.2.1 Battery Recharge

If the batteries are low, either the *discharge/recharge* or the *immediate recharge* procedure can be carried out.
In the first case, the batteries are first discharged and only subsequently recharged. This allows the batteries to maintain their original capacity over a long period.
To select **discharge/recharge, keep the "START STOP (MODIFY CHARGE/DISCHARGE)" button on the control panel pressed down for at least 2 seconds with the displayboard switched off** after connecting an external power source to the connector SUPPLY**.** The operation will take from a minimum 9hours to a maximum of about 13hours, depending on the initial battery charge level.
If you choose immediate recharge instead, the operation will last about 9 ours. However, although this type of recharge takes less time, it should only be used in exceptional circumstances as it shortens the life of the batteries.
To select **immediate recharge**, **keep the yellow "LAP RESET (SETUP DIRECT CHARGE)" button on the control panel pressed down for at least 2 seconds with the displayboard switched off** after connecting an external power source to the connector SUPPLY**.**
In both recharge modes it is possible **to interrupt the process** by pressing the START STOP and LAP RESET keys simultaneously.

The LOW BATTERY led on the control panel tells you the battery charge status, the type of power source used and the recharge operation status when the battery is being recharged.

EXTERNAL SUPPLY	
STATUS	LOW BATTERY LED
• ON/OFF Displayboard • Batteries *Charged*	Green – Green – Pause
• ON/OFF Displayboard • Batteries *Discharged*	Green – Red – Pause

INTERNAL SUPPLY (BATTERY)	
STATUS	LOW BATTERY LED
• OFF Displayboard • Batteries *Charged or Discharged*	OFF
• ON Displayboard • Batteries *Charged*	Green – Pause – Green – Pause
• ON Displayboard • Batteries *Discharged*	Red – Pause – Red – Pause

DISCHARGE/CHARGE	
STATUS	LOW BATTERY LED
• Start of Discharging	
• Discharging Over – Start of Recharging	Pause – Green – Pause – Green FAST
• Recharging Over	Green Continous
DIRECT CHARGE	
STATUS	LOW BATTERY LED
• Start of Recharging	Pause – Green – Pause – Green FAST
• Recharging Over	Green Continous

1.3 µTAB FIRMWARE

Every time it is switched on, µGRAPH displays the firmware version stored at that moment:

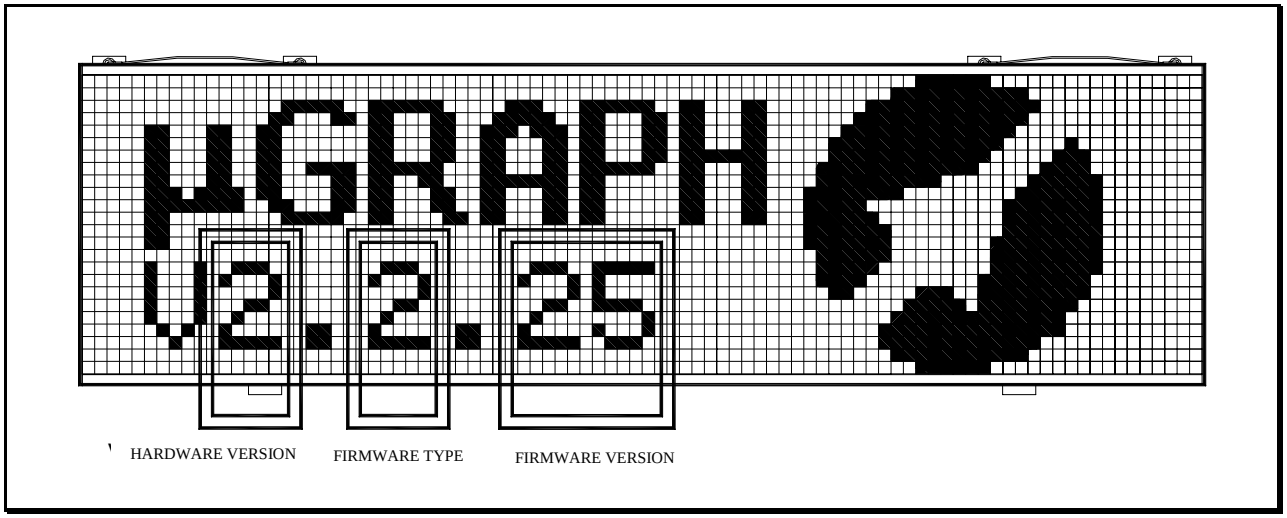


Figure 1

As can be noted in Figure 1, the numerical code of the firmware consists of 3 parts:

1. *Harware Version*, the first number: indicates the version of the motherboard which controls the displayboard.

2. *Type of Firmware*, set to 1 (Standard Firmware)

3. *Firmware Version*, the last two numbers: it is important to provide the MICROGATE staff with this number if you require technical assistance.

1.3.1 Updating of Firmware

Free µGRAPH Firmware updating is possible by downloading the latest versions from the site http://www.microgate.it or requesting them from MICROGATE.
Once the update file has been obtained, the operations to be performed are simple:

A. Switch off µGRAPH
B. Press the START STOP (MODIFY) e LAP RESET (SETUP) buttons simultaneously and, while keeping them pressed down, switch on the displayboard (attention, the power supply must be disconnected before switching on the displayboard); the led on the displayboard should slowly blink red-green.
C. Connect the PC serial to the µGRAPH SERIAL connector (using the 20m CAB010 cable or the 2m CAB001)
D. From the PC run the uFlasher program containing the latest Firmware version. During programming, the LOW BATTERY led on the displayboard turns ORANGE.
E. After about 2 minutes programming is over (uFlasher shows the message "Device successfully programmed"). At this point, the led turns GREEN.
F. The µGRAPH Firmware has been successfully updated. Now you can switch off the displayboard.

Any error in programming is indicated by the LOW BATTERY led on the displayboard, which turns RED. In the unlikely eventuality that this should happen, simply repeat the procedure indicated above.

1.4 PROGRAM

By selecting the Program it is possible to command µGRAPH through the SERIAL serial communication port or the RADIO connector.
The commands that can be given to µGRAPH are listed in par.0 Appendix A: µGRAPH Serial Frame on p.39. We strongly recommend that the less expert should exploit the versatility of the MICROGATE µBOARDS software to control µGRAPH correctly rather than making tedious attempts at direct programming.

Note: the commands identified as 'priority' or 'non-priority' (or 'strong' and 'weak') should be understood as being priority or non-priority in relation to the pause command. For example, a '*Weak reset*' command given after a command pause will be carried out only at the end of the pause. A '*Strong reset*' command, on the other hand, will be carried out directly.

Setup
In Program, setup allows you to re-initialize all µGRAPH parameters to standard values.
Keep LAP-SETUP pressed for at least two seconds to enter Setup

Font: ***SMALL*** Set the Font type required with START-MODIFY (SMALL 6x8 pixels, MEDIUM (proportional medium) 12xVariable pixels, HUGE 24x15 pixels, SPECIAL 24x15 pixels (only numbers e.g. M:SS.CC), MED.FIXED (non-proportional medium) 12x10 pixels, SPECIAL2 24x13 pixels (only numbers e.g. MM:SS.CC)
Press LAP-SETUP

Green: Test Pixel
Press START-MODIFY to perform the displayboard test, LAP-SETUP to exit without performing the test

Sure ? (Green)
Press START-MODIFY to confirm, LAP-SETUP to exit without performing the test

Green: INITIALIZE
Press START-MODIFY to confirm, LAP-SETUP to exit without initializing

Sure ? (Green)
Press START-MODIFY to confirm, LAP-SETUP to exit without initializing

1.5 µGRAPH SERIAL FRAME

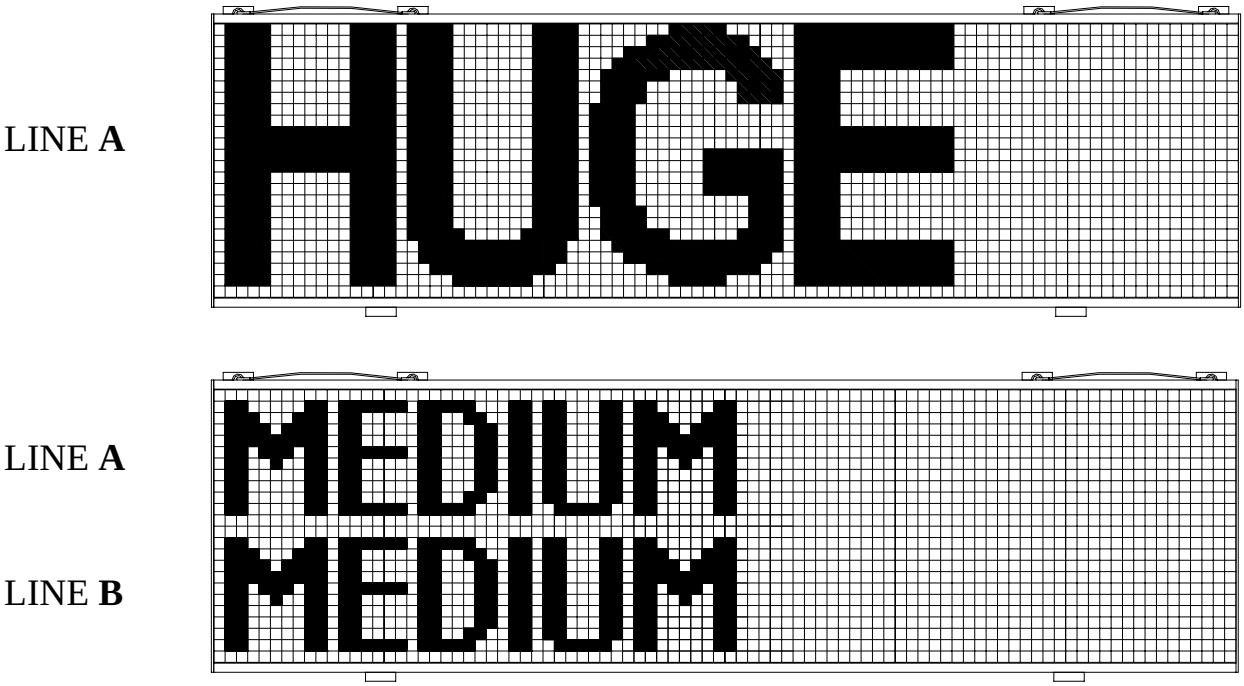
(8 BIT, 1 STOP, PARITY NONE)

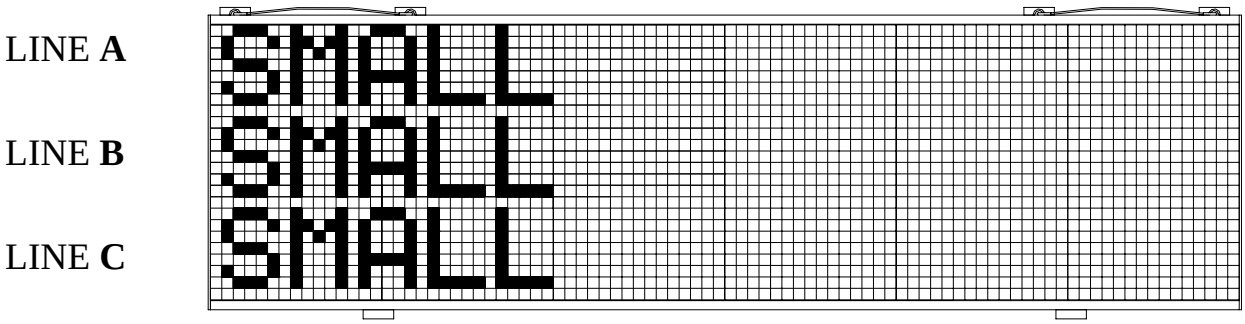
The serial speed is set at 9600 bit/s. The data transmitted consists of 8 bits of information and 1 bit of Stop, without parity check. There is no hand-shaking.
The µGRAPH protocol is based on 2 standard records:

- textual, which allows the display of simple strings of text
- graphic, which allows the display of strings of text and images

1.5.1 Text Frame

When textual records are used, the µGRAPH displayboard is divided up into a number of sections depending on the height of character employed. The textual record will be fully compatible with that of the µTAB alphanumeric displayboard.
The character used by the displayboard can be set manually by means of the *Setup* of program 0 (see Program on p.11), or the appropriate serial command.
3 character heights can be used, and the display area of each µGraph will be divided up like this:





As can be seen, when the HUGE character height is used, µGRAPH has a single line of display. If the MEDIUM character is used, the displayboard is divided into two parts (line A and line B). Finally, if the SMALL character is used, the displayboard is divided into 3 sections (line A, line B e line C).
As well as height, another important characteristic of the fonts is width. Some fonts can be shown in both proportional and non-proportional mode *(see chap.* Proportional and Non-Proportional Fonts on p. 20*)*.

1.5.1.1 Text Frame format and Command Table

The format of the text frame is as follows:

Field	Length	Conten.	Meaning
Start of Frame	1	ESC (0x1B)	Start of Command frame
Address	1	‘ ’(0x20)	Blank
Command	1	(Any)	Command to be sent to Displayboard (see below)
Data	Variable	Variable	Optional data area of command
End of frame	1	ETX (0x03)	End of Command frame
Checksum	1	Variable	7-bit checksum made on whole frame

The table below gives the various commands which can be used in the command field:

Command	Command Code	
• Date Display	A	Dec. 65 - Hex 41h
• Program start	B	Dec. 66 - Hex 42h
• Time setting sensitive to break	C	Dec. 67 - Hex 43h
• Time setting not sensitive to break	c	Dec. 99 - Hex 63h
• Break setting (it breaks the execution of following commands)	D	Dec. 68 - Hex 44h
• Date setting	d	Dec. 100 - Hex 64h
• Entry Point/Label for loops	E	Dec. 69 - Hex 45h
• Program end	K	Dec. 75 - Hex 4Bh
• Loop/Goto	L	Dec. 76 - Hex 4Ch
• Internal clock time setting (Real Time Clock)	M	Dec. 77 - Hex 4Dh
• Internal clock time display(Real Time Clock)	N	Dec. 78 - Hex 4Eh
• Running string writing	O	Dec. 79 - Hex 4Fh
• Stop running string	o	Dec. 111 - Hex 6Fh
• Internal hardware program execution	P	Dec. 80 - Hex 50h
• "Weak" displayboard reset (sensitive to Break)	R	Dec. 82 - Hex 52h
• "Strong" displayboard reset (not sensitive to Break)	r	Dec. 114 - Hex 72h
• Fixed string writing	S	Dec. 83 - Hex 53h
• Display of set time	T	Dec. 84 - Hex 54h

1.5.1.2 Syntax of Text Frame commands

1.5.1.2.1 DATE DISPLAY

Date display		
Command Code	**'A'**	
Data		
Item	***Length (bytes)***	***Notes***
Position (column)	2	00 = first character on the left
Mode	1	0=disabling 1=DD/MM/YY 2=DD MM YY

1.5.1.2.2 TIME SETTING SENSITIVE TO BREAK

Time setting sensitive to Break		
Command Code	**'C'**	
Data		
Item	***Length (bytes)***	***Notes***
HHMMSSCC	8	hours minutes seconds hundredths

1.5.1.2.3 TIME SETTING NOT SENSITIVE TO BREAK

Time setting not sensitive to Break		
Command Code	**'c'**	
Data		
Item	***Length (bytes)***	***Notes***
HHMMSSCC	8	hours minutes seconds hundredths

1.5.1.2.4 BREAK SETTING (IT BREAKS THE EXECUTION OF FOLLOWING COMMANDS)

Break setting (it breaks the execution of following commands)		
Command Code	**'D'**	
Data		
Item	***Length (bytes)***	***Notes***
Delay	5	Delay duration in hundredths

1.5.1.2.5 DATE SETTING

Date setting		
Command Code	**'d'**	
Data		
Item	***Length (bytes)***	***Notes***
Date	6	DDMMYY format
Day	1	1=Sunday 2=Monday ...

1.5.1.2.6 INTERNAL CLOCK TIME SETTING (REAL TIME CLOCK)

Internal clock time setting (Real Time Clock)		
Command Code	**'M'**	
Data		
Item	***Length (bytes)***	***Notes***
HHMMSSCC	8	hours minutes seconds hundredths

1.5.1.2.7 INTERNAL CLOCK TIME DISPLAY(REAL TIME CLOCK)

Internal clock time display (Real Time Clock)		
Command Code	**'N'**	
Data		
Item	***Length (bytes)***	***Notes***
Position (column)	2	00 = first character on the left
Mode	1	0 = disabling 1 = format HH:MM:SS 2 = format MM:SS 3 = format HH:MM 24h (e.g. 15.25) 4 = format HH:MM 12h (e.g. 3:25 PM)

1.5.1.2.8 RUNNING STRING WRITING

Running string writing		
Command Code	**'O'**	
Data		
Item	***Length (bytes)***	***Notes***
Position (column)	2	00 = first character on the left
N° of columns involved	2	0 < n <= 81
Delay of string motion	3	In hundredths
String	<=255	Characters to be written

1.5.1.2.9 STOP RUNNING STRING

Stop running string		
Command Code	**'o'**	
Data		
Item	***Length (bytes)***	***Notes***
HHMMSSCC	8	hours minutes seconds hundredths

1.5.1.2.10 INTERNAL HARDWARE PROGRAM EXECUTION

Internal hardware program execution		
Command Code	**'P'**	
Data		
Item	***Length (bytes)***	***Notes***
N° of program	2	00 = 1st program (as for switch)

1.5.1.2.11 "WEAK" DISPLAYBOARD RESET (SENSITIVE TO BREAK)

"Weak" displayboard reset (sensitive to Break)		
Command Code	**'R'**	
Data		
Item	***Length (bytes)***	***Notes***
None		

1.5.1.2.12 "STRONG" DISPLAYBOARD RESET (NOT SENSITIVE TO BREAK)

"Strong" displayboard reset (not sensitive to Break)		
Command Code	**'r'**	
Data		
Item	***Length (bytes)***	***Notes***
None		

1.5.1.2.13 FIXED STRING WRITING

Fixed string writing		
Command Code	**'S'**	
Data		
Item	***Length (bytes)***	***Notes***
Position (column)	2	00 = first character on the left
String	<=81	Characters to be written

1.5.1.2.14 DISPLAY OF SET TIME

Display of set time		
Command Code	**'T'**	
Data		
Item	***Length (bytes)***	***Notes***
Position (column)	2	00 = first character on the left
Mode	1	0 = disabling 1 = format HH:MM:SS 2 = format MM:SS 3 = format HH:MM 24h (e.g. 15.25) 4 = format HH:MM 12h (e.g. 3:25 PM)

The following 4 commands are used for setting "programs" (series of operations to be performed in sequence):

1.5.1.2.15 PROGRAM START

Program start		
Command Code	**'B'**	
Data		
Item	***Length (bytes)***	***Notes***
None		

1.5.1.2.16 PROGRAM END

Program end		
Command Code	**'K'**	
Data		
Item	***Length (bytes)***	***Notes***
None		

1.5.1.2.17 ENTRY POINT/LABEL FOR LOOPS

Entry Point/Label for loops		
Command Code	**'E'**	
Data		
Item	***Length (bytes)***	***Notes***
Label name	1	From 0 to 9

1.5.1.2.18 LOOP/GOTO

Loop/Goto		
Command Code	**'L'**	
Data		
Item	***Length (bytes)***	***Notes***
Label name	1	From 0 to 9
Loop number	2	00 = infinite loop

NOTE: numerical parameters with more than one digit must be padded (on the left) with zeros if they occupy fewer characters than those fixed.

EXAMPLE: running string ("Microgate") on line A, starting from first column, number of columns involved 9, delay 30 hundredths:

ESC - A - **O** - 00 - 09 - 030 - Microgate - ETX - Chk

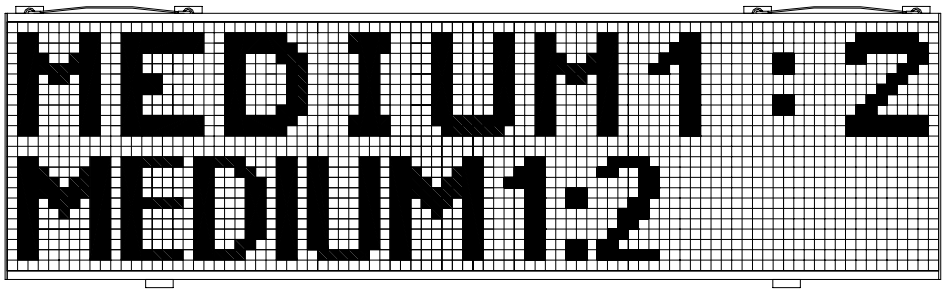
1.5.2 Proportional and Non-Proportional Fonts

In both textual and graphic frames, some fonts can be displayed in non-proportional and proportional mode:

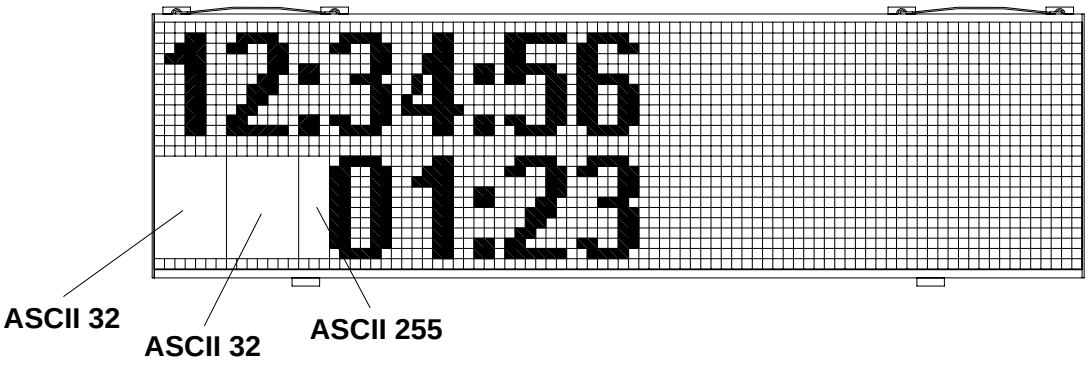
- non-proportional fonts have letters, figures, punctuation and spaces of the same width
- proportional fonts have: -figures of the same width
  -punctuation of the same width (less than that of the figures)
  -letters of variable width
  -space the same width as for figures
  -"short" space the same width as for punctuation and corresponding to the character ASCII 255

Font
Non-Proportional

Font
Proportional



The "brief" space of proportional fonts is very useful when times must be lined up on different lines:



As can be seen in the figure, to line up the time of the bottom line with that of the top line, two "normal" spaces and one "brief" space have been used.

1.5.3 Graphic Frame

The advantage of the graphic frame is that it allows you to display images and active objects as well as text strings.
The position of strings and images is not restricted by lines or columns. The positioning of each object will be completely free, will have as reference the upper-left corner of the µGraph and be given in pixel. The objects themselves will have their highest point on the left as reference (unless a different setting is made).
To use the displayboard in graphic mode, commands must be sent to the identifier (see Address field) '@'. If data is sent to the graphic displayboard with the address ' ', it will be interpreted as µTAB commands and treated as such.
The format of the command frame is different for commands sent to the graphic displayboard, so it is important not to get the identifiers mixed up. At the start of the Data area 2 words are inserted with the coordinates in pixels of the starting point of the command and a byte containing the Binary Operation to be made. If a position parameter is used in the command for µTab, this will not be transferred but will be replaced by the new format.

NOTE: The Binary Operator value will not in fact be used for all commands (for example, for the PAUSE command). However, it must be sent for all graphic displayboard commands.

1.5.3.1 Graphic Frame format and Command table

The format of the frame for the graphic displayboard will therefore become:

Field	Length	Content	Meaning
Start of Frame	1	ESC (0x1B)	Start of command
Address	1	@ (0x40)	Identifier of Graphic Displayboard
Command	1	Variable	Command to be sent to Displayboard
Start of Horizontal Coordinate	2	0-809	max 9 displayboards placed side by side (the first column is the one furthest to the left)
Start of Vertical Coordinate	2	0-383	max 16 placed one on top of the other (the first line is the top one)
Binary Operation	1	0-4	See relative table below
Font	1	0-3	Binary code 0=Default 1=6x8 non proportional (SMALL) 2=12xVariable proportional (MEDIUM) 3=24xVariable proportional (HUGE) 4=24x15 'full size', only numeric (SPECIAL) 5=12x10 non proportional (MEDIUM FIXED) 6=24x13 'full size narrow', only num. (SPECIAL2) Note: by adding 128 (0x80) to the font identifier, alignment to the right is activated. In this case the start position is in relation to the top right corner of the field.
Data	Variable	Variable	Command optional data area
End of Frame	1	ETX (0x03)	End of Command
Checksum	1	Variable	7-bit checksum made on whole frame.

The table below gives the identifiers of the binary operator applied. 'Source' means the bitmap transferred with the command, and 'destination' the area of the displayboard on which it is placed.

Code	Operation performed
0	No Operation: Copies the pixels, writing over the previous status
1	NOT: Inverts the source values and copies them on the destination
2	AND: Only the active pixels on both the source and destination stay switched on
3	OR: Only the switched-off pixels on both the source and destination are switched off
4	XOR: The pixel at destination is inverted if the corresponding pixel at source is switched on.

NOTE: by adding **128 (80 hex)** to the Binary Operator, the command will be regularly processed, but the displayboard will **not be updated**. This allows to send several commands (e.g. several strings placed at different positions), updating the visualization just when the last command is received.

The various commands which can be used in the Command field of the graphic record are given below:

Command	**Command Code**	
• Data display	A	Dec. 65 - Hex 41h
• Font Selection	F	Dec. 70 – Hex 46h
• Image Insertion	I	Dec. 73 – Hex 49h
• Command of digital outputs	i	Dec. 105 – Hex 69h
• set time display	N	Dec. 78 - Hex 4Eh
• Writing of Moving string	O	Dec. 79 - Hex 4Fh
• Resetting a Displayboard Area	Q	Dec. 81 – Hex 51h
• Writing of fixed string	G	Dec. 71 - Hex 4Ah
• set time display	T	Dec. 84 - Hex 54h
• Active object disactivation	t	Dec. 116 – Hex 74h

1.5.3.2 Active Objects

The display commands include “Active Objects”, that is, predefined objects which are managed autonomously by the graphic displayboard. Active objects can be of 4 different types:

- The internal time of the displayboard (Real Time Clock) in various formats: it is provided by the internal quartz clock of the displayboard, which functions also when power is off. It is usually synchronized to the time of day.
- The time of day in various formats: it is provided by the quartz precision clock of the displayboard which functions only when power is on. When switched on, it synchronizes itself with RTC
- Data in various formats
- Rolling texts

On every μGRAPH displayboard up to a maximum of 16 active objects can be displayed, each of which is characterized by an origin (coordinates x and y of the start pixel). If two active objects have the same origin, they can only be displayed one at a time.
The command for displaying active objects requires the use of a special “Graphic Header” (ESC - @ - command – x_start – y_start – binary operator – font).

1.5.3.3 Syntax of Graphic Frame commands

1.5.3.3.1 DATA DISPLAY

Data display – Active Object		
Command Code	**'A'**	
Data		
Item	***Length (bytes)***	***Notes***
Display format	1	1(binary)=DD/MM/YY 2(binary)=DD MM YY

1.5.3.3.2 IMAGE INSERTION

This command is used to display Bitmap images on the graphic displayboard. Each bit of data placed at '1' corresponds to a switched-on pixel in the image. The image is scanned vertically, with one column at a time being sent, aligned to the byte. No type of compression is used.

Image Insertion		
Command Code	**'I'**	
Data		
Item	***Length (bytes)***	***Notes***
Dimension X	2	Horizontal Dimension of image
Dimension Y	2	Vertical Dimension of image
Image Data	?	Each vertical line is scanned and sent, aligned to the byte. The Least Significative bit refers to the highest pixel.

1.5.3.3.3 COMMAND OF DIGITAL OUTPUTS

Command of digital outputs		
Command Code	**'i'**	
Data		
Item	***Length (bytes)***	***Notes***
Input/output control	1	Use bits from 0 to 3 to select the mode of digital I/Os from 0 to 3 (0 = output, 1 = input).
Value of digital outputs	1	Use bits from 0 to 4 to set the value on digital outputs from 0 to 4 (0 = OV, 1 = 5V).

1.5.3.3.4 INTERNAL CLOCK DISPLAY (RTC)

Display of internal clock (RTC) – Active Object		
Command Code	**'N'**	
Data		
Item	***Length (bytes)***	***Notes***
Display Format	1	1 (binary) = HH:MM:SS 2 (binary) = MM:SS 3 (binary) = HH:MM (24h) 4 (binary) = HH:MM (12h)
Delay	4	Time ahead or behind the setting of the internal clock, in thousandths of a second

1.5.3.3.5 WRITING OF MOVING STRING

Writing of moving string– *Active object*		
Command Code	**'O'**	
Data		
Item	***Length (bytes)***	***Notes***
Width of text	2	Width of text in Pixels (binary word)
Delay in display	2	Delay in running (Frame to Frame) in hundredths (binary word)
Width of display	1	Width of display in Pixels (binary)
Text	?	From 1 to 255 characters + 'null terminator'

1.5.3.3.6 WRITING OF FIXED STRING

Writing of fixed string		
Command Code	**'S'**	
Data		
Item	***Length (bytes)***	***Notes***
String	<=81	Characters to be displayed (with null terminator)

1.5.3.3.7 SET TIME DISPLAY

Set time display – *Active object*		
Command Code	**'T'**	
Data		
Item	***Length (bytes)***	***Notes***
Display format	1	1(binary)=HH:MM:SS 2(binary)=MM:SS 3(binary)=HH:MM (24h) 4(binary)=HH:MM (12h)
Lag	4	long integer (31 bit + sign) with time in front or behind indicated in relation to the internal clock (quartz precision), in thousandths.

1.5.3.3.8 ACTIVE OBJECT DISACTIVATION

Active object disactivation		
Command Code	**'t'**	
Data		
Item	***Length (bytes)***	***Notes***
None	-	None given: the object is identified by its position

ATTENTION: when an active object is written over one in the same position, the original object is automatically disactivated.

1.5.4 Resetting a Displayboard Area

This command allows you to cancel just one area of the graphic displayboard.
The graphic displayboards form a single surface composed of the sum of their areas. This command cancels one part of the surface, irrespective of which displayboards are involved in the operation.

Resetting a Displayboard Area		
Command Code	**'Q'**	
Data		
Item	***Length (bytes)***	***Notes***
Dimension X		Horizontal Dimension of the area to be canceled
Dimension Y		Vertical Dimension of the area to be canceled

1.5.5 Font Selection

If you intend to use the µTab graphic displayboard in compatible mode, you must set the character to be displayed. This command sets the character for all the displayboards and prepares them to receive commands with an identifier other than '@'. When you switch on, the default font is 6x8.

Font Selection		
Command Code	**'F'**	
Data		
Item	***Length (bytes)***	***Notes***
None (use the font field of the graphic header to set the font)		Fonts: • 48(30h): Font 6x8 (Equal to the one on the µTAB, 3 lines per displayboard., 15 characters per line) • 49(31h): Font 12xVariable (2 rows per displayboard, 11 characters per line) • 50(32h): Font 24xVariable (1 row per displayboard, 6 characters per line) • 51(33h): Font 24x15 (only numbers, 1 line per displayboard, 5 numbers and two punctuation marks per line) • 52(34h): Font 12x10 (2 lines per displayboard, 11 characters per line) • 53(35h): Font 24x13 (only numbers, 1 line per displayboard, 6 numbers and two punctuation marks per line)

1.5.6 Example

1.5.6.1 Fixed string writing with graphic frame

Data:

- String: MICROGATE
- Start of Horizontal Coordinate: 10
- Start of Vertical Coordinate: 6
- Font: MEDIUM

String to be sent:

Field	Byte	Conten. (hex)	Meaning
Start of Frame	0	**0x1B**	Start of command
Address	1	**0x40**	IDENTIFIER OF GRAPHIC DISPLAYBOARD
Command	2	**0x53**	FIXED STRING COMMAND
Start of Horizontal Coordinate (First byte)	3	**0x0A**	10 pixels -> in hex **0x0A**
Start of Horizontal Coordinate (Second byte)	4	**0x00**	
Start of Vertical Coordinate (First byte)	5	**0x06**	6 pixels -> in hex **0x06**
Start of Vertical Coordinate (Second byte)	6	**0x00**	
Binary operation	7	**0x00**	No operation
Font	8	**0x02**	Font MEDIUM
Data	9	**0x4D**	Character **M**
Data	10	**0x49**	Carattere **I**
Data	11	**0x43**	Character **C**
Data	12	**0x52**	Character **R**
Data	13	**0x4F**	Character **O**
Data	14	**0x47**	Character **G**
Data	15	**0x41**	Character **A**
Data	16	**0x54**	Character **T**
Data	17	**0x45**	Character **E**
End of Frame	18	**0x03**	End of Command
Checksum	19	**0x5E**	7-bit checksum made on whole frame: 0x1B+0x40+0x53+0x0A+0x06+0x02+0x4D+ 0x49+0x43+0x52+0x4F+0x47+0x41+0x54+ 0x45+0x03 = 0x35E 0x35E AND 0x7F = **0x5E**

Microgate S.r.l.
Via Kravogl, 8
I-39100 BOLZANO - ITALY
http://www.microgate.it