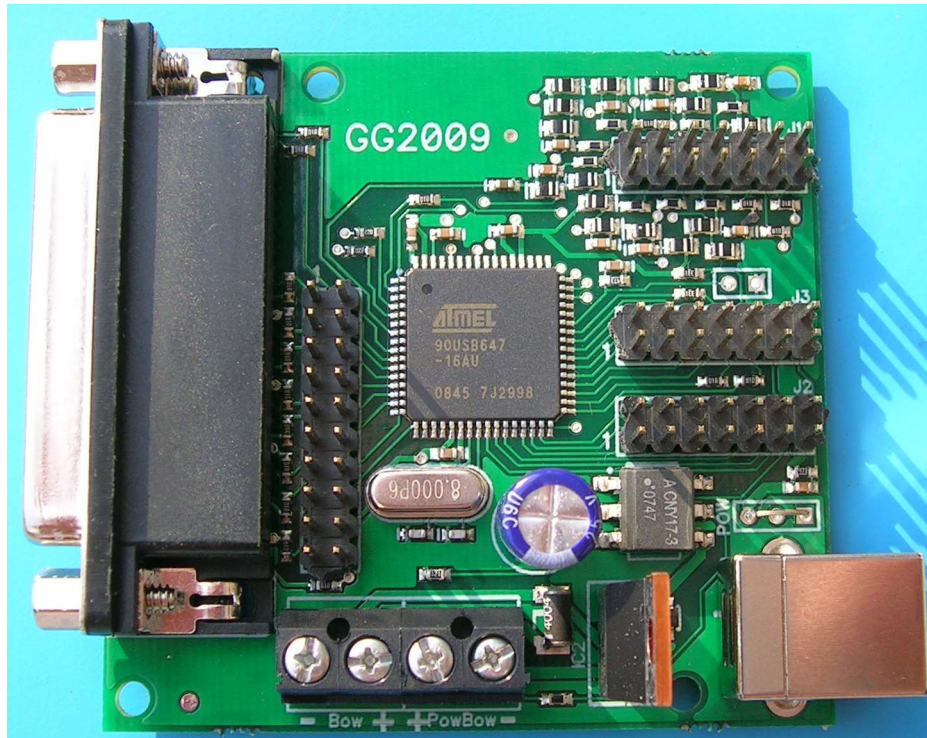


GGC V3 documentation

V3.00 - 16/10/2020

The GGC V3 adapter is an intelligent USB converter for driving a step motor interface. The GGC drives most step/dir compatible motor interfaces, such as the MM2001, and the MDLCNC bipolar motor controller.



1. Functionalities

The functionalities of the GGC V3 adapter are:

- Generation of a unique GMFC license key, independent of the PC and disk formatting.
- Works under any Windows version that supports USB from Win98 to Windows 10.
- 10, 20, or 50 KHz internal timer for implementing smooth accelerations. The internal timer supports motor interfaces that don't have a timer.
- Compatible with most existing step/dir unipolar and bipolar motor interfaces. Fully configurable Dir/Step outputs. Motor Enable on parallel port pin 1 or 17, with High/Low active level.
- Internal wire heat circuits.
- Optional LCD display. The GGC V3 support LCDs with 1 or 2 lines of 16 characters and the RepRap LCD with 4 lines of 20 characters.
- Automated recognition of the GGC parameters by GMFC (from version 3.89.14).
- Full compatibility with the former MMUSB interface.
- Programming of the MM2001 PIC firmware. The **V5.1** or **MM2001_GGC** versions of the MM2001 firmware are mandatory to work with the GGC V3 interface.

- Configuration through the **ggc_utility V3** software on the PC. The utility also allows the programming of the MM2001 PIC firmware.

Several internal parameters can be set by using **ggc_utility**: the configuration of the parallel port pinout, the timings of the motor pulse commands and the frequency of the internal timer.

The GGC adapter possesses two functioning modes: MMUSB and GGC.

2. MMUSB Mode

In the MMUSB mode, the GGC adapter is legacy compatible with the former MMUSB adapter and supports only the MM2001 motor interface. The LCD display is optional. The wire heat is managed by the MM2001 interface. No switch or LED is necessary for the GGC to work in this mode. When using a MM2001 with the GGC in MMUSB mode, the MM2001 must be programmed with the **V5.1** firmware.

In the MMUSB mode, the GGC internal parameters can't be changed and are set to the following values:

- Timer frequency: 100 microseconds (10 KHz)
- Motor step duration: 50 microseconds
- Time for data before step: 14 microseconds

If a LCD display is used, it successively displays after reset:

"GGC V3.00", where 3.00 is the GGC firmware version.

"Dir 14,Stp 50,T0", this screen gives the value of internal parameters. **Dir** is the time for the data to be set before the motor pulse command. **Stp** is the duration of the motor pulse command. **T** is the timer value: 0 means 10Khz.

"MMUSB Mode", this screen indicates that the GGC is operating in MMUSB mode.

"WAITING USB", this screen is shown until the GGC is recognized by Windows and ready to be used.

"MMUSB H50% MAN M", this screen is displayed during normal functioning. **H50%** shows the current heat (here, 50%), **MAN** (or **PC**) shows the position of the PC/MAN switch of MM2001 interface. **M** (if displayed) shows that the motors are powered on by GMFC.

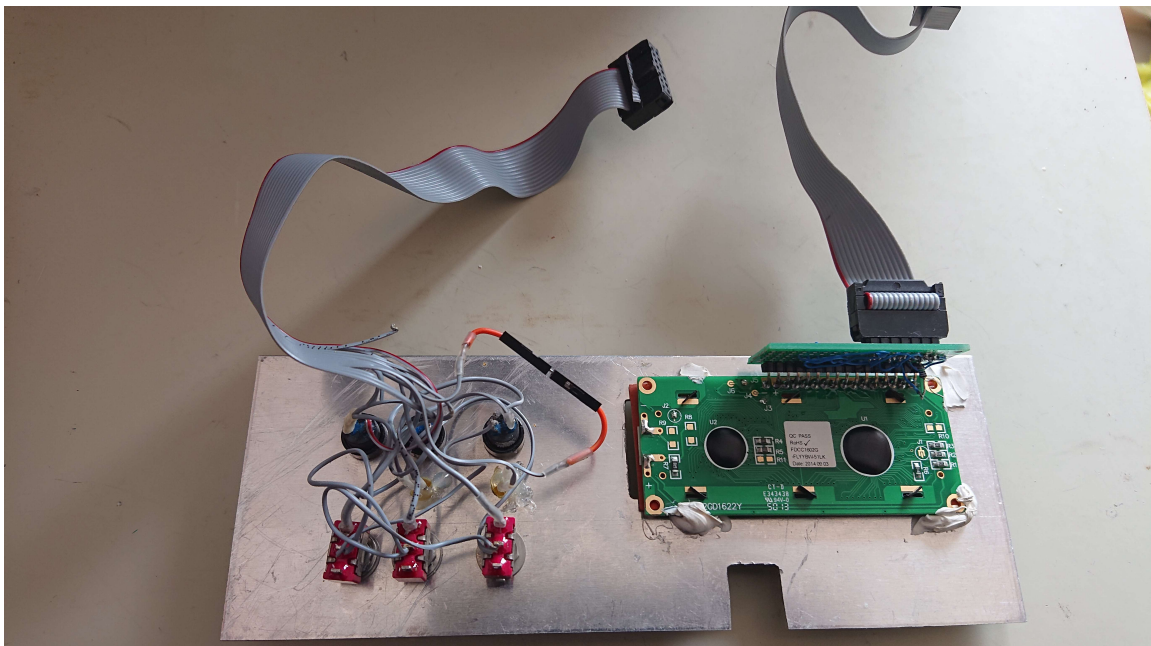
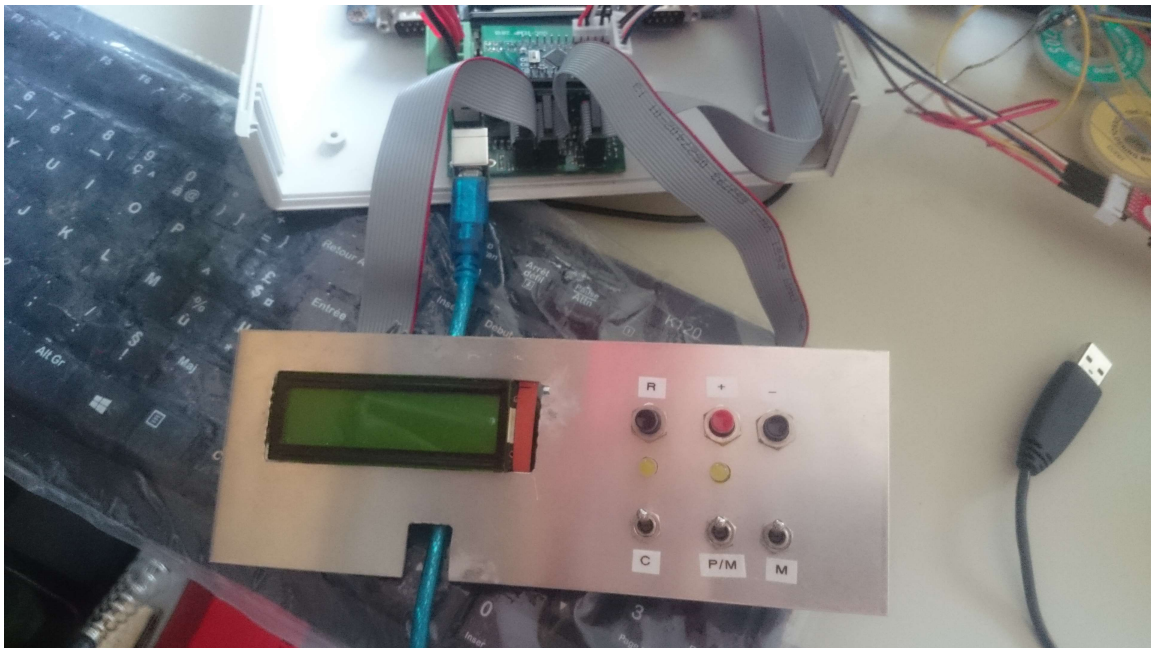
3. GGC Mode

In GGC mode, two different command panels are supported: the BP button and the RepRap panel.

3.1. The BP Button panel

The BP panel must be made by the user. The BP panel relies on three push buttons **BP+**, **BP-**, **Reset**, the led **LEDHeat** and 3 switches **HeatOn**, **PC/MAN** and **MotorOn**.

The **LEDHeat** is mandatory for the heat circuit to work. Buttons and switches are optional. When used, buttons and switches are connected through the J3 connector. The LCD is connected to J2. Both 1 line et 2 line 16 character LCDs can be used.



3.1.1 Buttons and switches

BP+, **BP-** are used to set the heat value when the heat value is controlled by the GGC. A short press on **BP+** increases the heat; a short press on **BP-** decreases the heat.

Pressing on **BP-** and **BP+** simultaneously enters a menu for setting the maximum possible heat value. The maximum heat prevents breaking the wire if it does not support a heat value of 100%. Pressing on **BP-** and **BP+** again will exit this menu. Note, that with versions of GMFC above 4.0, the maximum heat value is specified in the table configuration dialog of GMFC and overrides the maximum value in the GGC.

The optional **PC/GGC** switch controls the source heat value that is set either by the PC or the GGC interface. Even in the GGC position, the heat is only activated when GMFC sends a heat command.

The optional **HeatON** switch enables the heat. This switch is useful only for providing additional security in the addition of the GMFC heat command.

The optional **MotorOn** switch enables the motors. This switch is useful only for providing additional security. It can be used as an emergency stop switch.

The optional **Reset** button is used to reset the GGC interface.

3.1.2 LEDs

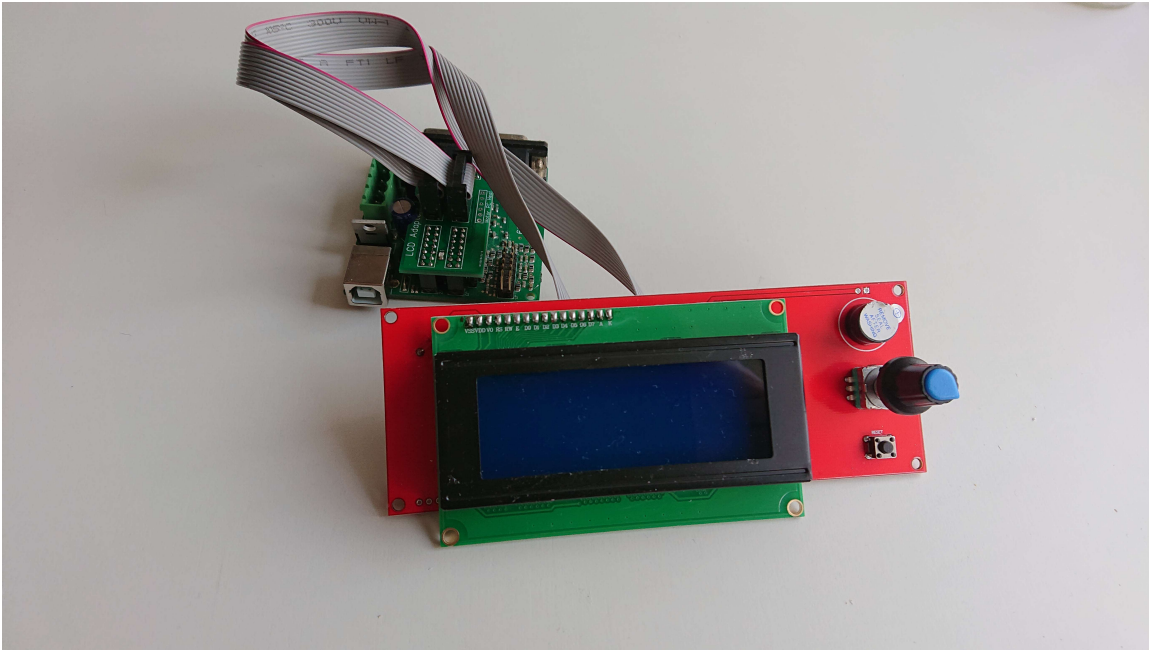
LEDHeat: this LED indicates that the heating circuit is functioning. Because control of the wire heat goes through the LED, it MUST be connected for the heat circuit to work.

LEDMotor: this optional LED indicates that there is power to the motors.

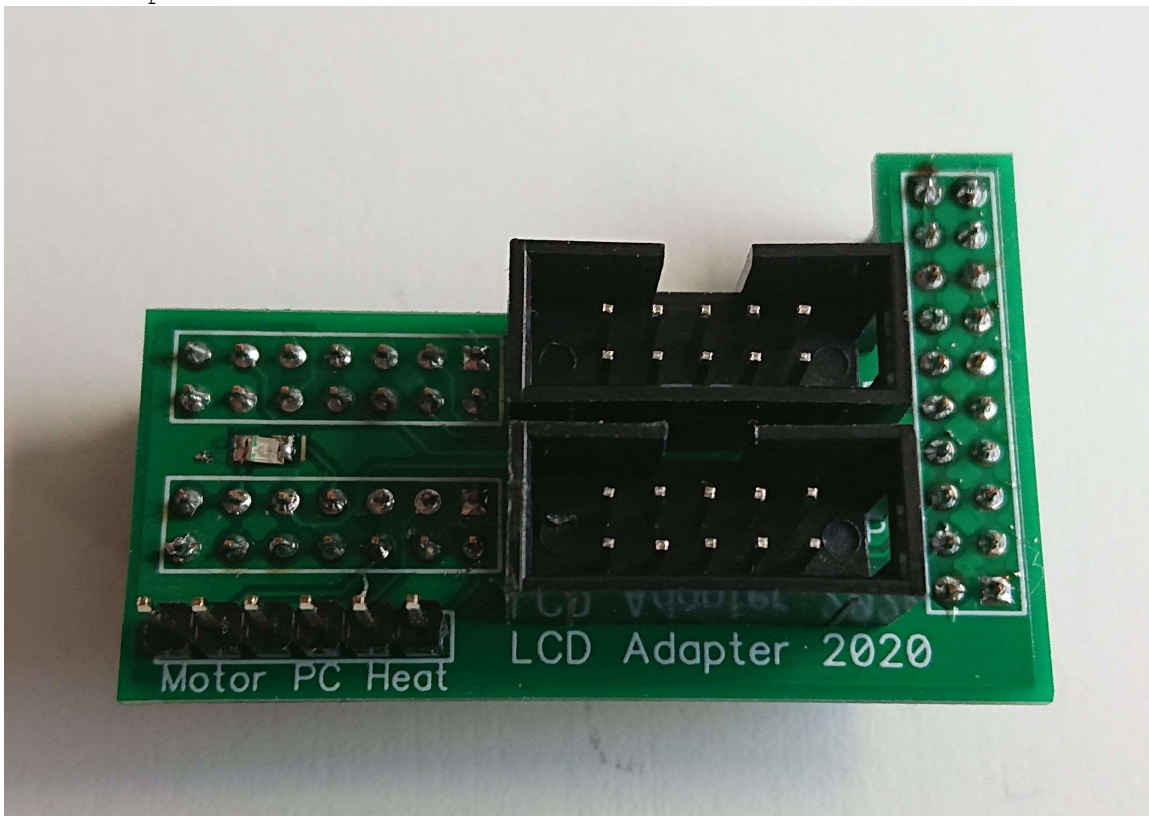
LEDon: this optional LED indicates that there is power to the GGC.

3.2. The RepRap panel

The RepRap panel contains a 4 line LCD, a rotary selector with a press button, and a reset button.



The RepRap panel must be connected to the GGC through the GGC LCD adapter which has to be bought in addition to the GGC. The adapter provides the **LEDHeat**, and is ready to use. Soldering is only required for the optional switches.



3.2.1 Buttons and switches

The rotary selector is used to set the heat value when the heat value is controlled by the GGC.

Pressing on the selector enters a menu for setting the maximum possible heat value. The maximum heat prevents breaking the wire if it does not support a heat value of 100%. Pressing on the selector again will exit this menu. Note, that with versions of GMFC above 4.0, the maximum heat value is specified in the table configuration dialog of GMFC and overrides the maximum value in the GGC.

The optional **PC/GGC** switch controls the source heat value that is set either by the PC or the GGC interface. Even in the GGC position, the heat is only activated when GMFC sends a heat command.

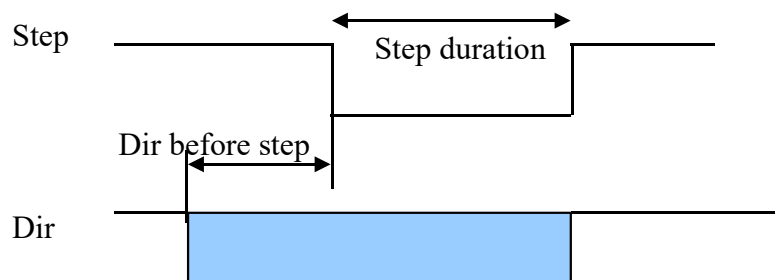
The optional **HeatON** switch enables the heat. This switch is useful only for providing additional security in the addition of the GMFC heat command.

The optional **MotorOn** switch enables the motors. This switch is useful only for providing additional security. It can be used as an emergency stop switch.

3.3. Motor interface parameters

The control of the motor interface can be adjusted by several parameters:

- Parallel port pinout of the step/dir signals.
- **Timer frequency:** 50KHz (20microseconds), 20KHz (50 microseconds) or 10KHz (100 microseconds).
- **Step pulse duration:** the step pulse is active low. Its value can be chosen with 2 microseconds increments.
- **Direction available before the step pulse:** value can be chosen with 2 microseconds increments.
- Motor enable signal on pin 17 (MM2001 compatible) or pin 1 (Chinese motor interfaces).
- Motor enable active low or high.



When using a MM2001 with the GGC in GGC mode, you must program the MM2001 with the **MM2001_GGC** firmware which is compatible with a 50 Khz

timer. Using the **MM2001_GGC** firmware, the MM2001 internal timer and heat functionalities are disabled.

3.4. LCD screens

3.4.1 Initialisation

If the LCD display is used, it successively displays after reset:

"GGC V3.00", where 3.00 is the GGC firmware version.

"Dir 2,Stp 2,T1", this message displays the value of internal parameters. **Dir** is the time for the data to be set before the motor pluse command. **Stp** is the duration of the motor pulse command. **T** is the timer value: 0 means 10KHz, 1 means 20KHz, 2 means 50KHz.

"HMAX SECURE:99%", this screen indicates the maximum allowable heat value to suit the type of cutting wire being used.

"WAITING USB", this message is displayed until the GGC is recognized by Windows on the PC and ready to be used.

3.4.2 Main menu

If using a one or two line LCD display:

"H:050% PC M* H*", this message is displayed during normal functioning. **H050%** shows the current heat value (here, 50%), **PC/GGC** tells that the heat value is given by the GGC or by the PC (GMFC). In GCC heat mode, the value can be adjusted by the **BP+** and **BP-** buttons.

M* shows that the motors are powered on. This means that GMFC is running a heat command. **M** (without *) is shown when motors are off.

H* shows that the heat is on. The LEDHeat is also on. **H** (without *) is shown when the heat is off.

If using a four line RepRap LCD display:

"Heat: 50% Max: 100%"

"PC Motor* Heat*"

Heat: 80% shows the current heat value (here 50%). **Max: 100%** shows the maximum heat value (here 100%),

PC/GGC tells that the heat value is given by the GGC or by the PC (GMFC). In GCC heat mode, the value can be adjusted by the rotary button.

Mot* shows that the motors are powered on. This means that GMFC is running a heat command. **Mot** (without *) is shown when motors are off.

Heat* shows that the heat is on. The LEDHeat is also on. **Heat** (without *) is shown when the heat is off.

3.4.2 Maximum Heat Menu

If using a one or two line LCD display:

"Heat Max: 100%"

This is the value of the maximum heat value. It is adjusted by the **BP+** and **BP-** buttons. Pressing both buttons simultaneously will return to the main menu.

If using a four line RepRap LCD display:

"Max Heat Setting"

"Max: 100%"

This is the value of the maximum heat value. It is adjusted by the rotary buttons. Pressing the button will return to the main menu.

4. GGC Connections

The GGC connects to the motor interface via the parallel port, and to the PC via a 'B' type USB cable as used for printers.

On board connectors -

J1 - 14 pin connector - Home and Limit switches

J2 - 14 pin connector - LCD display

J3 - 14 pin connector - LED, buttons and switches

J4 - 20 pin connector - Optional temperature sensor daughter board.

The pinout of J1, J2, J3 from top is:

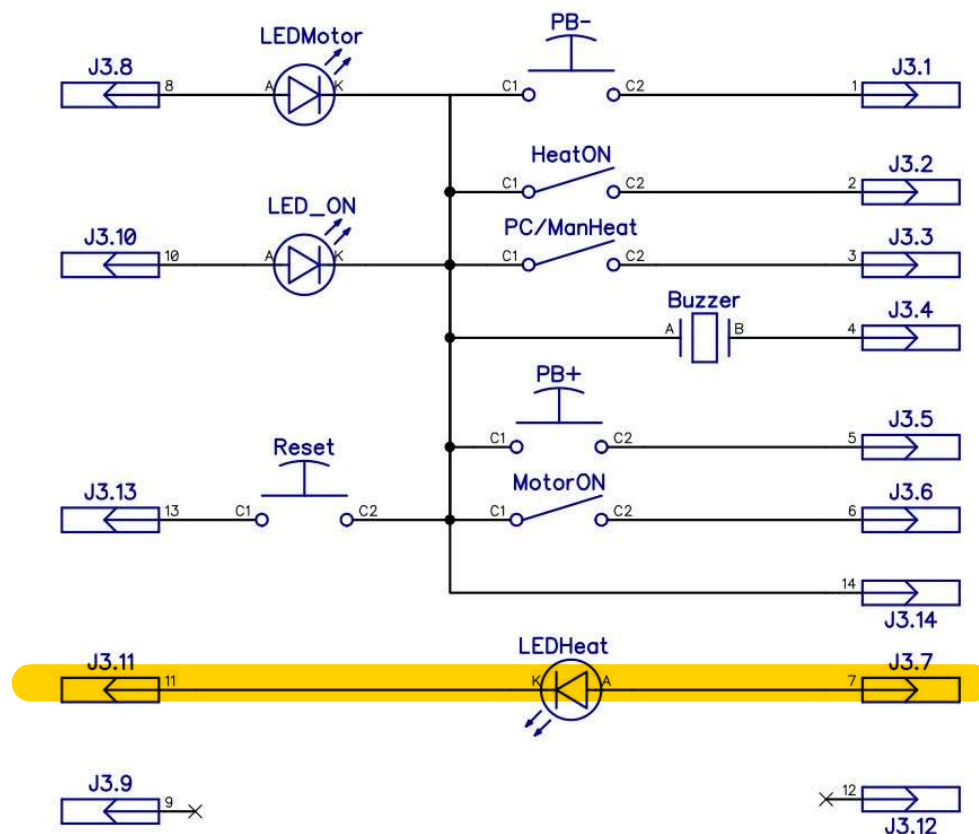
2	4	6	8	10	12	14
1	3	5	7	9	11	13

Pin 1 is marked on the board.

Powering the heat wire

The wire power supply is connected to the GGC via the **PowBow** connector. The maximum voltage is 50V DC. **Beware of polarities; wrong connection will damage the GGC.** The bow must be connected to the **Bow** connector.

J3 - LEDs and switches wiring



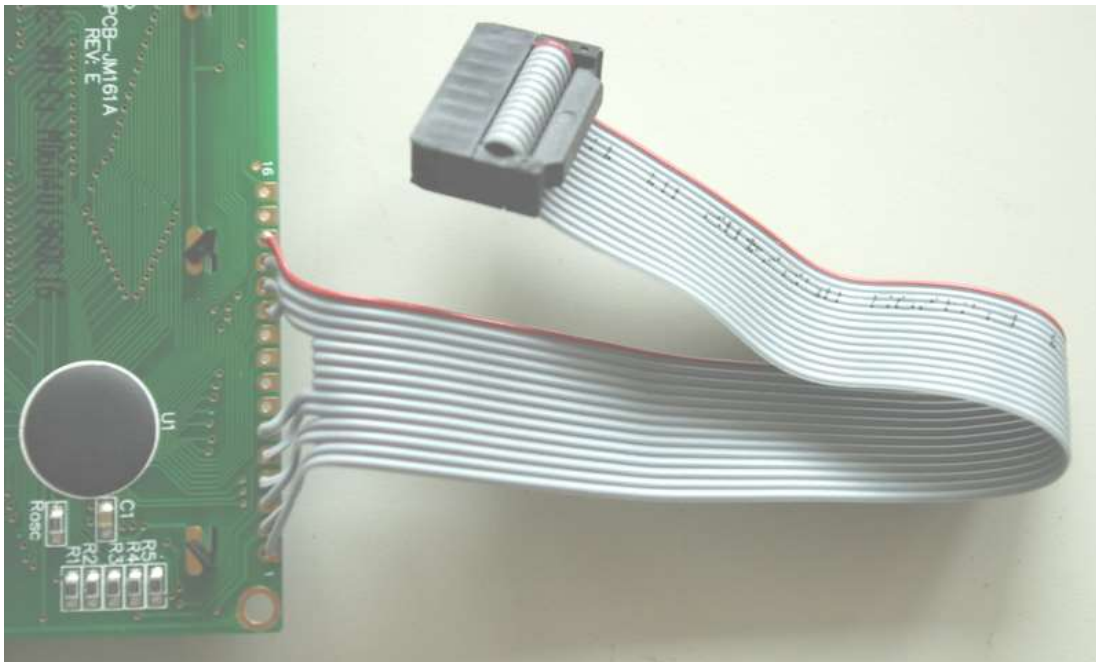
Buttons and switches are connected to GND through J3.14. **Do not connect J3.14 to power supply ground.**

J2 - LCD wiring

J2 is used to connect the one or two lines 16 character LCD Display Panel via a short length of ribbon cable wired as detailed below. All 16 character LCDs should be compatible.

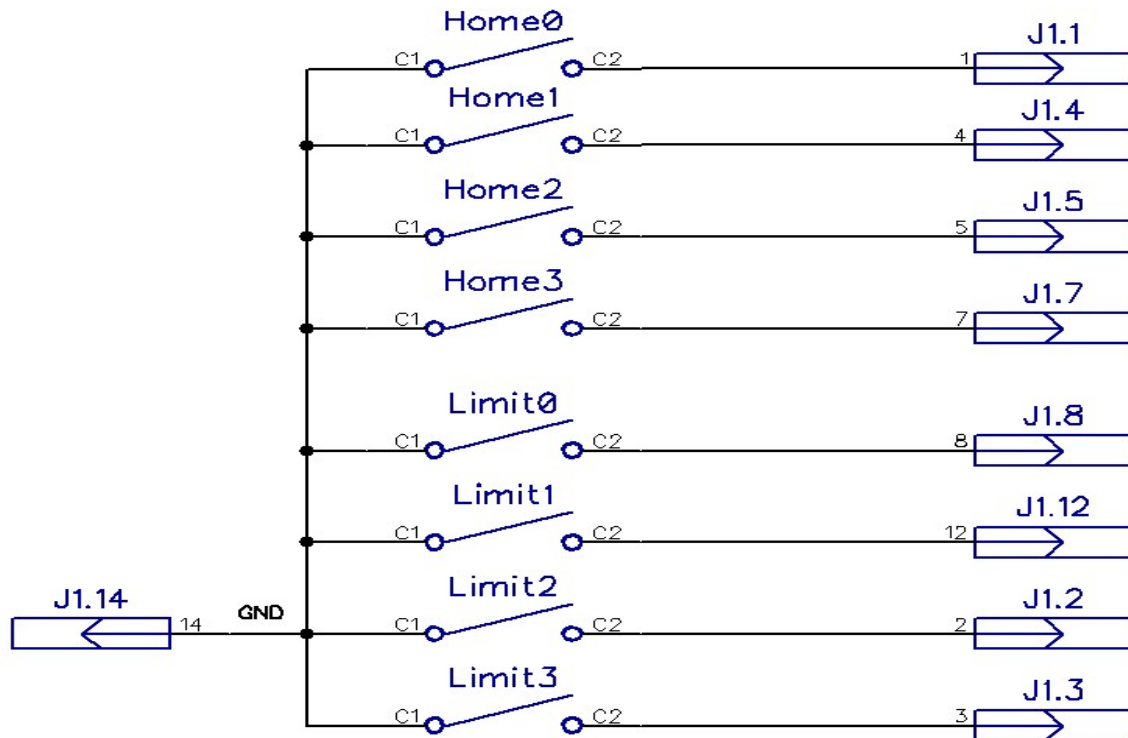
Connector Pin	Function	LCD Pin
J2.1	PD7	LCD14
J2.2	PD6	LCD13
J2.3	PD5	LCD12
J2.4	PD4	LCD11
J2.5	Non connected	
J2.6	Non connected	
J2.7	Non connected	
J2.8	Non connected	
J2.9	E	LCD6
J2.10	R/W	LCD5
J2.11	RS	LCD4
J2.12	GND	LCD3
J2.13	Vcc	LCD2
J2.14	GND	LCD1

Here is a picture of the J2 connecting cable:



Note that the red wire is connected to pin J2.1

J1 - Home & Limit Switch Wiring



The Home switches and Limit switches are used to prevent accidental over-run beyond the physical limits of the X and Y axes. The Home and Limit switch options are only functioning when running the GMFC EXPert software. They do not function under any other version of GMFC. Home and Limit Switches must be of type "Normally Open".

Important: Limit switches are internally connected to the input signals of the parallel port (10 timer, 11 Heat acquisition, 12 Man/Auto heat). If you want to use Limit switches, the GGC must be used in GGC mode, and you must be sure that these input signals are left open or not used by the motor interface. This can be done by cutting the parallel cable. If you are using an MM2001 motor interface, the **MM2001_GGC** firmware is mandatory.

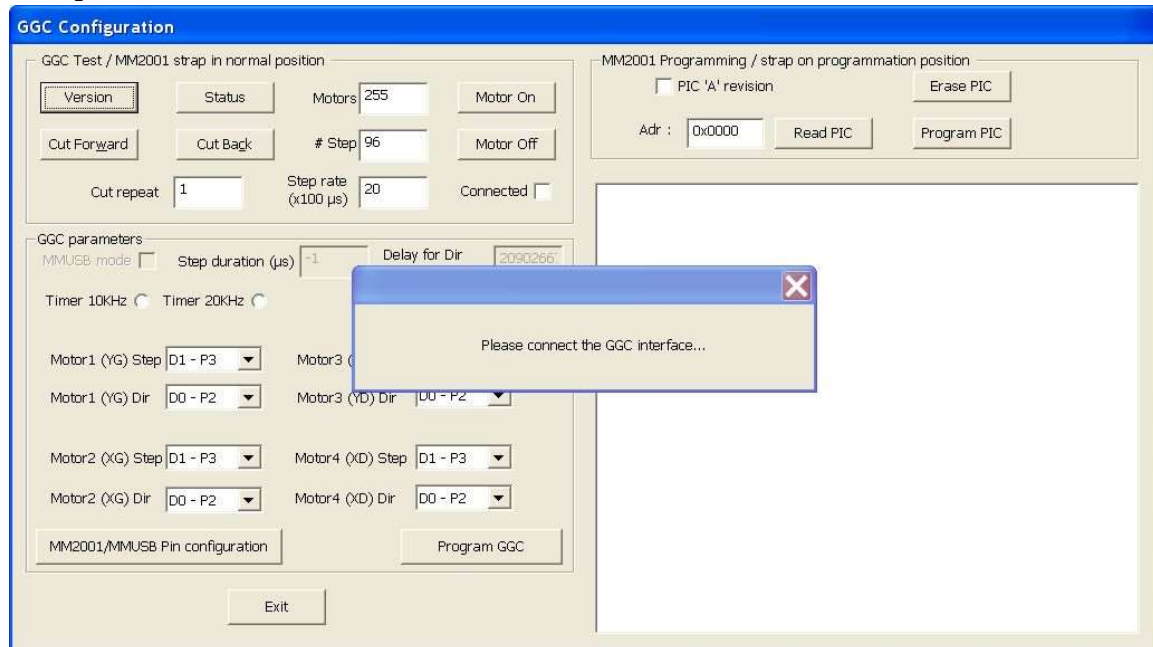
5. Configuration utility, ggc_utilityV3

The ggc_utilityV3 program allows the PC configuration of the GGC interface, testing and moving motors without GMFC. It also permits to program the PIC firmware of the MM2001.

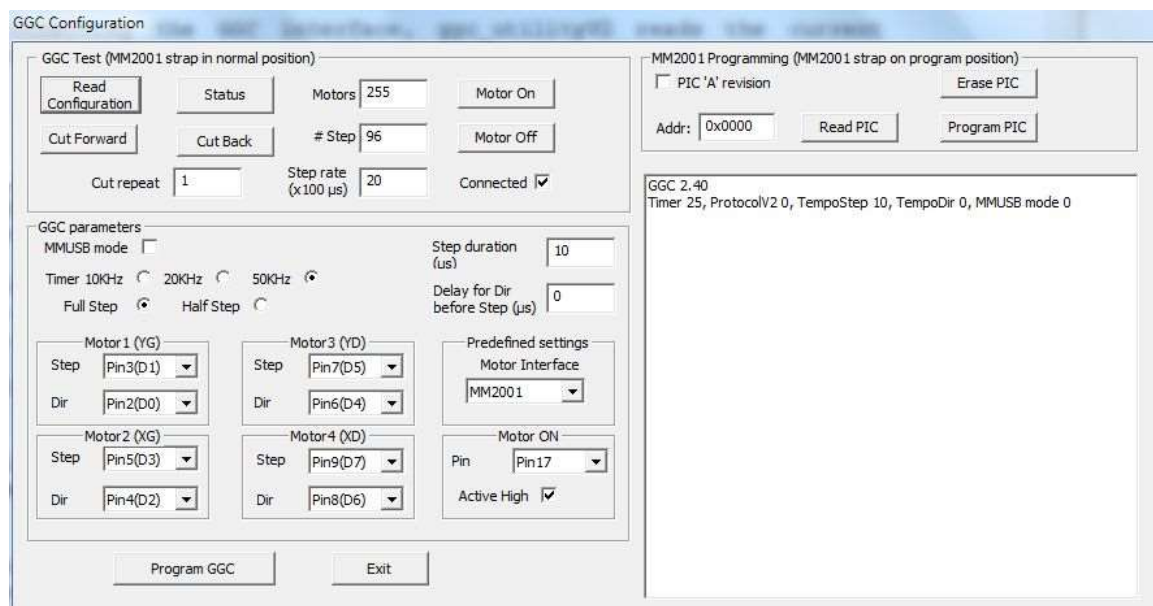
The program ggc_utilityV3 is available from <https://gmfcsoft.fr/shop/en/usb-support-for-gmfc-with-the-ggc-adapter/>.

5.1. Internal parameter configuration

Download the file **ggcV3.zip**, then uncompress it; run the executable file **ggcV3.exe**. If the GGC adaptor is not yet connected, the ggc_utility will ask you to connect it.



Press the "Read Configuration" button so that the ggc_utilityV3 displays the configuration of the GGC adaptor.



The GGC is shipped configured in MMUSB mode. If you want to switch to the GGC mode, you must uncheck the **'MMUSB mode'** box. Then you can modify the appropriate internal timer frequency and motor timing parameters. Additionally, to avoid rewiring motor cables, you can simply reconfigure the motor outputs. Finally, you can set the pinout for the "Motor On" signal (1 or 17) and also the active state ("Low" or "High").

The ggc_utilityV2 provides a list of predefined settings for well-known interfaces. Just select the interface in the list to set the predefined parameters. **Caution: this drop box is only for settings parameters, it does not display the type of the interface currently connected.**

To program the settings into the GGC V2, you have to press the **'Program GGC'** button. Wait until "done" is printed in the message window.

Note for the Gecko G540 controller - Choose setting for the MDLCNC interface. Then set the "step duration" to 2 and "Delay for Dir before Step" to 0. Set the Timer to 50Khz. You have to disable the charge pump on the Gecko since the GGC does not provide the required signal. There is no Motor ON signal for the Gecko, by disabling the charge pump, motors are always ON.

5.2. Moving motors with the ggc_utilityV3

To move the motors, you must first press the **'Motor On'** button to power the motors. Then, you have to set several parameters:

First, you must specify which motors to move using the field **'Motors'** and the following table:

X Left	X Right	Y left	Y Right
12	192	3	48
204		51	

Sum up values to move motors simultaneously. The value 255 will move all motors.

#Step: This is the number of steps performed by a move command. The maximum value is 128.

When starting to move the motors, a good value would be to use the number of steps required to rotate the shaft a half or one full revolution.

Step rate: This sets the interval between motor steps. The value is given in hundreds of microseconds. The default value of 20 (i.e., 2 milliseconds) should work with all motors. You should decrease the value for testing a higher speed. Still, precise speed calibration should be done within GMFC.

Cut Repeat: This is the number of times that the motor move will be executed when pressing the **Cut** button.

For example: if **#Step** equals the number of steps for rotating 1 revolution, and **Cut Repeat** equals **10**, the motors will turn 10 revolutions each time **Cut** is pressed.

Pressing the '**Cut Forward**' button will move the motors forward and up, if the motors are correctly connected (X to the end of the table, and Y up).

Pressing the '**Cut Back**' button will move the motors backward and down.

The '**Motors OFF**' button switches the motors off. When the motors are off, pressing any of the **Cut** buttons will have no effect.

5.3. Programming the MM2001 PIC firmware

This section is only for the users of the MM2001 interface. It does not apply to other motor interfaces.

You must **carefully** follow instructions related to the programming jumper and the RESET button of the MM2001 interface.

If you have a PIC type 16F874A or 16F877A, check the box '**PIC "A" Revision**'. If you forget to do so, the PIC may be blocked and further programming disabled.

Press the '**Write PIC**' button and follow the instructions carefully... Use the file **MM2001-V51.HEX** or later for correct operation of the MM2001 with the GGC interface. Programming the PIC with older firmware is possible, but the MM2001 will not run properly because the GGC interface requires a faster response time.

When programming a PIC type "A", it is necessary to erase it first using the '**Erase PIC**' button. If the programming fails, it is necessary to perform a further erase before each attempt to program the device.

For standard (none "A") PIC, erasing is not necessary.