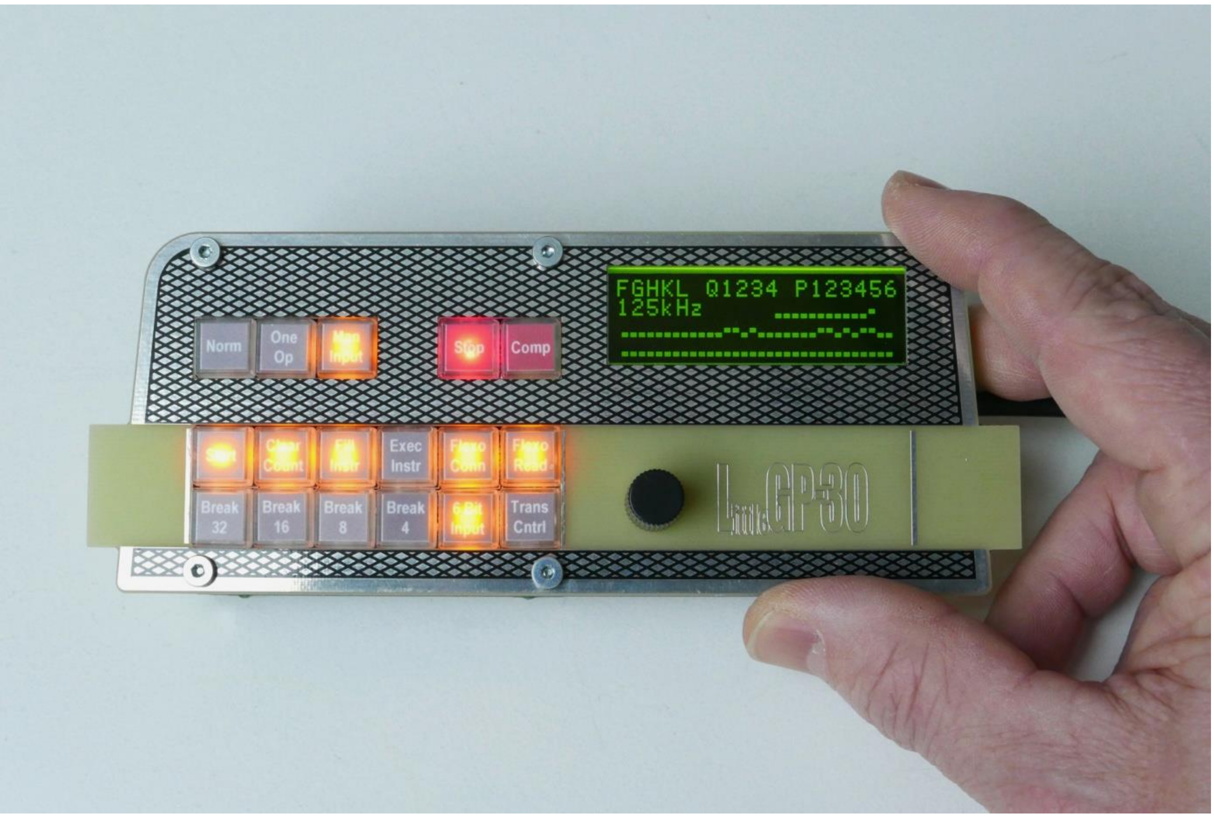
# LittleGP-30

Jürgen Müller, juergen@e-basteln.de

Document rev. 3.1
22nd October 2017

This document describes the updated “wide panel” version of the LittleGP-30,
board version 8/2017 or later, with on-board USB and RS-232 interfaces.

If you have an older version, please see www.e-basteln.de/lgp30 for documentation.

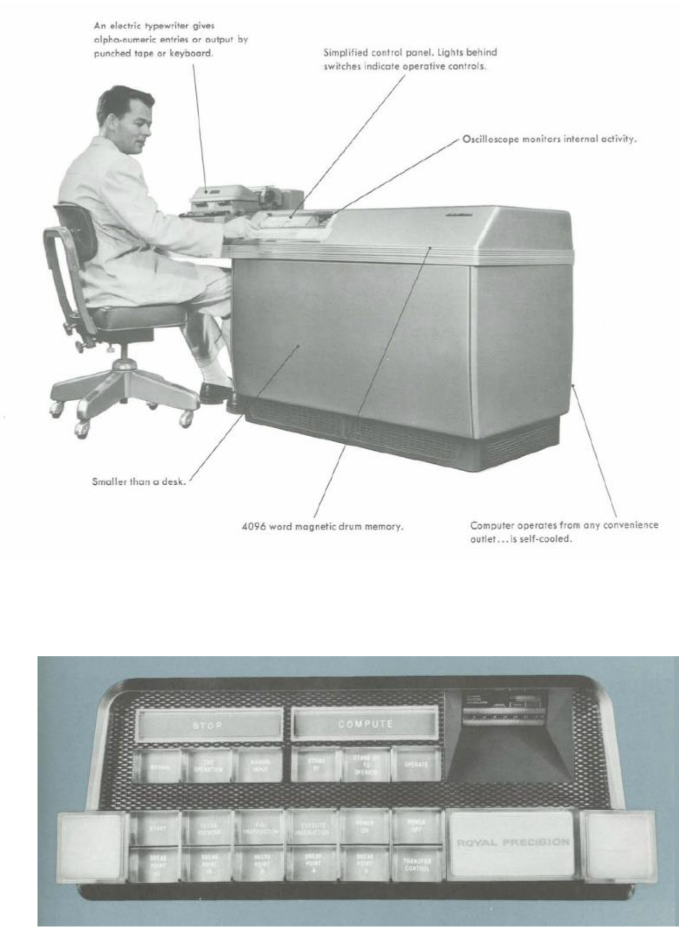


# Contents

# 1 What’s this?

The LGP-30 was a commercial computer, released in 1956. Due to its simple design and relatively low cost, it may be seen as the first “personal computer” – to be used by a single user as their “desk computer”. (It could sit by your desk, and happened to be the size of a desk.) Designed in the age of vacuum tubes, it needed only 113 tubes in total, of which only 24 were used in the CPU itself! This simplicity was achieved by a bit-serial CPU design, which was tightly integrated with the magnetic drum storage unit. The magnetic drum contained not only the main memory, but also the CPU’s three 32-bit registers, and several tracks with timing signals to control the instruction decoding and execution.

This document describes a replica that is true to the bit-serial implementation and its timing, but uses modern components. The CPU and the magnetic drum storage are recreated in an FPGA – by implementing the complete logic equations published by the LGP-30’s inventor (in a scientific paper and in the computer’s service manual). The magnetic drum, while implemented in on-board memory inside the FPGA, is made tangible via an optional video display of its contents.


An electric typewriter gives alpha-numeric entries or output by punched tape or keyboard.
Simplified control panel. Lights behind switches indicate operative controls.
Oscilloscope monitors internal activity.
Smaller than a desk.
4096 word magnetic drum memory.
Computer operates from any convenience outlet . . . is self-cooled.
STOP
COMPUTE
ROYAL PRECISION

*LGP-30 overview and front panel; from a contemporary brochure*

This way, one can play with all the quirks of the LGP-30, including the timing behavior of programs, which depends critically on the position of instructions and data on the magnetic drum. For those who want to really dive into the details of the bit-serial design, the clock rate can be slowed down (all the way to single step, bit-by-bit clocking), and the contents of the CPU’s tube-based flip-flops can be inspected on an LCD (all 15 bits of them!).

## 1.1 License terms and credits

This document, the PCB layout and the FPGA code are Copyright © 2016, 2017 Jürgen Müller, juergen@e-basteln.de. Use for commercial purposes requires my explicit permission in writing. Use for non-commercial purposes is granted free of charge, but I ask that you disclose the source and include this copyright notice. I would appreciate if you let me know when you re-use this stuff.

Use of the following third-party components is gratefully acknowledged. These may bring their own license restrictions; please check before you publish this code or use it commercially:

- USB Serial Data Transfer Core by Joris van Rantwijk, http://jorisvr.nl/usb/
- USB 1.1 PHY Layer by Rudolf Usselmann, rudi@asics.ws, http://www.opencores.org/cores/usb_phy/
- USB 1.1 PHY Layer (VHDL conversion) by Martin Neumann, martin@neumanns-mail.de, http://opencores.org/project,usb11_phy_translation
- Minimal DVI Encoder framework by Mike Field, http://hamsterworks.co.nz/mediawiki/index.php?title=Minimal_DVI-D
- TMDS Encoder by Jean P. Nicolle, http://www.fpga4fun.com/HDMI.html
- Serial UART by Ken Chapman, Xilinx Ltd, http://ohm.bu.edu/~dean/Xilinx/KCPSM6_Release7_30Sept13/UART_and_PicoTerm/UART6_User_Guide_and_Reference_Designs_29March13.pdf
- LCD font by Benedikt K., http://www.mikrocontroller.net/topic/54860

*Two ways to build your LittleGP-30: Enclosed (Hammond 515-0930 enclosure shown here) or bare boards.*

# 2 Building the LittleGP-30

Soldering the LittleGP-30 circuit board is quite straightforward. I recommend following the sequence in this chapter. If you are in a rush, please be sure to observe the following hints:

- Components are installed on both, the top and bottom side of the PCB. The component outlines on the silkscreen print will tell you where they go.
- For most resistors and capacitors that are installed on the bottom, the **leads must be clipped flush with the PCB to allow installation of other components on the top**. The same is true for the HDMI socket and for pin header P4. I recommend clipping the leads with a side cutter *before* soldering.
- If you want to use the front panel, **the LCD and the MAX3232 chip should *not* be socketed**, to keep their profile lower than the switches.
- Before installing the LCD, make sure all components below it have been installed (on the bottom side of the PCB). Please **test the correct function of the HDMI output before installing the LCD**, since soldering errors will be very hard to fix afterwards!

## 2.1 What you will need

- Numato “Mimas” Spartan 6 FPGA board: http://numato.com/product/mimas-spartan-6-fpga-development-board
- LittleGP-30 PCB and electronic components, see the bill of materials in appendix 4.5.
- Cables:
  - USB connection (for FPGA configuration, PC serial connection) and power supply: Mini-USB type B, straight connector, to regular USB type A.
  - Serial connection: 8-pin mini DIN (male) to 9-pin Sub-D (female), to connect to a standard PC RS-232 port. See section 3.2 for pinout details.
  - HDMI, length < 1.5m recommended
- Hand soldering tools
  - Soldering iron (pen or chisel tip), solder (resin core, 0.5mm diameter or less), wire cutters, tweezers etc.
- A PC running Windows is required (temporarily) to download the FPGA configuration. For later operation, any PC running a serial terminal program can be used, or of course a real serial terminal operating at 19200 Baud.

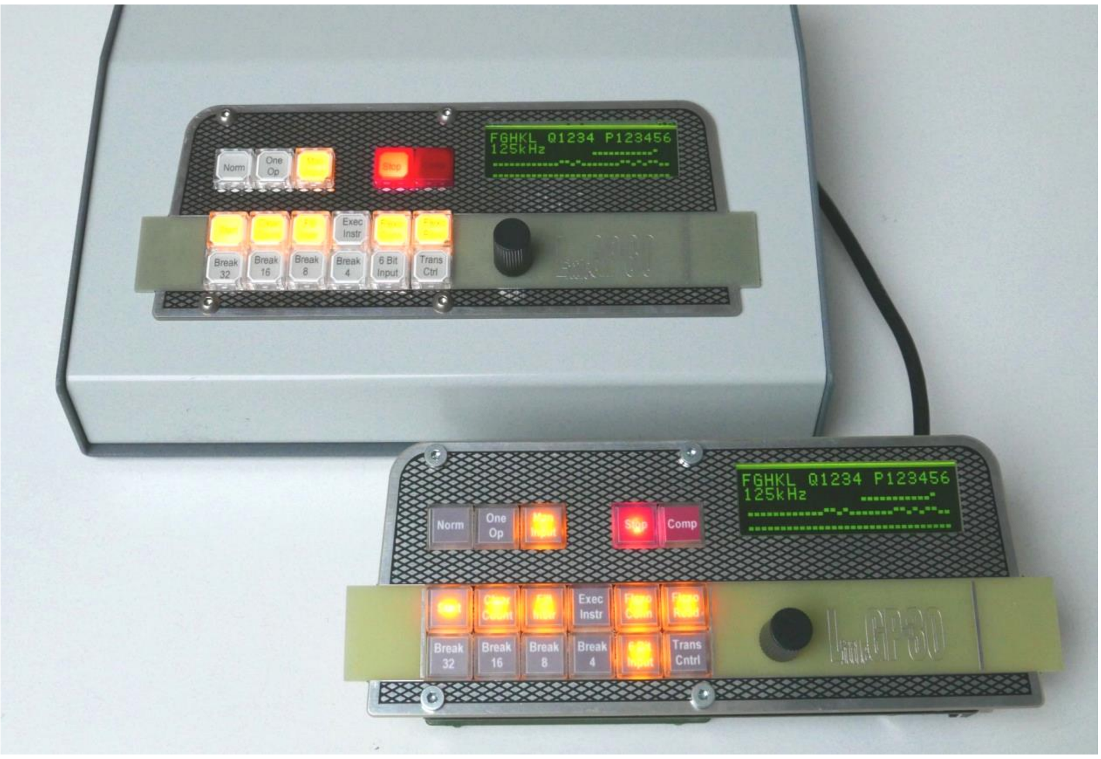
## 2.2 Builder’s choices

Before buying parts and building the replica, there are a few choices to be made: Which kind of switches do you want to install, do you want to interface a terminal via classic RS-232 and/or via USB, and do you want to build this kit for installation in an enclosure or for use as a bare board?

### 2.2.1 Switches – OMRON or Diptronics

The PCB is suitable for use with two different switch models. Both are back-illuminated, with 10*10 mm² square key caps which allow the insertion of printed labels to indicate button function. Both are momentary action switches; the latching action of the LGP-30 switches is recreated in software and indicated using the LED backlights. Which ones you prefer is a matter of taste:

- The OMRON B3W-9000 series switches provide a nice even illumination of the key cap, due to a built-in diffusor. On the other hand, the key tops show an octagonal outline inside the caps, which does not look quite right for the LGP-30. Also, I found that white-on-grey lettering, as used in the original LGP-30, is hard to read on the transparent film needed for the labels. The switches are more expensive, at approx. $2.50 each (17 switches are required).
- As a lower cost option, Diptronics TLL-6 switches and KTL-SAI key caps can be used. About $1.00 per pair, but the illumination is not as nice and shows a bright spot in the center. The key cap inserts are square, instead of the octagon shape of the OMRONs. White-on-grey lettering can be used on the paper labels, although it becomes a bit difficult to read when a button is illuminated.



*LittleGP-30 with enclosure (Hammond 515-0930) and OMRON switches, and as a bare-board version with Diptronics switches.*

The bill of materials in the appendix gives the exact type numbers for the recommended switches. The OMRON switches are more widely available, but come in many different configurations – be sure to get models with a single LED and 10mm clear key caps. Diptronics switches are only available from selected distributors, often in large quantities only. I am looking into sourcing options; Rapid Electronics in the UK is the only supplier of small quantities I have identified so far.

Several Chinese relatives of the TLL-6 switches are available at even lower cost. Also, remaining stock of an obsolete OMRON switch (B3E-1100), which has the same footprint as the Diptronics switches, is available from Jameco. The catch is that the clips for holding the key-cap are shaped slightly different in these switches, and that I have not found any matching transparent caps for either of them. If you come across a good low-cost option, please let me know!

### 2.2.2 USB and/or RS-232 connection?

A serial terminal, or PC-based terminal emulation program, is needed to program and run the LittleGP-30. The LittleGP-30 supports two options to connect this: Via a classic RS-232 interface, or via USB for use with a Windows, Linux or Apple Mac computer. You can choose to populate either one of the interfaces, or both.

Connector X100 and all components with 10x part numbers are for the USB interface. Connector X200 and all components with 20x part numbers provide the RS-232 interface. For space reasons, X200 uses a mini-DIN jack as introduced by Apple in the 1980s, but it is electrically compatible with standard RS-232 interfaces. (And with 1980s and 1990s Macs too!)

Parts count for the USB interface is very low, and the USB connector also serves as a convenient power jack. Hence, I recommend to always populate the USB port. On the other hand, if you plan to use your LittleGP-30 only with computers via USB, you may want to omit the RS-232 interface. You can always add it later if desired.

### 2.2.3 Enclosure or bare board?

The LittleGP-30 can be built as an open "single board computer" – well, a stack of two boards actually –, optionally adding the front panel for cosmetics. Or you can install the LittleGP-30 in an enclosure of your choice, using the front panel board to cover the cutout in the enclosure.

The only choice you need to make early on is whether to install the on-board mini-DIN connector X200 for RS-232, or whether to solder a short cable (to an enclosure-mounted Sub-D jack) directly to the PCB instead. The latter allows for a slightly more compact installation, which may be important if you want to keep the enclosure as narrow as possible.

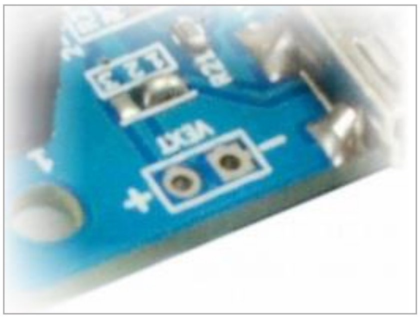
## 2.3 Preparing the Numato Mimas board

### 2.3.1 Power supply connection

The Mimas board and front-end PCB require a +5V regulated supply, with a current rating of 500 mA or more. (Up to 350 mA current consumption with an HDMI monitor connected, depending on the number of LEDs that are on.)  Any regular USB port or small USB-style power supply should be adequate.

The Mimas board can be powered through its USB connector, and generates the 3.1V supply for the FPGA and most components on the LittleGP-30 front end. But a few components on the LittleGP-30 board require a +5V supply, which is not present on the Mimas headers P1 and P2. The required connection will be prepared in this step:

- Populate the "VEXT" connector on the Numato Mimas board with a 2-pin female header. This will mate



with the single +5V pin P4 on the LittleGP-30 board. The ground connection will be established via the two large headers P1 and P2.

- On the power selection solder jumper, located right behind the VEXT connector, connect all three pins with a fat solder bridge. Power can now be supplied either via USB (for testing and FPGA programming) or through an external power jack connected via the LittleGP-30 piggyback board and the VEXT connector.

Attention: Do *not* supply power via USB and the VEXT connector at the same time! This would result in the two power supplies fighting over the definition of "+5V", and could damage a power supply. If you plan to install the LittleGP-30 in an enclosure with an external power jack, it's easy to forget about this restriction if you want to connect USB for reprogramming at some point. To prevent future problems, you can change the power selection jumper later to connect pins 2 and 3 only: This will disconnect the USB power, so +5V will always have to be supplied via the LittleGP-30 board, and no conflicts can occur. But for initial configuration and testing, it is more convenient to be able to use USB power.

### 2.3.2 Configuring the FPGA

To configure the FPGA with an available .BIN binary file, it is *not* necessary to install the Xilinx development environment. (See section 2.7 for details on that package.)

A simple downloading tool from Numato, the supplier of the FPGA board, is required. This tool is only available as a Windows program. You only need to perform these steps once; the FPGA will then store its configuration in flash memory:

- Obtain the .BIN file for this project from www.e-basteln.de/lgp30.
- Download the FPGA Configuration Tool, mimasConfig.exe, from Numato: http://productdata.numato.com/assets/downloads/fpga/mimas/mimasConfig.exe
- Download the USB Driver from Numato. Unpack the ZIP file, but don't install it yet: http://productdata.numato.com/assets/downloads/common/numato_lab_usb_cdc_driver.zip
- Connect the Numato board to the PC via a USB cable.
- When prompted on the PC, install the USB driver.
- Run mimasConfig.exe, which does not require installation. Select the correct COM port (check in the device manager if unsure) and .BIN binary file, and click "Program".
- Further documentation from Numato can be found at https://docs.numato.com/doc/mimas-spartan-6-fpga-development-board/

### 2.3.3 Preliminary function test

After programming is complete, the FPGA will automatically start up. The LGP-30 replica is already running, but you can't see much of it as the custom control panel is not there yet.

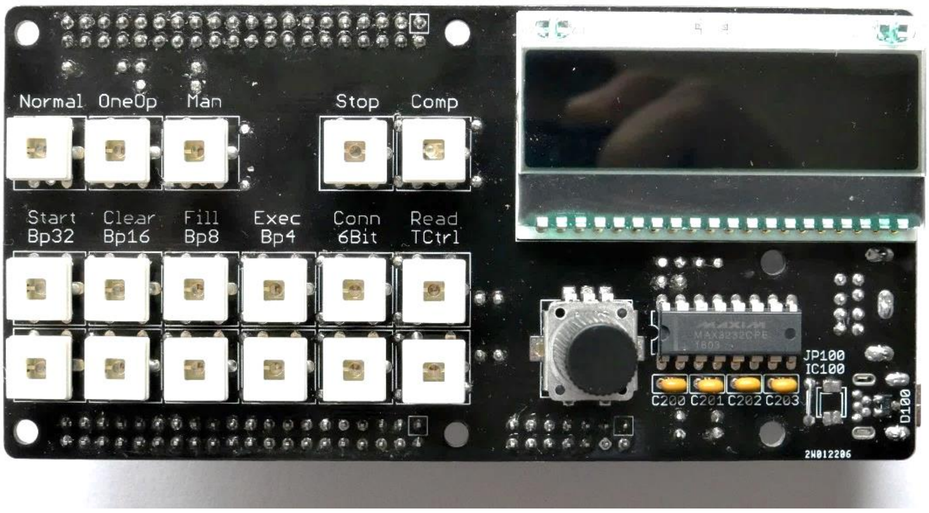
But you can check the correct function of the on-board switches on the Mimas board (which serve as option switches for a few user interface options): Each of the four buttons should toggle the LED next to it on and off.

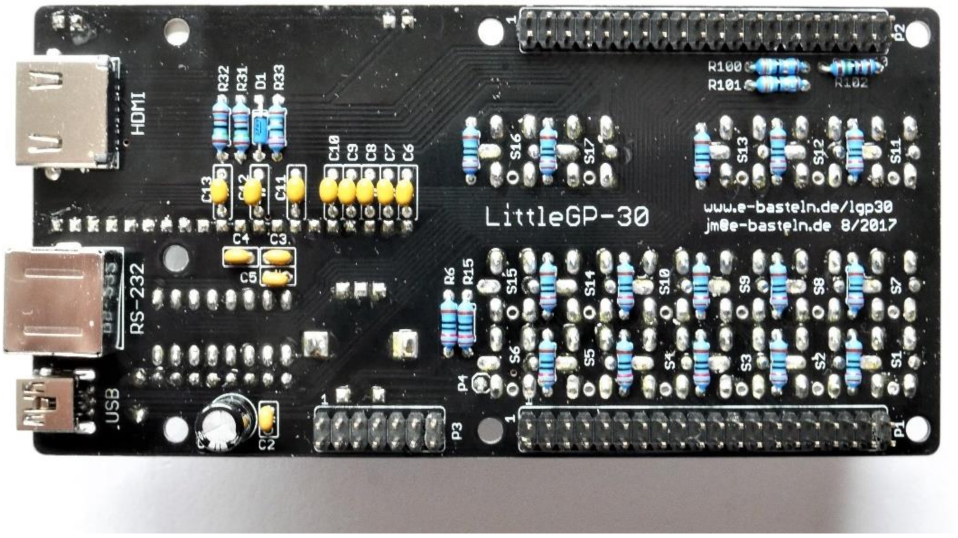
If you don’t get the expected behavior, please verify that you have selected the correct .BIN file and repeat the download steps above.

## 2.4 Populating the LittleGP-30 circuit board

Populating this board is quite straightforward. Except for two optional protection circuits for the USB port, no SMD (surface mount) parts are used. I recommend populating the board in the order of the following sections.

For reference, here’s what the final, populated board should look like. Diptronics switches are used in this example, with their labels and caps not installed yet.





### 2.4.1 Bottom: Discrete components

Populate the discrete components – resistors, capacitors, one diode – mounted on the bottom side first. Start with the lowest-profile components (resistors and diode), then work your way up towards the taller ones (small and larger capacitors), and finally the electrolytic capacitor. For the diode and electrolytic cap, be sure to observe the polarity as indicated on the silkscreen.

For all components that are installed below the LCD, it is important to clip their leads very short, flush with the PCB. If you use the OMRON switches, the same is true for the resistors mounted below these switches. The Diptronics switches have a smaller footprint which does not overlap with the resistors, hence there is no need to trim resistor leads flush below these switches.

I recommend that you

- insert the passive components,
- turn the board over and place it on a flat surface,
- clip all leads level with the PCB, using side cutters,
- then solder.

Since the through-holes all have plated sides, there will be ample surface for the solder to securely connect and hold the parts to the PCB.

### 2.4.2 Bottom: HDMI connector

The PCB is designed for a through-hole HDMI jack which arranges the 19 pins in three rows, in a 1.5mm pitch. This makes it reasonably easy to solder. The connector you buy may have a threaded flange for chassis mounting, which is not needed. You can clip the flange off if you prefer.

Install the HDMI connector. As it is also mounted below the LCD, its leads must also be clipped short, including the two mounting tabs. Clip flush with the PCB before soldering, as outlined above.

Inspect your soldering carefully, preferably under a magnifier, to catch any missing connections or solder bridges.

### 2.4.3 Bottom: Pin headers and USB connector

Install the pin headers P1 to P4. Expansion connector P3 is optional. See section 2.5 for its possible uses, to decide whether you want to populate it or leave it off for now.

To ensure that those headers which mate with the Numato Mimas board (P1, P2, P4) are nicely aligned, it is recommended to insert all of them into the corresponding Mimas sockets, place the LittleGP-30 PCB on top, and solder.

The single pin at P4 is located below a switch. If you use the OMRON switches, which have a larger footprint, **clip off P4, flush with the PCB's top side, before soldering**.

USB connector X100 could be omitted, if you want to control the LittleGP-30 via the RS-232 interface only. I recommend populating the USB interface in any case, as it offers additional flexibility and a convenient power supply connection. It also has a very low parts count; only X100 and resistors R100..R102 are needed. X100 is a type B mini USB jack (through-hole). These come with either two or four mounting tabs; either version can be used. Do solder the mounting tabs into the PCB slots for stability!

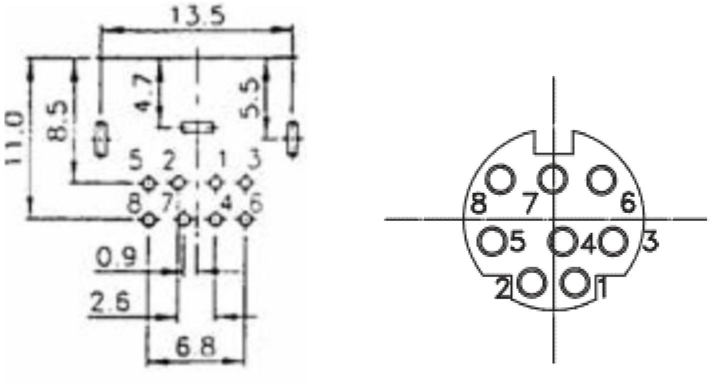
### 2.4.4 Bottom: RS-232 connector and cable

Populating the RS-232 interface (jack X200 and all other components with 2xx reference numbers) is optional. If you want to use the LittleGP-30 with a PC with USB interface only, the RS-232 interface is not required. If you decide to skip its installation now, it can still be added later, after completion of your build.

For space reasons, the RS-232 interface does not use a Sub-D connector, but the 8-pin mini DIN jack introduced by Apple in the 1980s. The pinout used by the LittleGP-30 is the same as used in the old Macintosh computers and peripherals. Plugs and jacks for PCB mounting are still readily available. A shielded jack is recommended, because its metal mounting tabs can be soldered into the PCB slots for a robust assembly.

Pre-made serial cables with an 8-pin DIN male plug (for the LittleGP-30) and a 9-pin Sub-D female plug (for a standard PC RS-232 jack) are also available, although not very common these days. These are the same cables that would be used to connect an old Macintosh computer to a PC for data transfer, or to connect an old Macintosh peripheral with serial interface to a PC. If you want to solder your own cable, please see the pinout and connection details below.

If you want to install the LittleGP-30 in an enclosure, with an enclosure-mounted Sub-D jack, you may prefer to omit X200 and solder a short cable directly to the PCB instead. Please see the pinout on the PCB (solder side) below.



*Circuit board footprint (seen from the solder side) and jack pinout (seen from the front).*

| Mini DIN-8 male | | Sub D-9 female | |
|---|---|---|---|
| Pin | Signal | Signal | Pin |
| 1 | Handshake out | CTS | 8 |
| 2 | Handshake in | RTS | 7 |
| 3 | TxD - | RXD | 2 |
| 4 | Gnd | GND | 5 |
| 5 | RxD - | TXD | 3 |
| 6 | TxD+ | -- | |
| 7 | GPIO | -- | |
| 8 | RxD+ | -- | |
| | | DSR – DTR | 6 – 4 bridge |

### 2.4.5 Top: MAX3232 and capacitors

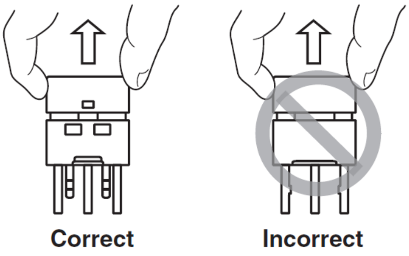
Install IC200, the MAX3232 level converter, and its capacitors C200 to C203. The IC should *not* be placed in a socket, to keep its profile well below the height of the switches.

Again, these components can be omitted if only the USB interface is to be used.

### 2.4.6 Top: Switches

**OMRON switches**

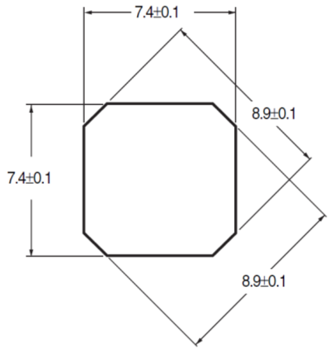
Before installing the OMRON switches, remove their key-caps. OMRON advises against pulling off the caps once the switches have been soldered, to avoid putting stress on the pins. Grab the caps on the side away from their mating latches, as illustrated.



The "Compute" and "Stop" indicators (S16, S17) are not switches in the original LGP-30, but only indicator lamps. The LittleGP-30 replica does use them as switches to select user interface options – see the use instructions below. In the bill of materials, I have suggested switches with green and red LEDs and higher actuation force for these two. Don't get them mixed up with the regular switches after you have removed the caps. An abbreviated model number is also printed on the side of the switch body.

The pinout of the OMRON switches is not symmetrical. They can only be inserted in one orientation, so no need to worry about LED polarity. Before soldering the switches, **please double-check that they sit nice and flat on the PCB, and are not tilted due to a resistor pin which is not clipped short enough.**

Before putting the key caps back on, print and insert labels. For the OMRON switches, these look best when printed on clear transparent film or translucent paper. OMRON's official dimension drawing is shown on the right; these labels fit into a recessed area inside the key caps. I have found that, unless they are printed on very thick film, the small labels can move around inside the caps. Larger labels which fill the full width of the caps stay in place better. A Powerpoint template with labels in both sizes is available for download on the LittleGP-30 page.



If you get a label wrong, it is feasible to carefully pry the key caps back off (violating OMRON's advice).

**Diptronics switches**

These ship with the caps packaged separately. I recommend soldering the switches first and installing the caps later.  Again, the two switches with green and red LEDs are meant for S16 and S17 ("Compute" and "Stop" indicators).

The Diptronics switches have a symmetrical pinout, and you must install them the right way around to get the LED polarity right. One leg of the LED is painted either red, green or yellow. This indicates the LED color, and also marks the cathode (negative terminal). **The colored LED leg must be con-**

**nected to ground. It must be oriented away from the series resistor corresponding to the switch, and away from the LCD display.**

For the Diptronics switches, print the labels on white paper (needed as a diffuser). Be careful when installing the key caps:

- Pry the ivory insert and transparent cap apart before installing them on the switches. I used a small screwdriver blade, inserted into the slits in the insert, to pull the insert out.
- Push the ivory inserts onto the switches first. Observe the correct orientation of the little plastic latches, and push the insert straight down. This step requires quite a lot of force and **is not reversible**; the latches will probably break if you try to pull the inserts back up!
- Position the label and push on the transparent cap. This step can be reversed with care if needed, but try to get it right first time to avoid damage to the switch.

### 2.4.7 Top: Encoder

The encoder can only be installed in one orientation. Do solder its mounting tabs too, for good mechanical support.

A small knob will help in manipulating the encoder. The one on the parts list (appendix 4.5) is cheap and unobtrusive, but needs to be drilled out with a 7mm drill if you don’t want it to sit very high. The required depth of drilling depends on the height at which the front panel is installed – please check and drill after the final assembly. Do not drill too deep; the knob should sit at least 0.5mm above the front panel to allow the pushbutton to be actuated! Before clamping the knob in a vise for drilling, wrap it with a few turns of masking tape – the plastic is quite soft.

### 2.4.8 Top: Optional USB protection or jumper wire

Above USB jack X100, two optional components for electrical protection of the USB input can be mounted: The dual diode D100 protects the USB data lines from transient voltages. The polymer fuse and protection diode, IC100, protects against over-voltages, over-currents, or reverse voltages on the +5V supply (if applied via the USB input).

While this extra protection is nice to have, it is not needed. When using the LittleGP-30 in a domestic environment, I recommend that you omit both components: Being the only SMD components on the board, they can be tricky to solder – especially IC100, which is intended for reflow soldering and difficult to hand-solder. They are also a bit pricey, and may be harder to source than most other components.

If IC100 is not installed, be sure to install jumper wire J100 instead; otherwise no power can be supplied via the USB jack. On the other hand, if you *do* choose to install IC100, do *not* install J100, as it would bypass the fuse.

### 2.4.9 Visual inspection and functional test

Congratulations – your LittleGP-30 replica is nearly complete! Before you move on to installing the LCD, I recommend a check of your work so far. Start with a careful visual inspection:

- Have all discrete components been populated, and with the right values?

- Pay particular attention to the components located below the LCD (C6-C13, D1, R31-R33). These will be difficult to fix once the LCD has been installed. Check values, diode polarity, and solder joints.
- Any solder bridges? Inspect the HDMI connector in particular. The headers P1, P2 and P3 also deserve a closer look.
- Have all pins been soldered? It is easy to miss some of the push-buttons' pins, or some of the pins on the large headers. In addition to malfunctions right from the start, this can lead to annoying problems with intermittent connections later.

Now plug the LittleGP-30 board onto the Numato Mimas for a first function test. This assumes that you have prepared the Mimas with FPGA firmware and VEXT power connector, as described in section 2.3.

- Orient the LittleGP-30 such that its one-pin connector P4 meets the VEXT +5V connector on the Mimas. Push down firmly to seat the P1, P2 and P4 connections.
- Connect power via the Mimas USB port.
- You should see several LEDs lighting up: Manual, Stop, Start, Clear, Fill, Conn(ect) and Read.
- Pressing the buttons in the bottom row should each toggle the respective LED on and off. The mode buttons – Normal, OneOp and Manual – should act as "radio buttons": Pressing one of them will light its LED and disable the LEDs in the other two. Various combinations of buttons in the middle row will also be lighted depending on the mode selected, showing whether they are enabled.
- Connect a monitor to the HDMI output. An image in XGA format (1024*768 pixels) should come up, displaying the contents of the LGP-30's magnetic drum.
- Try rotating the encoder (but don't push the encoder button). The display contents should move left and right.

**The HDMI function test is the most important one, as you cannot access the HDMI connector once the LCD has been installed.**

### 2.4.10 Top: LCD display

Before you install the LCD display, please perform the visual inspection and function test outlined above. Soldering errors on the components located below the LCD, and specifically on the HDMI connector, are hard to fix afterwards!

**The LCD should be installed *without sockets*.** The LCD and backlight are 5.5mm high; if you add a 4mm low-profile socket and the 1.2mm front panel thickness, the complete assembly will be nearly as high as the OMRON switches. Unless you like your switches nearly flush with the front panel, there is not enough room for a socket. If you use the lower profile Diptronics switches, there is definitely not enough room for a display socket.

There is one M3 mounting hole below the LCD. This hole is *not* used when installing the LittleGP-30 in an enclosure or attaching feet to a bare-board stack in the recommended ways (see section 2.6). **If you have any special plans for mounting the boards, which require attaching something to the bottom of the PCB via this hole, insert an M3 screw before populating the LCD.** In that case, chamfer the hole from the top side and insert a countersunk screw.

The DOGM-132 displays (132*32 pixel graphical LCDs) come in different versions. The recommended EA DOGM132S-5 is optimized for inverted operation, with bright pixels on black background, and is meant to be used with a backlight. EA LED55X31-G is a yellow/green backlight, which results in a color scheme similar to the original LGP-30 oscilloscope. Other models of the DOGM-132 series, including transflective versions without a backlight, can be used alternatively.

To install a display with backlight:

- Remove the protective foils from the display back and backlight front.
- Insert all display pins into the backlight's corresponding holes.
- Insert the LCD+backlight package into the PCB through-holes.
- Adjust the height of the display above the PCB:
  - For minimum height: Insert a strip of cardboard between the LCD and the PCB. This will be removed after soldering, and ensures a safety distance of 0.5mm or so to avoid mechanical damage to the LCD backlight.
  - For custom height: Mount your front panel at the desired height (using the spacers and enclosure wall thickness you are planning for). Turn the PCB stack over, let the LCD rest against the front panel, and make sure that it sits level.
- Solder the 20+2+2 pins of the LCD from the bottom side.
- Solder the 2+2 LED supply pins from the top, to electrically connect the backlight board to these pins.

## 2.5 Expansion port options

The 14-pin expansion connector P3 provides optional connections for

- 5V DC power supply (in or out),
- eight digital I/O lines – FPGA pins, which are configured as 3.3V CMOS by default. See section 2.5.2 below for information on their function in the standard FPGA setup.
- 3.1V DC power supply (out) for attached peripheral circuits.

At this time in the build, you probably don't want to bother with expansion options; but supplying power via P3 may be of interest if you want to install your LittleGP-30 in an enclosure.

| Pin | Function | Function | Pin |
|---|---|---|---|
| 1 | +5V in/out | +5V in/out | 2 |
| 3 | GND | GND | 4 |
| 5 | TEST6 I/O | TEST7 I/O | 6 |
| 7 | TEST4 I/O | TEST5 I/O | 8 |
| 9 | TEST2 I/O | TEST3 I/O | 10 |
| 11 | TEST0 I/O | TEST1 I/O | 12 |
| 13 | +3.1V out | +3.1V out | 14 |

*Pinout of the expansion port connector*

### 2.5.1 Power supply

Pins 1/2 and 3/4 can be used to connect an external power supply.  For LittleGP-30 systems installed in an enclosure, this may be more convenient than supplying power via the USB port on the LittleGP-30 or Mimas board. See section 2.3 for details on the Mimas power selection jumper.

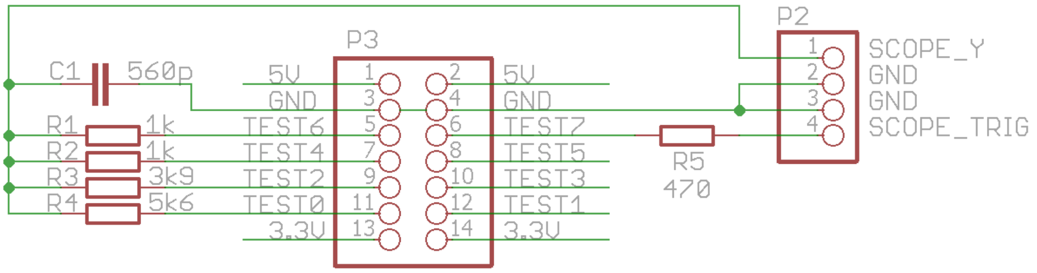
A regulated +5V supply, with a current rating of 350 mA or more, is required (500 mA or more recommended). I have measured up to 350 mA current consumption with an HDMI monitor connected, depending on the number of LEDs that are on. Operation without an HDMI monitor reduces this by ~40 mA. Any regular USB port or small USB-style power supply should be adequate.

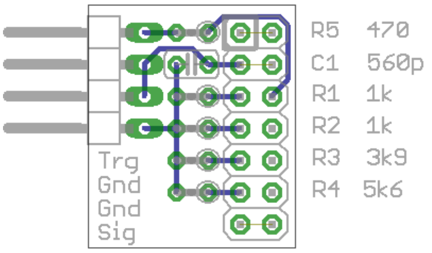
### 2.5.2 Oscilloscope connection

The original LGP-30 used an oscilloscope to display the register contents – accumulator, instruction register, and program counter. This type of display was needed since, due to the bit-serial architecture, the bits only showed up briefly, one at a time, at 125 kHz clock rate.

The LittleGP-30 replica has an LCD to display the same graphical information, so the oscilloscope is not needed. If you want to connect one just for kicks, here's how. The oscilloscope is actually a better display than the LCD, since it follows changes quite a bit faster! I recommend skipping this section in the initial build, and revisiting it later if you like.

Connector P3 provides the required signals: The register output, which alternates between bit-streams of the A, R, and C registers, and two offset outputs, which create the vertical displacement between the three signal traces . All signals need to be added via a simple resistor network – please see the circuit diagram and layout suggestion below. A trigger signal, which occurs at the beginning of each register word, is also provided.



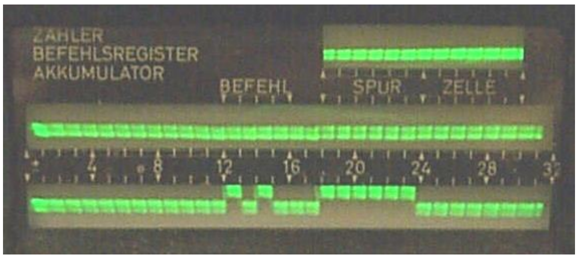


*Schematic and layout for an oscilloscope adapter for the expansion port. The Y signal and trigger outputs can directly drive 50 Ohm cable (at ~ 10:1 attenuation).*

*This can be simply hand-wired, as shown below.*

*Oscilloscope adapter, hand-wired on perfboard*

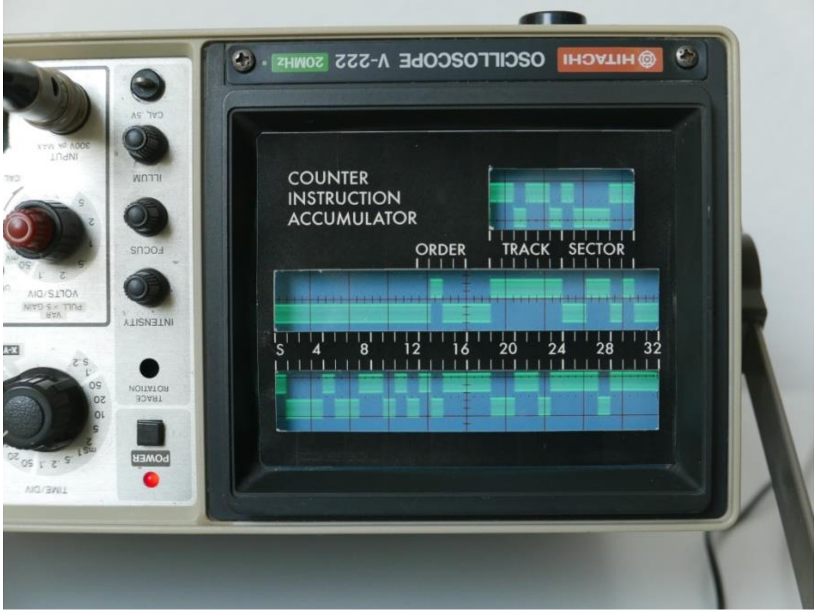
Some images of working LGP-30 displays show the traces having a large vertical width. While this may be an artefact of the photograph or a defect of the LGP-30 pictured, I find that the wide traces make the display easier to read. Hence, a “fat trace” component (5 MHz signal) is output on the connector and added to the Y signal via R4 to make the trace appear wider. You can adjust this to taste: To reduce the width of the trace, increase the value of R4, or vice versa.



*LGP-30 Display.*
*Image from http://computermuseum.informatik.uni-stuttgart.de/dev/lgp30/lgp30_3.html*

The polarity of the provided signals is such that the traces will appear in the same order as in the LGP-30: Counter on top, instruction register in the middle, accumulator below. However, the bit order will be the wrong way around: The LGP-30 always processes and outputs the low bit first (needed for ripple carry during addition and subtraction). Its oscilloscope scanned *from right to left* to show the bits in the conventional order, with the low bit on the right. Your oscilloscope is probably fixed to scanning from left to right; and unfortunately, the FPGA cannot easily output a ramp signal for use with an X/Y scope.

There is an easy solution to this problem: Turn your scope upside down! ;-)
You will then also need to invert the scope’s Y input to correct the orientation.



*This oscilloscope scans from right to left, showing low bits on the right. ;-)*

## 2.6 Enclosures and mounting hardware

You can either install your LittleGP-30 in an enclosure, or use it as a “bare board stack”. In either configuration, I personally like a tilted control panel, similar to the real LGP-30. As the final installation is a matter of personal taste, the parts list does not include any enclosure or mounting parts, and I will only give some general hints here.

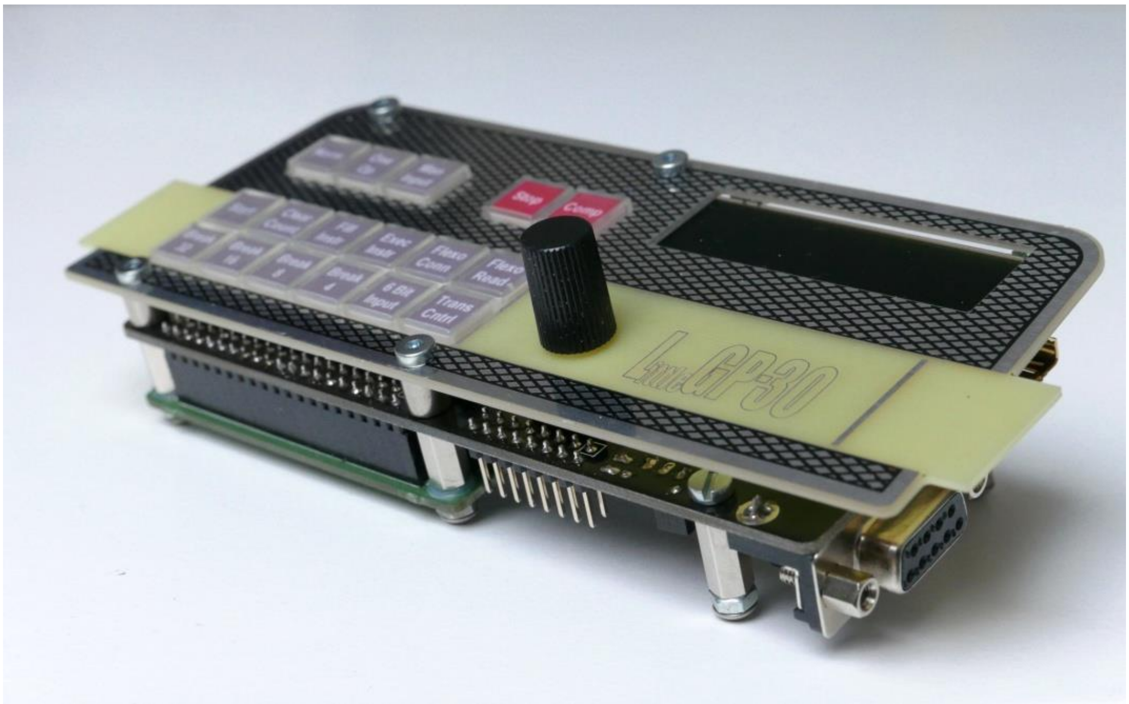
### 2.6.1 General considerations

- Be careful when using metal hardware to mount the boards. If the solder mask gets damaged, metal screw heads or nuts may cause short circuits. The use of Nylon washers is recommended.
- The minimum distance between the LittleGP-30 PCB and the front panel is 6 mm, defined by the encoder and the LCD. Spacers (brass or plastic tubes with 3.x mm inner diameter) are available in 5 mm or, less common, 6 mm length. You may find that you need to add a washer for a bit of extra clearance. On the other hand, the thickness of an enclosure’s front will count towards this distance. If you buy plastic spacers, you can simply cut them to the right length.
- Four screw heads will remain visible on the front panel. Personally, I like the look of DIN/ISO 7380 screws here, with button head and hex socket. You may prefer to chamfer the holes and use countersunk screws, but keep in mind that the front panel material is quite thin.

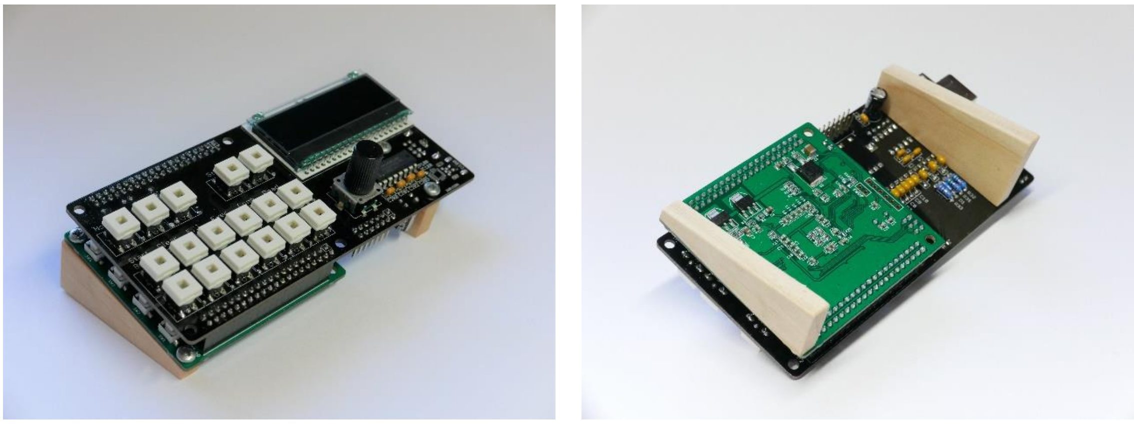
### 2.6.2 Bare board stack

Assembling the LittleGP-30 without an enclosure should be straightforward. The connectors on the right-hand side are all ready to use. You only need to attach the front panel with M3 screws and spacers, and add feet on the bottom to position your LittleGP-30 – either flat on the table or tilted up in the back.

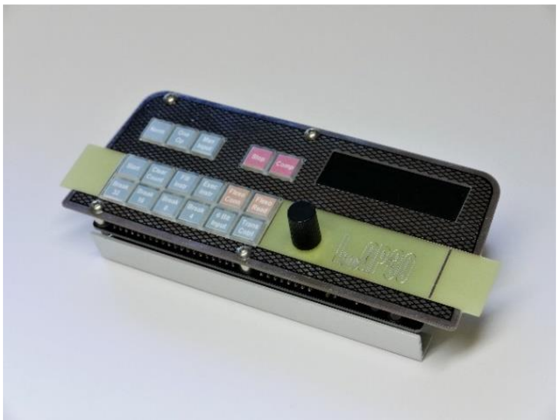
I have tried three different ways of supporting the board stack, which are pictured below. I used a tilted position in all cases, but obviously all three approaches can also be used with a horizontal placement of the boards.

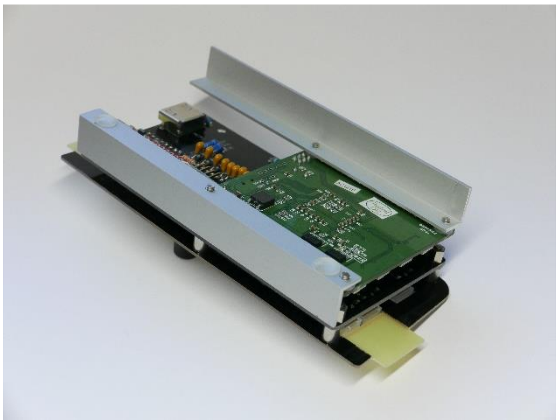


*A "bare bones" build of the LittleGP-30. This version simply uses PCB spacers for feet, with longer feet in the back to tilt the whole unit. Use the mounting hole beneath the LCD for best stability; but you can also get away with the two holes in front of the LCD.*



*Wooden support wedges don't use the mounting hole beneath the LCD.*
*The PCBs provide a component-free "corridor" for placing the supports.*





*Aluminum L profiles (10*15 mm) attach to the Numato FPGA board only. Use washers to keep the aluminum away from the pins of the two 40-pin connectors. – I tapped threads into the aluminum profiles here, but using countersunk M3 screws from below and nuts above the PCBs should work just as well.*

### 2.6.3 Mounting in an enclosure

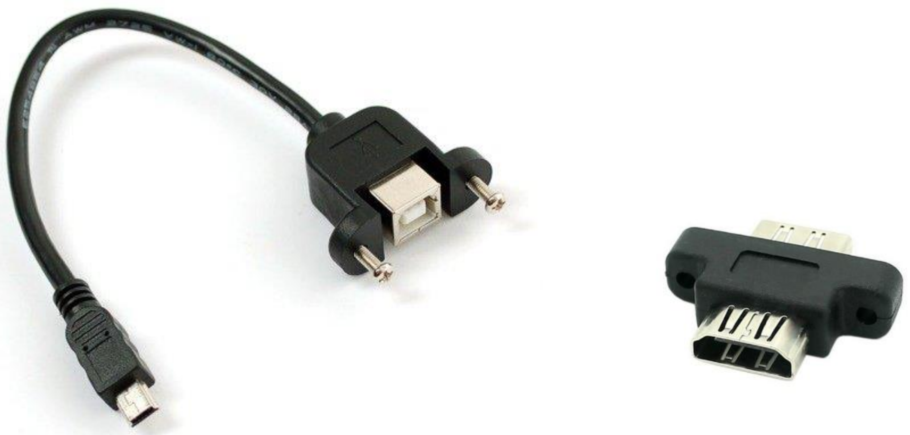
If you prefer an enclosure, these two fit nicely – there are many others of course:

- Hammond 515-0930, steel/aluminum. The top and back, where you need to make cut-outs, are aluminum; the bottom and sides are steel, with vents already in place. Comes powder-coated in two-tone grey. Available e.g. here, approx. €25: https://www.tube-town.net/ttstore/Gehaeuse/Hammond-Gehaeuse/Hammond-515-Serie/Hammond-Chassis-515-0930::3444.html
- Combiplast CP-18-34, grey ABS. Available e.g. from tme.eu, 5.95€. www.tme.eu/de/katalog/#search=cp-18-34

Suggested dimensions for the front panel cut-out are provided in appendix 4.3.

You will probably want to place connectors for RS-232 and power, and/or USB and optionally HDMI, on the back of the enclosure, using short cables inside the enclosure. A pure power connector is most easily routed to expansion connector P3. For RS-232 you can solder your own extension cable, either with a mini-DIN plug on the PCB side, or soldering it directly to the PCB. See section 2.4.4 for the pinout.

To connect USB and HDMI, I recommend ready-made short extension cables; these are available cheaply via various online outlets. USB cables with panel-mount jacks are readily available. HDMI cables with a panel-mount jack are also offered – but only with a straight plug on the other end, it appears. If space is tight and you need to use an angled HDMI plug on the PCB side, use a cable with plugs on both ends, and a separate HDMI panel connector (with jacks on both sides). Please see the pictures below for examples.



*USB extension with panel-mount jack (left), and HDMI dual-jack panel mounted adapter, for use with short HDMI plug/plug extension cables (right).*

## 2.7 FPGA development – optional

If all you want to do is download the provided LittleGP-30 .BIN file to the FPGA, you do not need the Xilinx development tools. Please see section 2.3.2 for the required steps.

But if you want to dive into modifying the FPGA code, the source files are provided at www.e-basteln.de/lgp30, and free Windows development tools for the Spartan-6 FPGA family are available from Xilinx. The tools are a bit unwieldy and no longer well-supported, however:

- A Xilinx Account is needed to download the tools. Free registration under https://secure.xilinx.com/webreg/createUser.do
- Download Xilinx ISE Webpack (a 6 GByte download!) https://secure.xilinx.com/webreg/register.do?group=dlc&htmlfile=&emailFile=&cancellink=&eFrom=&eSubject=&version=14.7&akdm=1&filename=Xilinx_ISE_DS_Win_14.7_1015_1.tar
- Xilinx ISE is no longer maintained and is not fully compatible with Win8 and Win10. The License Manager and Project Navigator both just close when you try to open a file. This can be fixed as follows, quoted from http://www.eevblog.com/forum/microcontrollers/guide-getting-xilinx-ise-to-work-with-windows-8-64-bit/:
    - Fixing Project Navigator, iMPACT and License Manager.
      Note: I am assuming you are using ISE 14.7 and have installed it to the default location
    - Open the following directory: C:\Xilinx\14.7\ISE_DS\ISE\lib\nt64
    - Find and rename libPortability.dll to libPortability.dll.orig
    - Make a copy of libPortabilityNOSH.dll (copy and paste it to the same directory) and rename it libPortability.dll
    - Copy libPortabilityNOSH.dll again, but this time navigate to C:\Xilinx\14.7\ISE_DS\common\lib\nt64 and paste it there
    - In C:\Xilinx\14.7\ISE_DS\common\lib\nt64 Find and rename libPortability.dll to libPortability.dll.orig
    - Rename libPortabilityNOSH.dll to libPortability.dll
- A brief tutorial for first steps with Xilinx ISE: http://langster1980.blogspot.de/2014/10/elbert-v2-fpga-tutorial.html

# 3 Using the LittleGP-30

Congratulations – your LittleGP-30 is ready to use!
The following sections will guide you through the next steps to get started:

- Familiarize yourself with the real LGP-30 – section 3.1
- Connect a power supply: +5V, 350mA or more.
  Use either your computer's USB port or an external supply, and connect it
  - via the USB mini-B connector on the LittleGP-30 (recommended), or
  - via the USB mini-B port on the Mimas board, or
  - via expansion connector P3 on the LittleGP-30 board (see 2.5.1).
- Connect an external ASCII terminal to simulate the Flexowriter – section 3.2.
- Briefly familiarize yourself with the LittleGP-30 control panel and LCD – sections 3.3 and 3.4.
- Start loading and running programs – section 3.5.

Further functions are optional and can be explored later:

- HDMI monitor for drum display – section 3.6;
- XModem file transfer to load and store drum images – section 3.7;
- Oscilloscope for old-style register display – section 2.5.2.

## 3.1 The real LGP-30

If you have decided to build this replica, you have probably familiarized yourself with the design and operation of the original LGP-30 to some extent. Several nice web pages with LGP-30 descriptions, technical documentation, and reminiscences of the early days of programming are easily found online.

The University of Stuttgart hosts the most comprehensive site, with technical documentation, a software collection on ASCII-converted paper tapes, and a description of their own, working LGP-30. Please see Appendix 4.1 for links.

The LGP-30 Operating Manual is specifically recommended as a starting point for your exploration of the LGP-30: ftp://bitsavers.informatik.uni-stuttgart.de/pdf/royalPrecision/LGP-30/LGP-30_Operations_Manual.pdf

## 3.2 Serial terminal

An ASCII terminal is required to use the LGP-30 replica. It replaces the Flexowriter and its paper tape reader and punch. If a terminal program with XModem capability is used, it can also be employed to store and recall the contents of the magnetic drum memory – discussed in section 3.7 below.

The terminal can either be connected via a serial (RS-232) connection or, when using a terminal program on a PC, directly via USB. The LittleGP-30 will automatically detect which port is in use when it receives the first character from the terminal. The following paragraphs describe the connections and the recommended configuration of the terminal.

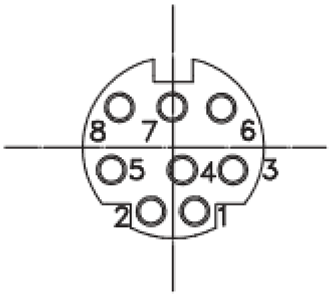
### 3.2.1 RS-232 connection

Set the RS-232 parameters to

- 19200 Baud
- 8 data bits
- 1 stop bit
- No parity
- Hardware handshake (RTS/CTS)
- No local echo

The physical data rate is chosen quite high, to enable XModem transfers of complete drum contents at reasonable speed. (19200 Baud is the fastest rate handled by real VT-100 terminals, in case you want to connect one.) But the effective transfer speed in regular LGP-30 operation will be much slower: Approx. 200 cps (characters per second) when reading, and 10 cps when printing. This matches the speeds of the high-speed tape reader and the Flexowriter, respectively. The slower speeds are required because the CPU timing during input and output operations needs to match the I/O timing. When running the CPU at lower or higher clock rates, I/O speed is adjusted accordingly.

| Mini DIN-8 male | | Sub D-9 female | |
|---|---|---|---|
| Pin | Signal | Signal | Pin |
| 1 | Handshake out | CTS | 8 |
| 2 | Handshake in | RTS | 7 |
| 3 | TxD - | RXD | 2 |
| 4 | Gnd | GND | 5 |
| 5 | RxD - | TXD | 3 |
| 6 | TxD+ | -- | |
| 7 | GPIO | -- | |
| 8 | RxD+ | -- | |
| | | DSR – DTR | 6 – 4 bridge |



*RS-232 cable pinout, to connect to PC serial port (9 pin Sub-D connector).*
*Jack pinout, seen from the front.*

### 3.2.2 USB connection

The USB connection should be straightforward: Run a USB cable with “mini B” connector to a Windows, Linux or Max OS computer. The LittleGP-30 FPGA contains a USB CDC device (communications device class). This is supported by on-board drivers in all mentioned operating systems, and shows up as a virtual serial port. Windows 8 and earlier versions require a .INF file for driver installation, which is available from www.e-basteln.de/lgp30; the other operating systems should recognize the USB serial device right away.

- Baud rate and handshake options can still be set in the PC-side driver, but this is only for backwards compatibility with client programs. The USB data transfer will work no matter what baud rate is set, and will always provide data flow control.
- Disable local echo in your terminal program, as the LittleGP-30 echoes each character when it is processed.

USB data transfer has been tested under Windows 7, Windows 10 and Linux (Ubuntu 16.4). The Windows USBSER.SYS driver apparently has some issues with flow control, which hinder the upload of text files (“paper tapes”) under certain conditions. This seems to depend on the timing and block sizes sent by the client program, but is easily worked around: When using the Tera Term program (see 3.2.5), I found that a 1 ms delay after transmitting each character was required for a stable USB handshake. Other terminal programs worked right out of the box.

USB transfer under Mac OS, Windows 8.x and Windows XP has not been tested, but should work. I appreciate your feedback if you should try one of these configurations.

### 3.2.3 Quick function test

Run a terminal program on your PC (or connect a real serial ASCII terminal!), configure as indicated above, and connect to your LittleGP-30. If you type characters, they should be scrolled into the A register, 4 bits at a time. You can see this on the LCD display. You should also see the characters being echoed back to the terminal program, indicating that the terminal can also receive data from the LGP-30.

### 3.2.4 Special characters and control buttons

The replica translates the Flexowriter 6-bit code to ASCII. Plain ASCII text is used, with one exception: The Flexowriter’s “color shift” code, which switches between black or red typewriter ribbon, is translated into VT-100 escape sequences to toggle between normal and boldface characters. Use of these escape sequences can be disabled by option switch SW3 (on the Mimas FPGA board); it is enabled by default.

The ESC key has a special function: It simulates the “Start” key on the Flexowriter, which is identical in function to the “Start” button on the control panel.

Reading a “ ‘ ” (conditional stop) character from the keyboard or from a text file also triggers a “Start” signal. In addition, it stops further input by clearing the CTS flow control signal. Input will be re-enabled by the next “Input” instruction executed by the LGP-30.

Two buttons on the replica’s control panel provide controls which were originally located on the Flexowriter:

- The “Read” button, located on the replica’s control panel, is a toggle switch. It replaces the “Stop Read” and “Start Read” switches on the original Flexowriter. It can be used to manually start and stop input from the serial terminal by changing the state of the CTS handshake signal. Note that, different from the real Flexowriter, the replica will not accept input from the keyboard while the “paper tape” is stopped. (Since the CTS signal will of course stop all RS-232 input.)
- The “Connect” button, also on the control panel, represents the “Connect” lever on the Flexowriter. The “Start” signals triggered by the ESC key or the “ ‘ ” conditional stop character will only be sent to the logic unit when the Connect switch is on. The Connect switch

should always be on in regular operation, but needs to be switched off e.g. for a manual, tape-assisted bootstrap. (See section 3.5.3.)

- There is no “Conditional Stop” control on the replica. On the real Flexowriter, this can be used to suppress the “stop the tape reader” function of the “ ‘ ” character. I have not found any use for this in the replica and assume that it was only needed if one wanted to copy a tape on the Flexowriter. So, the “conditional stop” character is actually an “unconditional stop” on this replica… If you run into the need to disable the stop function, please let me know!

### 3.2.5 Terminal programs; configuration and tweaks

A real VT-100 terminal should be usable, but I recommend a terminal emulation program on a PC. These can also send and receive text files, to replace the Flexowriter’s paper tape reader and punch. The paper tape files available on the internet are directly usable, specifically those from the Stuttgart computer museum: ftp://ftp.informatik.uni-stuttgart.de/pub/cm/lgp30/papertapes/

If the terminal program supports the XModem file transfer protocol, this can be used to load and store complete images of the magnetic drum – see chapter 3.73.6 on XModem below.

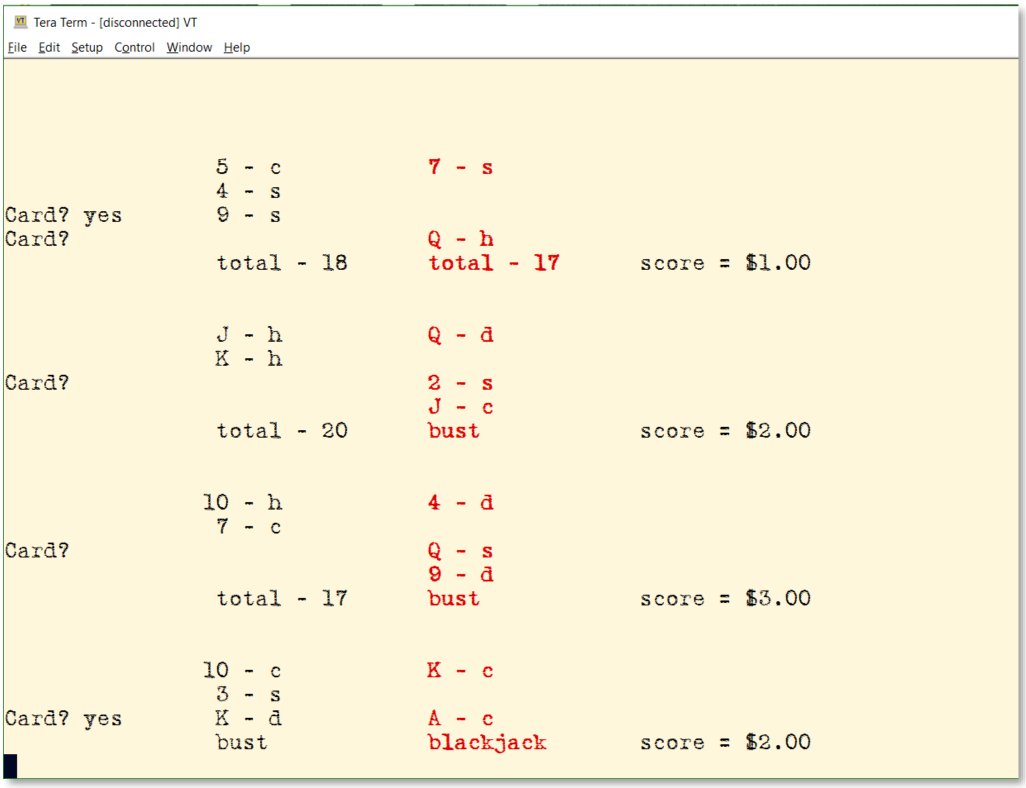
Under Windows, I like Tera Term, a free program which is still actively maintained: https://ttssh2.osdn.jp/index.html.en. If you use Tera Term, you can use some of its many configuration options to make the integration with the LittleGP-30 as smooth as possible. Insert or edit the following entries in teraterm.ini:

```
; XMODEM receive command
XModemRcvCommand=\

; Black text and yellowish background colors
; for normal characters
VTColor=0,0,0,255,255,229

; Set “bold” typeface to red, to match LGP-30 color shift
EnableBoldAttrColor=on
VTBoldColor=222,0,0,255,255,230

; Transmit delay per character (in msec)
; Workaround for Windows USB driver issue, not needed for RS-232
DelayPerChar=1
```

The Flexowriter – at least some models – had tabulator stops that were easily adjustable depending on the current application. The complete set of tab stops was arranged on a removable tabulator rack (http://homepage.cs.uiowa.edu/~jones/flexo/tour.shtml). While I am not sure whether this was the case for the Flexowriter model shipped with the LGP-30, it appears that tab settings different from the 8-character increments common today were indeed used. For example, the output from the well-known Blackjack program is correctly aligned for a tabulator spacing of 12 or larger.

The VT-100 terminal has user-adjustable tab stops as well: The “ESC [3g” sequence tells the terminal to clear all pre-defined tab stops; “ESC H” sets a tab stop at the current cursor position. If your terminal program correctly emulates the VT-100, you can prepare text files to set the tabs at various positions. This will allow you to emulate the Flexowriter’s quickly exchangeable tab racks by replaying the desired tab-setting file.

Tera Term has a command line option to replay a file upon startup: "ttermpro.exe /R=tab12.tty” will start a terminal with tabs set every 12 characters, if that’s how you have set up the tab12.tty file. I have prepared such files for download at [www.e-basteln.de/lgp30](http://www.e-basteln.de/lgp30).


Tera Term - [disconnected] VT
File Edit Setup Control Window Help

```
                5 - c           7 - s
                4 - s
Card? yes       9 - s
Card?                           Q - h
                total - 18      total - 17      score = $1.00


                J - h           Q - d
                K - h
Card?                           2 - s
                                J - c
                total - 20      bust            score = $2.00


               10 - h           4 - d
                7 - c
Card?                           Q - s
                                9 - d
                total - 17      bust            score = $3.00


               10 - c           K - c
                3 - s
Card? yes       K - d           A - c
                bust            blackjack       score = $2.00
```

*With the right color scheme, font, and tab setting, the terminal does a credible job at conveying the Flexowriter look.*

## 3.3 Control panel

### 3.3.1 LGP-30 Control Panel

All relevant buttons and indicators from the LGP-30 control panel are present. See the LGP-30 operation manual for a detailed explanation of their function.

- Top row: Mode switches (Normal, One Operation, Manual Input). Pushing one button selects its mode and releases any other mode. The LEDs indicate the active mode. “Stop” and “Compute” are pure status indicator lights in the original LGP-30. The replica uses switches instead, which can control user interface options (see section 3.3.4 below).
- Middle row: Operation buttons (Start, Clear Counter, Fill Instruction, Execute Instruction). Pushing a button triggers an operation. Depending on the Mode selection, only some of these buttons are active. The LEDs indicate which buttons are active.
- Bottom row: Toggle switches (Breakpoints, 4/6 Bit Input, Transfer Control). Each push toggles their state, and the LEDs show the current state.

The LGP-30’s power-up, standby and shutdown functionality is not replicated.

### 3.3.2 Flexowriter controls

The control panel includes two buttons normally located on the Flexowriter:

- “Connect” switch (toggle): Controls whether the “Start” button on the Flexowriter or a “conditional stop” character read by the Flexowriter are passed to the LGP-30. This should be enabled most of the time, but will need to be disabled e.g. for a semi-manual bootstrap procedure. (The bootstrap involves reading word for word from the tape, each followed by manual fill/execute instruction commands).
- “Read” toggle: Combines the functions of the Flexowriter’s “Start Read” and “Stop Read” controls. It will start and stop the paper tape reader – in case of the replica by setting or clearing the “CTS” signal on the RS-232 port.

### 3.3.3 Encoder

The Encoder is not present in the original LGP-30, but is a feature of the replica only. It operates in two modes; you can toggle between them by pushing down on the encoder knob. The active mode is indicated in the LCD clock rate display:

- Normal mode (active after startup): Shift the HDMI drum display left and right.
- Clock mode (clock rate display in the LCD is inverted): Control the LGP-30 CPU clock rate, i.e. the rotation speed of the simulated drum, in bits/second.

The standard bit clock is 125 kHz. It can be increased up to 1.25 MHz and decreased down to 1.25 Hz in three steps per decade.

At speeds of 250 Hz and below, the HDMI drum display will switch to animated mode, showing the drum rotation and the position of the read/write heads. Turning the encoder further left (slower) below 1.25 Hz enters single step mode: The drum will stop, and each further encoder step to the left will move the drum by one bit. Note that this is much slower than the LGP-30 “One Operation” mode, which executes one complete instruction per step.

As a third function, the encoder allows you to adjust the LCD contrast. To do so, push and hold the knob down while turning it. This setting will rarely be needed: Unless you use the LCD under uncommon angles or lighting conditions, the default contrast setting works fine. As a side effect, pushing and holding the knob will also be registered as a toggle between normal and clock mode. Briefly push the knob again to switch back to the desired mode.

### 3.3.4 Option switches

The switches on the front edge of the Numato Mimas board may be used to control options of the replica. Options can also be toggled via the switches in the “Compute” and “Stop” status lights, so they are accessible in enclosed systems. In the standard FPGA configuration, the following are supported:

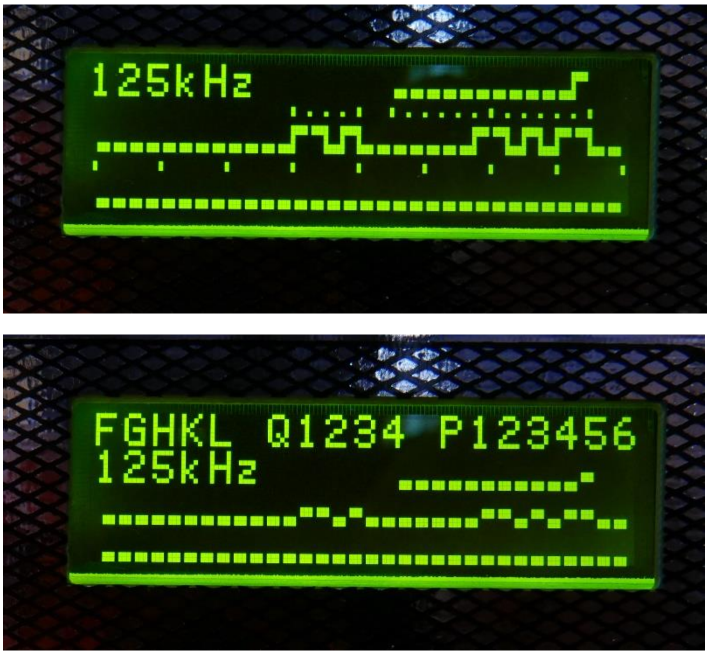
- SW1 and “Stop” switch:  Toggle extended information in the HDMI display. The display of the 15 internal flip-flops, at the bottom of the screen, scan be toggled off and on.
- SW4 and “Comp” switch: Toggle extended information in the LCD display. Switches to a more compact display of the CPU register contents and adds a display of the 15 internal flip-flops in the top line.
- SW3: In the RS-232 output, disable the VT-100 ESC sequence for black/red color shifting. This may be useful when using a non-VT-100 terminal, or when capturing LGP-30 output in a text file (“punching a tape”).

## 3.4 LCD display

The LCD replicates the oscilloscope display in the LGP-30: It shows the CPU registers – from top to bottom: Program counter C, instruction register R, accumulator A. For the R and A registers, 32 bits are displayed, including the spacer bit in the rightmost position. The most significant bit (sign bit) is on the left. The program counter is 12 bits long (6-bit track address, 6-bit sector address).

In between the three traces, little tick marks indicate the position of the opcode, track and sector fields in the C and R registers, and denote groups of four bits in the R and A registers. In the original LGP-30, these markings were printed on a screen overlay to make the display easier to read. In the upper left, the LCD displays the current CPU clock rate, which can be adjusted via the encoder (see section 3.3.3). This is a feature of the replica which was not present on the LGP-30.

An extended display mode, which also shows the status of the LGP-30’s 15 electronic flip-flops, can be enabled via option switch SW4 on the Mimas board, or via the switch in the “Compute” status light. This display was not present on the original LGP-30’s user control panel, although the flip-flop tube modules had neon bulb indicators for troubleshooting. Observing the flip-flop status may be of interest if you want to follow the LGP-30’s instruction decoding step by step, at very slow clock rates or in single-step mode. If you have connected an HDMI display, that provides a more convenient way to observe the flip-flops.



*LCD in regular and extended mode*

## 3.5 Loading programs

There are three ways to load programs into the LGP-30 replica:

- Reading "paper tapes":
  - Text files are uploaded via your terminal program to simulate paper tapes.
  - See 3.5.1 for details.
- Loading previously stored drums:
  - Complete drum contents can be stored and re-loaded if you use a terminal program that supports the old XModem file transfer protocol.
  - A few ready-made drum images are available for download from www.e-basteln.de/lgp30.
  - See the "XModem" section 3.7 for details.
- Manual bootstrap from an empty drum:
  - The hard way, for "real programmers" ;-)
  - See 3.5.3 for a walk-through.

### 3.5.1 Loading a paper tape

- Text files are uploaded via your terminal program to simulate paper tapes.
- The LGP-30 boot loader program, called 10.4, is already preinstalled on the replica's drum. Note that this is a German version from the Stuttgart archives – it prints "MIST" instead of "ERROR" if a checksum error occurs.
- Press the following control panel buttons to start the loader:
  **Manual Input – Clear Counter – Normal Operation – Start**
- Tell your terminal program to send the paper tape text file.

- Observe the transfer on the terminal's screen. Due to a buffer on the sending PC's side, the file transmission may proceed rapidly. Processing by the LGP-30 will lag behind, and the echoed characters on the screen will indicate the speed at which the LGP-30 digests the characters. Wait until the transfer is complete and no further output appears.
- Most paper tapes will end with a command that sets the program counter to the start address, so you simply press **Start** once more to start the newly loaded program.
- Alternatively, set the start address manually. This is done by filling a jump instruction ("unconditional transfer" U) into the R register and executing it. For example, to start the Blackjack program at address 2k00, enter the following on the control panel and the terminal:
  **Manual Input**
  (Terminal:) **000u2k00**
  **Fill Instruction – One Operation – Execute Instruction**
  **Normal Operation – Start**
- (Note: The Blackjack program only initializes its variables upon the first Start, and then stops for input. To continue, press Start again, or the ESC key on the terminal's keyboard.)

General hint: Paper tape files should *not* have a carriage return (or other characters) after their final "conditional stop" character! When reading a paper tape, this character will remain in the RS232 input buffer and may get in the way of subsequent operations, e.g. an XModem download. Many of the Stuttgart paper tape files do have a final carriage return; I recommend removing it. To flush the buffer of pending characters, switch the LGP-30 to Manual mode and activate "Read" – possibly several times, until no further input is received.

### 3.5.2 Compiling and running an ACT-V program

The "Algebraic Compiler and Translator", ACT, provides a compiled high-level language for the LGP-30, roughly on the same level of abstraction as an early Fortran dialect. The Stuttgart archives (see appendix 4.1) have manuals and paper tapes for various versions of ACT, but only ACT-V seems to have complete paper tapes *and* manuals.

The following steps load the ACT-V compiler and libraries, and compile and run the "lunar lander" ACT-V program. The instructions are adapted from Christian Corti's simulator instructions at ftp://ftp.informatik.uni-stuttgart.de/pub/cm/lgp30/simulator/lgp30_en.txt. All mentioned paper tape files are available from the Stuttgart server. Please note that only a German version of the program, "Mondlandung", is available.

0. The following paper tapes are needed (each as a file on disk):
   - ACT-V compiler – act5.tx,
   - ACT-V basic subroutines – act5bs.tx,
   - source code for "lunar lander" – mondl.act,
   - square-root subroutine – wurzl.tx.

1. Load the drum with the 10.4 boot loader –
   no action needed, as this is already preloaded upon power-up of the LittleGP-30.

2. Load the ACT-V compiler. This is a regular paper tape transfer using the 10.4 loader:
   **Manual Input – Clear Counter – Normal Operation – Start**
   (Terminal:) Send text file > act5.tx

3. Load the ACT-V basic subroutines. Another paper tape transfer using the 10.4 loader.
   **Manual Input – Clear Counter – Normal Operation – Start**
   (Terminal:) Send text file > act5bs.tx

4. If you want to experiment with your own ACT-V programs in the future, now is a good time to store an image of the drum with the compiler and libraries.

   Simply tell your terminal program to receive a file in XModem (128-byte blocks), and store it e.g. as act5.drum. The transfer should proceed automatically. The LittleGP-30 will display "XModem" in place of the clock rate for a while, then switch back to regular operation.

5. Read and compile the source program.
   First start the compiler at address 2000, via a 10.4 instruction:
   **Manual Input – Clear Counter – Normal Operation – Start**
   (Terminal:) **.0002000**
   **Start**

   The LGP-30 will stop immediately (HALT) because some break points need to be set, input must be switched to 6 bit and the source program must be made ready for reading. So, enter:
   **Bp8 – Bp4 – 6Bit**
   **Start**
   (Terminal:) Send text file > contact

   The source program is read and compiled. At the end of the compilation the compiler will print the symbol table.

6. Load the square subroutine.
   The symbol table shows that the label s000 (= last code address) is at address 1042. This information is needed to find a place for the square root. A good place is 1045.

   We will tell the 10.4 loader to load the subroutine there (the ';' instruction below), and relocate it by adding a suitable offset to all relative addresses on the tape (the '/' instruction). Towards its end, the square root tape also contains an instruction to patch a (relocated) jump to the subroutine to an address somewhere in the ACT-V library, so the library knows where to find it...

   **Bp8 – Bp4 – 6Bit** (release keys, lights should be off)
   **Manual Input – Clear Counter – Normal Operation – Start**
   (Terminal:) **;0001045**
   **Start**
   (Terminal:) **/0001045**
   **Start**
   (Terminal:) Send text file > wurzl.tx

7. Finally: Start the compiled program
   **Manual Input – Clear Counter – Normal Operation – Start**
   (Terminal:) **.0000300**
   **Start**
   **Start**

Now have fun playing this game!
(Don't forget: Always enter numbers with the sign, e.g. +30 instead of 30!)

**8.** If you plan to play the game again in the future you should save the drum, so that you can simply reload it and press Start to play. Perform the following before or after step 7:

**Manual Input – Clear Counter – Normal Operation – Start**
(Terminal:) **.0000300**
**Start**
Terminal: XModem Receive > [name, e.g. LUNAR.DRUM]

After powering the LittleGP-30 the next time, the game is simply started with
(Terminal:) XModem Send [name, see above]
**Normal**
**Start**

### 3.5.3 Loading the 10.4 bootstrap manually

The hard way, for "real programmers"! ;-)

The paper tape for the 10.4 loader begins with an interesting self-extending bootstrap sequence. The first words are entered in a semi-manual way; then the boot loader takes over to read the main program from the remainder of the tape. It is worthwhile trying this once. (But then it gets old quite fast…)

The bootstrap process is described in the 1963 edition of the Programming Manual. German version available here – see page 27: ftp://ftp.informatik.uni-stuttgart.de/pub/cm/lgp30/docs/EC13-022_Programmierungsanleitung_Apr1963.pdf.

1. Obtain the ASCII-converted paper tape p104.tx from the Stuttgart server.
2. To start from scratch and remove the pre-installed 10.4 loader, first load an empty drum image via XModem. A drum image "empty.drum" is provided at www.e-basteln.de/lgp30
3. On the control panel, set **Flexo Conn(ect)** to OFF. While the initial bootstrap commands will be read from the paper tape, we will need to control the execution of instructions manually.
4. On the terminal, start sending p104.tx as a text file.
   The word 'manual' will be printed, and the Read indicator on the control panel will be switched OFF.
5. On the control panel, switch to **Manual Input** and press the **Flexo Read** button.
   The first instruction will be printed on the terminal and shifted into the accumulator.
6. On the control panel, press **Fill Instruction** and then **Flexo Read**.
   The instruction will be transferred to the R register, and the next data word printed and shifted into the accumulator.
7. On the control panel, switch to **One Operation** and press **Execute Instruction**.
   The instruction in register R will execute, using the data in accumulator R.
8. Repeat steps 5 to 7 five more times.
   The last printed data word should be Z000'.
9. On the control panel, switch to **Manual Input** and press **Flexo Read**.
   The word 'normal' will be printed, indicating that the manual bootstrap entry is complete and the boot loader is ready to take over.
10. On the control panel, switch to **Normal Operation** and set **Flexo Conn(ect)** to ON.

11. On the control panel, press **Start**.
    The remainder of the paper tape file will be read automatically.
12. When the paper tape stops, the 10.4 loader is ready to use! To start the loader and load an application paper tape, continue as described in section 3.5.1.

Understanding how the self-extending bootstrap sequence actually works is left as an exercise to the reader. ;-) The self-extending, self-modifying code extends well into the instruction sequence which is loaded automatically in step 11. It is quite clever indeed, and I am not sure whether I have fully figured it out…

## 3.6 Drum memory and flip-flop display (HDMI) – optional

Use of the drum memory display is entirely optional. The real LGP-30 did not have such a display, and the replica can be fully used without it. But when you really want to follow the operation of the LGP-30 on a bit level, the display may be useful.

A monitor with an HDMI input is required. (DVI inputs can also be used, with a cheap passive adapter.) The video output has XGA resolution, 1024*768 pixels. Hence older 4:3 monitors should work nicely, but the picture should also look fine when scaled to higher resolution or at 16:9 format.

The display always shows all 64 tracks of the LGP-30 main memory. Register contents and control signal tracks are shown below the data. Track numbers are indicated on the left and right side of the screen, sector numbers on top. Note that the display shows the sectors in their physical order on the drum; therefor the logical sector numbers follow the 7-step interleave pattern used by the LGP-30.

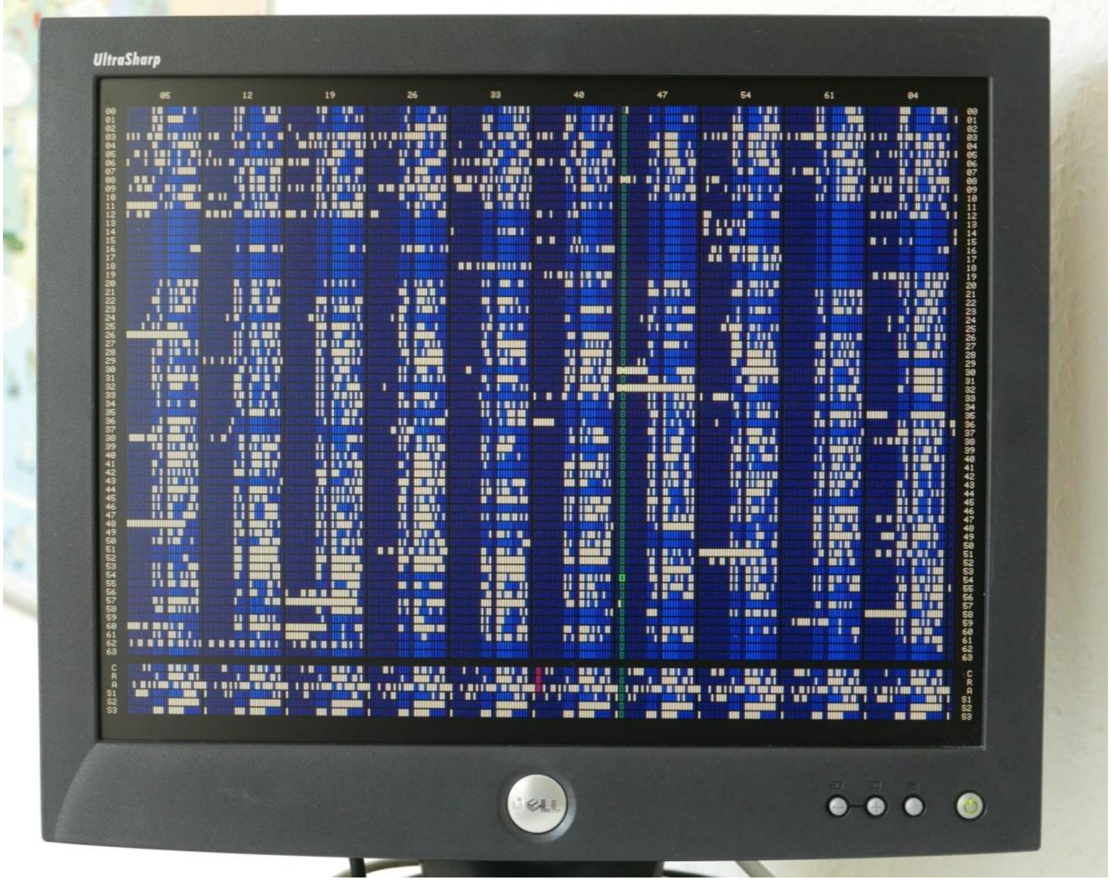
The display shows a 10-word segment of each track. Low bits are shown on the right. Word boundaries and word structure are delineated by blue bands, with track, sector and opcode fields highlighted in brighter shades.

In normal operation, the display is static and can be scrolled left and right with the encoder knob. (Encoder must be in normal mode, clock rate in the LCD not highlighted.)

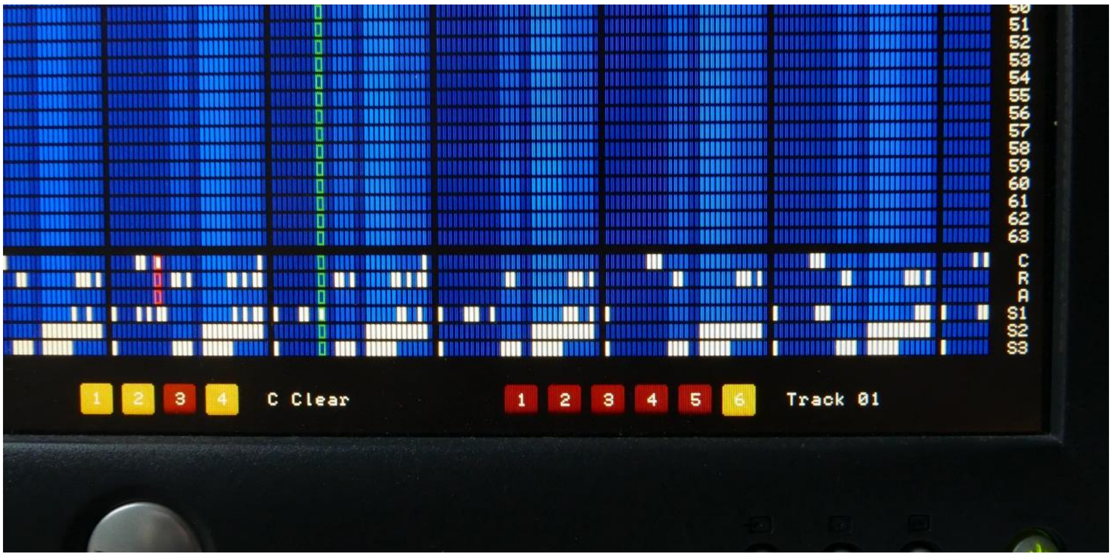
At slow CPU clock rates (250 Hz and below), the display will switch to scrolling mode. A stationary column of 64 read/write heads is shown in green, and the drum's bit pattern scrolls across it from left to right. The active read head is indicated by a highlighted frame; during write operations this frame switches to red color.

The three dedicated write heads for the re-circulating registers are shown in red, upstream from the read heads. When register contents change, you can observe how the write head refreshes the whole track content with the same repeating pattern, by continuously re-writing what the read head sees 32 bits downstream. The additional read head for the accumulator, used during multiplication, is not shown – the drum does not "know" when the logic unit reads this head anyway.

At the bottom of the screen, the status of the LGP-30's 15 electronic flip-flops is displayed and decoded. This may be of interest if you want to follow the LGP-30's instruction decoding step by step, at very slow clock rates or in single-step mode. This display can be toggled off and on via option switch SW1 on the Mimas board, or via the switch in the "Stop" status light. In the original LGP-30, the user could not normally observe the flip-flop contents, but the internal flip-flop tube modules had neon bulb indicators for troubleshooting.



*The drum display shows all 64 tracks as well as the CPU register and sync tracks, but only 10 sectors at a time. Rotate left or right with the encoder knob.*



*When the logic unit's clock is slowed down to 250 Hz and below, the read/write heads will be displayed, and the drum will move across the heads in real time. This picture also shows the flip-flop display enabled in the bottom.*

## 3.7 Drum memory storage (XModem) – optional

The magnetic drum of the real LGP-30 retains its contents – main memory and registers – when the LGP-30 is stopped and shut down. After restarting the computer, a program can be resumed at the exact point it had been stopped.

In contrast, this replica stores the drum content in RAM memory; data are lost when the FPGA is powered off. To make the use of complex programs easier – e.g the ACT compiler and libraries, which have to be read from multiple tapes – drum contents can be stored and retrieved via XModem file transfers. To use this feature, you must use a terminal program that supports the classic XModem protocol:

### 3.7.1 Operation

- The classic XModem protocol is used, with 128 Byte blocks and 1-byte checksums (not CRC).
- Main memory (64*64 words) and register tracks (A, R, C) are stored.
- Download of the drum contents from the FPGA is started via the terminal program. The simulator will recognize the NACK control character sent by the terminal program and will automatically start the transfer. This is part of the standard XModem protocol and should work for all terminal programs that implement XModem.
- Upload of drum contents to the FPGA is also started from the terminal program. It requires sending the reserved "\" character to signal a transfer to the replica, which will then enter reception mode and send a NACK. This function of the "\" character is *not* an XModem standard feature, but a convention adopted for the LGP-30 replica only.

  The "\" can either be sent manually from the terminal program; in this case the upload must be started max. 100 seconds afterwards. Alternatively, some terminal programs are configurable to send an XModem upload command automatically. See the "Serial Terminal" section 3.2.5 for an example configuration for the Tera Term program.

### 3.7.2 Data format

As long as you just store and retrieve drum files, you don't need to care about the data file format. Here it is anyway. Note that for technical reasons, the file format is *not* compatible with the format used by Christian Corti's LGP30SIM.EXE!

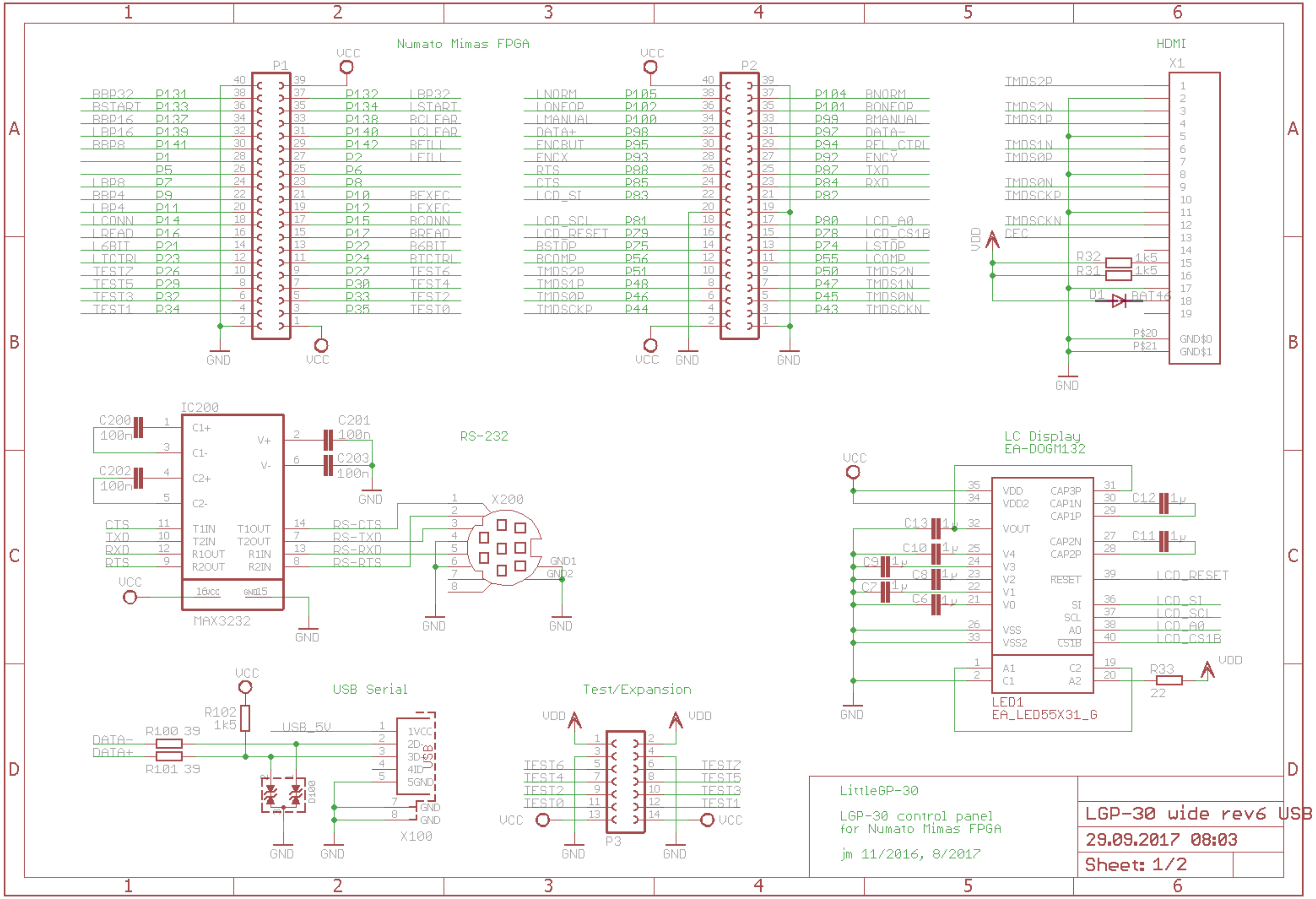
- Data files are 18,432 byte binary files.
- First 16,384 bytes are the main memory data:  
  Track 0..63 in ascending order; each starting with logical sector 00, followed by sectors in physical order (i.e. logical sectors 00, 57, 50, …); low byte first.
- Following 2,048 bytes are the register data:  
  Each stored byte contains one bit each from the A, R, C register tracks, with the following structure: "0 0 0 0 0 A R C" (five padding zeros in the high bits, register data in the low bits). Bytes are stored in the order in which they appear on the physical tracks, starting with drum position 0 (logical sector 0 on the data tracks).

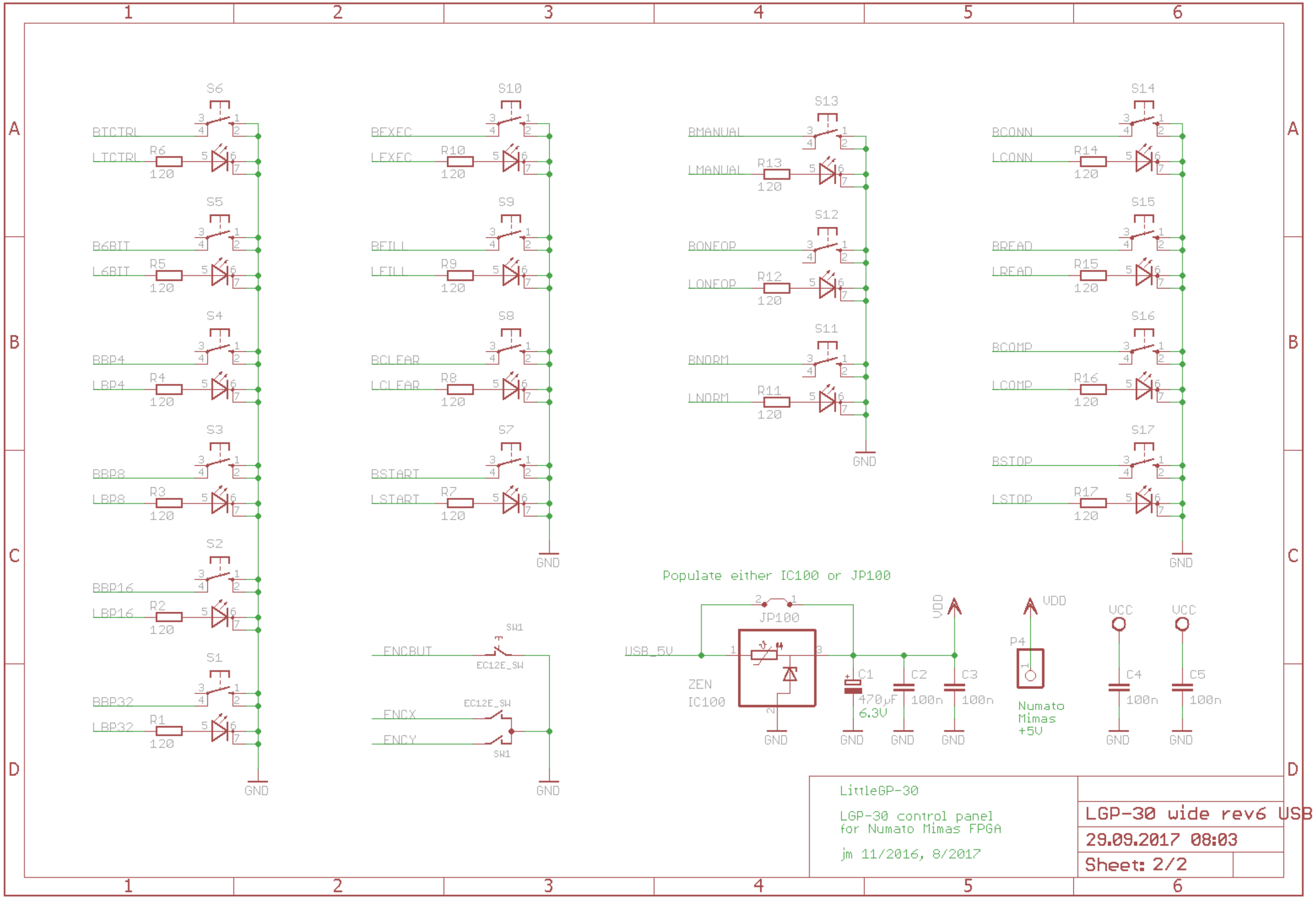
# 4 Appendix

## 4.1 Download links

- All my project files (FPGA source code and binary, and the latest version of this document) can be downloaded from www.e-basteln.de/lgp30.
- Tools and documentation for the Numato Mimas Spartan 6 development board are available from http://numato.com/mimas-spartan-6-fpga-development-board/.
- For download links to the Xilinx development tools, see section 2.7 on FPGA development.
- The University of Stuttgart's computer museum has a great collection of LGP-30 information:
    - Program documentation and ASCII-converted paper tapes: ftp://ftp.informatik.uni-stuttgart.de/pub/cm/lgp30/.
    - Manuals and more program documentation: ftp://bitsavers.informatik.uni-stuttgart.de/pdf/royalPrecision/LGP-30/
    - History and repair of their original, working LGP-30: http://computermuseum.informatik.uni-stuttgart.de/dev_en/lgp30/lgp30.html.
- The Operation Manual and Programming Manual are specifically recommended as starting points for exploring the LGP-30:
    - Operation Manual ftp://bitsavers.informatik.uni-stuttgart.de/pdf/royalPrecision/LGP-30/LGP-30_Operations_Manual.pdf
    - Programming Manual ftp://bitsavers.informatik.uni-stuttgart.de/pdf/royalPrecision/LGP-30/LGP-30_Programming_Manual_Apr57.pdf

## 4.2 Schematics



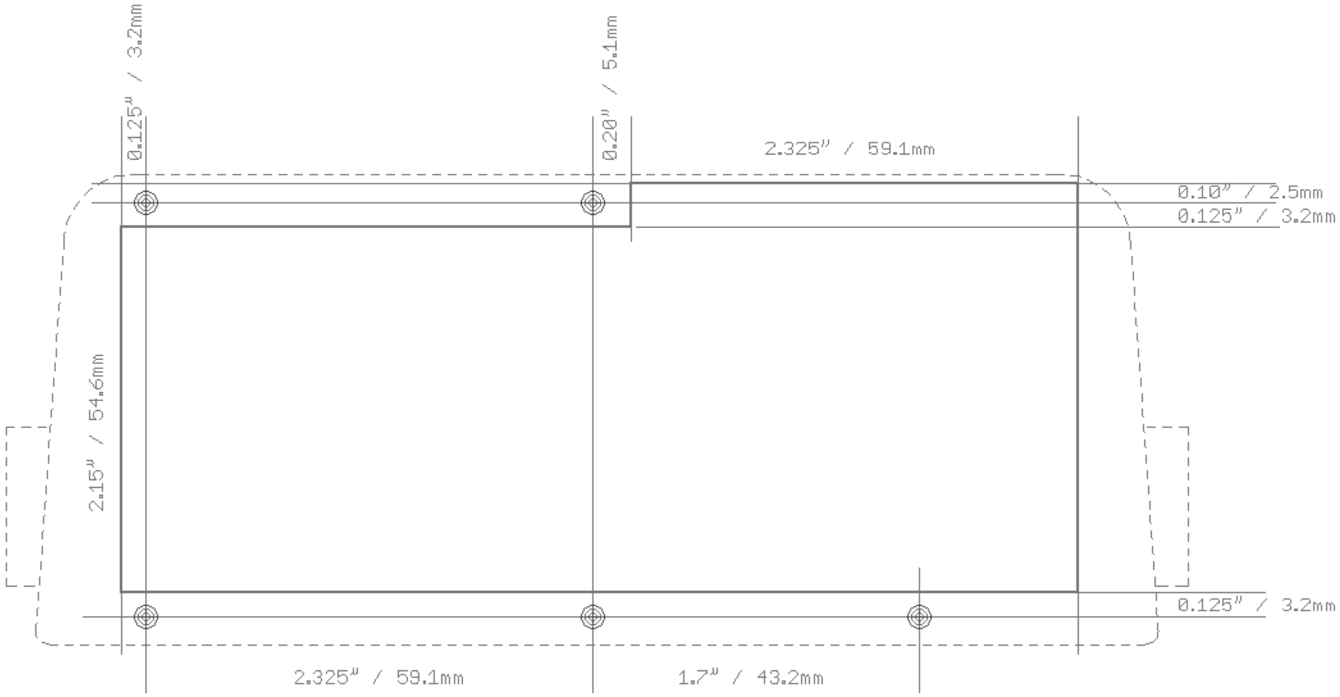


S6
BTCTRL
LTCTRL
R6
120
S5
B6BIT
L6BIT
R5
120
S4
BBP4
LBP4
R4
120
S3
BBP8
LBP8
R3
120
S2
BBP16
LBP16
R2
120
S1
BBP32
LBP32
R1
120
GND
S10
BEXEC
LEXEC
R10
120
S9
BFILL
LFILL
R9
120
S8
BCLEAR
LCLEAR
R8
120
S7
BSTART
LSTART
R7
120
GND
ENCBUT
SW1
EC12E_SW
ENCX
ENCY
EC12E_SW
SW1
GND
S13
BMANUAL
LMANUAL
R13
120
S12
BONEOP
LONEOP
R12
120
S11
BNORM
LNORM
R11
120
GND
S14
BCONN
LCONN
R14
120
S15
BREAD
LREAD
R15
120
S16
BCOMP
LCOMP
R16
120
S17
BSTOP
LSTOP
R17
120
GND
Populate either IC100 or JP100
JP100
USB_5V
ZEN
IC100
GND
C1
470µF
6.3V
GND
C2
100n
GND
C3
100n
GND
VDD
VDD
P4
Numato
Mimas
+5V
VCC
C4
100n
GND
VCC
C5
100n
GND
1
2
3
4
5
6
A
B
C
D
LittleGP-30
LGP-30 control panel
for Numato Mimas FPGA
jm 11/2016, 8/2017
LGP-30 wide rev6 USB
29.09.2017 08:03
Sheet: 2/2

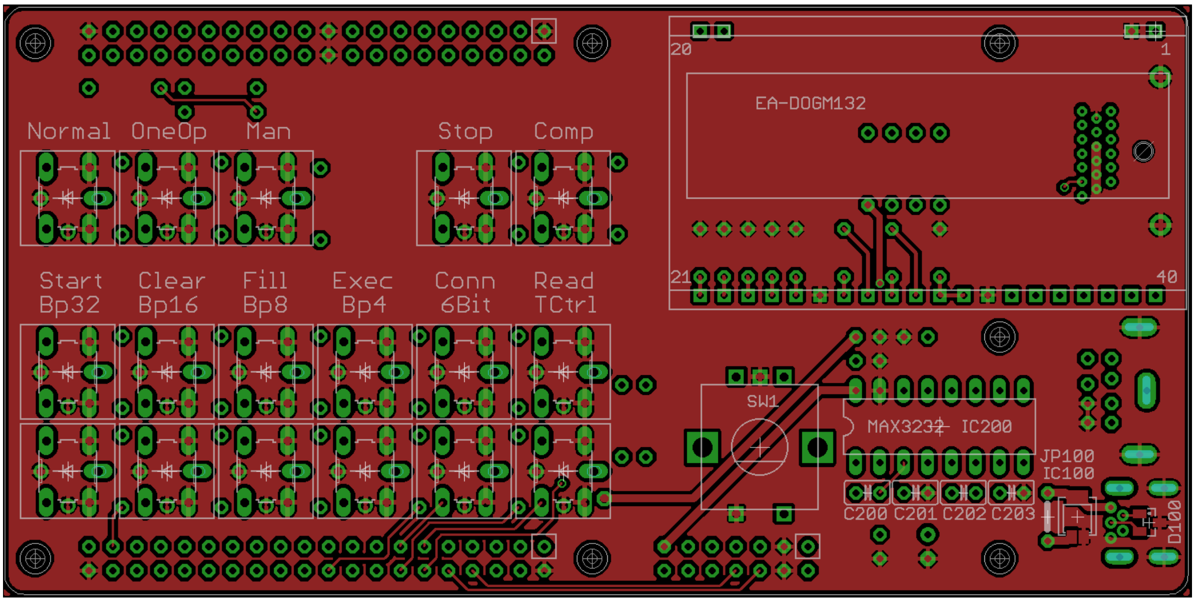
## 4.3 Front panel cutout

Suggested mounting holes and cutout for an enclosure-mounted board.

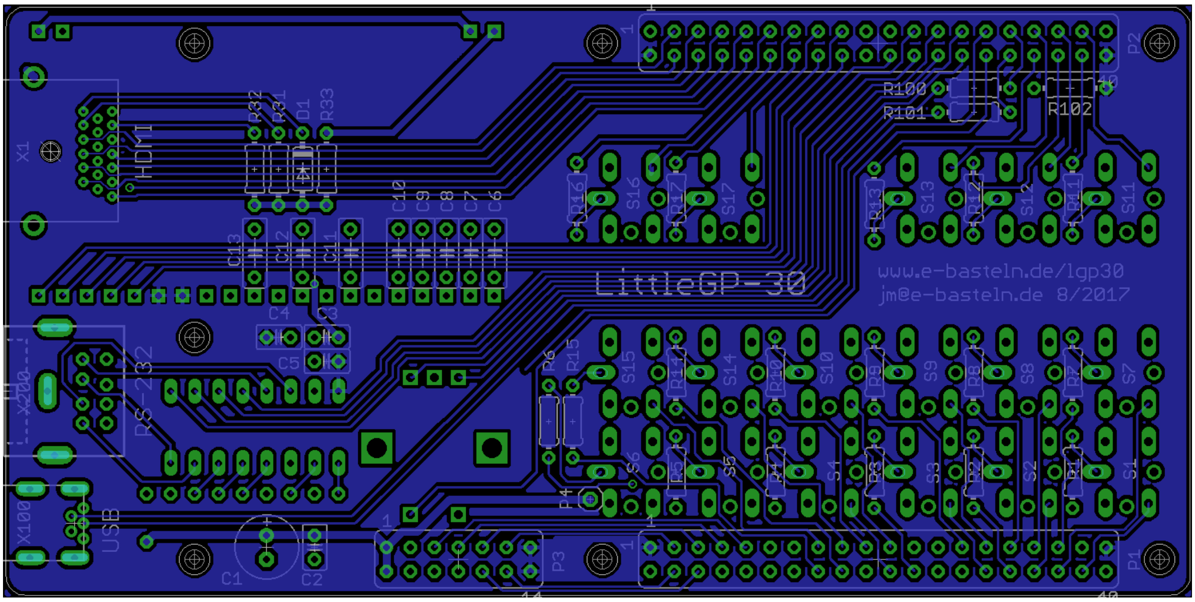
This drawing should be reproduced here at 1:1 scale, hence can be printed and used as a template to transfer dimensions to an enclosure.



## 4.4 PCB layout



*Top side*



*Bottom side*

## 4.5 Bill of materials

Part numbers refer to www.reichelt.de, and are for reference only. Their online catalog has further details and data sheets if any designation is ambiguous. (Catalog can be switched to English.)

| Ref | Quant | Item | Reichelt No. | Comment |
|---|---|---|---|---|
| | | | | |
| C1 | 1 | Electrolytic Cap 470µF 6.3V | NHG-A 470U 6,3 | 2.5mm spacing, 6..7mm diameter |
| C2-C5 | 4 | Capacitor 100nF, 2.5mm spacing | Z5U-2,5 100N | multilayer ceramic capacitor |
| C6 - C13 | 8 | Capacitor 1µF, 5mm spacing | Z5U-5 1,0µ | multilayer ceramic capacitor |
| R1 - R17 | 17 | Resistor 120 Ohm, 0.6W | METALL 120 | series resistors for LEDs, ~9mA |
| R31, R32 | 2 | Resistor 1k5, 0.6W | METALL 1,50K | I²C termination HDMI |
| R33 | 1 | Resistor 22 Ohm, 0.6W | METALL 22,0 | series resistor LCD backlight |
| D1 | 1 | Schottky diode BAT46, DO35 | BAT 46 | UF = 0,25V, 150 mA. For HDMI 5V line. |
| | | | | |
| IC1 | 1 | LCD, Electronic Assembly DOGM132-5 | EA DOGM132S-5 | transmissive, inverse, FSTN |
| LED1 | 1 | LED backlight for EA DOGM | EA LED55X31-G | green |
| | | | | |
| S1 - S15 | 15 | Illum. tactile button, 10*10mm | B3W-9000 Y1C | OMRON - 1 yellow LED, clear cap, low force |
| S16 | 1 | Illum. tactile button, 10*10mm | B3W-9002 G1G | OMRON - 1 green LED, green cap, high force |
| S17 | 1 | Illum. tactile button, 10*10mm | B3W-9002 R1R | OMRON - 1 red LED, red cap, high force |
| SW1 | 1 | Rotary encoder with push-button | STEC12E08 | ALPS EC12E24244 |
| (none) | 1 | Knob for 6mm flattened axis | KNOPF 10-150E | for encoder, optional |
| | | | | |
| X1 | 1 | HDMI connector | HDMI 090P | through-hole, 7+6+6 pins, 1.5mm pitch |
| P1, P2 | 1 | Pin header 2*40, 2.54mm pitch | SL 2X40G 2,54 | split into 2 headers, 2*20 each |
| P3 | 1 | Pin header 2*7, 2.54mm pitch | SL 2X10G 2,54 | optional. Part is 2*10, shorten to 2*7 |
| P4 | 1 | Pin header, 1 pin | MPE 087-1-002 | Mates with Mimas +5V socket. Use 1 pin. |
| (none) | 1 | Pin socket, 2 pins (Mimas +5V) | BL 1X05G7 2,54 | populate on Mimas board!<br>Use 2 pins only for +5V and GND |
| | | | | |
| *optional, for on-board USB* | | | | |
| X100 | 1 | USB jack Mini-B | USB BWM | horizontal, through-hole |
| R100, R101 | 2 | Resistor 39 Ohm, 0.6W | METALL 39,0 | Series resistors USB D+, D- |
| R102 | 1 | Resistor 1k5, 0.6W | METALL 1,50K | Pullup resistor USB D+ |
| D100 | 1 | PESD5V0U2BT (NXP) or similar | e.g. Conrad | Dual TVS diode SOT-23<br>(ESD protection, optional) |
| IC100 | 1 | ZEN056V130A24LS or ZEN056V115A24LS | e.g. Conrad | Polyzen supply voltage protection, optional |
| JP100 | 1 | (solder a jumper wire) | -- | (use if IC100 not populated) |

| Ref | Quant | Item | Reichelt No. | Comment |
|---|---|---|---|---|
| | | | | |
| *optional, for on-board RS-232* | | | | |
| IC200 | 1 | MAX3232CPE, DIL16 | MAX 3232 CPE | RS-232 level converter, 3.3V supply |
| C200-C203 | 4 | Capacitor 100nF, 2.5mm spacing | Z5U-2,5 100N | multilayer ceramic capacitor |
| X200 | 1 | Mini-DIN jack, 8 pins | EB-DIOS M08 | e.g. Assmann A-DIO-FS08, Lumberg TM 0508 A/8 |
| | | | | |
| *alternative switches (replacing the OMRON switches above)* | | | | |
| S1 - S15 | 15 | tactile button, yellow LED | TLL-63BY-S-V-B | Diptronics, available from rapidonline.com |
| S16 | 1 | tactile button, green LED | TLL-63BG-S-V-B | Diptronics, available from rapidonline.com |
| S17 | 1 | tactile button, red LED | TLL-63BR-S-V-B | Diptronics, available from rapidonline.com |
| | 17 | key cap (clear cap, ivory insert) | KTL-SAI | Diptronics, available from rapidonline.com |
| | | | | |
| *enclosure suggestions* | | | | |
| (none) | 1 | Hammond 515-0930 | www.tube-town.net/ | Aluminium, 21.80€ tube-town.net |
| (none) | 1 | Combiplast CP-18-34 | www.tme.eu | ABS grey, 5.95€ tme.eu |