

Q68

MAINBOARD USER'S MANUAL

Version 11/2017

Chapter 1: Overview

Key Features

The Q68 mainboard is a medium-performance Sinclair QL successor mainboard. It is based on a specialized chip, which includes both the CPU and the system controllers on a single integrated circuit. This chip includes a 68000 compatible microprocessor, video controller and most system components.

Like the original QL mainboard, the Q68 is a complete solution including graphics, peripherals and sound. It can be fitted into an available case. The Q68 does not need any parts from an existing QL, nor QL expansion or speedup cards.

Instead of the traditional QL mass storage devices, the Q68 has two SDHC card drives. The graphics remain compatible to the original QL, but add high colour and high resolution modes. Video output is suitable for contemporary monitors. Peripherals are modernized, while the overall system maintains a high degree of QL compatibility.

CPU

The microprocessor inside the Q68 is code compatible to the 68000, and runs at 40 MHz clock rate. In addition to the higher clock rate, many instructions are executed in less cycles than by a 68000. Unlike the 68000, it has a 32-bit address bus. If desired, the Q68 can slow down the system speed by hardware.

RAM

The Q68 has 32 MB of on-board synchronous dynamic RAM (SDRAM). 4 MB of this memory is dedicated screen memory and not available for other purposes. Additionally, it has a small 12 KB static RAM (SRAM) area which runs at higher speed.

ROM

The Q68 has a 96 KB ROM area at the start of the address range. This area is used by the integrated bootloader, and also as ROM emulation for a loadable operating system.

Graphics

The Q68 has an integrated graphics controller with 16 bit colour and highspeed FIFO RAM. It works non-interlaced in all modes, using dual / triple scan for the low resolutions.

Video modes

0. Standard QL MODE 8 256x256 pixel (8 colours)
1. Standard QL MODE 4 512x256 pixel (4 colours)

2. Q40/Q60 highcolour mode 512x256 pixel (65536 colours)
3. Q40/Q60 highcolour mode 1024x512 pixel (65536 colours)*

4. High resolution QL MODE 4 1024x768 pixel (4 colours)
5. Aurora mode 1024x768 pixel, 256 (colours)*

6. Medium resolution highcolour mode 512x384 pixel (65536 colours)
7. High resolution highcolour mode 1024x768 pixel (65536 colours)*

In all of the above modes, the output on the VGA connector is provided as a 1024x768 VESA standard signal for maximum PC compatibility.

Note: Due to memory bandwidth limitations, using the video modes 3, 5 and 7 slows down the system significantly. The loss of speed is approximately 37%, 28% and 56%. These modes are not recommended when speed is relevant.

For full speed, use the Standard QL video modes 0 or 1. In the remaining modes 2, 4 and 6 only a minimal speed loss occurs in benchmark figures. This is not noticeable during normal use. These modes are recommended as a compromise between colours / resolution and work speed.

Monochrome display

In the standard QL modes and the high resolution mode 4, the Q68 offers special monochrome display functions, even if you attach a colour monitor. This is implemented in hardware and does not require any software changes.

You can make the Q68 screen look like a QL connected to a historic monochrome monitor in **black & white**, **amber** or **green** colour. The Q68 colour translation unit is simply configured by

POKE \$1C01C, x

Where x is 0 for default colours, 1 for black & white, 2 for amber, 3 for green.

SDHC Drive Interfaces

The Q68 contains hardware for two independent SDHC card interfaces. Fullsize card slots are provided for easier handling and copying of data between two cards.

Keyboard and Mouse Interface

The Q68 comes with ports for PS/2 keyboard and mouse. The mouse scrollwheel is supported.

Sound

The Q68 provides a stereo sampled sound system. Digital-analog conversion uses 20kHz sample rate and hardware FIFO buffering for up to 1024 samples. Combined line / headphone output on jack socket.

Realtime Clock

- DS1374C-33 real time clock, typical oscillator accuracy approximately 10ppm at 25°C
- Longlife super-capacitor for oscillator supply backup during power-off
- Integrated recharge capability

I/O Ports

- Serial port with up to 115200 (460000) Baud, external 9 pin SubD connector
- I²C Interface, internal 4 pin connector

Extension Bus

The Q68 mainboard contains a parallel extension bus with 8 bit data/address bus and control signals for the possibility of add-ons.

Ethernet

The Q68 has an on-board CP2200 ethernet controller with integrated IEEE 802.3 MAC and 10 Base-T PHY. At the time this manual was written, no Q68 specific software support for the CP2200 was released.

Board Size

The Q68 mainboard has a size of 8x10 cm and can be mounted in an available case.

Power Supply

The Q68 is powered by a standard regulated 5V supply with 2.1 mm coaxial plug.

QL Compatibility

QL screen modes 4 and 8 are directly hardware compatible. The 50 Hz frame interrupt and those QL registers which are still useful with the modernized peripherals are available in hardware. Memory map and interrupt handling are similar. Microdrives and the original (unreliable) serial ports are not supported. The corresponding registers are currently not implemented.

Unpacking & Handling instructions

The Q68 mainboard package should contain the following items:

1. The mainboard
2. This User's Manual
3. Splitter cable for PS/2 mouse (optional)
4. SDHC card with pre-installed operating system
5. Power supply (optional)

The Q68 mainboard is sensitive to static electricity. Please do not remove the mainboard from its packing until you are ready to install it. Please do not touch the integrated circuits when handling the mainboard.

To charge the buffer of the realtime clock, it is recommended to keep the Q68 powered on for more than 24 hours after the first use, or if it has not been used for a long time. For this purpose, it does not matter if any operating system is running. The realtime clock is buffered by a large capacitor, so no specific procedure has to be followed to charge it.

Important: On some (but not all) PS/2 splitter cables from PCs, the labels and colours for keyboard and mouse do not match the Q68. In this case, the Q68 ROM Loader may stop without error message, so please swap and retry.

Chapter 2: Hardware

Mainboard

Please make sure that the power supply is turned off before making any connections to the Q68 board.

When attaching any system components to the mainboard via the mainboard connectors, please pay attention to the correct orientation. In the figure **Mainboard Component Locations**, the number 1 pin locations of the *internal* connectors **Power**, **Extension Bus**, **I²C** and **JTAG** are marked in darker grey. The *external* connectors only fit if they are turned into the correct orientation.

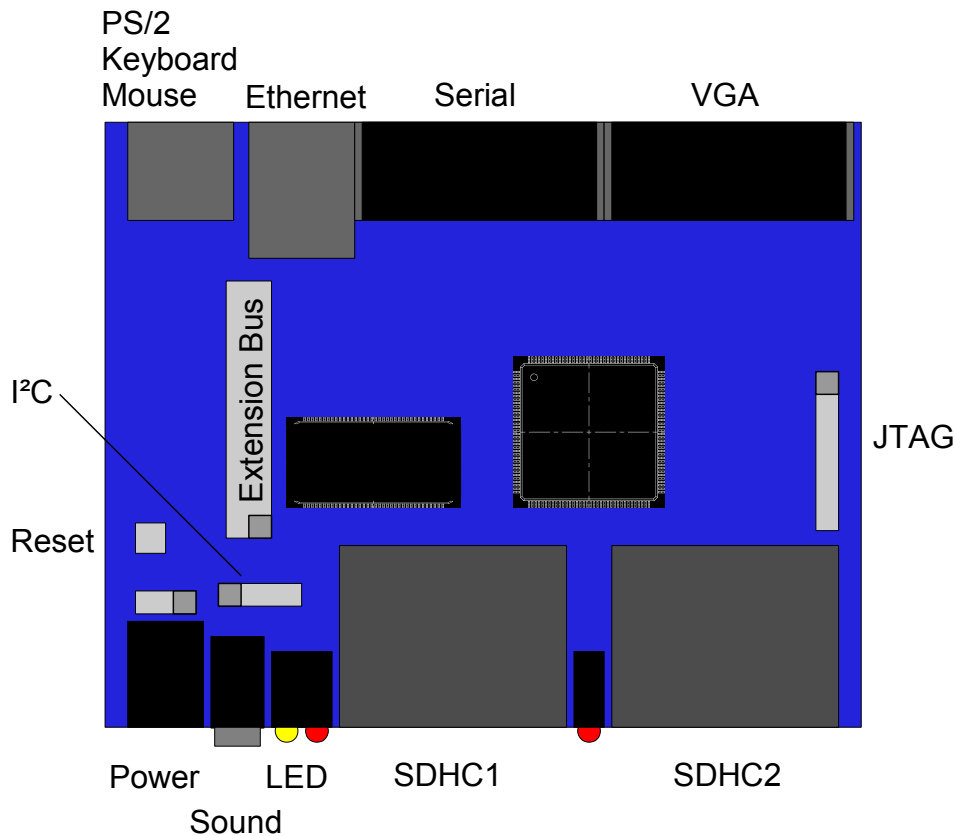


Figure: Mainboard Component Locations

Power

The supply current of the Q68 mainboard itself is only about 160 mA. Nevertheless it is recommended to provide at least 1A for the devices that are usually attached, for example keyboard, mouse and SDHC cards.

External Coaxial Connector

DC plug with with 2.1mm inner diameter.

Contact	Description
Inner	+5V stabilized power input, 1A recommended
Outer	Ground

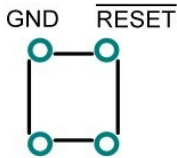
Internal Pin Header

This 3 pin header primarily allows to connect a switch between the external power plug and the Q68 internal power. In order to supply the Q68, pins 2 and 3 of this header need to be connected, using a switch, jumper, or solder connection. It is also possible to connect the power ground on pin 1, for example to add further internal components.

Pin	Description
1	Ground
2	+5V external power from coaxial plug
3	+5V internal power

Reset

It is possible to solder an optional Reset button at this location. Ground and Reset signal (low-active) are available on the upper two pads. A reset button is not required to operate the Q68. Normally, only a power switch is used to restart the system.



Extension Bus

The mainboard contains a parallel extension bus with 8 bit data bus and 8 bit address bus. Providing chip select, data strobe, read/write, interrupt request and reset lines. The signals can be used to attach external circuitry with 3.3V TTL or CMOS signals.

In the Q68 memory map, the extension bus is located at \$1D800 ... \$1DFFC. The external lines A0...A7 are internally connected to A2...A9 of the CPU, accessible every 4th byte (longword-aligned).

The extension bus also provides a hardware signal to prepare for singlewire networking (QLNET) with few external components.

Pin	Signal	Description
1	GND	Ground
2	+3.3V	Power output 3.3V, continuous current max. 200 mA, short peaks max. 400 mA
3	$\overline{\text{RESET}}$	Reset output
4	QLNET	Bidirectional singlewire networking, requires external components
5...12	D0...D7	Bidirectional data bus
13...20	A0...A7	Address bus output
21	$\overline{\text{CS}}$	Chip Select output
22	$\overline{\text{IRQ}}$	Interrupt input
23	$\overline{\text{DS}}$	Data Strobe output
24	$\overline{\text{R/W}}$	Read/Write output

I²C Interface

Provides an internal I²C master-only interface. It is also connected to the Q68 realtime clock DS1374, which occupies the binary I²C bus address 1101000. The Q68 supports I²C communication by two simple bitwise registers:

I2C_SCL at \$1C1C0: I²C Serial Clock output, 0=Low, 1=High

I2C_SDA at \$1C1C4: I²C Serial Data input / output, 0=Low, 1=High (open-drain pullup)

The Q68 normally expects the connected I²C slave devices operated at 3.3V. By an internal weak clamp diode, usage with 5V devices is also possible, provided the overall pullup resistance does not fall below 470 Ω.

The power output on the I²C bus connectore is shared with the extension bus. The maximum supply current specified there must not be exceeded by the sum of both currents.

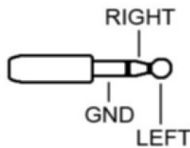
Pin	Description
1	Ground
2	Power output +3.3V
3	I ² C Clock
4	I ² C Data, internal 10 kΩ pullup resistor to 3.3V

JTAG

This connector is for manufacturing use only.

Sound

Audio output with 3.5mm stereo jack socket. Designed for headphones, speakers or line inputs with an impedance of 32 Ω or more. Lower impedance will not damage the mainboard, but may not reach normal loudness. The corresponding jack plug should have signals assigned as follows:



LED

Programmable LED in yellow colour. This is **not** a power supply indication, and can be freely used.

POKE \$1C100, 0 → LED off

POKE \$1C100, 1 → LED on

SDHC Sockets

The Q68 uses SDHC memory cards for mass storage. These must be **SDHC** cards, simple SD cards are not compatible. There are two sockets into which you can insert the cards. To the left of each socket is a red colour LED to display read or write activity.

SDHC1 is the primary mass storage socket of the Q68. The Q68 will try to boot from a card inserted here. SDHC2 is the secondary socket and may be left empty.

PS/2 Keyboard and Mouse

6-pin female Mini DIN connector.

Keyboard and mouse ports are separate, but to save space, a combined connector with signals for both ports is provided. A standard PS/2 keyboard can directly be plugged into the mainboard. If a mouse is also used, an additional PS/2 splitter cable is required.

Note: Many (but not all) splitter cables for PC have the mouse signals where the Q68 has keyboard. **In this case, the connectors must be swapped.**

There is no risk of damage when mouse and keyboard are interchanged, they just won't work on the Q68 then. The ROM Loader might stop without error message if keyboard and mouse are mismatched.

Mouse versions with scrollwheel are supported, but the LEDs inside the keyboard are currently not used.

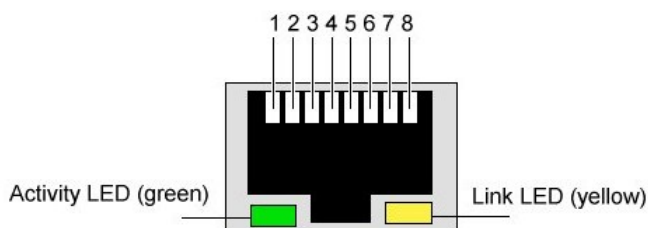
Some USB mice may also work with a USB/PS2 adapter, but only if they also support PS/2 electrically.

Pin	Description
1	Keyboard Data
2	Mouse Data
3	Ground
4	+5V
5	Keyboard Clock
6	Mouse Clock

Ethernet

The Q68 ethernet port uses a RJ45 8-pin connector (tab down). It has automatic polarity detection and correction. The 10 Base-T ethernet port is fully compatible with 100/1000 BASE-T networks.

Pin	Signal
1	TX+
2	TX-
3	RX+
6	RX-
4,5,7,8	Not connected



Serial Port

A serial port is provided on a PC standard male 9 pin SubD connector. It uses RS-232 compatible transceivers capable of up to 460 kBaud in hardware. Software support was tested up to 115200 Baud.

Please note that the Q68 does **not** use the **original QL** SER1 / SER2 pinout. The hardware delivers serial mouse power supply. The Q68 operates at a 7 or 8 bit rate. There is no parity nor hardware handshake.

Pin	Description
2	Receive Data (RxD) Input
3	Transmit Date (TxD) Output
4	Data Terminal Ready (DTR) Internally pulled up for serial mouse
5	Ground
7	Request To Send (RTS) Internally pulled up for serial mouse

VGA

VGA graphics output is provided by a PC standard 15 pin HDsub VGA connector.

Output signals are optimized for flatscreen monitors or HDMI converters which support 1024x768 VESA standard input at 60 Hz vertical refresh rate.

Analogue cathode ray tube (CRT) monitors with at least 48 kHz horizontal frequency can also be attached, but may show some flicker. This is due to the relatively low vertical refresh rate, which allows higher video signal quality in connection with digital monitors.

Some HDMI monitors might be connected to this with a suitable VGA/HDMI converter.

Pin	Description
1	Red
2	Green
3	Blue
11	Monitor ID Bit 0
13	Horizontal Sync
14	Vertical Sync
5,6,7,8,10	Ground

CPU Cycle Timer

The unsigned longword register at \$1C060 contains a high resolution timer in 25 ns units, equivalent to 40 MHz CPU clock cycles. It rolls over from $2^{32}-1$ to 0 about every 107.3 seconds. However, if two timer values are directly subtracted with a longword instruction, the time difference is always correct, as long it is shorter than 107.3 seconds.

Chapter 3: Software

Mass Storage

The Q68 relies on SDHC memory cards for mass storage. For use with the Q68, they must be partitioned and the first primary partition must be in FAT32 format.

The different files the Q68 needs, must be put into that FAT32 partition, which should be possible by copying from any Linux, Windows or MAC machine. However, make sure to understand the special precautions for these files as described in the following chapter.

Q68 Container Files

The Q68 is not an emulator which indirectly accesses files by the underlying PC or MAC operating software. Instead, it is a QL-native hardware device, and files in the FAT32 partition serve only as containers for direct lowlevel access by QL-native drivers.

To access these containers efficiently, the Q68 expects the container files not to be fragmented in the FAT32 file structure. Once the Q68 has found the beginning at a container file, it assumes the rest of that container file lies in continuous sectors on the card. Moreover, these container files must be located within the first 16 root directory entries of the FAT32 partition.

The best way to achieve this, is to make sure that before writing the the container files, the card is freshly formatted. Then write each container file immediately after formatting the card.

The container files follow a specific naming scheme. The file name of a container file must have 1 to 8 characters, followed by a decimal point and a three letter extension. Names for ROM or RAM images (usually holding an operating system) are machine specific. Therefore they have the „**Q68_**“ prefix and the extension „**SYS**“, indicating a system file. Containers with a *QLWA* filing system use the extension „**WIN**“.

At the time of writing this manual, the Q68 comes with the SMSQ/E operating system. SMSQ/E can either be in a RAM image container named „**Q68_RAM.SYS**“ or in a small boot filesystem container named „**Q68_SMSQ.WIN**“. The main filesystem for SMSQ/E is recommended to be named „**QLWA.WIN**“ by default.

While the „WIN“ extension reminds of the „QXL.WIN“ filename known from emulators, the Q68 demands continuous physical sectors - in contrast to those emulators. Therefore, the string „QLWA“ is used, which was used in the past as SMSQ/E partition type name for harddisks in the Atari und Q40/Q60 machines.

Preparing an SDHC Card

This cannot be done on the Q68 itself yet. To prepare a new card, the following procedure is recommended:

- Linux: Use gparted or the command-line version parted.
Then format the entire disk as FAT32, or at least the first partition if you have more.
- Windows: Download and install the SD Association's Formatting tool from:
https://www.sdcard.org/downloads/formatter_4/eula_windows/
Then open the installed tool, set "Format Size Adjustment" to „On“ and click „Format“
- MAC: Download and install the SD Association's Formatting tool from:
https://www.sdcard.org/downloads/formatter_4/eula_mac/
Then open the installed tool, select "Overwrite Format" and click „Format“

Now you can write the container files. First the operating system container, then the QL filesystem container.

Q68 ROM Loader

The Q68 has a 96 KB ROM area at the start of the address range. This area has a double usage. At power-up, it is first used by the internal hardware ROM, containing a small bootloader firmware. This bootloader will then, as a second step, load the actual operating system from SDHC card into an emulated ROM area at the same location.

The emulated ROM behaves like a physical ROM in a QL, and is booted by an actual hardware reset. It is typically used for ROM-based operating system like QDOS Classic or Minerva*. It is also possible to use general, non QL-specific 68000 system ROMs.

In cases where the 96 KB ROM area is not large enough, the bootloader can optionally load an operating system directly into RAM at \$30000 and jump there without a reset. This is typically used for the SMSQ/E operating system.

Before starting the operating system, the ROM loader will do some minimal system checks and display the result on screen. These are:

- RAM Check: Testing main memory in steps of 1 MB. For each successful test passed, a dot is printed on screen. Otherwise an error message is printed.
- Realtime clock: Time and date from the buffered realtime clock are read over the Q68 I²C bus and placed into the QL-specific *PC.CLOCK* register. The value of this register is printed on screen. If the clock is correctly set, one can expect a 10 digit decimal number here. A smaller number indicates a possible loss of data by a discharged buffer capacitor.
- PS/2 mouse: Presence of a PS/2 mouse is checked and the result printed on screen. If the mouse contains a scrollwheel, it will be displayed as an *extended* PS/2 mouse, otherwise as *standard* PS/2 mouse.
- SDHC card: Presence of an SDHC card in the first slot is checked. If not successful, the message „No valid SDHC card found“ will be displayed and the Q68 waits to possibly load a ROM binary over the serial port.

If an SDHC card was detected, it is first attempted, to boot from ROM image, then from RAM image, and at last from an SMSQ/E boot filesystem.

Booting from ROM image

If a file named „**Q68_ROM.SYS**“ is present on the SDHC card, it will be loaded at address \$0 to the Q68 ROM emulation area. The maximum allowed ROM length is 96 KB. If the file is present, but too long, an error message will be printed and booting from serial port will be attempted.

After the ROM image was loaded, the CPU will automatically be restarted by a 68000 hardware reset. Therefore, like a physical ROM, the image file must contain the 68000 vector table, especially the initial supervisor stackpointer at address 0 and the initial program counter at address 4.

Booting from RAM image

If a file named „**Q68_RAM.SYS**“ is present on the SDHC card, it will be loaded at hexadecimal address \$30000 to the Q68 RAM. There is no length limitation except the amount of available RAM.

After the RAM image was loaded, the CPU jumps directly to the code at hexadecimal address \$32000 in supervisor mode. No reset is performed, the CPU registers are not initialized, and the ROM area is not changed. If an operating system is booted from RAM image, its code has to do all initializations or changes of the vector table by itself.

Booting from SMSQ/E filesystem

If a file named „**Q68_SMSQ.WIN**“ is present on the SDHC card, the Q68 will at first check whether it contains a valid SMSQ/E filesystem image. If successful, the Q68 will then look up the first file inside the root directory of this SMSQ/E filesystem image. It will then load the contents of this „inner“ file at hexadecimal address \$32000 into RAM. Booting then continues like directly from RAM image.

*These older operating systems are not distributed with the Q68, since SMSQ/E with much better features has become available.

Booting from serial port

If no ROM or RAM image is found, or no valid SDHC card is present, the Q68 will print the message „Load ROM from SER“ and attempt to boot from serial port. A fixed baudrate of 115200, no parity, no hardware handshake is expected.

The remote station must first send a header of 4 Bytes to the Q68 which contains the length of the following data stream as a big endian 32 bit unsigned integer. Then it sends the data. If the Q68 receives the data correctly, it will respond with a byte containing the ASCII code 'Q', otherwise the ASCII code 'E' for error. During the transfer, the programmable LED of the Q68 will blink.

The serial data stream will be loaded at address \$0 to the Q68 ROM area and started by a hardware reset like a ROM image from SDHC card.

Q68 Clock Utility

Under SMSQ/E, the Q68 clock can be set by the **SDATE** command. But without further action, this setting would be lost after the power supply is turned off.

In order to keep the new clock setting while the machine is powered off, it is required to write it back to the buffered real time clock chip of the Q68. This is done by executing a clock utility program named „Q68SETRTC“ which should have been distributed with the Q68.

For example, if the „Q68SETRTC“ program is in the root directory, one would type:

EW WIN1_Q68SETRTC

It is sufficient to do this once after a new clock setting was made.

Chapter 4: Memory Map

Below you can see the QL and the Q68 memory map as an overview about the physical hardware locations. More detailed documentation for developers is also available on request, but is not included in this User's Manual.

Address(hex)	QL	Q68
0000 0000 0000 BFFF	Internal ROM 48K	ROM 96K (Normal operation: ROM image from SDHC card or SER. Powerup: 4K Flash)
0000 C000 0000 FFFF	External ROM 16K	
0001 0000 0001 7FFF	External I/O 32K	
0001 8000 0001 8FFF	Internal I/O 16K (QL registers)	Internal I/O 4K (QL registers)
0001 9000 0001 BFFF		Fast RAM 12K (Powerup: 12K Flash)
0001 C000 0001 CFFF	External I/O 16K	Internal I/O 4K (LED, PS/2, I ² C, SER, SDHC, Sound)
0001 D000 0001 FFFF		Extension Bus 12K (Ethernet and extension bus)
0002 0000 0002 7FFF	Screen 0 32K	Screen 0 32K
0002 8000 0003 FFFF	RAM 96K	RAM 28640K (28 MB)
0004 0000 000B FFFF	Expansion RAM 512K	
000C 0000 000F FFFF	Peripheral ROMs 16 x 16K	
0010 0000 01C0 0000		
FE80 0000 FEC0 0000		
FF00 0000 FFFF FFFF		Screen 1 4096K (4 MB)
		Internal I/O 16384K (Q40 compatible registers)

Chapter 5: Copyright and disclaimer

The Q68 hardware design, programmable logic and bootloader firmware are Copyright Peter Graf. The programmable logic also contains the TG68K.C core, which is Copyright Tobias Gubener under the terms of the GNU Lesser General Public License as published by the Free Software Foundation.

The Q68 is a hobby project intended for retro-computing and fun purposes, not to process important data.

It is distributed in the hope that it will be useful, but without warranty of merchantability or fitness for a particular purpose. No responsibility is accepted for the loss of data or consequent damage of any kind resulting from the use of this design.

All specifications are subject to change without notice. No responsibility is assumed for inaccuracies or errors.