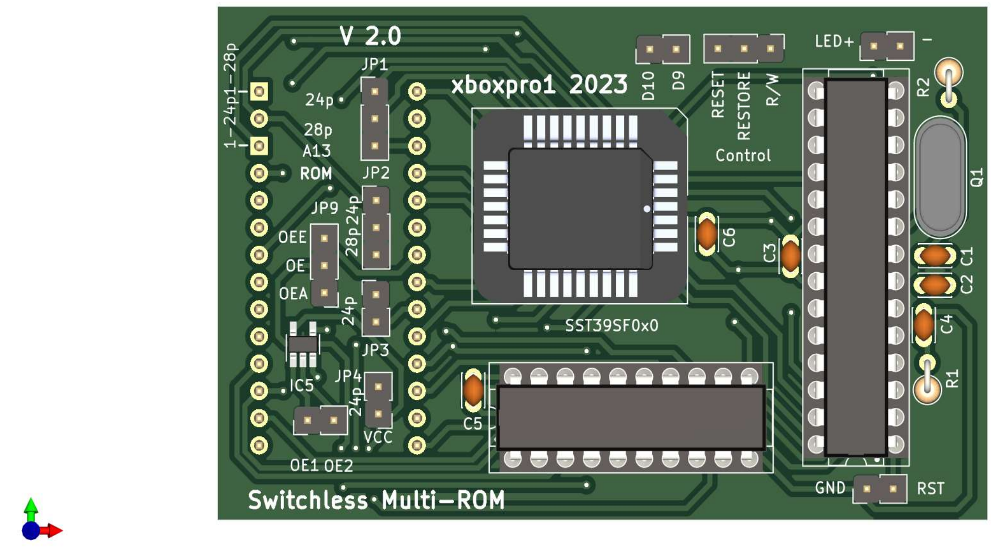
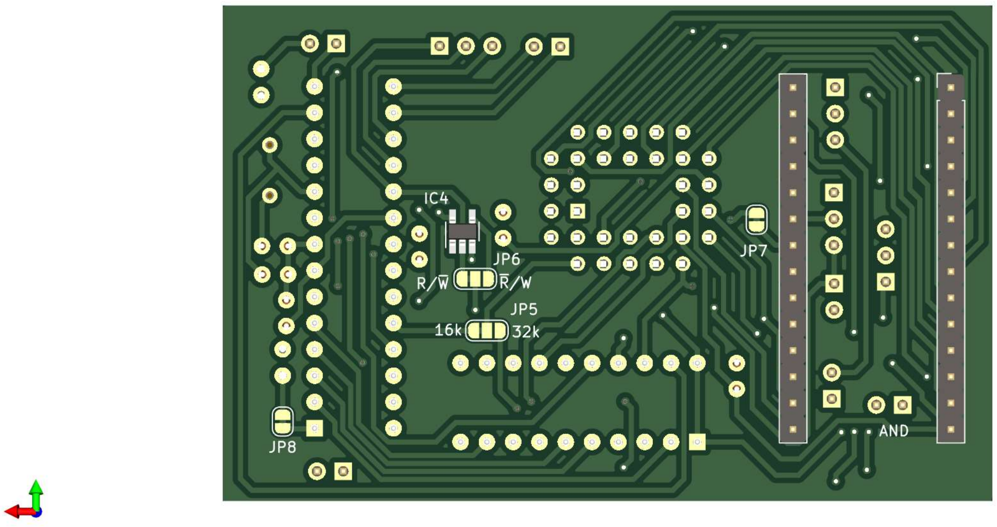
CBM Switchless Multi ROM

for 2364 & 27128 & 27256 ROMS





Hardware Info

Table of Contents

Functional Description

At power on, the AVR microcontroller (referred below as MCU) in the Switchless mini Kernal switch holds the reset line active-low while it reads address 0 from its internal EPROM to find out which Kernal ROM it should select, sets the A13-A18 address lines on the flash chip accordingly and releases the reset line to make sure the CBM computer is booted using the correct Kernal ROM image.

If at any time, Restore key is pressed and held for a few seconds the MCU will detect this and switch to Kernal ROM 0 which is the Switchless mini Kernal menu image.

When the user selects a Kernal image from the menu, the Switchless mini Kernal writes a predefined switching command to the data bus along with a byte that indicates the selected image number. It does this over and over.
Example command: CBMROM64#1 where 1 is a byte with value 0x01 that tells the Switchless mini Kernal that we want to switch to Kernal ROM image 1.
The Kernal switcher now captures the command from the data bus, writes the selected Kernal number to the MCU EPROM for future use and it sets the address lines A13-A18 accordingly. It then resets the computer using the new Kernal image by pulling the reset line low. The computer will continue to start up using this new Kernal ROM until a new choice is made.

Even at a clock speed of 20 MHz, reading and analyzing data on a 1MHz data bus can be quite a challenge so the data on the computer data bus is buffered in the Switchless mini Kernal switcher by a 74HCT273 flip-flop that is clocked by the write signal (r/w) from the CBM CPU. By clocking the data with the r/w signal we filter out irrelevant bus traffic and the flip-flop holds the relevant data until the next r/w cycle. This gives the MCU a little more time to read the data from the flip-flop.

Switching C64 and C128 Kernals

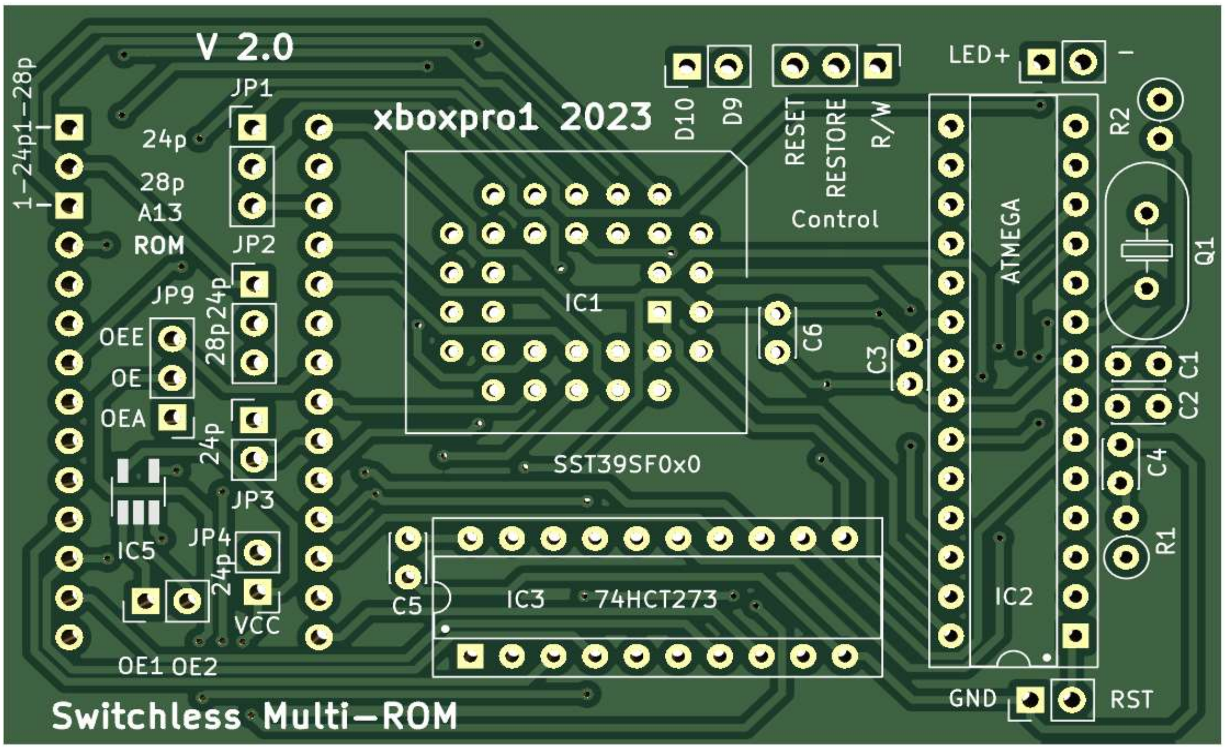
Hold the Restore key for about three seconds until the Kernal menu is activated. Once in menu you can use cursor up/down to move the Kernal ROM you want to switch to and press Return. The switch resets the computer with the new Kernal ROM and saves your choice for next power-on.

Switching C64 Kernal in a C128

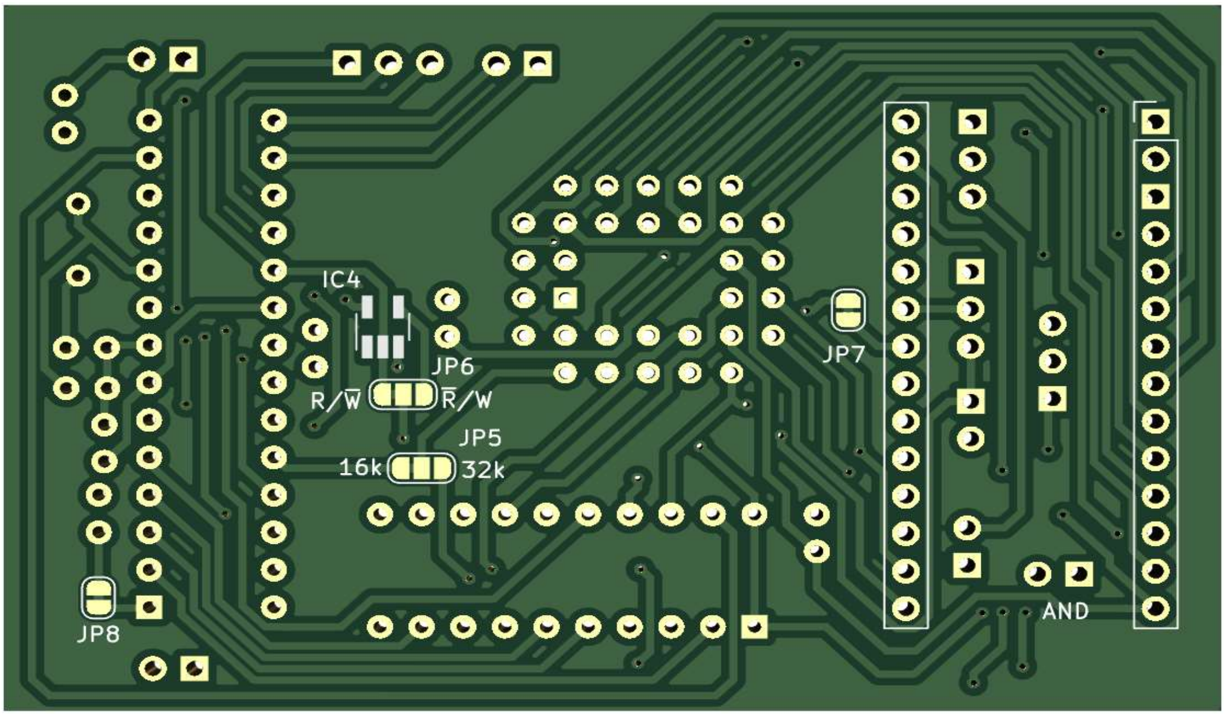
When switching the C64 Kernal in a C128, hold the CBM and Restore keys for about three seconds until the C64 Kernal menu is activated. Keep holding the CBM key while selecting a Kernal with the cursor up/down and Return keys. While pressing Return you should hold the CBM key until the reset is done to be taken to C64 mode. Otherwise you will end up in C128 mode after switching. For the C64 Kernal you have to do a GO64 or holding CBM and Restore key again. The switcher saves your choice for next power-on.
In the a C128DCR you can use either C128 or C64 Kernal menu to switch the Kernal ROM but both C64 and C128 Kernal ROM will be switched at the same time.

PCB

Front side



Back side

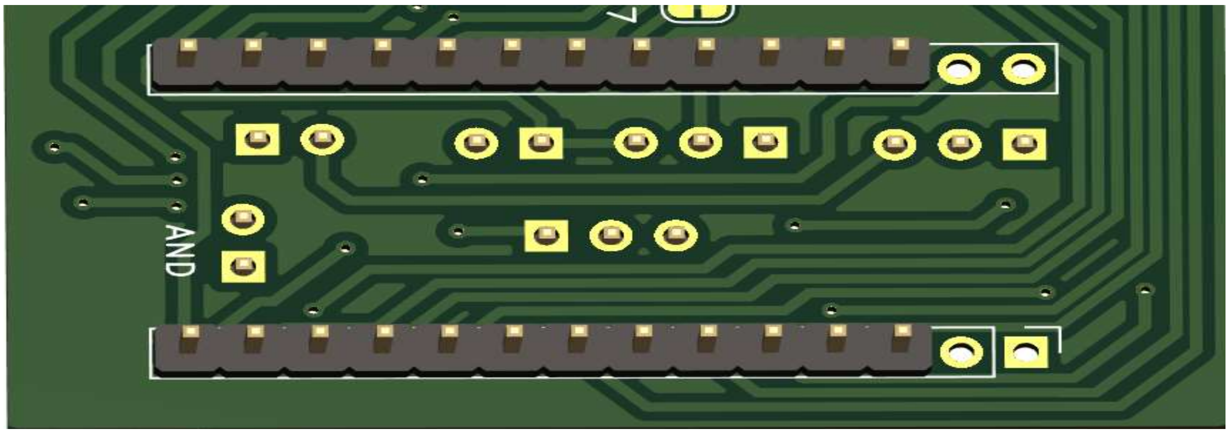


PCB Soldering

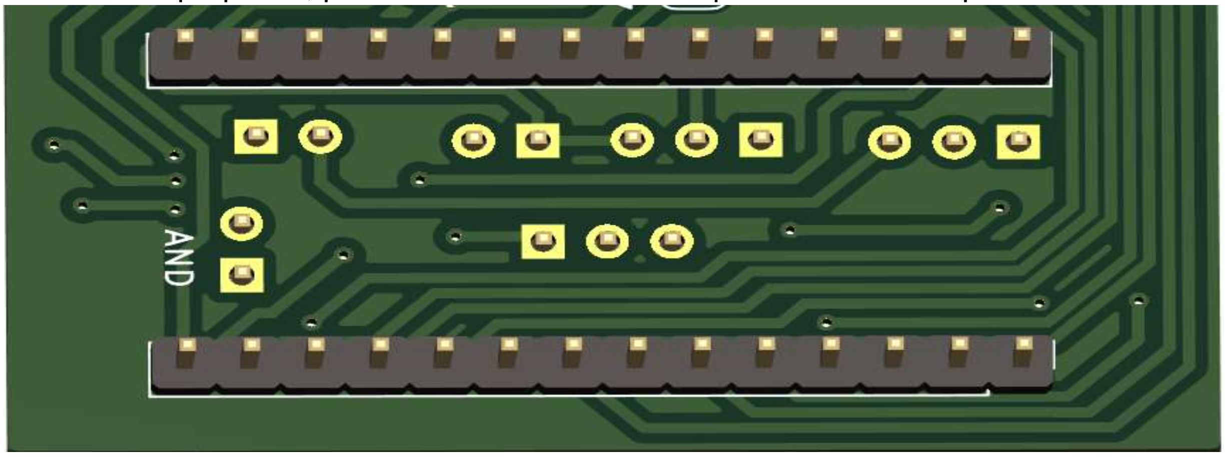
Recommended soldering order, first the SMD parts, then the two machined pin headers for the CBM ROM socket and then the rest in any order.

SMD
If you want to use the Kernal ROM switcher in a CBM C128DCR, 32k ROM, the signal r/w must be inverted. For this, the SMD IC4 must be soldered on the PCB back side. SMD IC5, JP9, the OE1/OE2 and D9/D10 connectors on the PCB front side are not needed unless you want to experiment with the Switchless multi ROM.

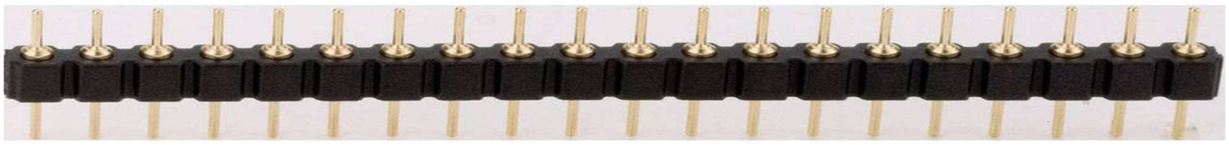
Pin headers
If you only want to use the Switchless multi ROM in a CBM computer with an 8k, 24-pin ROM. You just need to solder the machined pin headers in 24-pin mode.



For all other purposes, please solder the machined pin headers in 28-pin mode.



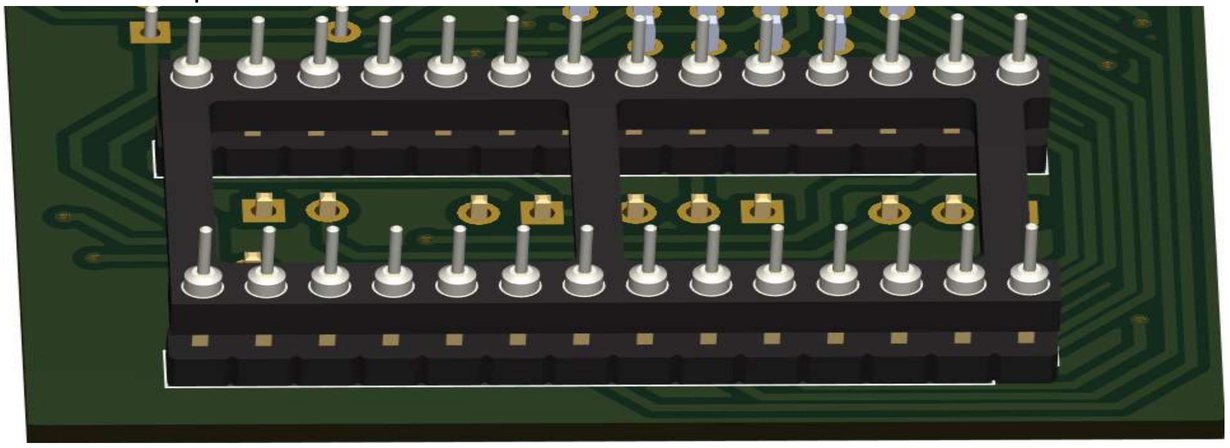
A machined pin header is round not square. Image below is a machined pin header.



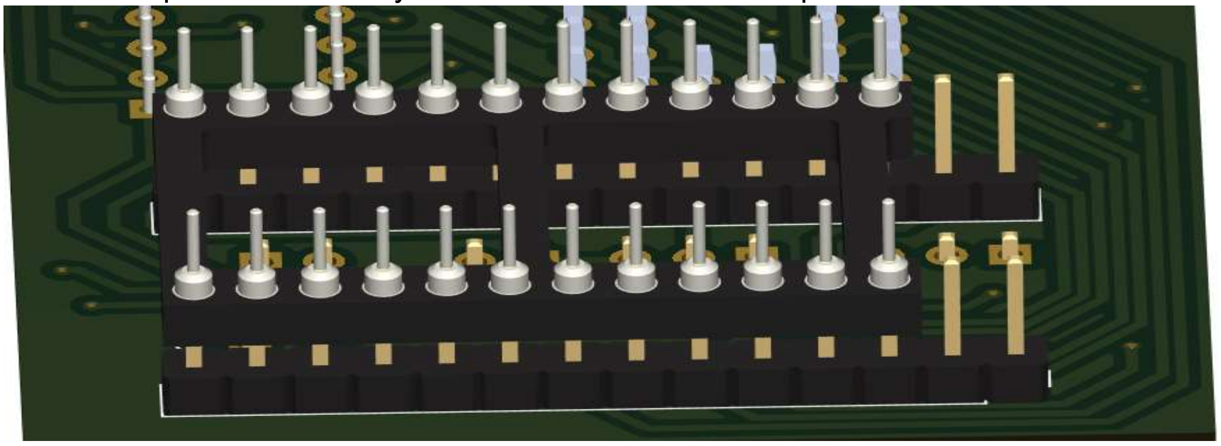
ROM Socket

It is most likely that an additional machined socket for the ROM socket is required to increase the gap between a CBM mainboard and the Switchless multi ROM PCB.

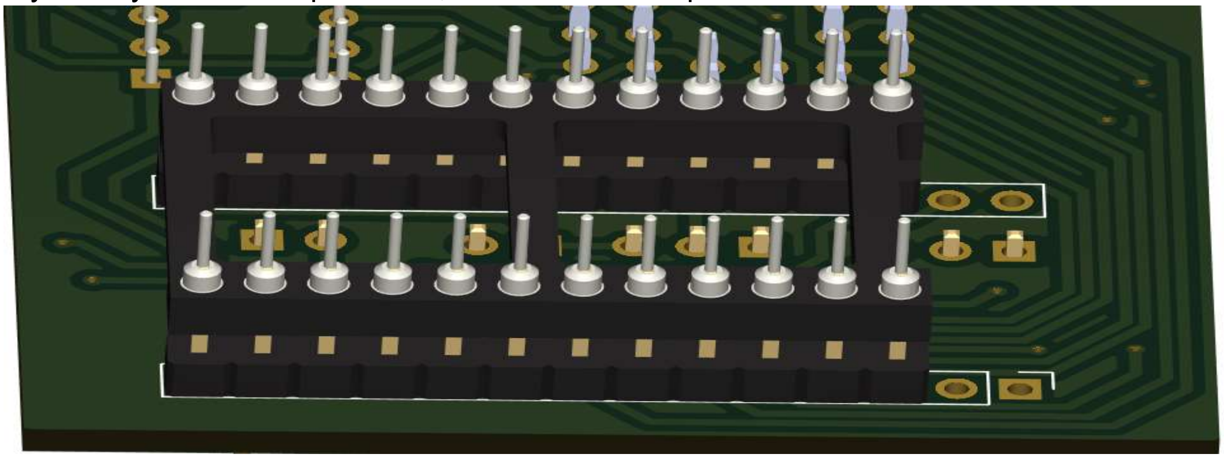
Additional 28-pin machined socket.



Or for a 24-pin ROM socket you can use an additional 24-pin machined socket.

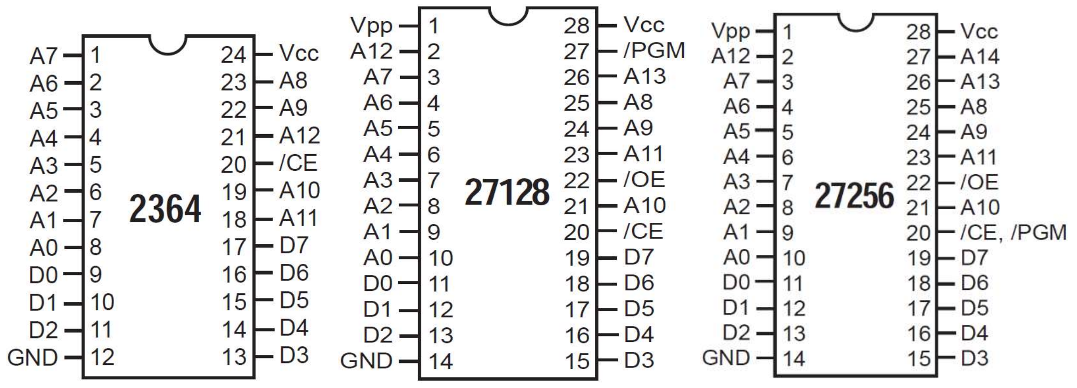


If you only use the 24-pin mode, an additional 24-pin machined socket is sufficient.



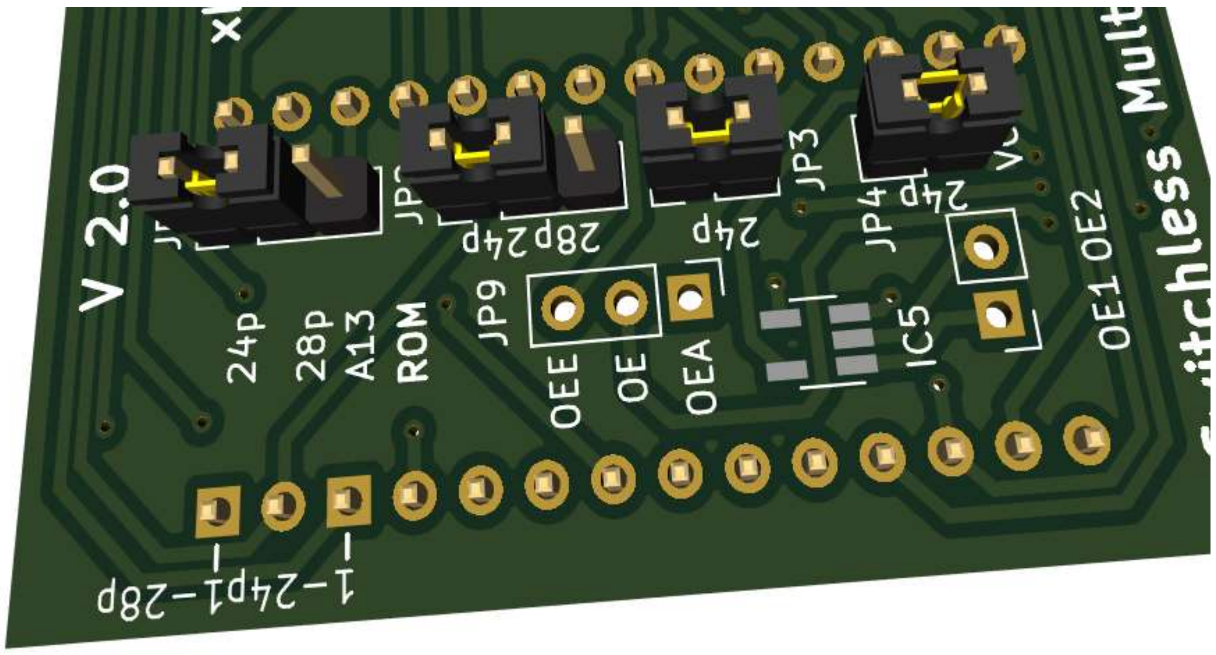
Jumper Settings

The pins of the 2364, 27128 and 27256 ROMs are slightly different.
The mode must therefore be set using jumpers.


2364
A7 1, A6 2, A5 3, A4 4, A3 5, A2 6, A1 7, A0 8, D0 9, D1 10, D2 11, GND 12
24 Vcc, 23 A8, 22 A9, 21 A12, 20 /CE, 19 A10, 18 A11, 17 D7, 16 D6, 15 D5, 14 D4, 13 D3
27128
Vpp 1, A12 2, A7 3, A6 4, A5 5, A4 6, A3 7, A2 8, A1 9, A0 10, D0 11, D1 12, D2 13, GND 14
28 Vcc, 27 /PGM, 26 A13, 25 A8, 24 A9, 23 A11, 22 /OE, 21 A10, 20 /CE, 19 D7, 18 D6, 17 D5, 16 D4, 15 D3
27256
Vpp 1, A12 2, A7 3, A6 4, A5 5, A4 6, A3 7, A2 8, A1 9, A0 10, D0 11, D1 12, D2 13, GND 14
28 Vcc, 27 A14, 26 A13, 25 A8, 24 A9, 23 A11, 22 /OE, 21 A10, 20 /CE, /PGM, 19 D7, 18 D6, 17 D5, 16 D4, 15 D3

VIC20, C64, 24-pin, 8k mode

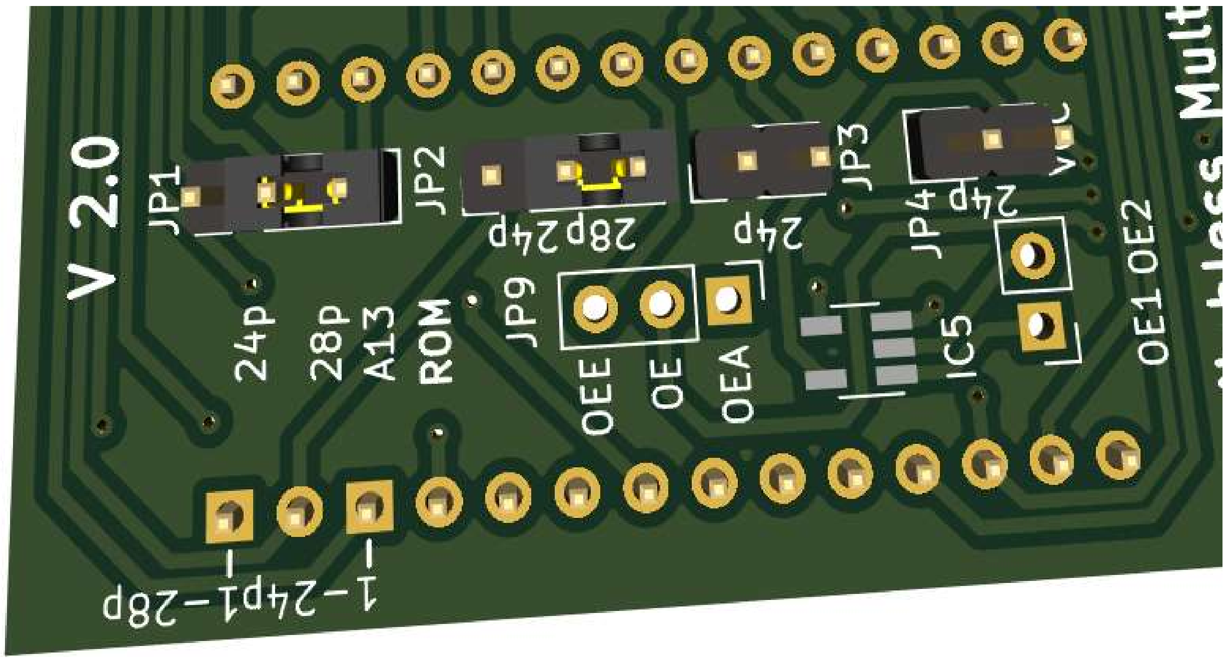
JP1: 24p (1-2) AVR MCU controls address line A13
JP2: 24p (1-2) CBM CPU controls address line A12 on ROM socket pin 23
JP3: 24p (1-2) CBM CPU controls address line A11 on ROM socket pin 20
JP4: 24p (1-2) CBM ROM socket pin 26 is connected to VCC



Jumper Settings

C64C, C128, 28-pin, 16k mode

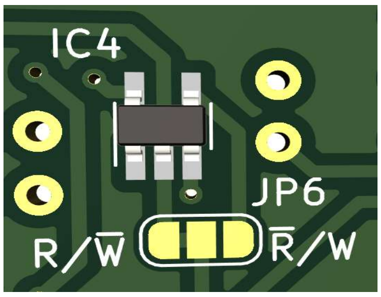
JP1: 28p (2-3) CBM CPU controls address line A13 on ROM socket pin 26
JP2: 28p (2-3) CBM CPU controls address line A11 on ROM socket pin 23
JP3: open Attention, it won't work without an open jumper.
JP4: open Attention, it won't work without an open jumper.



Jumper Settings

C128DCR, 28-pin, 32k mode

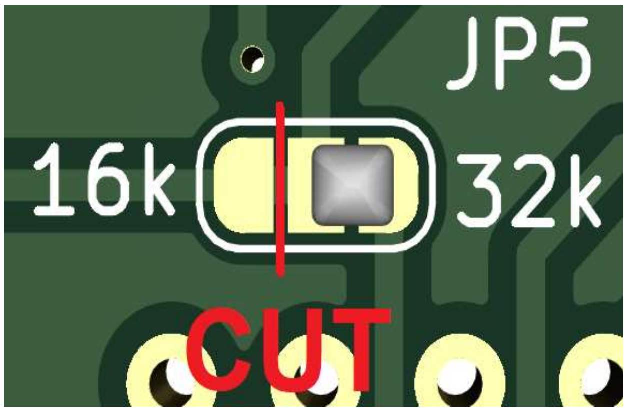
If you want to use the Switchless multi Kernal ROM in a C128DCR, 32k ROM, the signal r/w must be inverted. For this, the SMD IC4 must be soldered on the PCB back side.

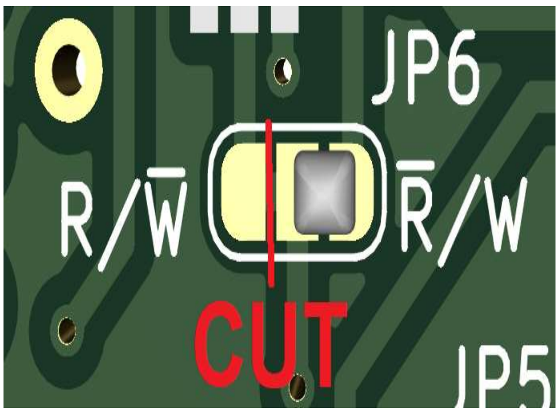


JP1 – JP4 are the same as 28-pin, 16k mode

JP5:
JP5 is connected for 16k mode by default.
Cut the connection between pin 1 and pin 2 with a cutting knife.
Use a multimeter to make sure there is no connection between pin 1 and pin 2.
Make a solder bridge between pin 2 and pin 3 for the 32k mode.
CBM CPU controls address line A14 on ROM socket pin 27

JP6:
JP6 is connected not to invert the r/w signal by default.
Cut the connection between pin 1 and pin 2 with a cutting knife.
Use a multimeter to make sure there is no connection between pin 1 and pin 2.
Make a solder bridge between pin 2 and pin 3 to invert the r/w signal.
The r/w signal is inverted for IC3.





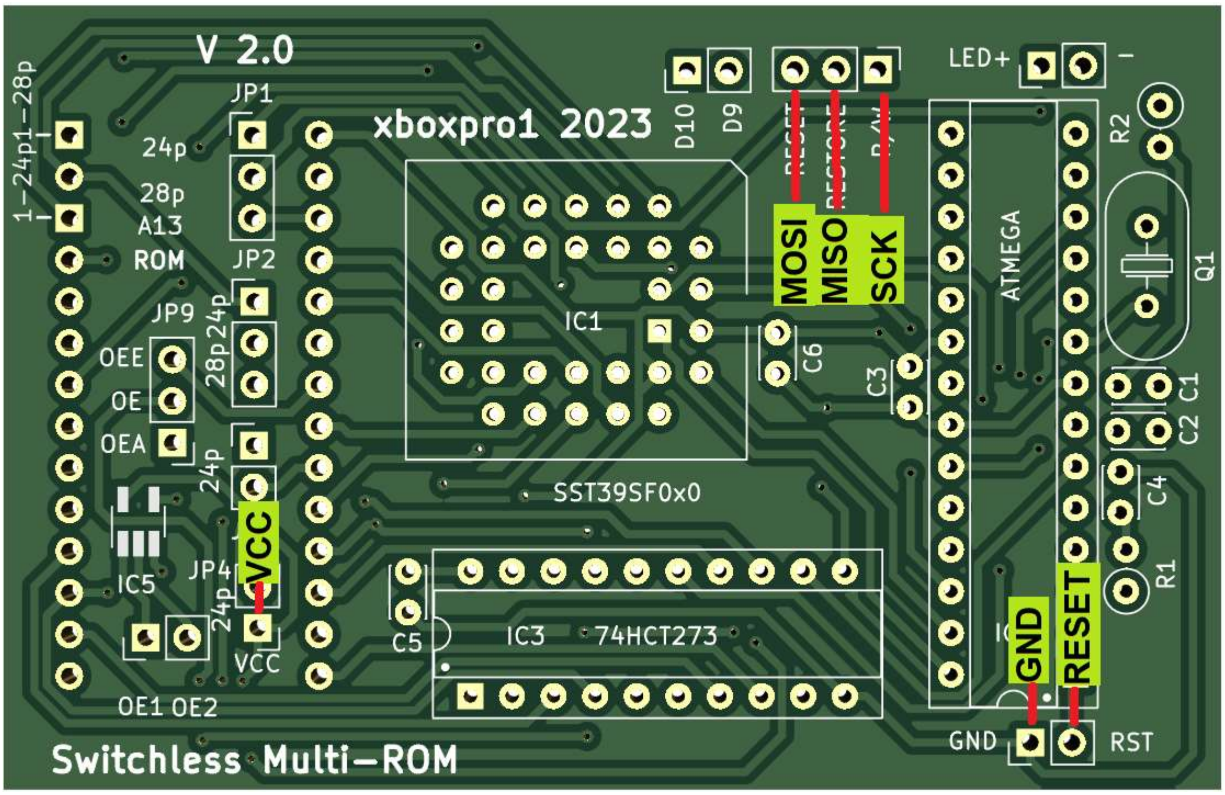
LED

The LED connector is used for the case's power LED to display the ROM blink codes and Restore key presses. The LED can be deactivated in the firmware in order to be able to use the MCU pin 8 (LED pin) for another purpose.

ISP Connection

To program the MCU via ISP, the ISP programmer must be connected as follows.

MOSI	→	PCB RESET
MISO	→	PCB RESTORE
SCK	→	PCB R/W
RESET	→	PCB RST
GND	→	PCB GND
VCC	→	PCB JP4 – pin 1

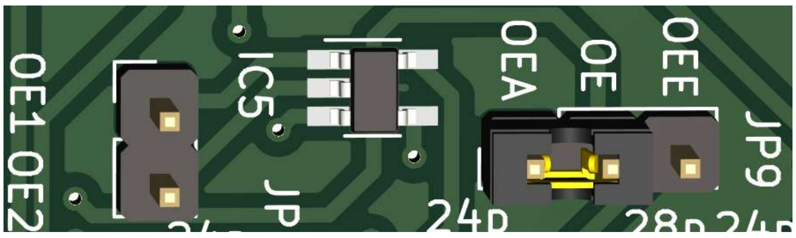


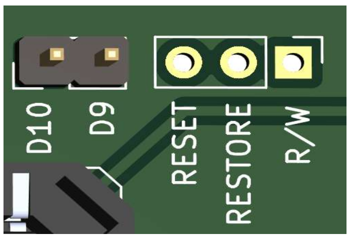
Experimental

Parts

Some parts on the PCB are for experimental testing and are not needed for normal operation. Unless you want to experiment too.

These parts are: IC5, JP9, connector OE1/OE and D9/D10.





Additional MCU pins

There are two additional MCU pins (9 and 10) on the D9/D10 connector for other purposes.

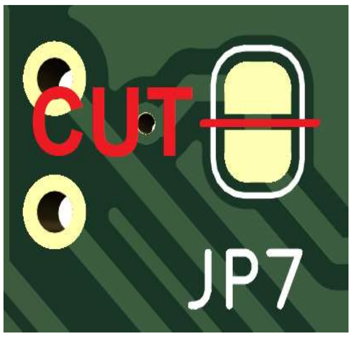
Jumper

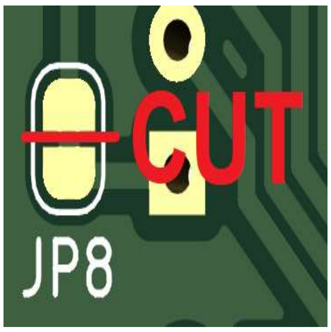
There are two additional jumpers on the PCB back side which are connected by default.

JP7 connects the $\overline{\mathrm{OE}}$ signal from the ROM socket pin 22 (OEE) with the flash-chip $\overline{\mathrm{OE}}$ signal on the PLCC socket pin 24 (OE). If you cut the connection it won't work at all. Unless you make a solder bridge on JP7 or solder JP9 and connect the pins OEE and OE with a jumper.

JP8 connects the resistor R1 to the AVR MCU external reset pin 1. If you cut the connection, you have thanks to the MCUdude MiniCore library an additional MCU pin 22. For this, reset must be disabled via the fuse settings when flashing the MCU.

Attention:
ISP programming of the MCU is no longer possible if reset is disabled via fuse settings. In order to be able to flash the MCU you have to use the USB universal programmer.





Experimental

JP9 connects the $\overline{\mathrm{OE}}$ signal from the ROM socket pin 22 (OEE) or the output from IC5, a single “AND” Gate, with the flash-chip $\overline{\mathrm{OE}}$ signal on the PLCC socket pin 24 (OE).

IC5 is a single “AND” Gate for experimental testing. Theoretically, the flash chip could accommodate Character ROM, Basic ROM and Kernal ROM at the same time.
To do this, the MCU firmware would have to be experimentally changed.

OE1/OE2 connector
The OE1/OE2 connector is for the two inputs of the single “AND” Gate.