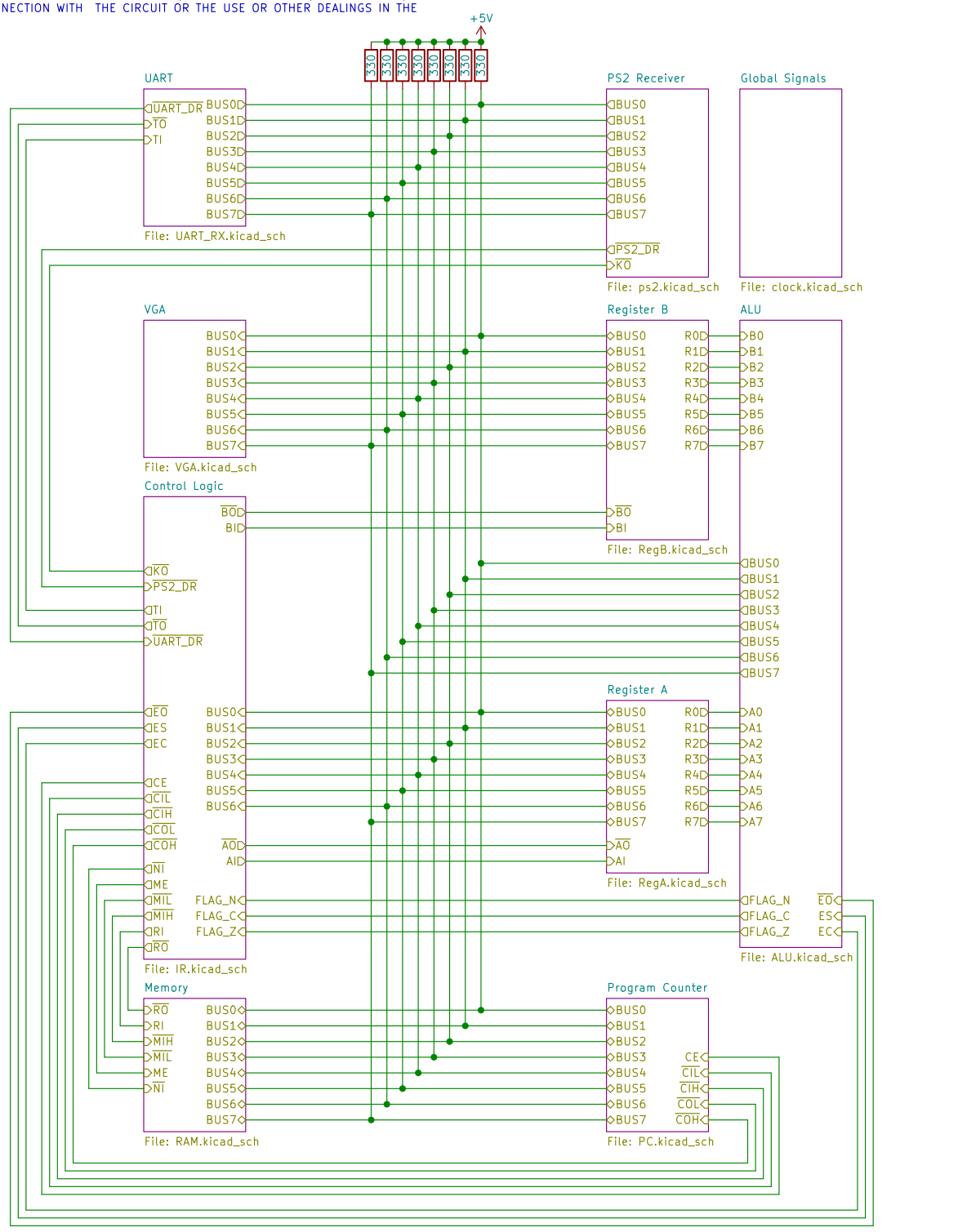
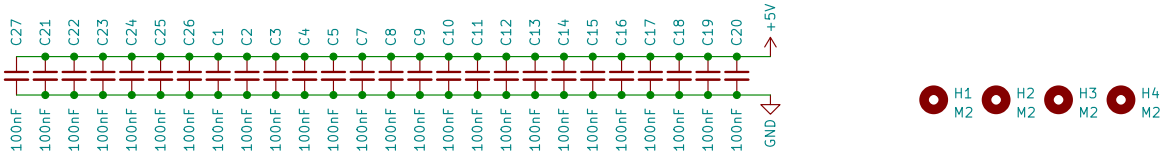




H1 M2 H2 M2 H3 M2 H4 M2

Supply voltage range 4.3V to 5.5V,
Supply current 140mA @ 4.3V, 175mA @ 5.0V, 200mA @ 5.5V

UART bitrate 250kbps, 1 start bit,
8 data bits, 2 stop bits, new line LF,
transmit delay 10ms/line.

You can use RTS/CTS flow control
and omit the line delay with a
USB-to-serial converter using
CH340G (not FT232, CP2102).

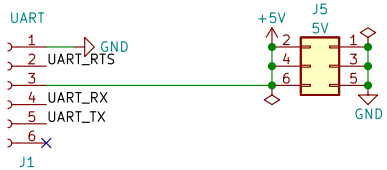
Author: Carsten Herting (slu4)

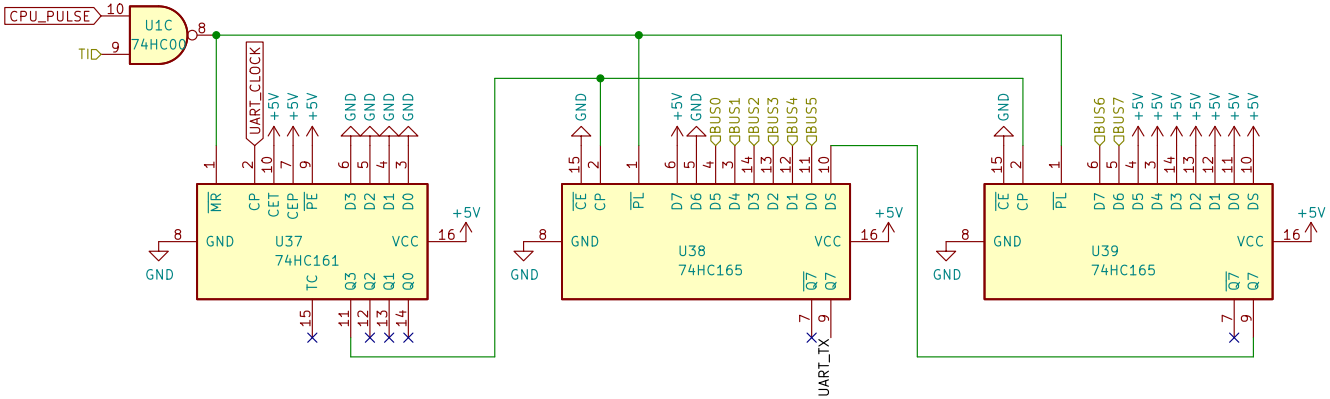
Sheet: /
File: 8-Bit CPU 32k.kicad_sch

Title: Minimal 64 Home Computer

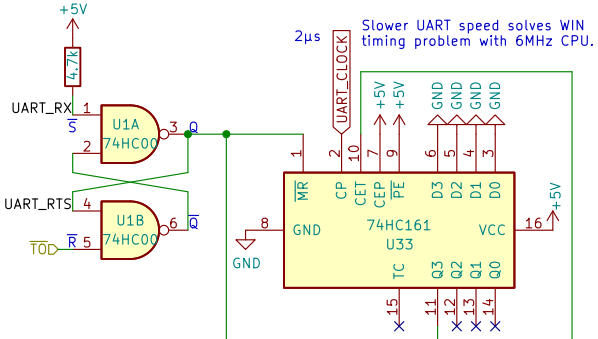
Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 1/11

By connecting pin 3 to 5V, you can also power the board directly from the USB (UART) port. But only use a single voltage source at any given time.



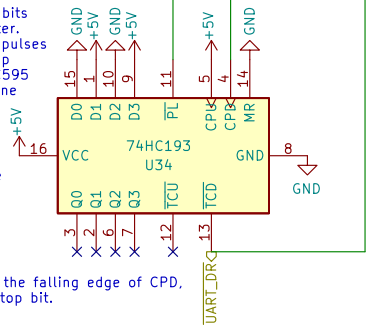


You can connect the RTS output ("I am ready to receive") to the CTS input ("You are clear to send" input) of your receiver IC and use RTS/CTS flow control. Only CH340G ICs seem to work.

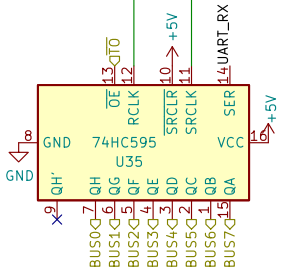


We need to shift 9 bits into the shift register. We count 10 clock pulses here (until high stop bit) since the 74HC595 output register is one clock pulse late.

74HC161 cannot be used since it has a synchroneous PL.



TCD goes LOW with the falling edge of CPD, at the end of the stop bit.



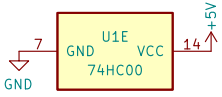
The bit order is intentionally reversed.

DESCRIPTION OF THE DATA ACQUISITION CYCLE:

1) Prior to receiving a datum: Q=LOW is reseting the timer and loading "10" into the bit counter. This also implies TCD(UART_DR)=HIGH, enabling counting in general and also signaling "no data is present".
2) An incoming datum, starting with a falling edge at RX, sets Q=HIGH, ending the reset of the timer and also ending the loading of the bit counter. As long as TCD=UART_DR remains HIGH, counting is now enabled. 10 bits will enter the shift register. The uninteresting start bit is shifted out. The output register is always one clock begind the shift register.
3) Upon hitting "0" (reading of high stop bit) the bit counter's TCD=UART_DR goes LOW with the falling edge of CPD, which is also stoping the timer via CET. The datum *prior* to this last shift remains in the 74HC595's output register since it is always one step late, and subsequently arriving data have no effect.
4) UART_DR is sampled by the flags register and fed into the control logic If UART_DR=LOW, the microcode of the INP instruction activates TO=TC in ..., BI|FI, TO|AI, EO|ES|EC|FI, IC.
5) The received datum is automatically compared to 0xff. Z=1 indicates that no datum other than 0xff is present.
6) Polling then looks like this:

```
loop:     INP BEQ loop     ; loop back if 0xff was read
          ; ...            ; process datum other than 0xff
          JPA loop         ; wait for next datum
```

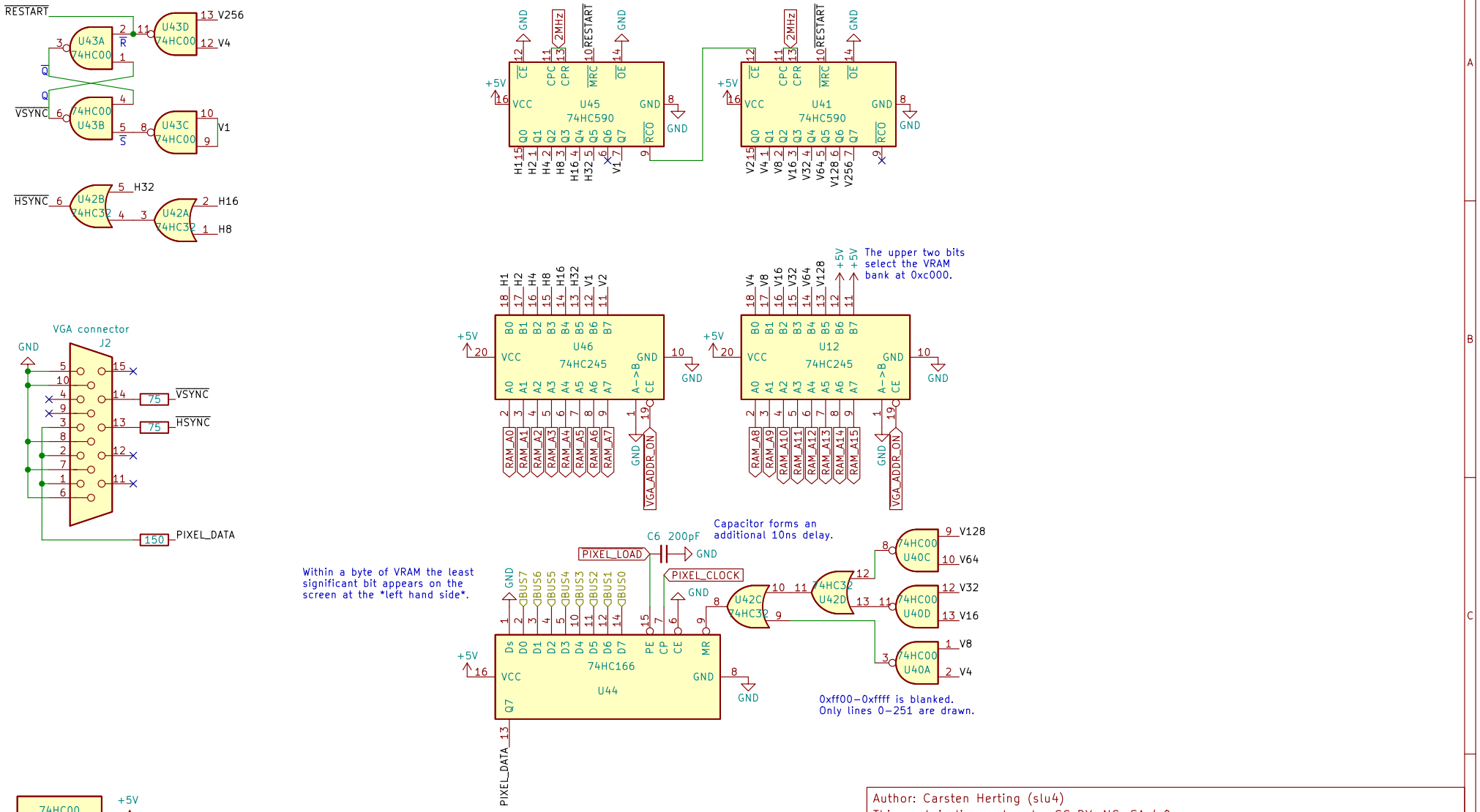
The loop samples (reads and resets) UART_DR every 11 clock cycles and thus ensures that a datum can be read before the 2nd stop bit (16 cycles) is over and the receiver has to be armed again.



Sheet: /UART/
File: UART_RX.kicad_sch

Title: UART Section

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 2/11

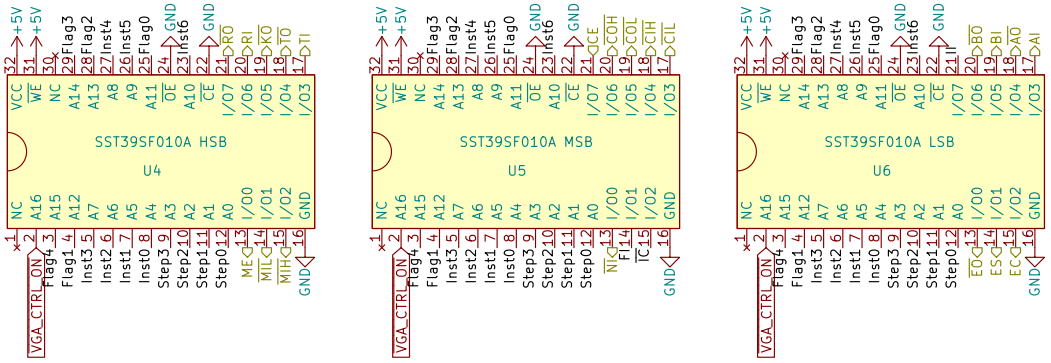


Sheet: /VGA/
File: VGA.kicad_sch

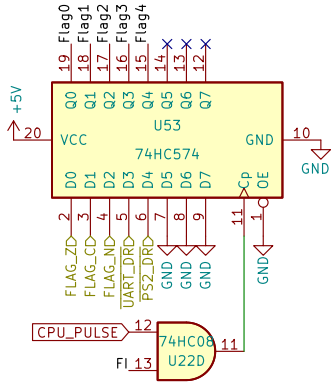
Title: VGA Circuit

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 3/11

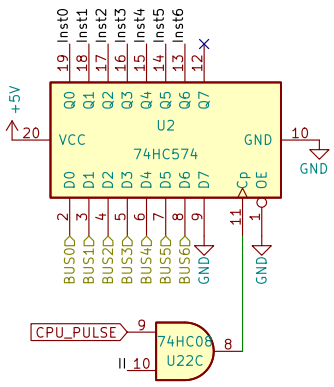
You can also use the larger SST39SF020A or SST39SF040A
here by connecting pins 1 and 30 of each IC to GND.



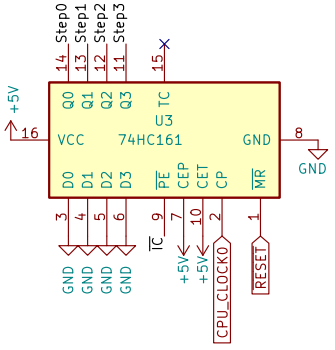
FLAGS REGISTER



INSTRUCTION REGISTER



STEP COUNTER



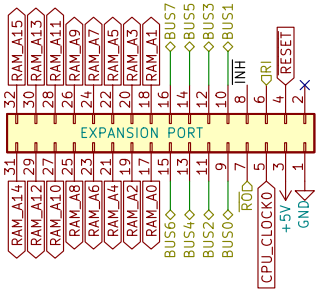
IC now works synchroneously ;-)

Sheet: /Control Logic/
File: IR.kicad_sch

Title: Control Logic

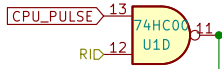
Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 4/11

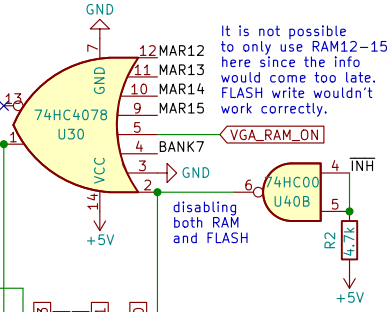
The expansion port has direct access to the RAM address lines and therefore "sees" the VGA's RAM access as well as the processor's acccess. Depending on the state of RAM_Axx, the expansion card will disable RAM/FLASH and put it's own data onto the bus. As long as the expansion card's memory-mapped address is outside VRAM, video output will not be affected.



This pin is intentionally unused.

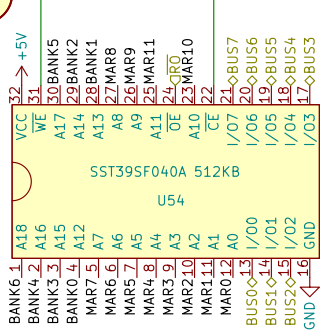
Use it in case you desparately need a special signal on your expansion card that is not already provided on the port.

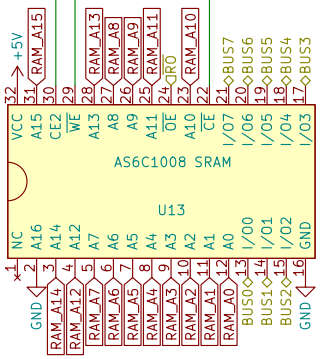




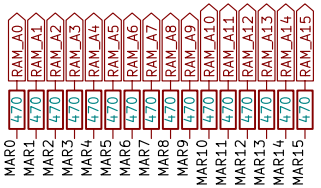
It is not possible to only use RAM12-15 here since the info would come too late. FLASH write wouldn't work correctly.

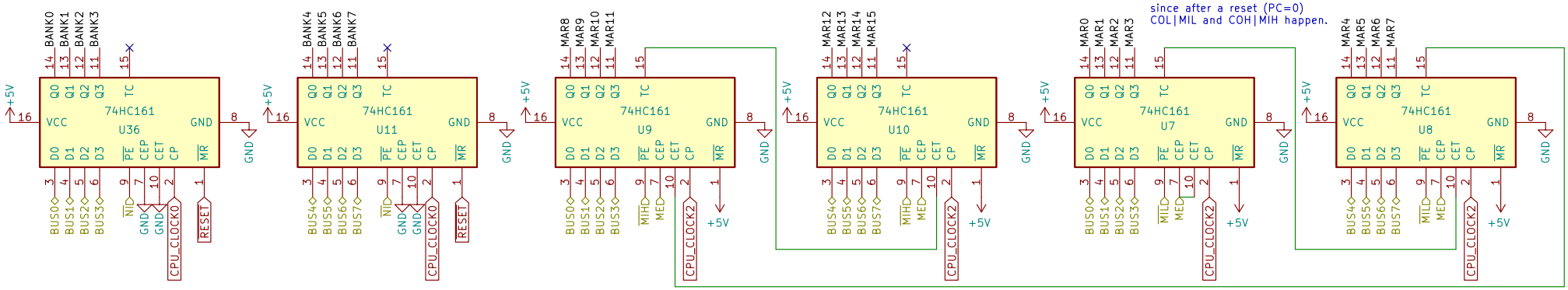
disabling both RAM and FLASH





55ns or 70ns versions are both tested okay





A MAR reset is not needed since after a reset (PC=0) COL|MIL and COH|MIH happen.

BANK REGISTER

I am using 74HC161 instead of 74HC173 because of MR.

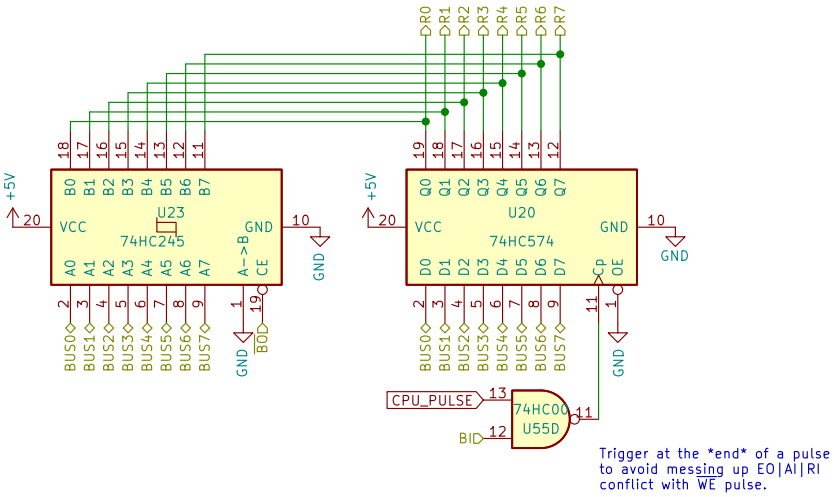
MEMORY ADDRESS REGISTER MSB

MEMORY ADDRESS REGISTER LSB

Sheet: /Memory/
File: RAM.kicad_sch

Title: RAM, ROM and MAR

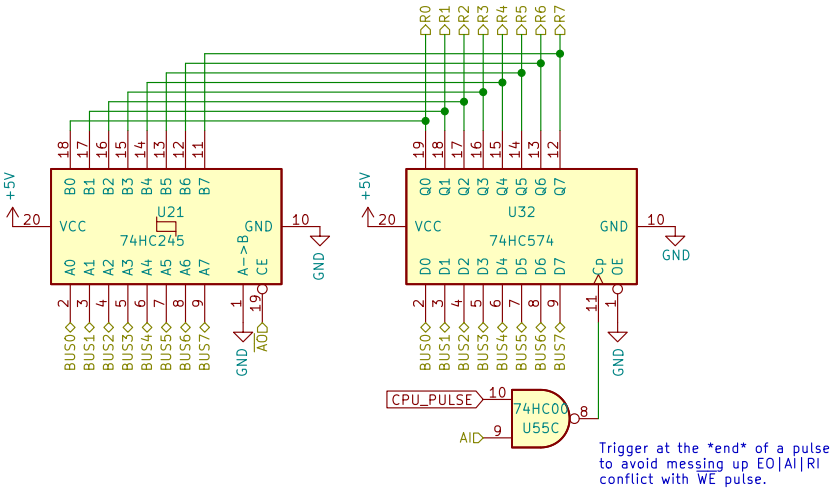
Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 5/11

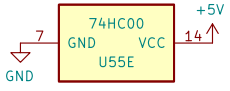


Sheet: /Register B/
File: RegB.kicad_sch

Title: B Register

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 7/11

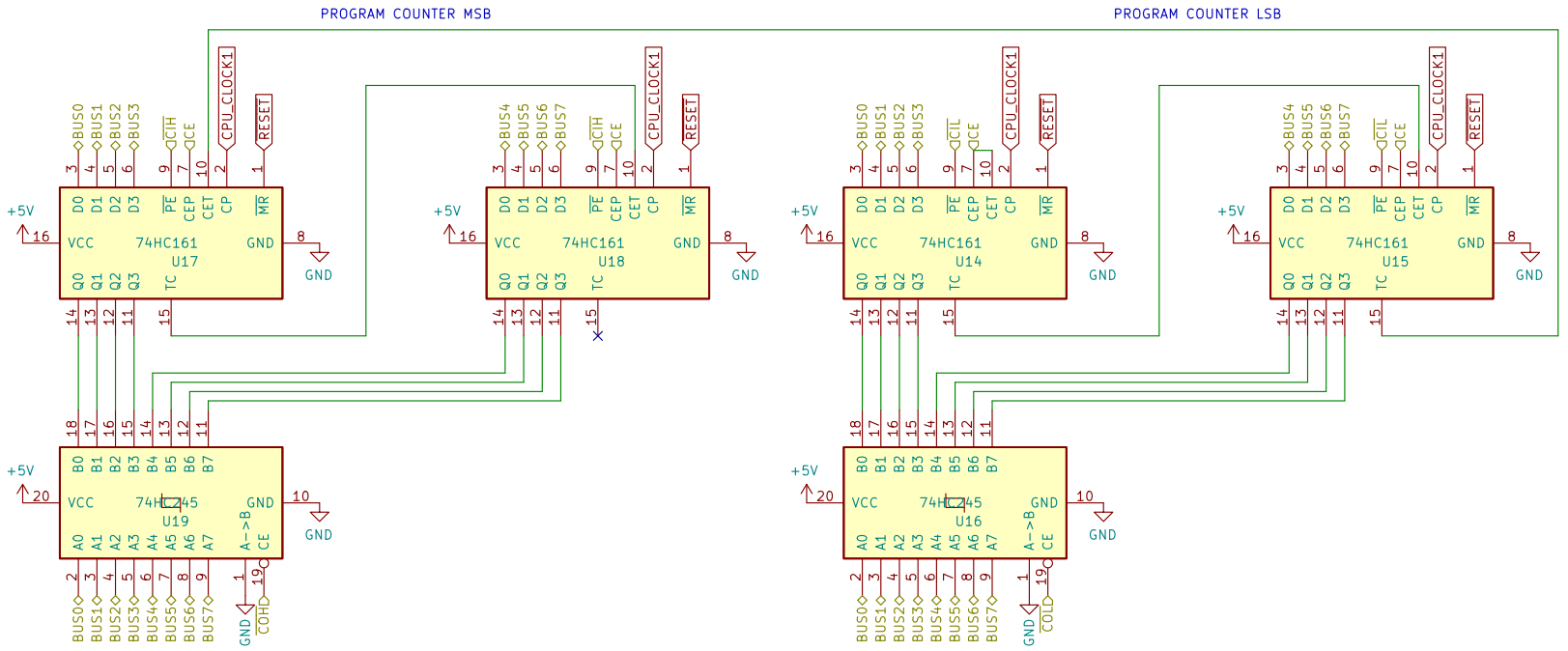




Sheet: /Register A/
File: RegA.kicad_sch

Title: A Register

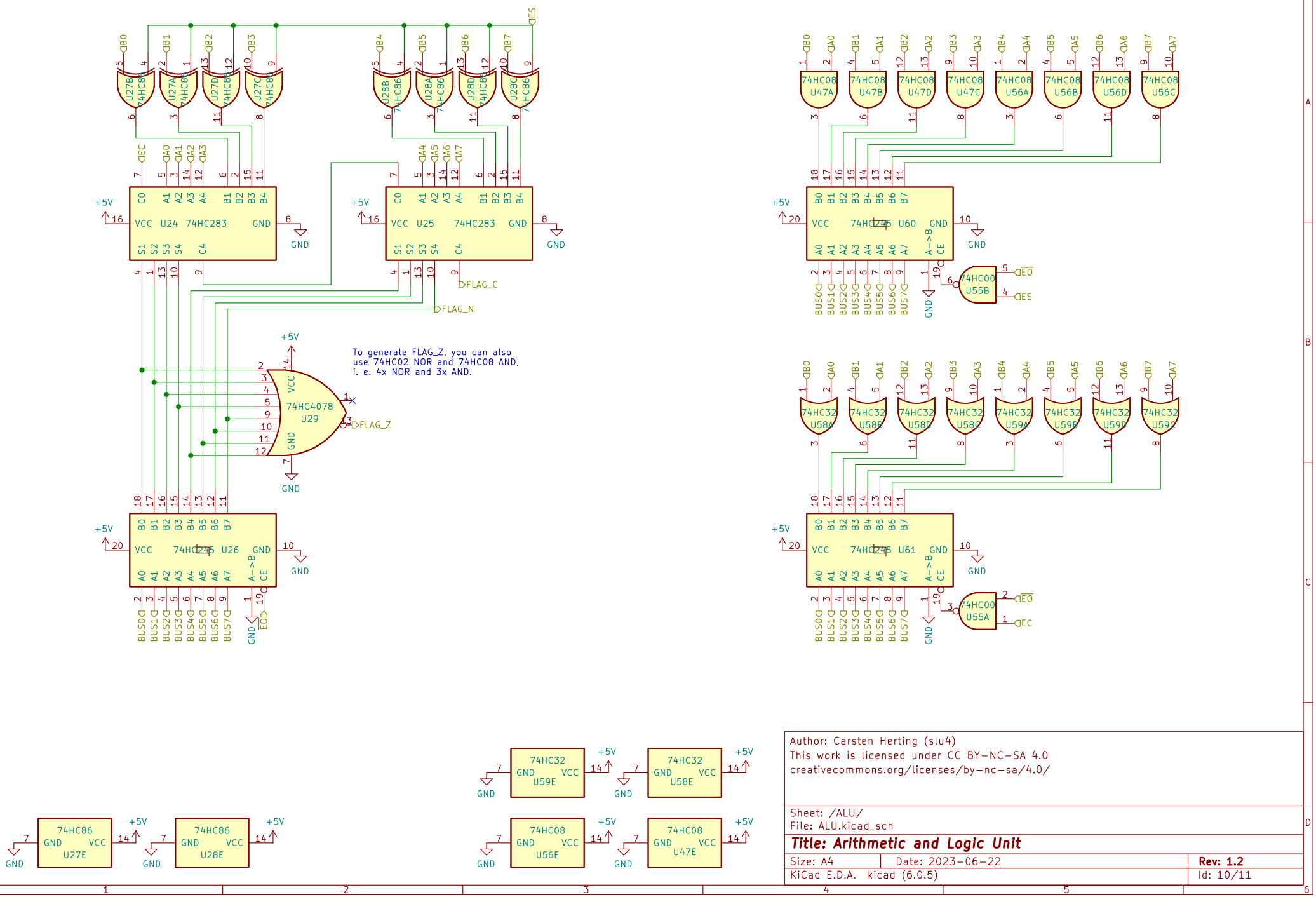
Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 8/11



Sheet: /Program Counter/
File: PC.kicad_sch

Title: Program Counter

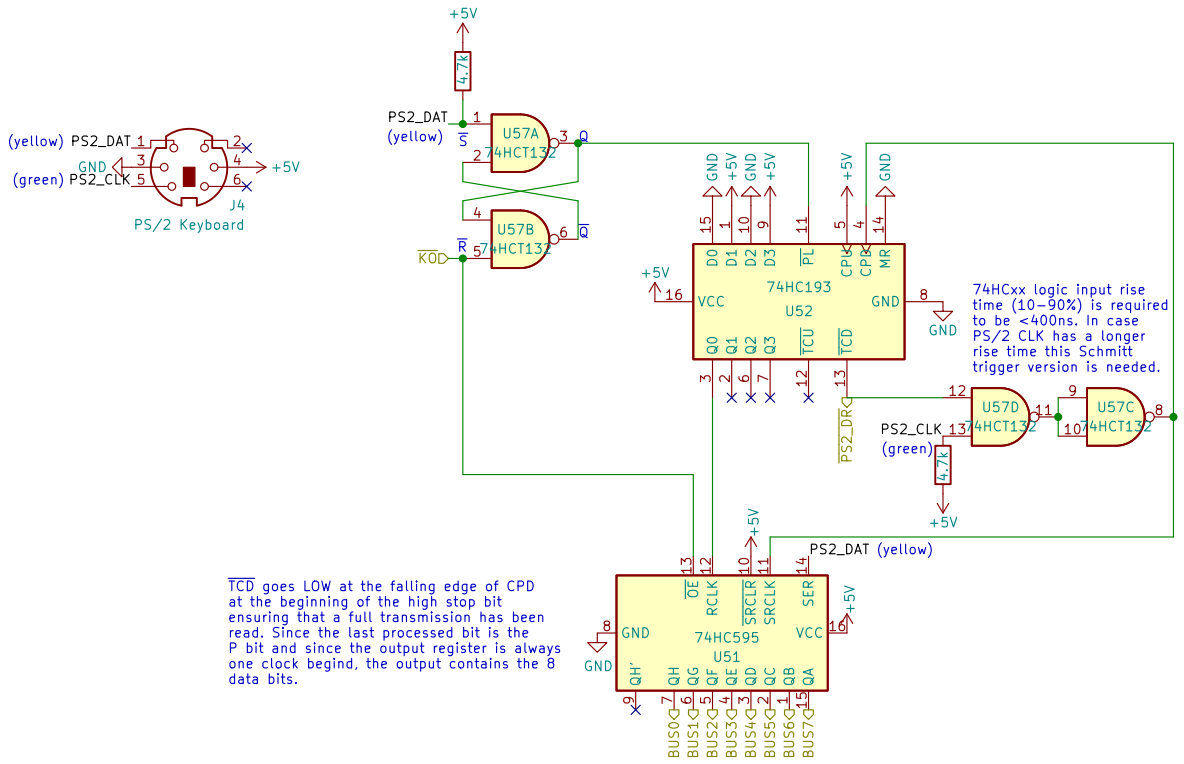
Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 9/11



Sheet: /ALU/
File: ALU.kicad_sch

Title: Arithmetic and Logic Unit

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 10/11

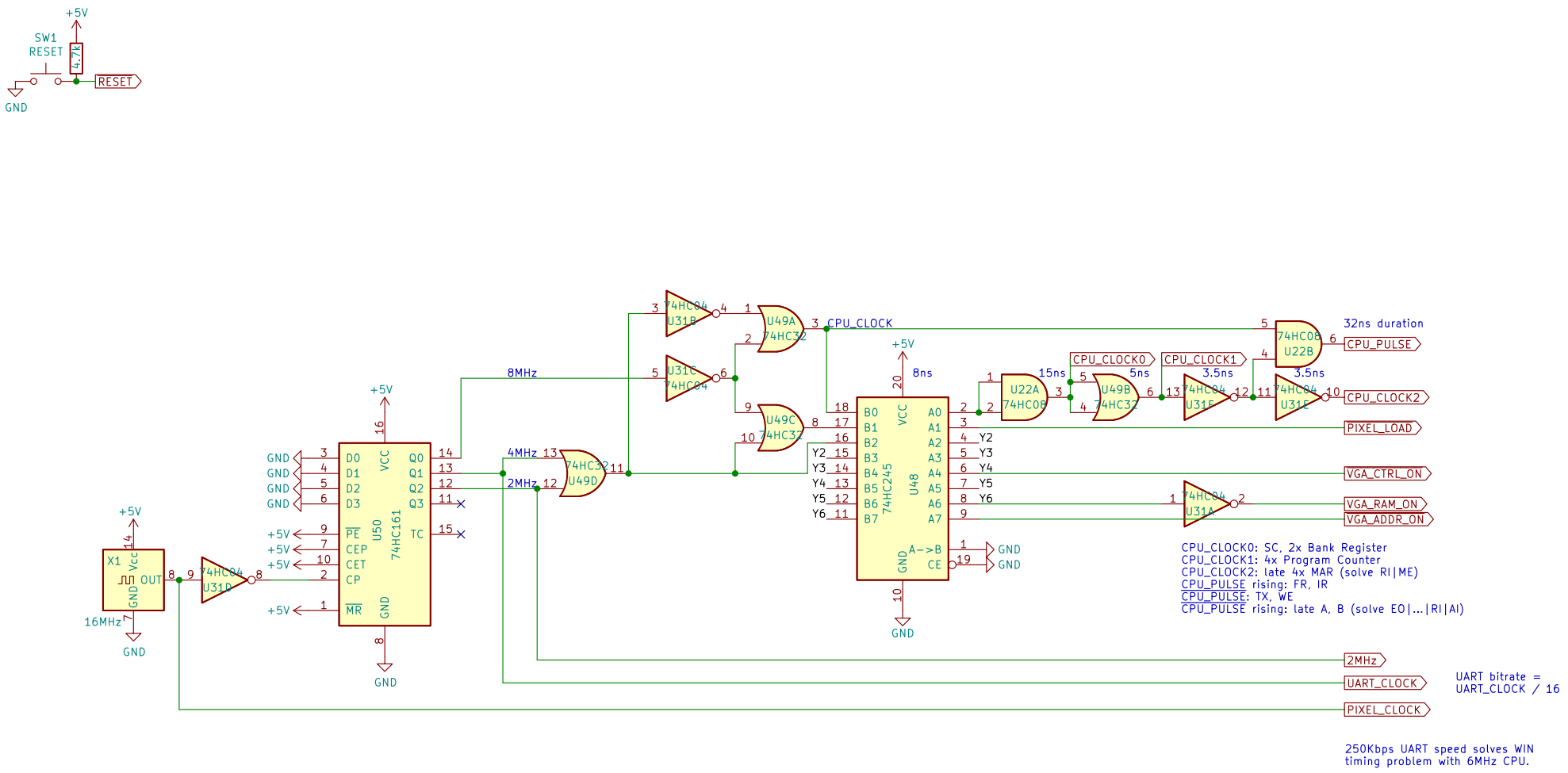


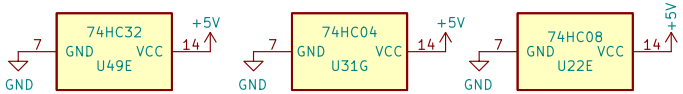
I am using a 74HC132 (NAND with Schmitt-trigger inputs) rather than the standard 74HC00 here, since PS/2 signals can have rather large rise and fall time.
For PS/2 devices with lower (3.3V) output use 74HCT132.

Sheet: /PS2 Receiver/
File: ps2.kicad_sch

Title: PS/2 Receiver

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 11/11





Sheet: /Global Signals/
File: clock.kicad_sch

Title: Global Signals

Size: A4	Date: 2023-06-22	**Rev: 1.2**
KiCad E.D.A. kicad (6.0.5)		Id: 11/11