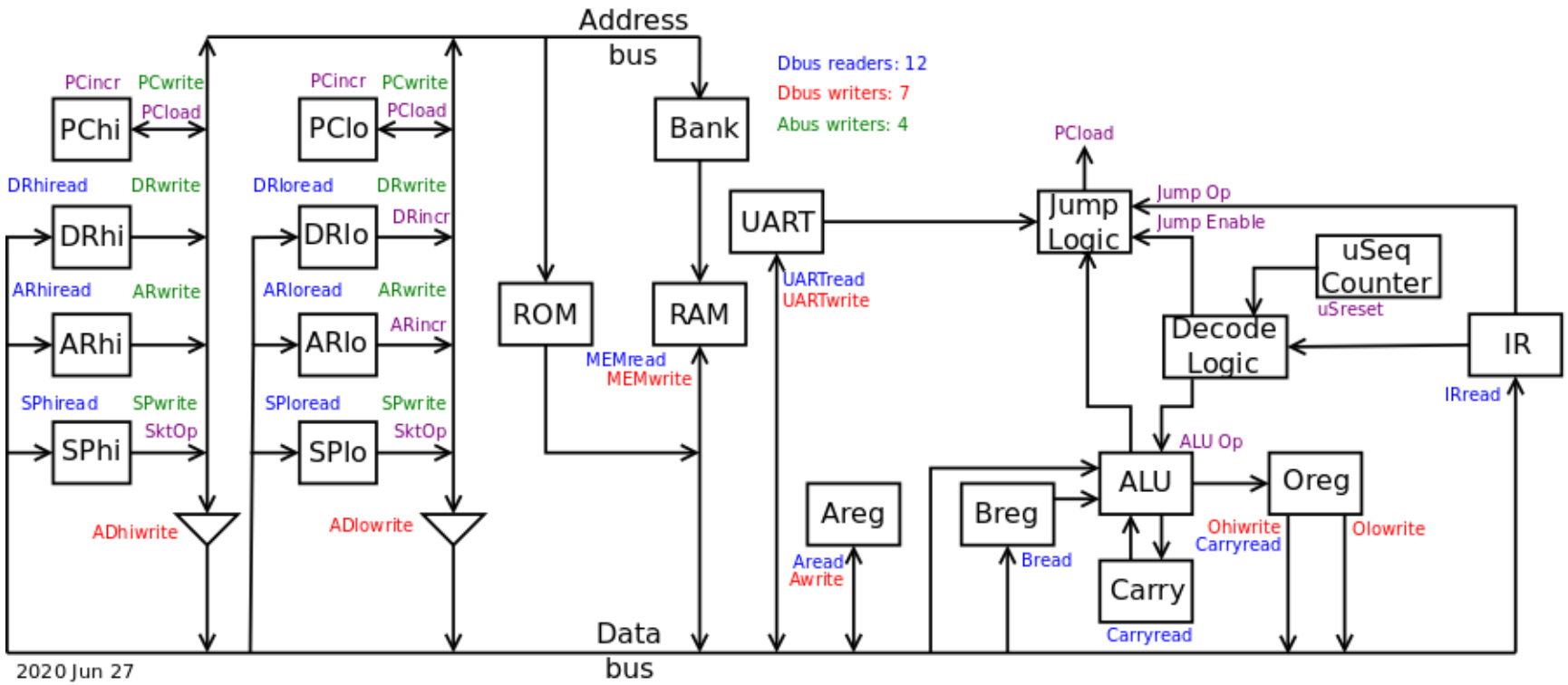


Sheet: Memory
File: Memory.sch

Sheet: Data
File: Data.sch

Sheet: Address Registers
File: AddrRegs.sch

Sheet: Control
File: Control.sch

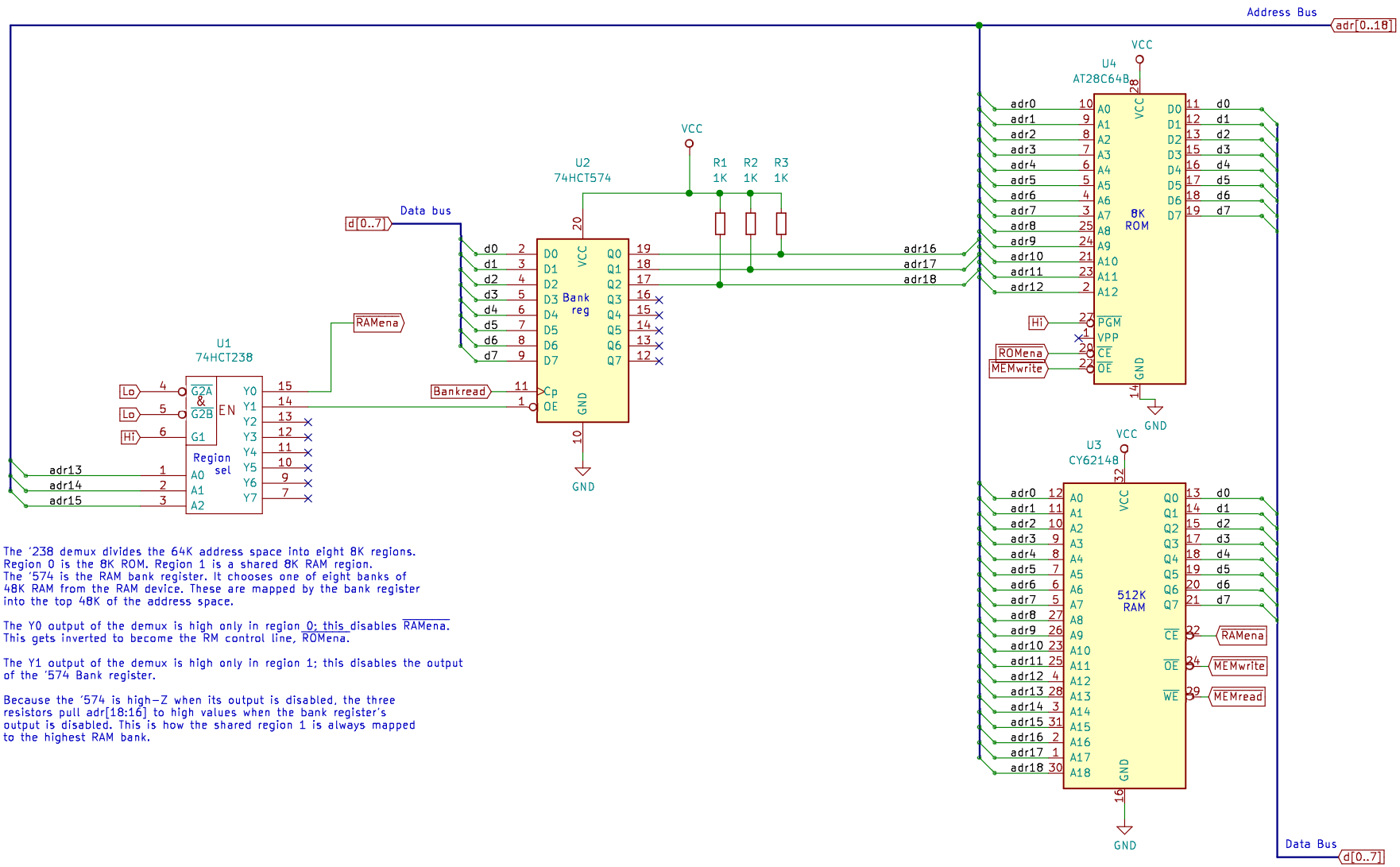
Sheet: Misc
File: Misc.sch

This is the FISC2 TTL CPU. Each one of the fivesheets on the left holds the main five sections of the design.

Sheet: /
File: FISC2.sch

Title: FISC2 TTL CPU

Size: A4	Date: 2020-06-29	**Rev:**
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1		Id: 1/6



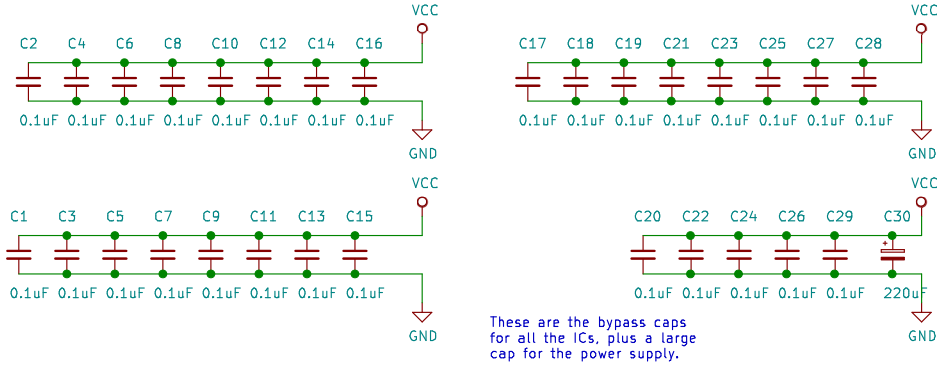
The '238 demux divides the 64K address space into eight 8K regions.
Region 0 is the 8K ROM. Region 1 is a shared 8K RAM region.
The '574 is the RAM bank register. It chooses one of eight banks of
48K RAM from the RAM device. These are mapped by the bank register
into the top 48K of the address space.

The Y0 output of the demux is high only in region 0; this disables $\overline{\text{RAMena}}$.
This gets inverted to become the RM control line, ROMena.

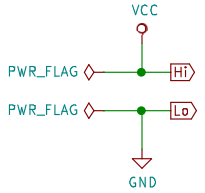
The Y1 output of the demux is high only in region 1; this disables the output
of the '574 Bank register.

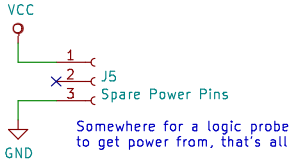
Because the '574 is high-Z when its output is disabled, the three
resistors pull adr[18:16] to high values when the bank register's
output is disabled. This is how the shared region 1 is always mapped
to the highest RAM bank.

Sheet: /Memory/	
File: Memory.sch	
Title: FISC2 TTL CPU	
Size: A4	Date: 2020-06-29
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1	Id: 2/6



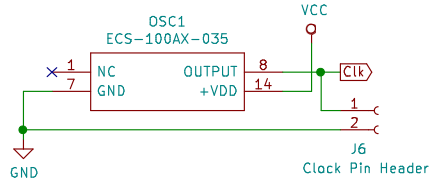
These are the bypass caps for all the ICs, plus a large cap for the power supply.



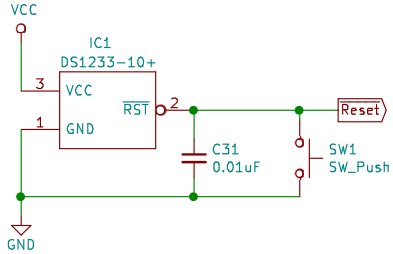


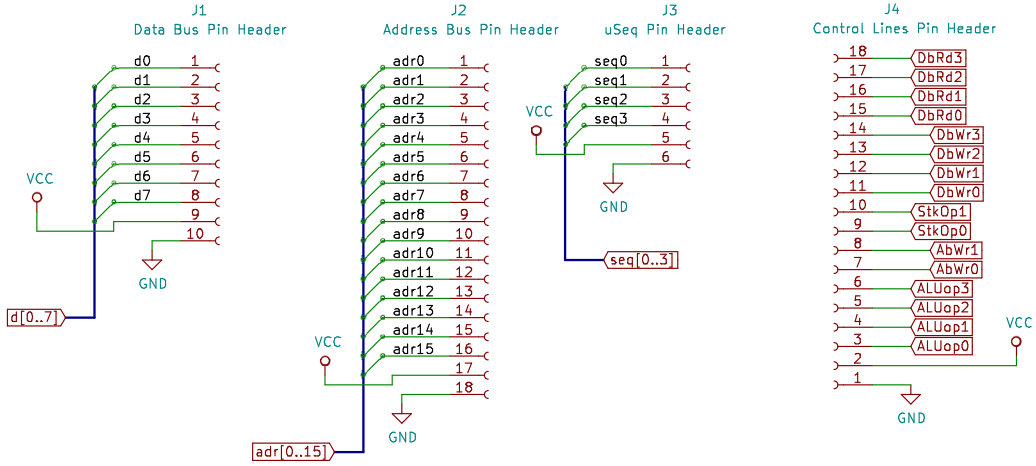
Somewhere for a logic probe to get power from, that's all

A 3.57MHz oscillator generates the main clock pulse.

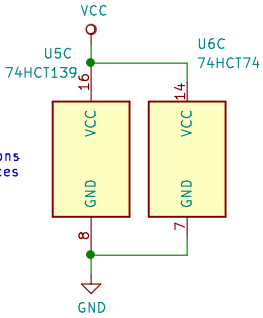


The DS1233 holds down the $\overline{\text{Reset}}$ line until power settles. The pushbutton allows for a manual reset.





These four pin headers allow the address bus, data bus, control lines and microsequence value to be examined by external equipment.



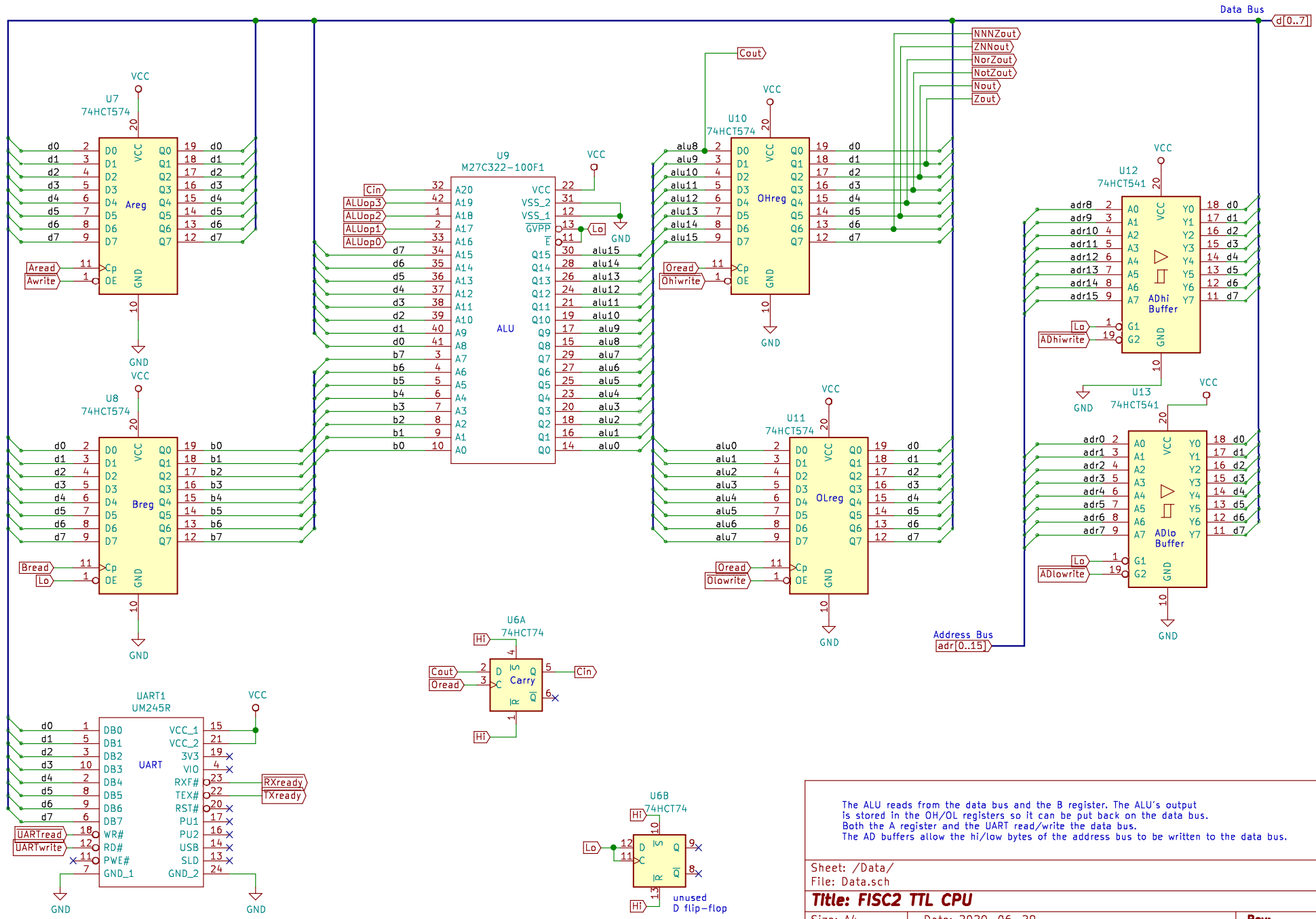
The power sections for several devices

Miscellaneous components in the design.

Sheet: /Misc/
File: Misc.sch

Title: FISC2 TTL CPU

Size: A4	Date: 2020-06-29	**Rev:**
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1		Id: 3/6

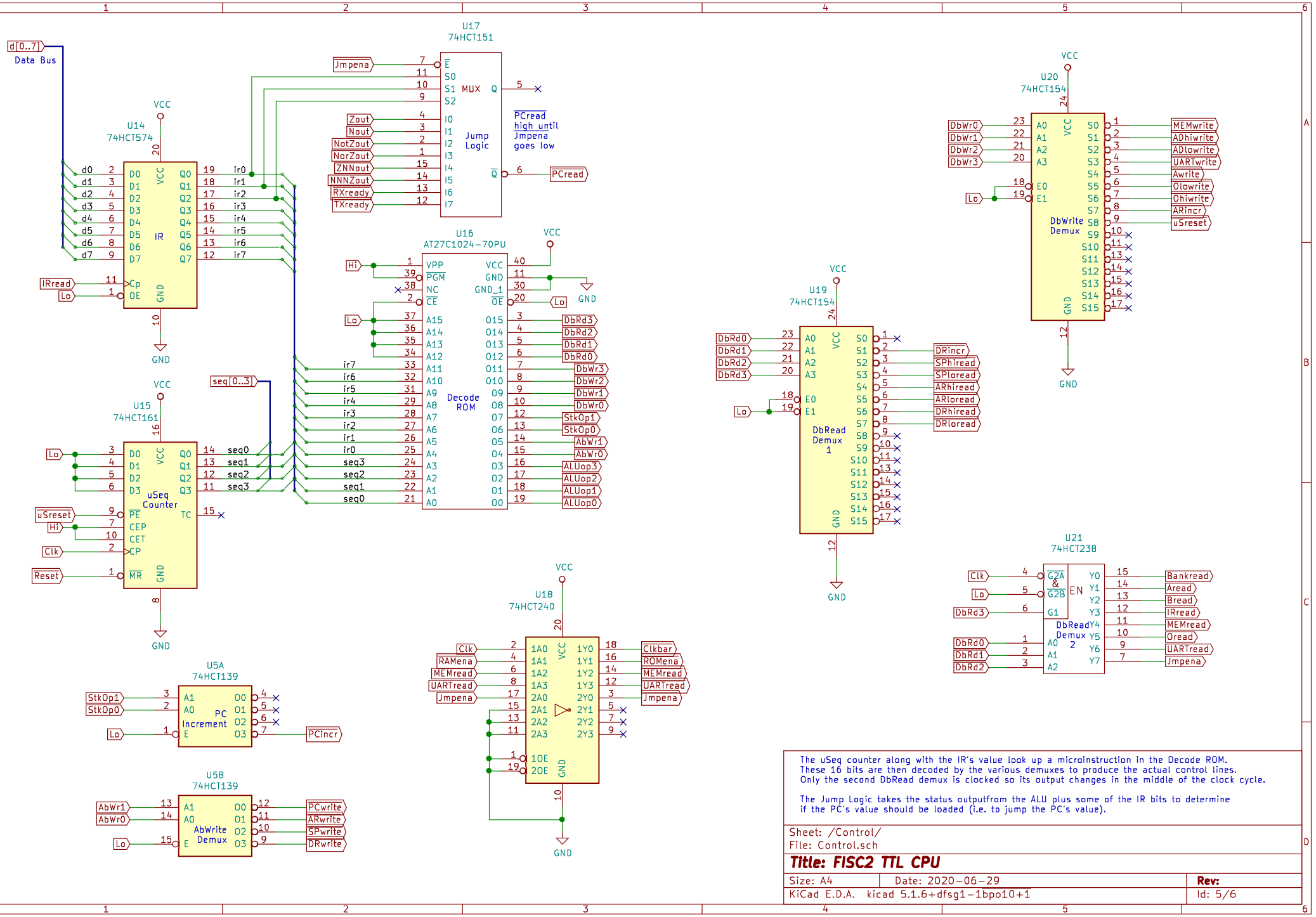


The ALU reads from the data bus and the B register. The ALU's output is stored in the OH/OL registers so it can be put back on the data bus.
Both the A register and the UART read/write the data bus.
The AD buffers allow the hi/low bytes of the address bus to be written to the data bus.

Sheet: /Data/
File: Data.sch

Title: FISC2 TTL CPU

Size: A4	Date: 2020-06-29	Rev:
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1		Id: 4/6



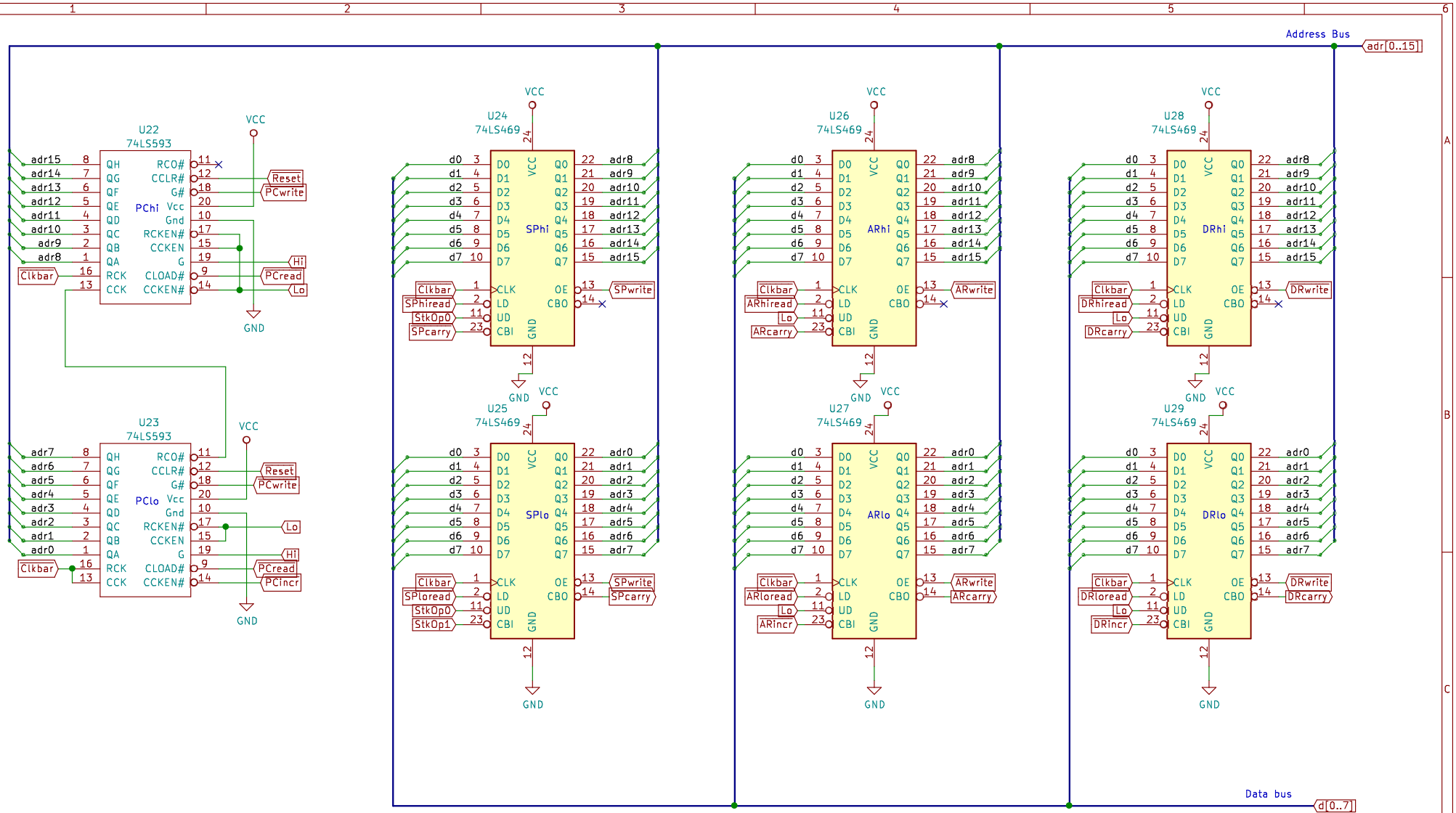
The uSeq counter along with the IR's value look up a microinstruction in the Decode ROM.
These 16 bits are then decoded by the various demuxes to produce the actual control lines.
Only the second DbRead demux is clocked so its output changes in the middle of the clock cycle.

The Jump Logic takes the status outputfrom the ALU plus some of the IR bits to determine
if the PC's value should be loaded (i.e. to jump the PC's value).

Sheet: /Control/
File: Control.sch

Title: FISC2 TTL CPU

Size: A4	Date: 2020-06-29	Rev:
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1		Id: 5/6



These four registers write to the address bus. The Program Counter (PC) is a 16-bit register which can load from the address bus (jump), hold or increment. The Stack Pointer (SP) can load from the data bus, hold, increment or decrement. The AR register typically holds the address of a data item in memory. The DR register is typically the destination address of data. Both the AR and DR can increment, allowing us to create instructions on 16-bit data items.

Sheet: /Address Registers/
File: AddrRegs.sch

Title: Address Registers

Size: A4	Date: 2020-06-29	**Rev:**
KiCad E.D.A. kicad 5.1.6+dfsg1-1bpo10+1		Id: 6/6