Sheet: Output Display

File: Output Display.sch

Sheet: Register XY

File: Register XY.sch

Sheet: Speed Control

File: Speed Control.sch

Sheet: Program Counter

File: Program Counter.sch

Sheet: Clock

File: Clock.sch

Sheet: Instruction Fetch

File: Instruction Fetch.sch

Sheet: Power-on Reset

File: Power-on Reset.sch

Sheet: Data RAM

File: Data RAM.sch

Sheet: Instruction Decode

File: Instruction Decode.sch

Sheet: Register A

File: Register A.sch

Sheet: Serial

File: Serial.sch

Sheet: Execute

File: Execute.sch

Sheet: Register B

File: Register B.sch

Sheet: Register D

File: Register D.sch

Sheet: Bus Display

File: Bus Display.sch

Sheet: ALU

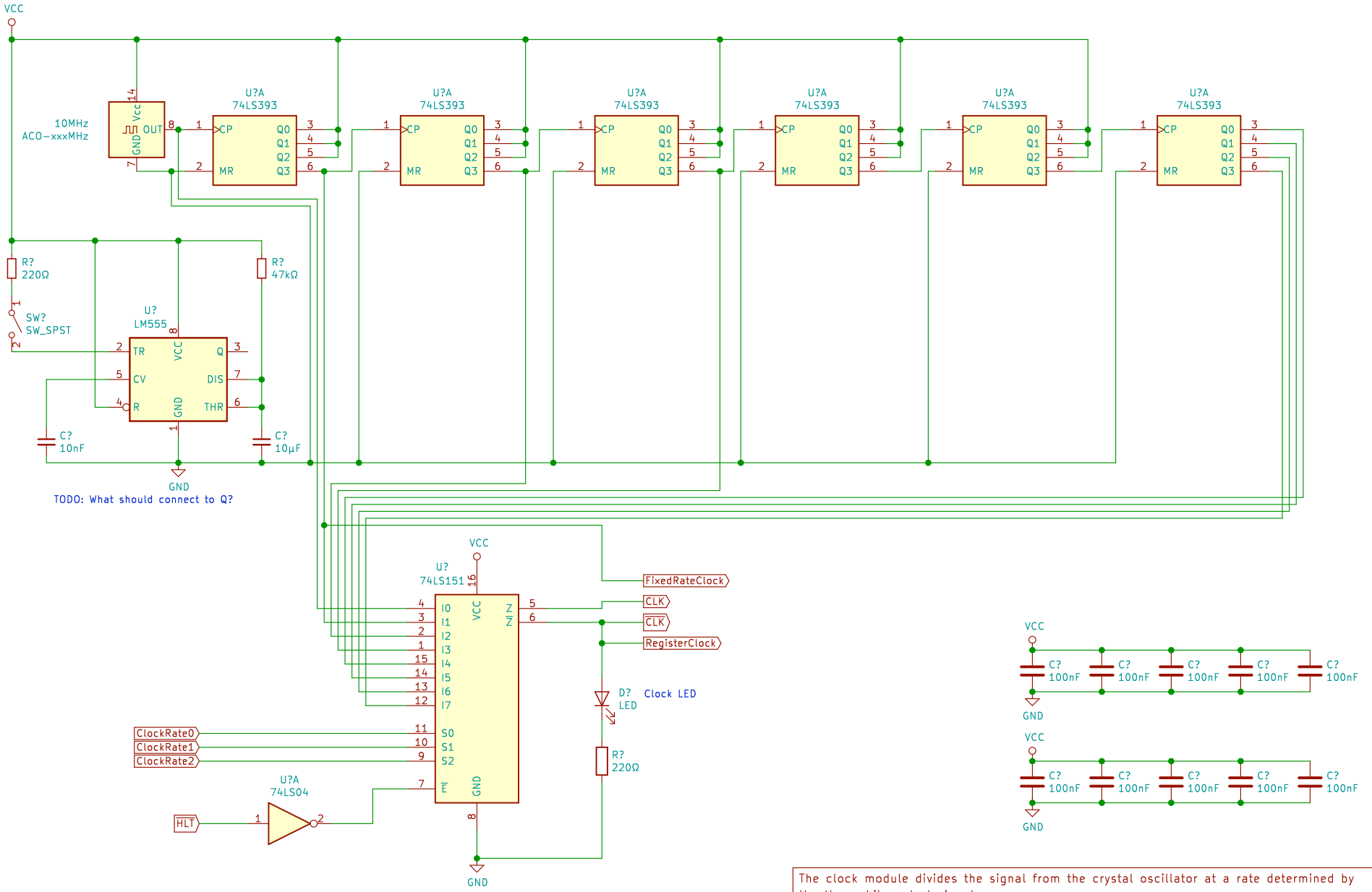
File: ALU.sch

TTL microcomputer built from 74xx series logic chips.

Sheet: /
File: TurtleTTL.sch

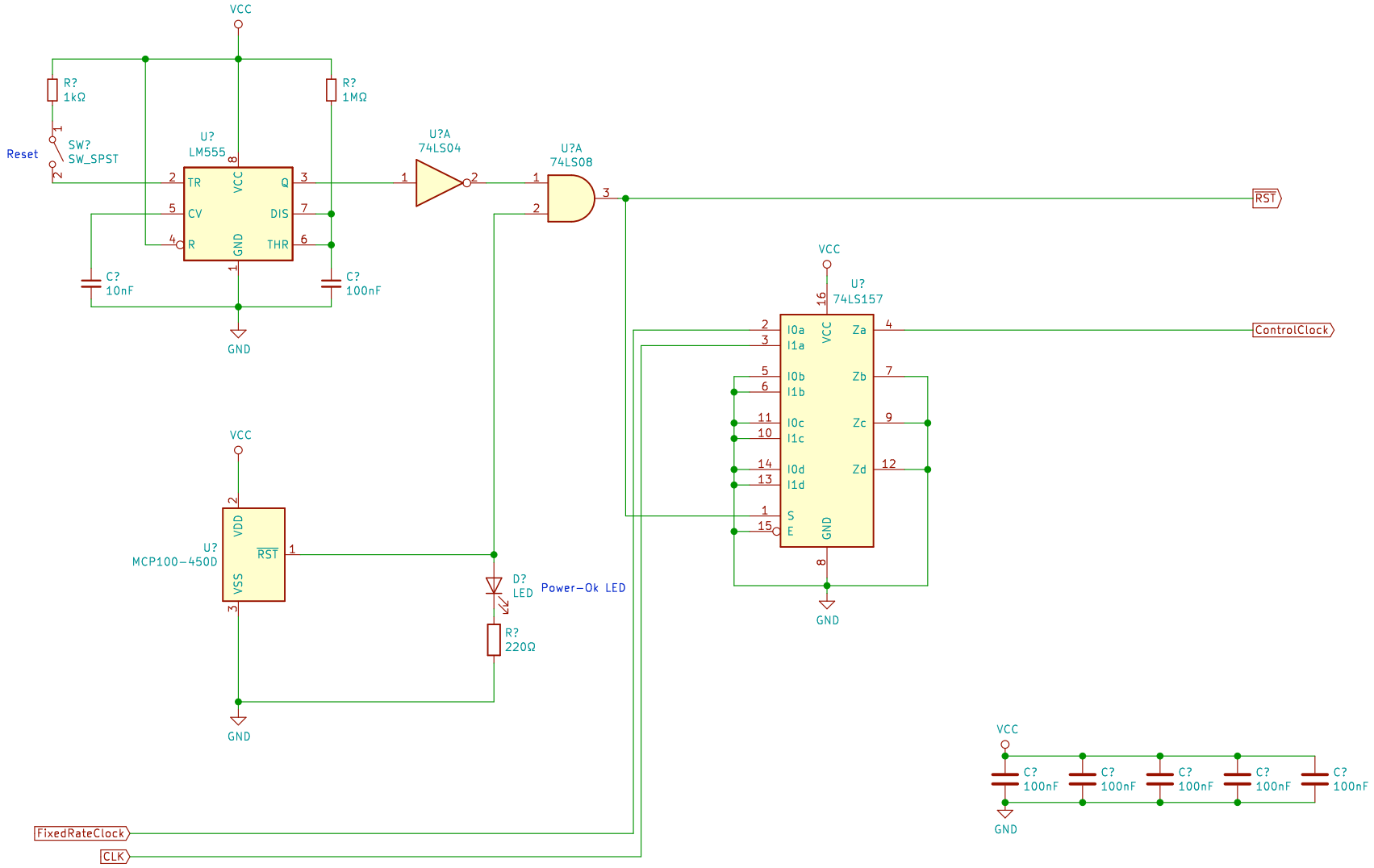
Title: Turtle TTL

Size: A	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 1/24



The clock module divides the signal from the crystal oscillator at a rate determined by the three-bit control signal.

Sheet: /Clock/	
File: Clock.sch	
Title: Clock	
Size: A	Date:
KiCad E.D.A. kicad (5.1.0-0)	Rev:
	Id: 2/24

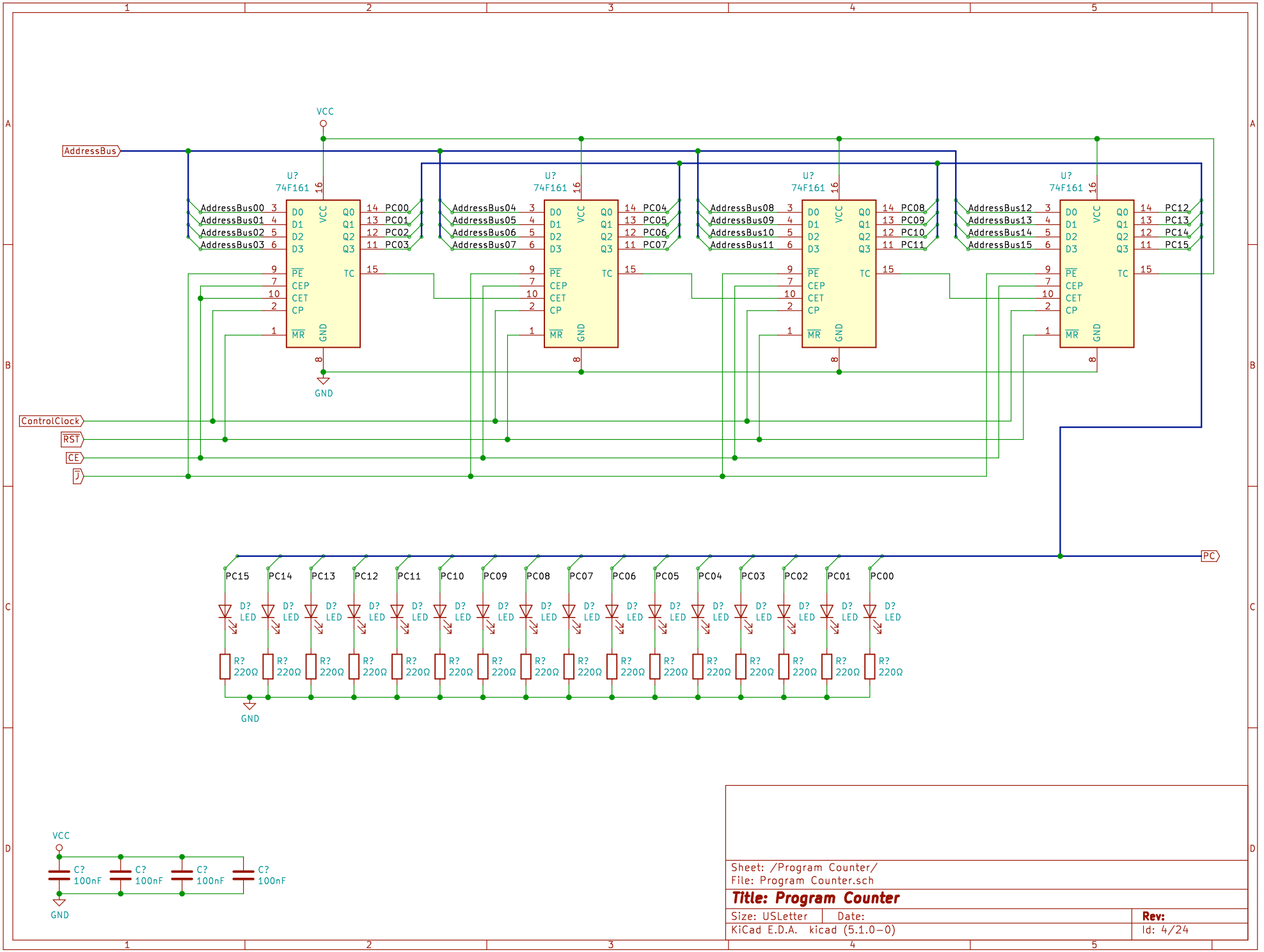


The MCP100 provides Power-on Reset functionality.
A button is also provided to manually reset the machine.
During reset, the control clock is pulsed repeatedly to flush the pipeline.

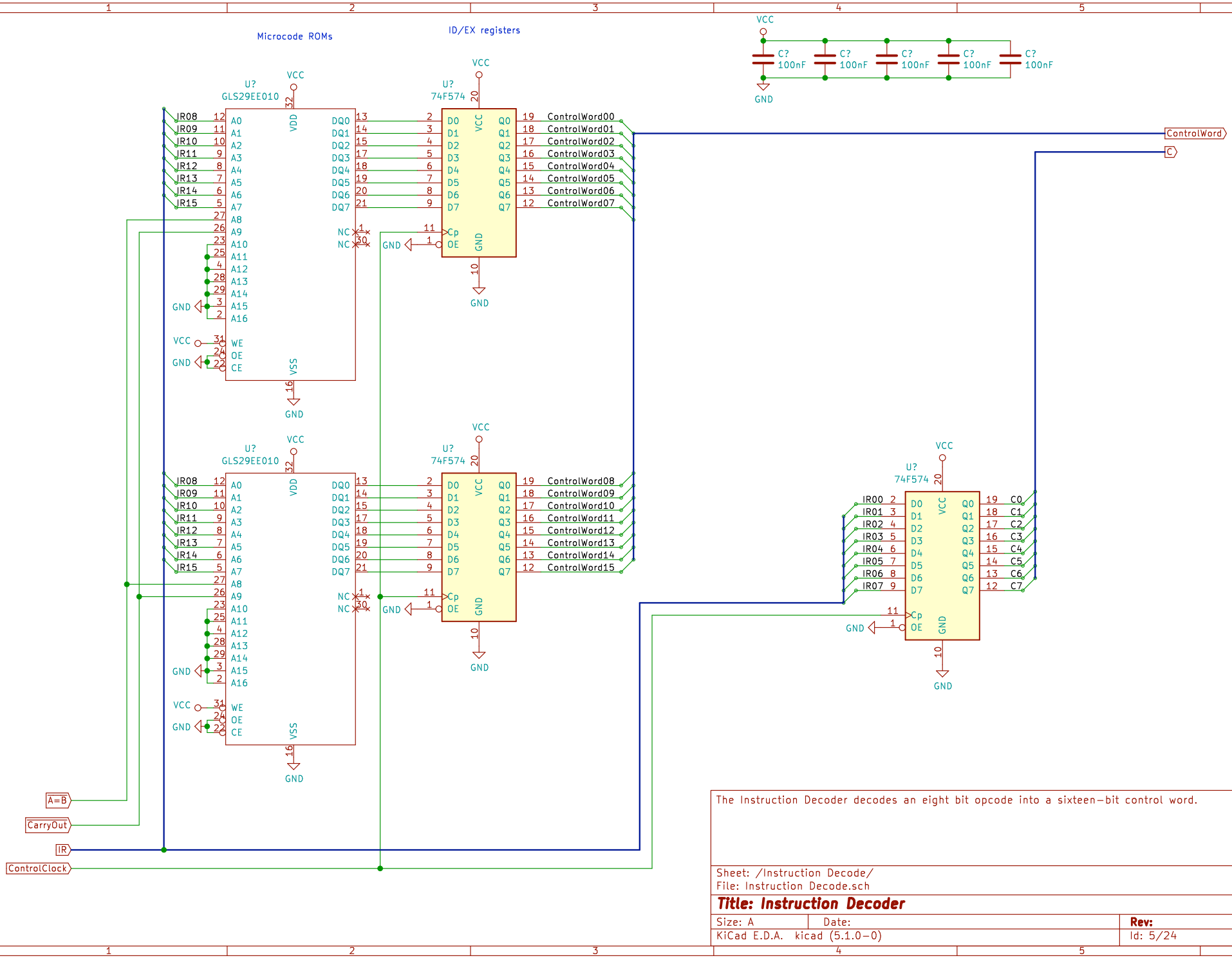
Sheet: /Power-on Reset/
File: Power-on Reset.sch

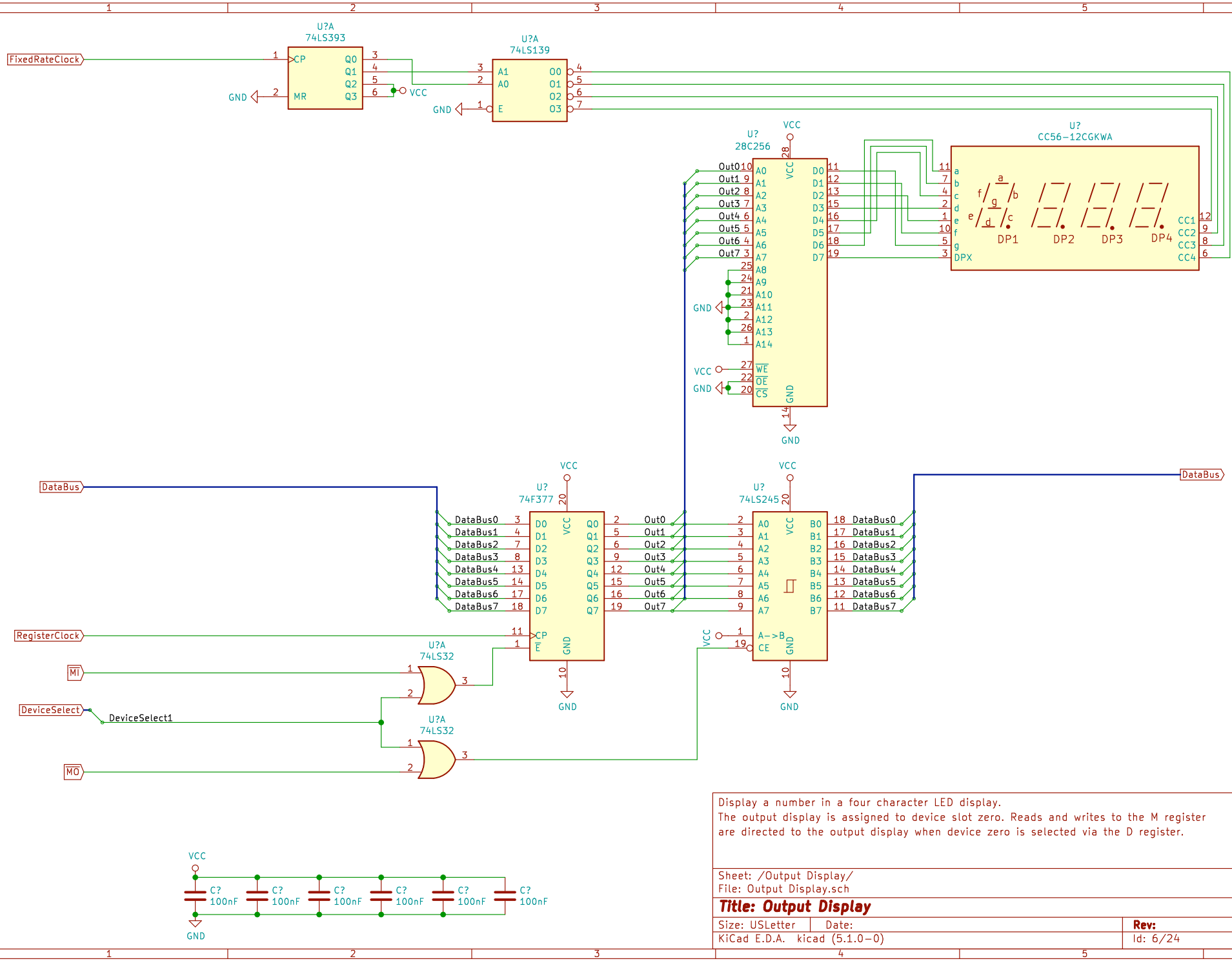
Title: Power-on Reset

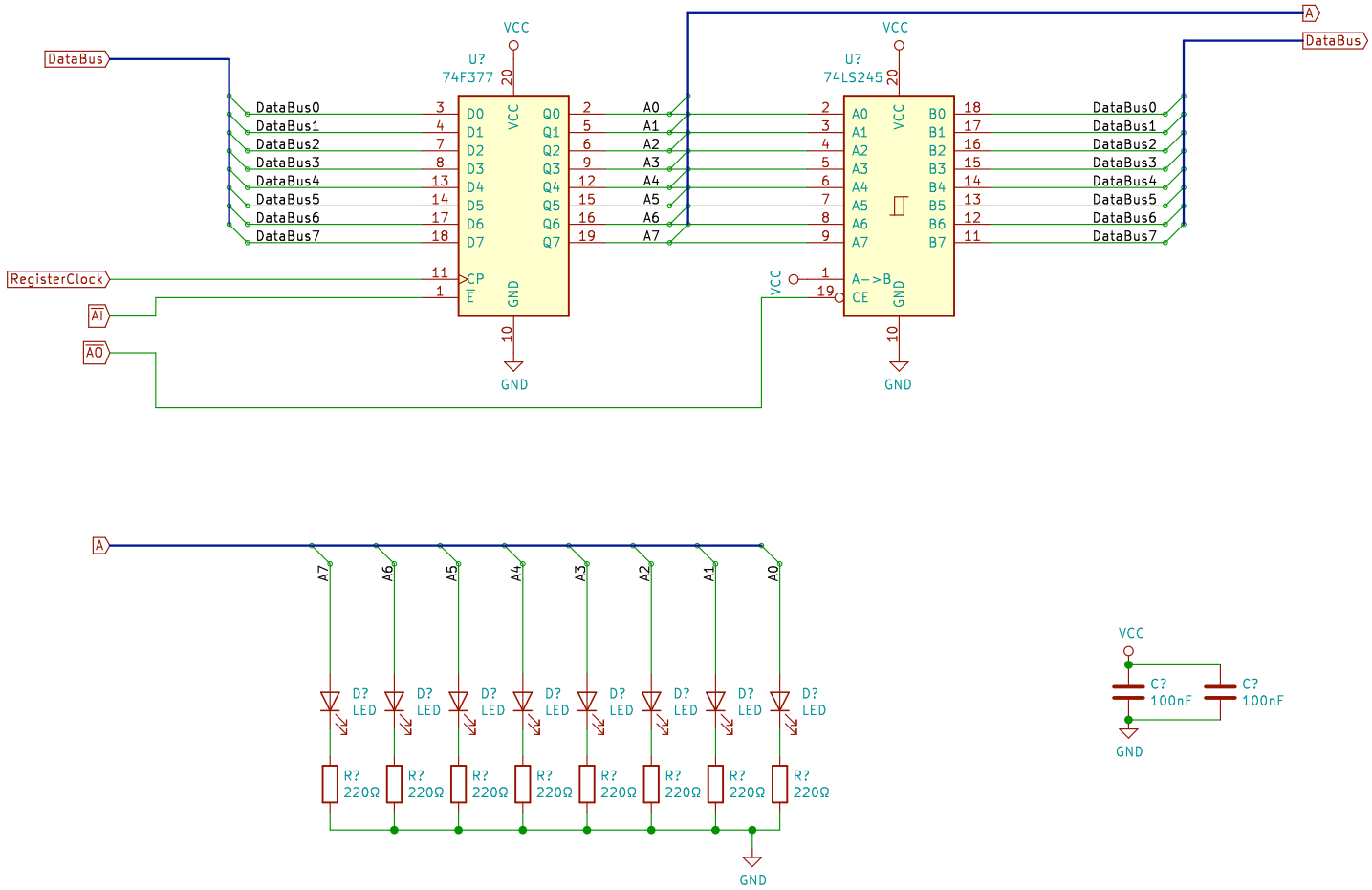
Size: A	Date:	Rev:
KiCad E.D.A. kicad (5.1.0-0)		Id: 3/24



Sheet: /Program Counter/	
File: Program Counter.sch	
Title: Program Counter	
Size: USLetter	Date:
KiCad E.D.A. kicad (5.1.0-0)	Id: 4/24


Microcode ROMs
ID/EX registers
U?
GLS29EE010
U?
74F574
VCC
GND
C?
100nF
IR08
IR09
IR10
IR11
IR12
IR13
IR14
IR15
ControlWord00
ControlWord01
ControlWord02
ControlWord03
ControlWord04
ControlWord05
ControlWord06
ControlWord07
ControlWord08
ControlWord09
ControlWord10
ControlWord11
ControlWord12
ControlWord13
ControlWord14
ControlWord15
ControlWord
C
IR00
IR01
IR02
IR03
IR04
IR05
IR06
IR07
C0
C1
C2
C3
C4
C5
C6
C7
A=B
CarryOut
IR
ControlClock
The Instruction Decoder decodes an eight bit opcode into a sixteen-bit control word.
Sheet: /Instruction Decode/
File: Instruction Decode.sch
Title: Instruction Decoder
Size: A
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 5/24


U?A
74LS393
FixedRateClock
CP
MR
Q0
Q1
Q2
Q3
GND
VCC
U?A
74LS139
A1
A0
E
O0
O1
O2
O3
U?
28C256
Out0
Out1
Out2
Out3
Out4
Out5
Out6
Out7
A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
WE
OE
CS
D0
D1
D2
D3
D4
D5
D6
D7
U?
CC56-12CGKWA
a
b
c
d
e
f
g
DP1
DP2
DP3
DP4
DPX
CC1
CC2
CC3
CC4
DataBus
U?
74F377
DataBus0
DataBus1
DataBus2
DataBus3
DataBus4
DataBus5
DataBus6
DataBus7
U?
74LS245
A->B
CE
B0
B1
B2
B3
B4
B5
B6
B7
RegisterClock
MI
DeviceSelect
DeviceSelect1
MO
U?A
74LS32
C?
100nF
Display a number in a four character LED display.
The output display is assigned to device slot zero. Reads and writes to the M register are directed to the output display when device zero is selected via the D register.
Sheet: /Output Display/
File: Output Display.sch
Title: Output Display
Size: USLetter
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 6/24

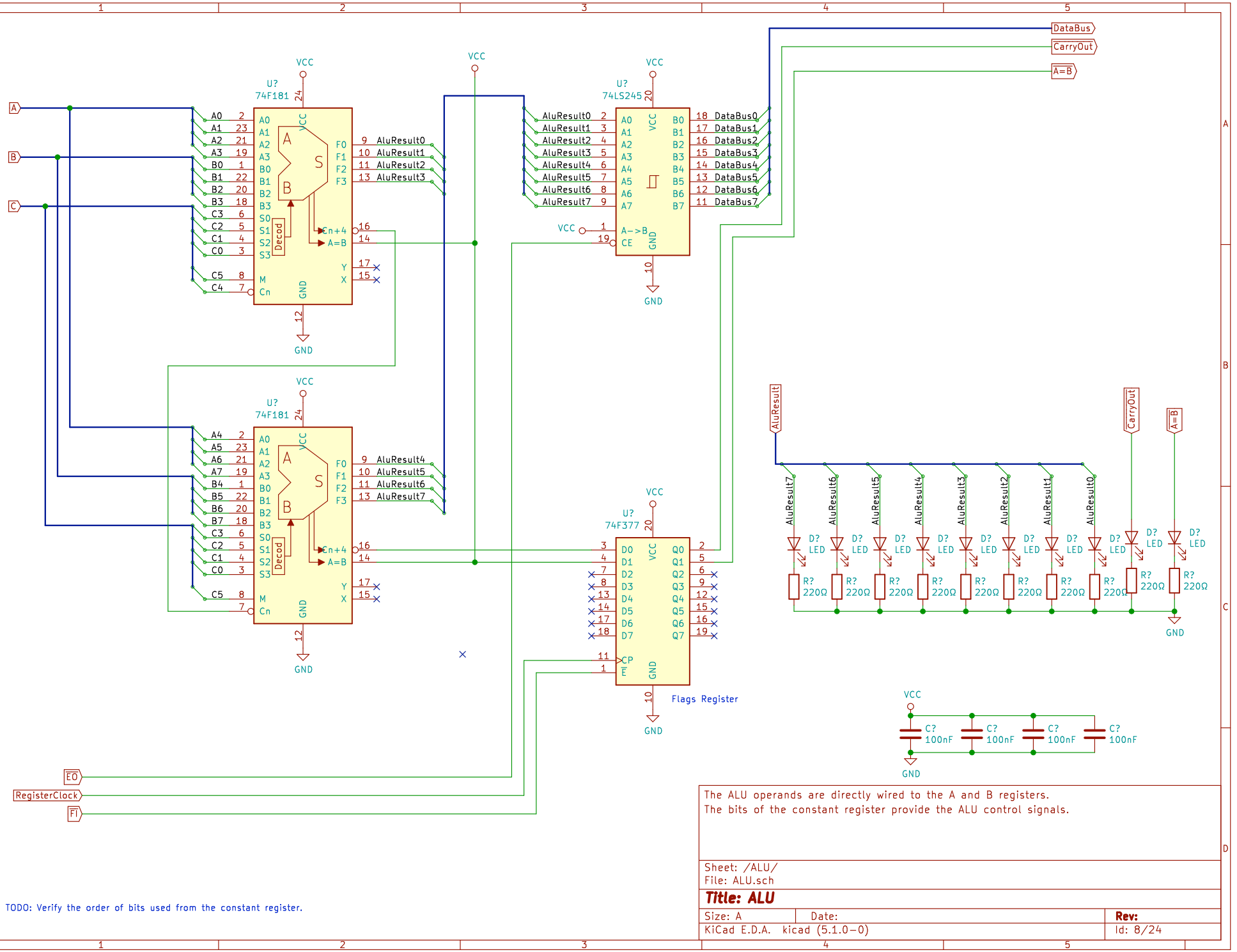


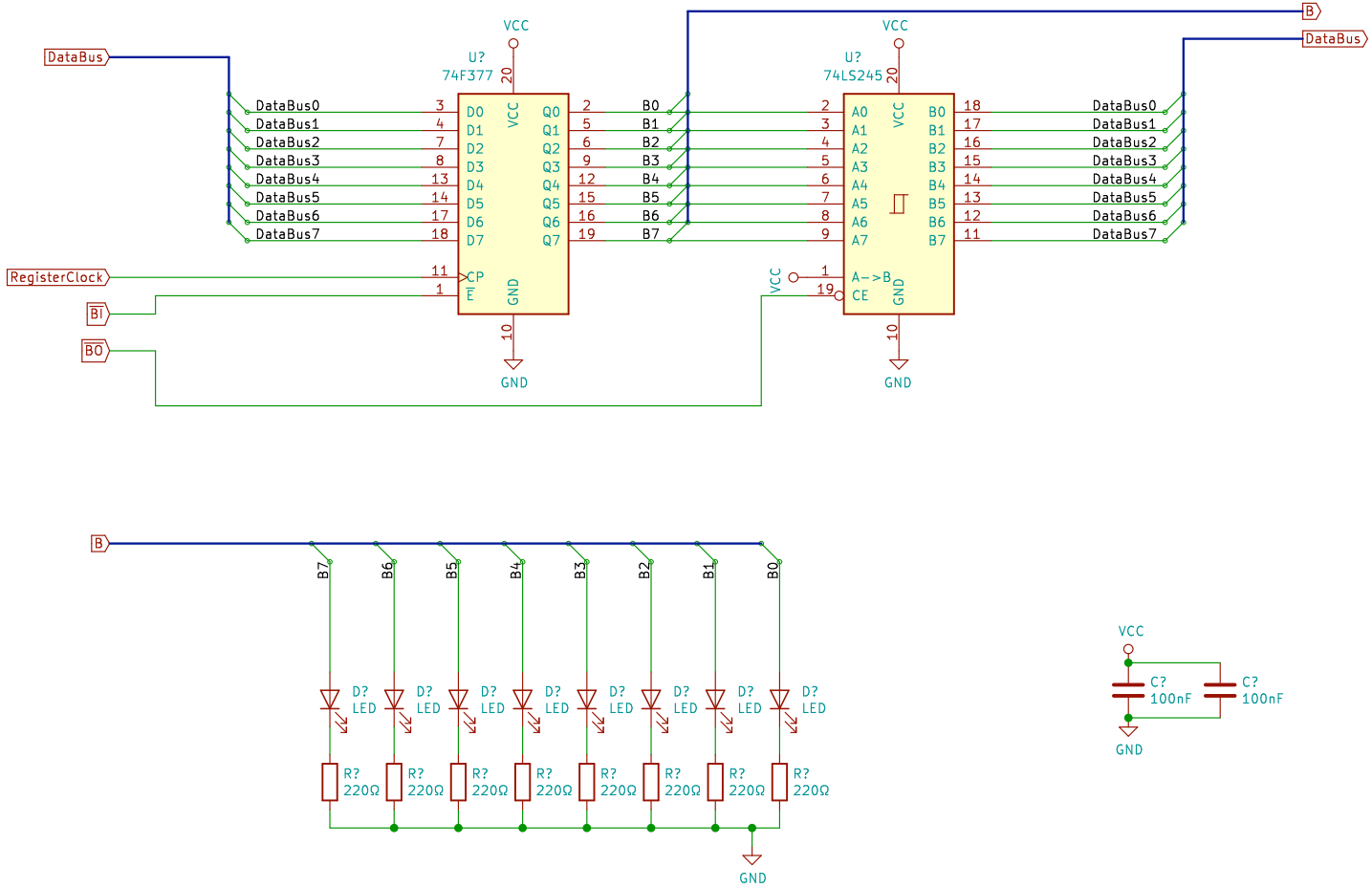
Register A is wired to the ALU's A operand.

Sheet: /Register A/
File: Register A.sch

Title: Register A

Size: A	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 7/24


DataBus
CarryOut
A=B
A
B
C
U?
74F181
VCC
GND
AluResult0
AluResult1
AluResult2
AluResult3
AluResult4
AluResult5
AluResult6
AluResult7
U?
74LS245
DataBus0
DataBus1
DataBus2
DataBus3
DataBus4
DataBus5
DataBus6
DataBus7
A->B
CE
U?
74F377
Flags Register
AluResult
LED
R?
220Ω
C?
100nF
EO
RegisterClock
FI
The ALU operands are directly wired to the A and B registers.
The bits of the constant register provide the ALU control signals.
Sheet: /ALU/
File: ALU.sch
Title: ALU
Size: A
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 8/24
TODO: Verify the order of bits used from the constant register.

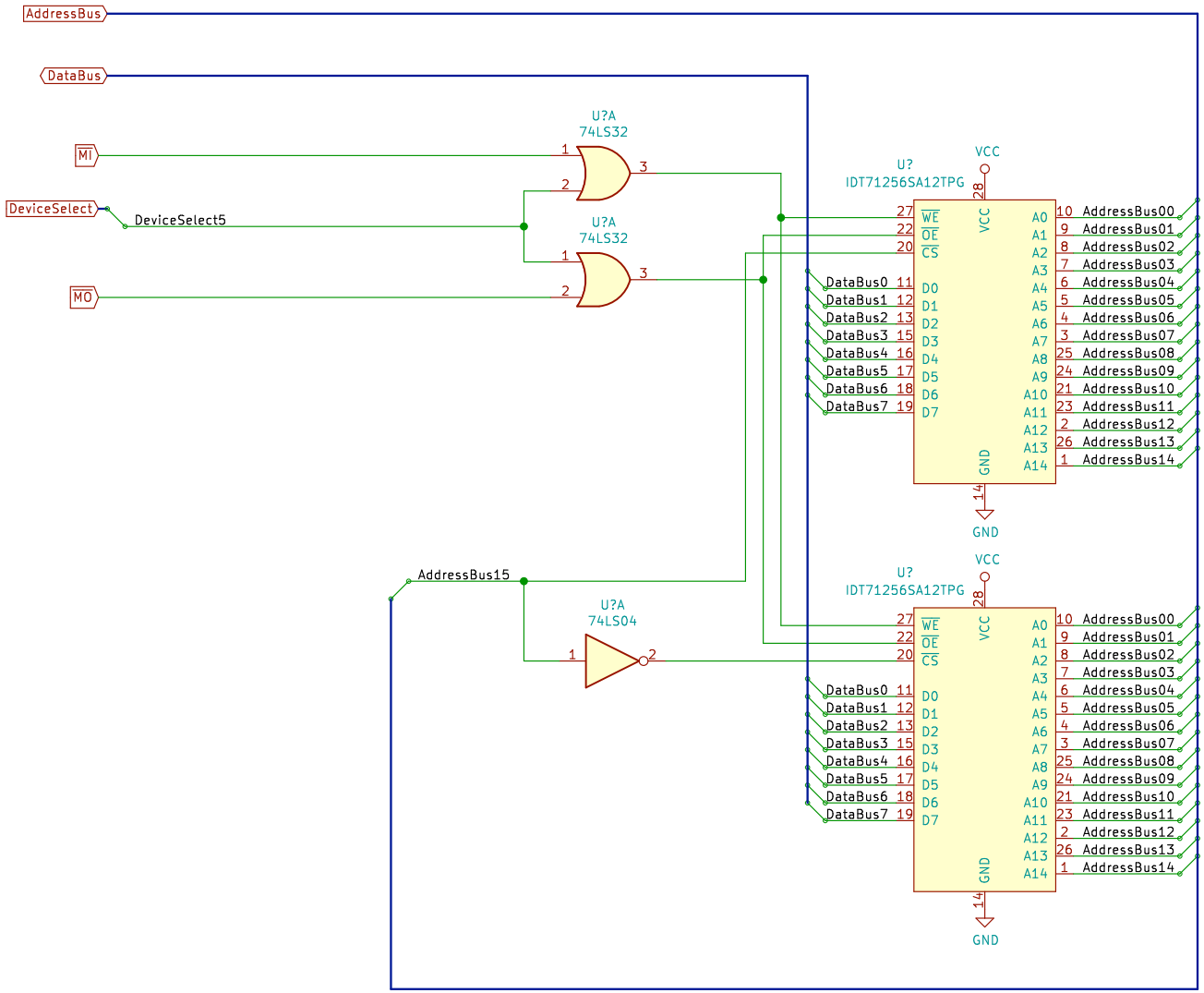


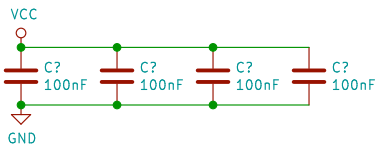
Register B is wired to the ALU's B operand.

Sheet: /Register B/
File: Register B.sch

Title: Register B

Size: A	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 9/24



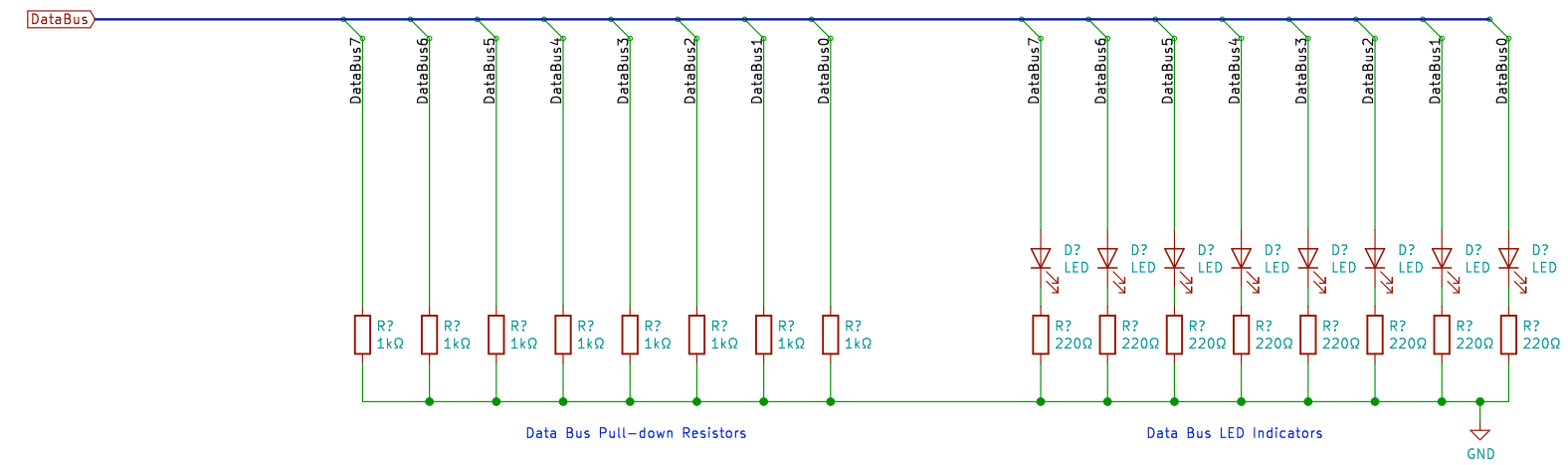


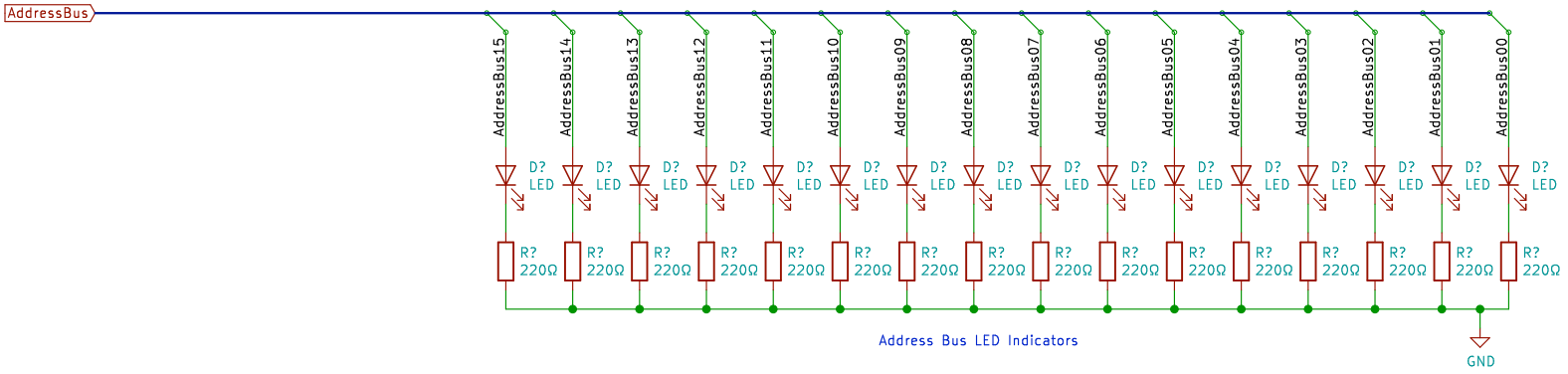
64KB Data RAM is connected to device slot five.
Reads and writes to the M register are directed to Data RAM when device 5 is selected via the D register.
The MSB of the address selects between two 32KB SRAM ICs.

Sheet: /Data RAM/
File: Data RAM.sch

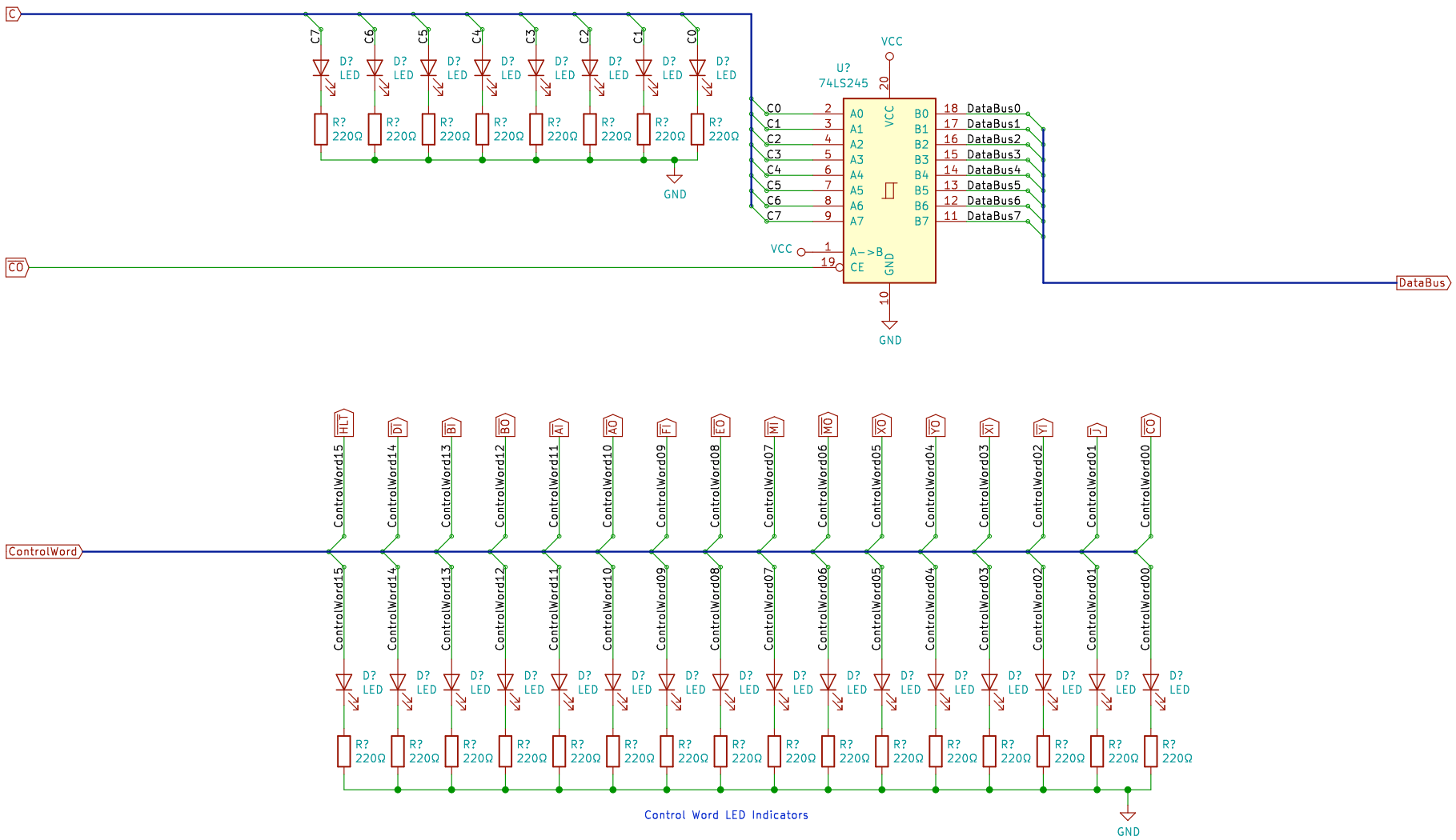
Title: Data RAM

Size: USLetter	Date:	Rev:
KiCad E.D.A. kicad (5.1.0-0)		Id: 10/24



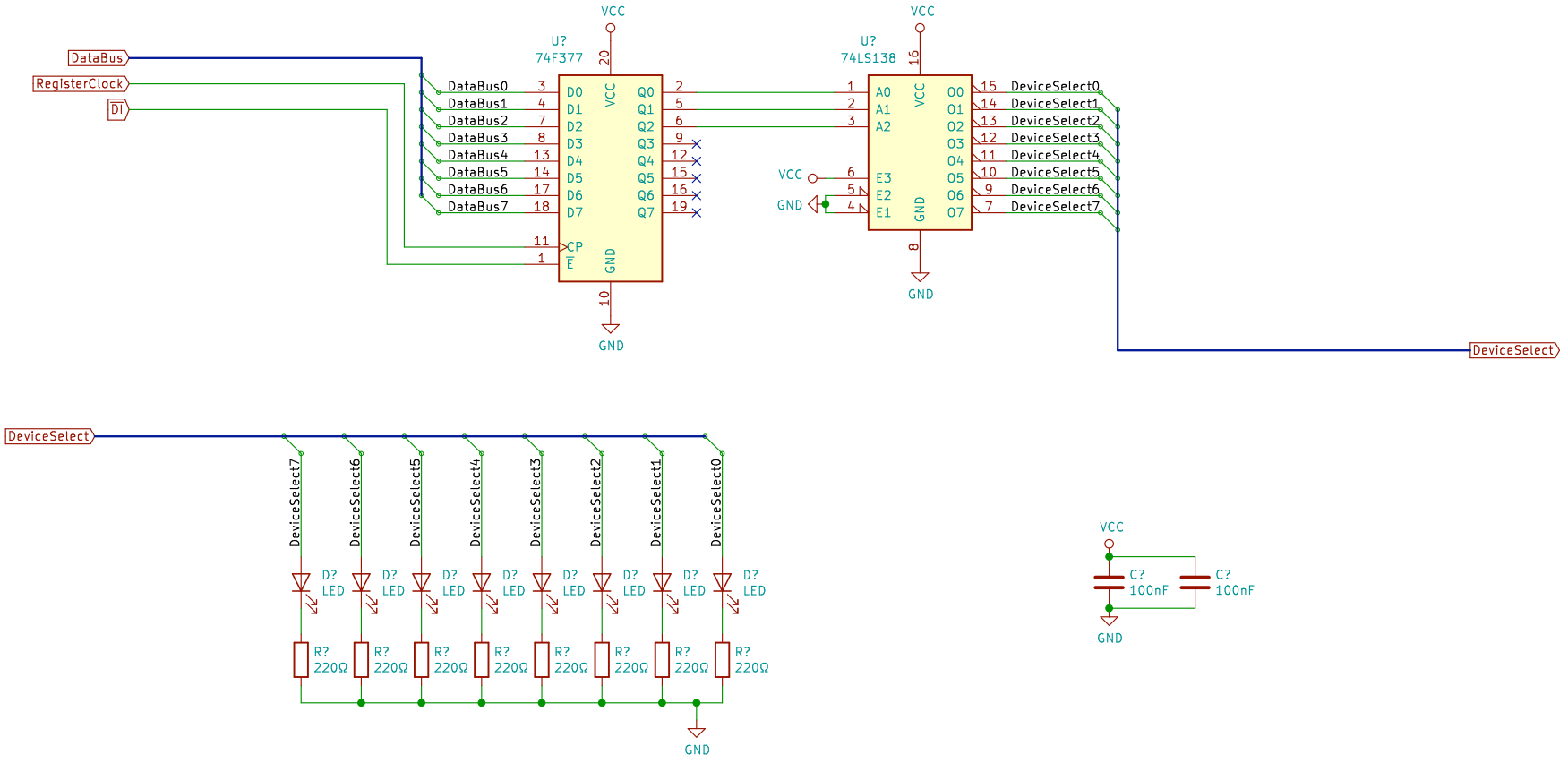


Sheet: /Bus Display/		
File: Bus Display.sch		
Title: Bus Display		
Size: A	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 11/24



The lower eight bits of the instruction register can be expressed onto the data bus as a constant value.

Sheet: /Execute/ File: Execute.sch		
Title: Execute		
Size: A	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 12/24



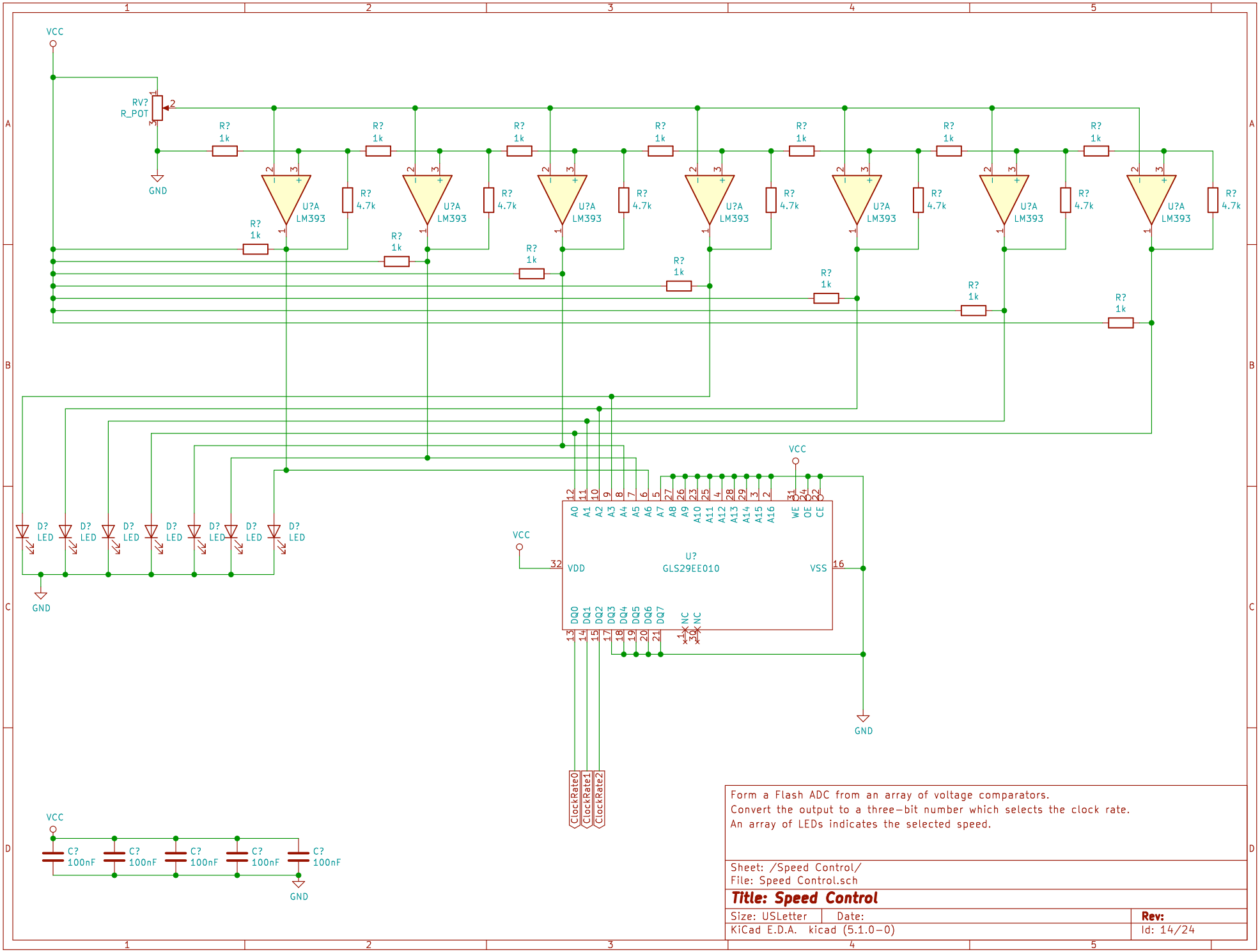
Register D controls the device select lines for peripherals and memory.

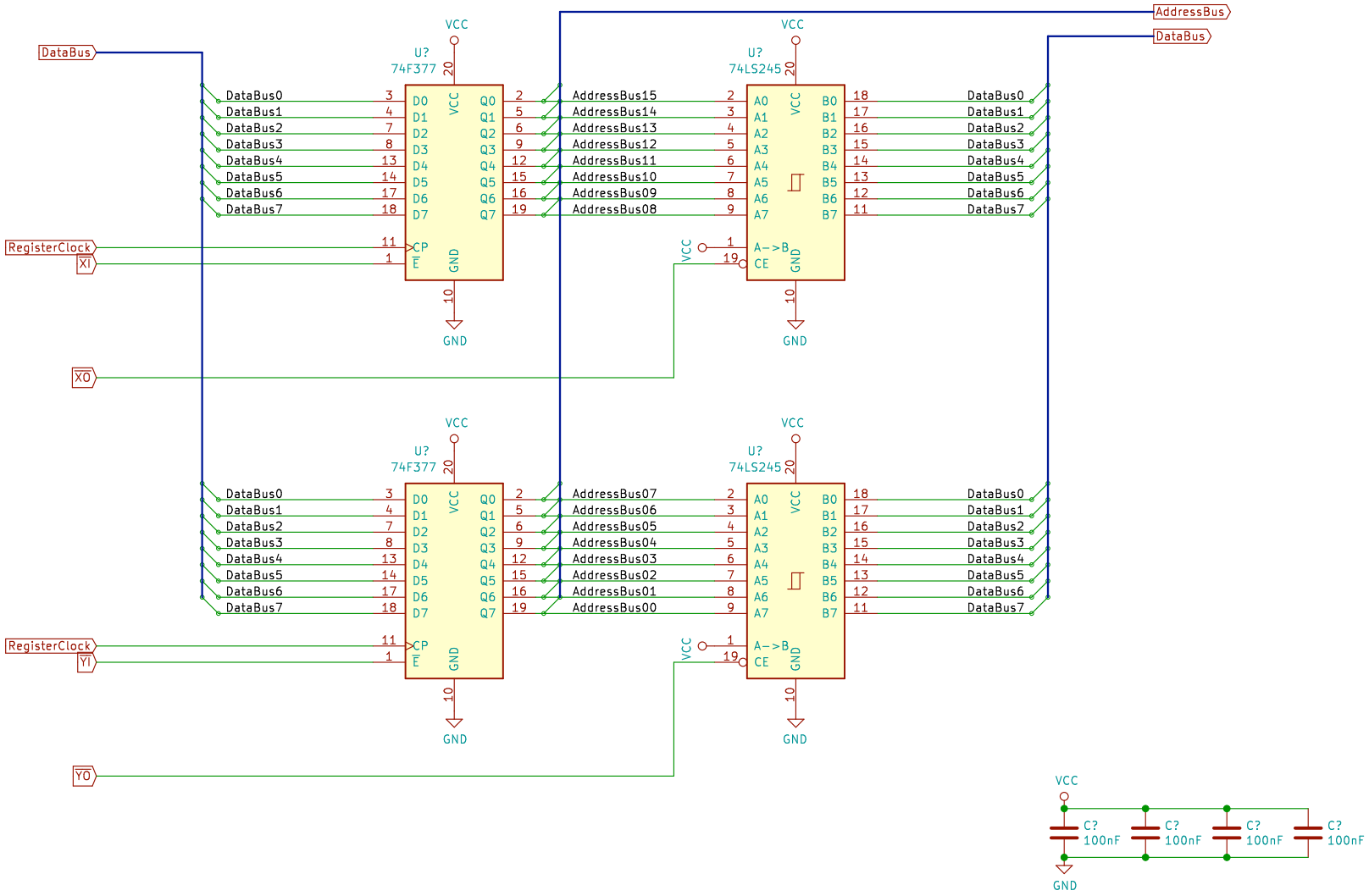
Sheet: /Register D/
File: Register D.sch

Title: Register D

Size: A | Date: | **Rev:**

KiCad E.D.A. kicad (5.1.0-0) | Id: 13/24


VCC
RV?
R_POT
GND
R?
1k
4.7k
U?A
LM393
D?
LED
U?
GLS29EE010
VDD
VSS
A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
A15
A16
WE
OE
CE
DQ0
DQ1
DQ2
DQ3
DQ4
DQ5
DQ6
DQ7
NC
ClockRate0
ClockRate1
ClockRate2
C?
100nF
Form a Flash ADC from an array of voltage comparators.
Convert the output to a three-bit number which selects the clock rate.
An array of LEDs indicates the selected speed.
Sheet: /Speed Control/
File: Speed Control.sch
Title: Speed Control
Size: USLetter
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 14/24

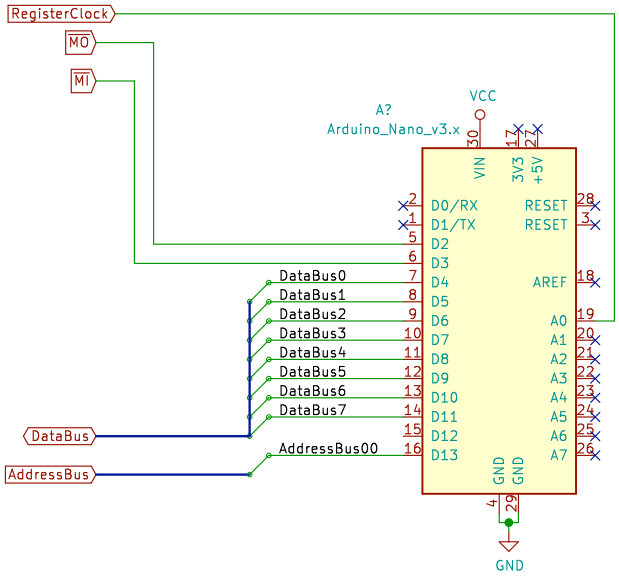


The XY register pair sets the value on the sixteen-bit Address Bus.

Sheet: /Register XY/
File: Register XY.sch

Title:

Size: USLetter | Date: | **Rev:**
KiCad E.D.A. kicad (5.1.0-0) | Id: 15/24



An Arduino Nano serves as a serial interface module.

Sheet: /Serial/
File: Serial.sch

Title:

Size: USLetter | Date: | **Rev:**
KiCad E.D.A. kicad (5.1.0-0) | Id: 16/24

Sheet: PC/IF

File: PC_IF.sch

Sheet: Instruction RAM Address

File: Instruction RAM Address.sch

Sheet: Instruction ROM

File: Instruction ROM.sch

Sheet: Instruction RAM

File: Instruction RAM.sch

Sheet: Instruction Register

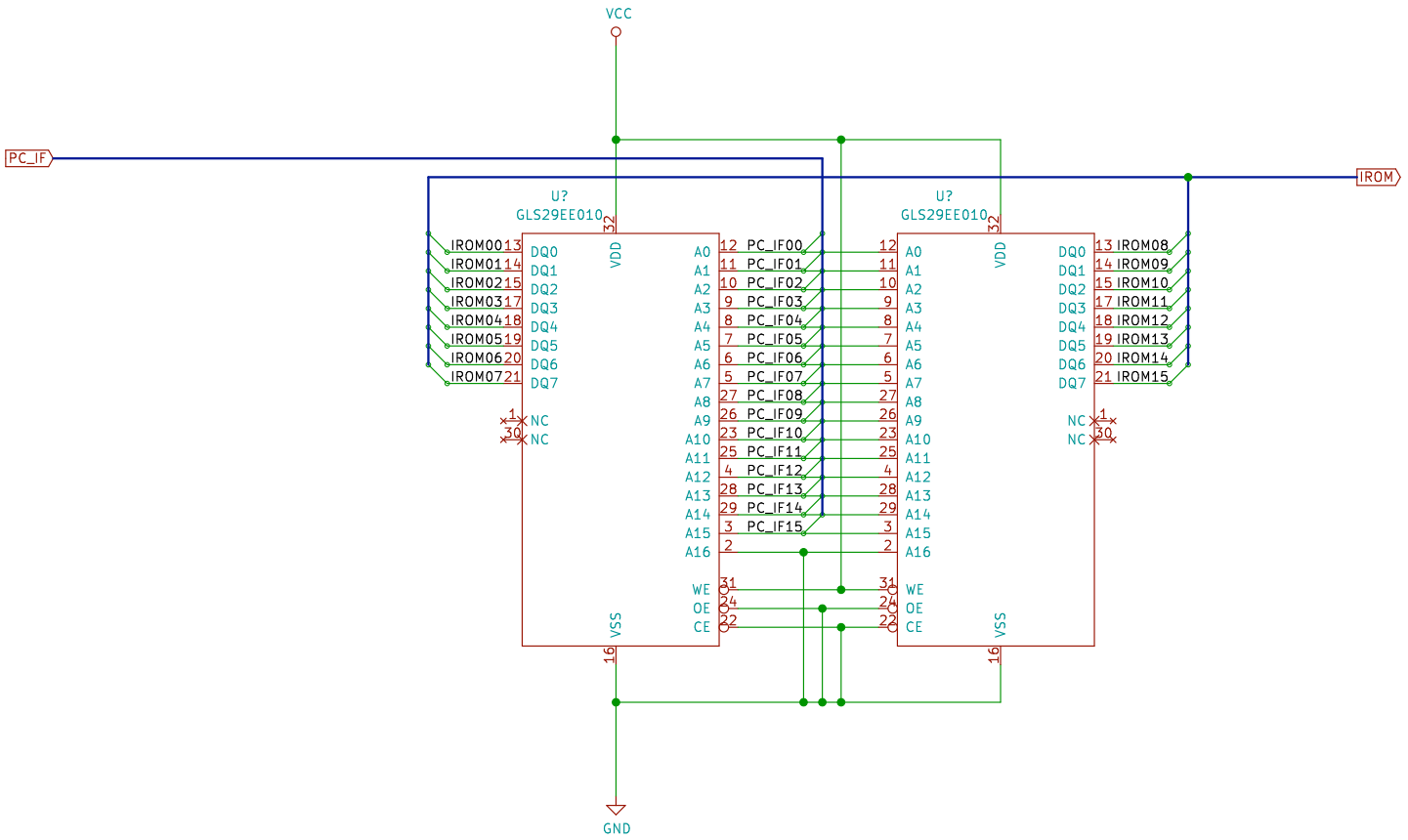
File: Instruction Register.sch

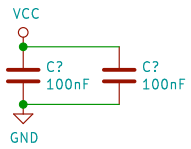
Instructions can be fetched from either Instruction ROM or Instruction RAM.
The lower 32KB of the address space is mapped to ROM, the remainder to RAM.

Sheet: /Instruction Fetch/
File: Instruction Fetch.sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 17/24



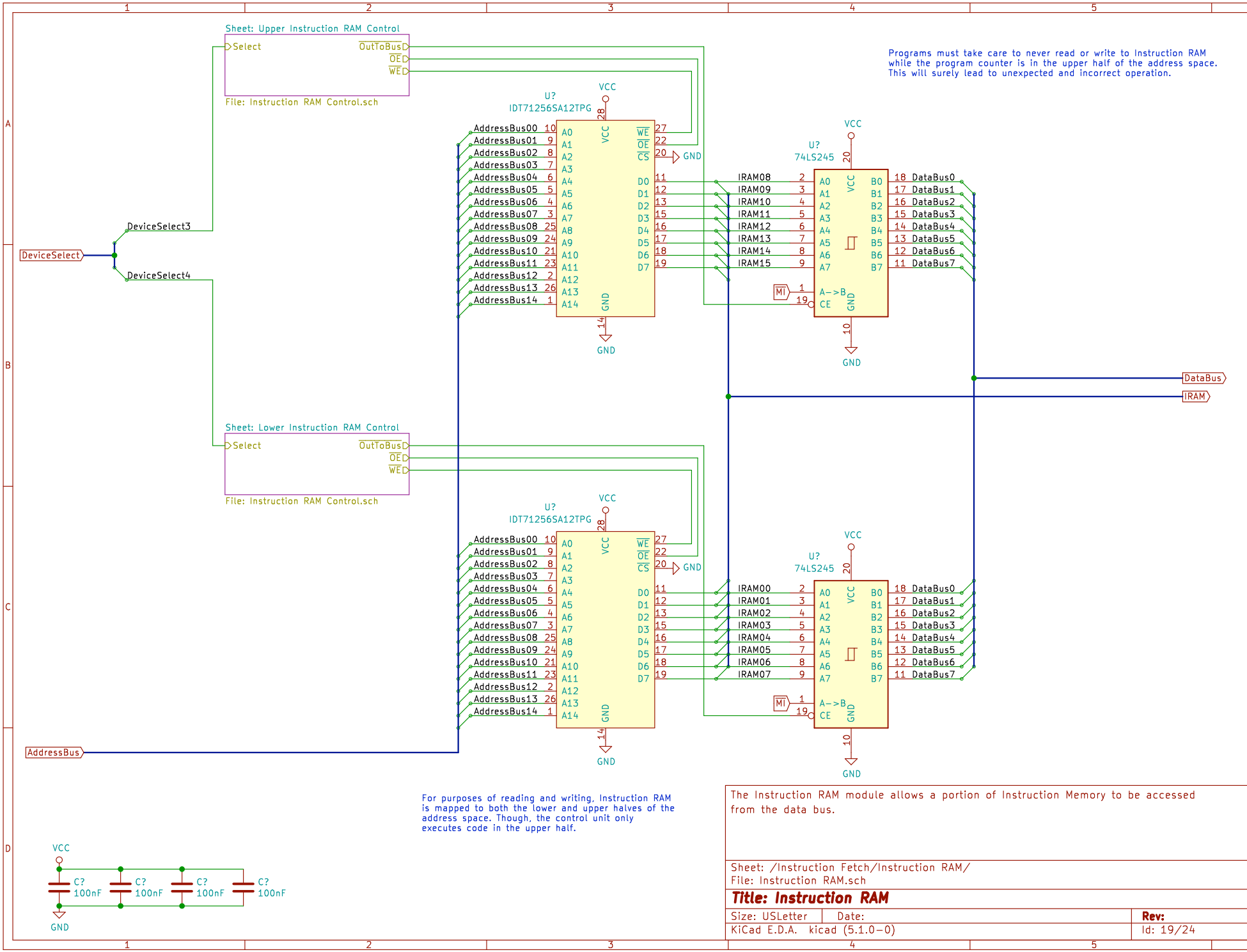


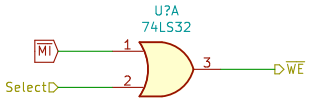
16-bit instructions are read from ROM at the address specified by the Program Counter.

Sheet: /Instruction Fetch/Instruction ROM/
File: Instruction ROM.sch

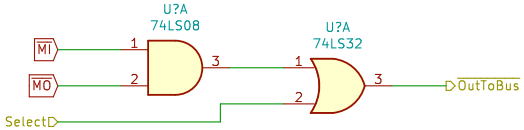
Title: Instruction ROM

Size: USLetter	Date:	Rev:
KiCad E.D.A. kicad (5.1.0-0)		Id: 18/24


Sheet: Upper Instruction RAM Control
Select
OutToBus
OE
WE
File: Instruction RAM Control.sch
Programs must take care to never read or write to Instruction RAM while the program counter is in the upper half of the address space. This will surely lead to unexpected and incorrect operation.
U?
IDT71256SA12TPG
VCC
GND
AddressBus00
AddressBus01
AddressBus02
AddressBus03
AddressBus04
AddressBus05
AddressBus06
AddressBus07
AddressBus08
AddressBus09
AddressBus10
AddressBus11
AddressBus12
AddressBus13
AddressBus14
U?
74LS245
IRAM08
IRAM09
IRAM10
IRAM11
IRAM12
IRAM13
IRAM14
IRAM15
DataBus0
DataBus1
DataBus2
DataBus3
DataBus4
DataBus5
DataBus6
DataBus7
DeviceSelect3
DeviceSelect
DeviceSelect4
DataBus
IRAM
Sheet: Lower Instruction RAM Control
Select
OutToBus
OE
WE
File: Instruction RAM Control.sch
U?
IDT71256SA12TPG
U?
74LS245
IRAM00
IRAM01
IRAM02
IRAM03
IRAM04
IRAM05
IRAM06
IRAM07
AddressBus
For purposes of reading and writing, Instruction RAM is mapped to both the lower and upper halves of the address space. Though, the control unit only executes code in the upper half.
C?
100nF
The Instruction RAM module allows a portion of Instruction Memory to be accessed from the data bus.
Sheet: /Instruction Fetch/Instruction RAM/
File: Instruction RAM.sch
Title: Instruction RAM
Size: USLetter
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 19/24

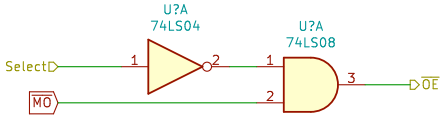


Only write to the Instruction SRAM when the MI signal is active and the appropriate plane of Instruction RAM is selected.



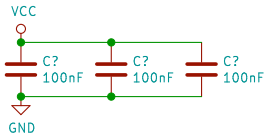
The bus transceiver should connect the Instruction SRAM to the Data Bus when the plane is selected and wither MO or MI is active.

The direction of the bus transceiver must also be set appropriately. This is done elsewhere.



Always output from Instruction SRAM when the plane of Instruction RAM is deselected. This allows RAM to be read for use in IF.

Else, if the plane is selected then only read from SRAM when the MO signal is active.

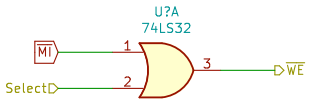


Logic for the control signals which drive Instruction RAM.

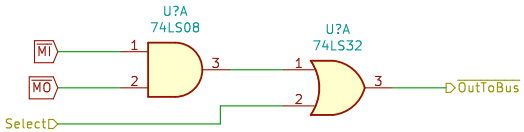
Sheet: /Instruction Fetch/Instruction RAM/Lower Instruction RAM Control/
File: Instruction RAM Control.sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 20/24

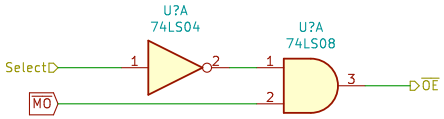


Only write to the Instruction SRAM when the MI signal is active and the appropriate plane of Instruction RAM is selected.



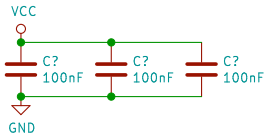
The bus transceiver should connect the Instruction SRAM to the Data Bus when the plane is selected and wither MO or MI is active.

The direction of the bus transceiver must also be set appropriately. This is done elsewhere.



Always output from Instruction SRAM when the plane of Instruction RAM is deselected. This allows RAM to be read for use in IF.

Else, if the plane is selected then only read from SRAM when the MO signal is active.

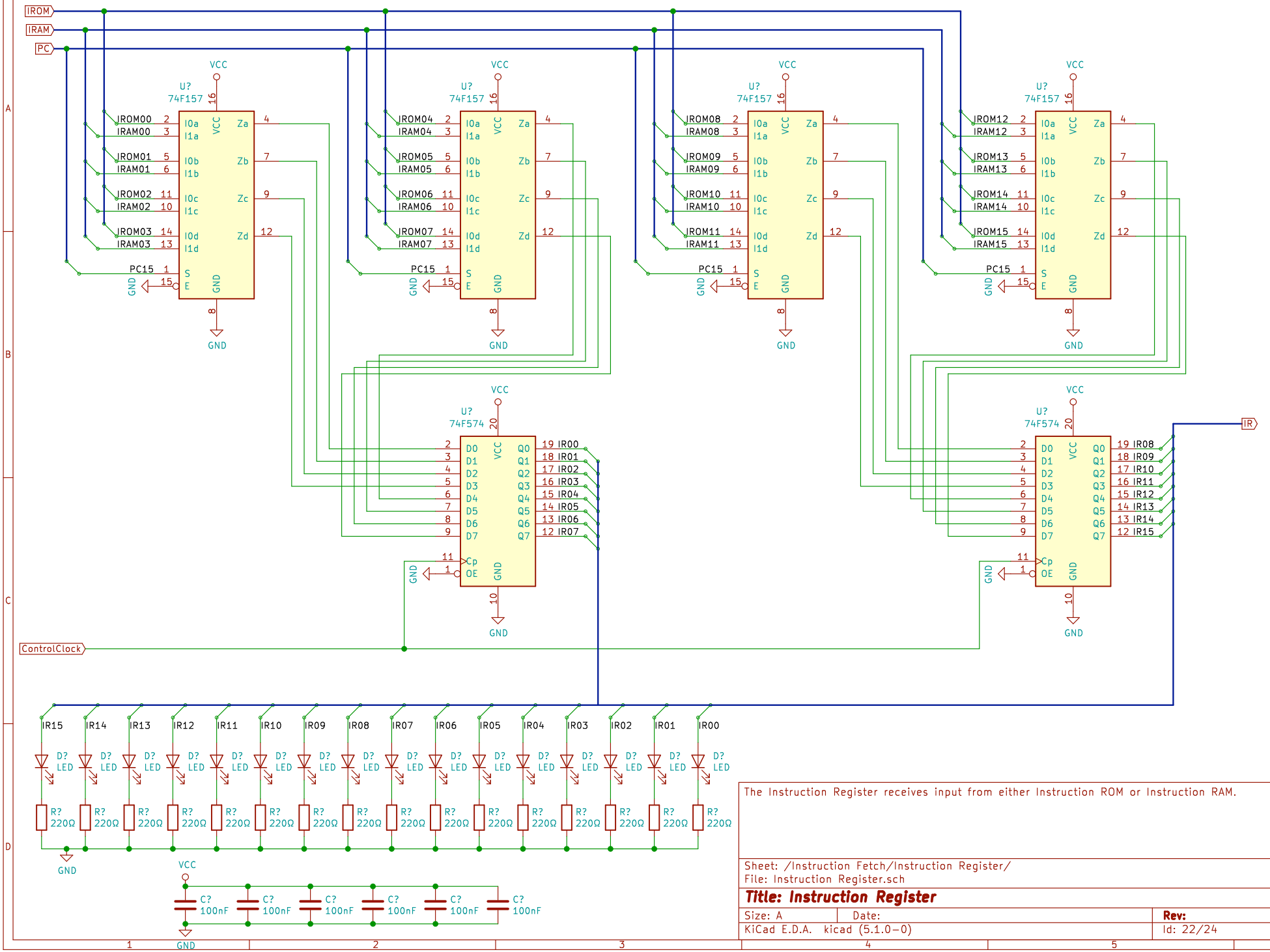


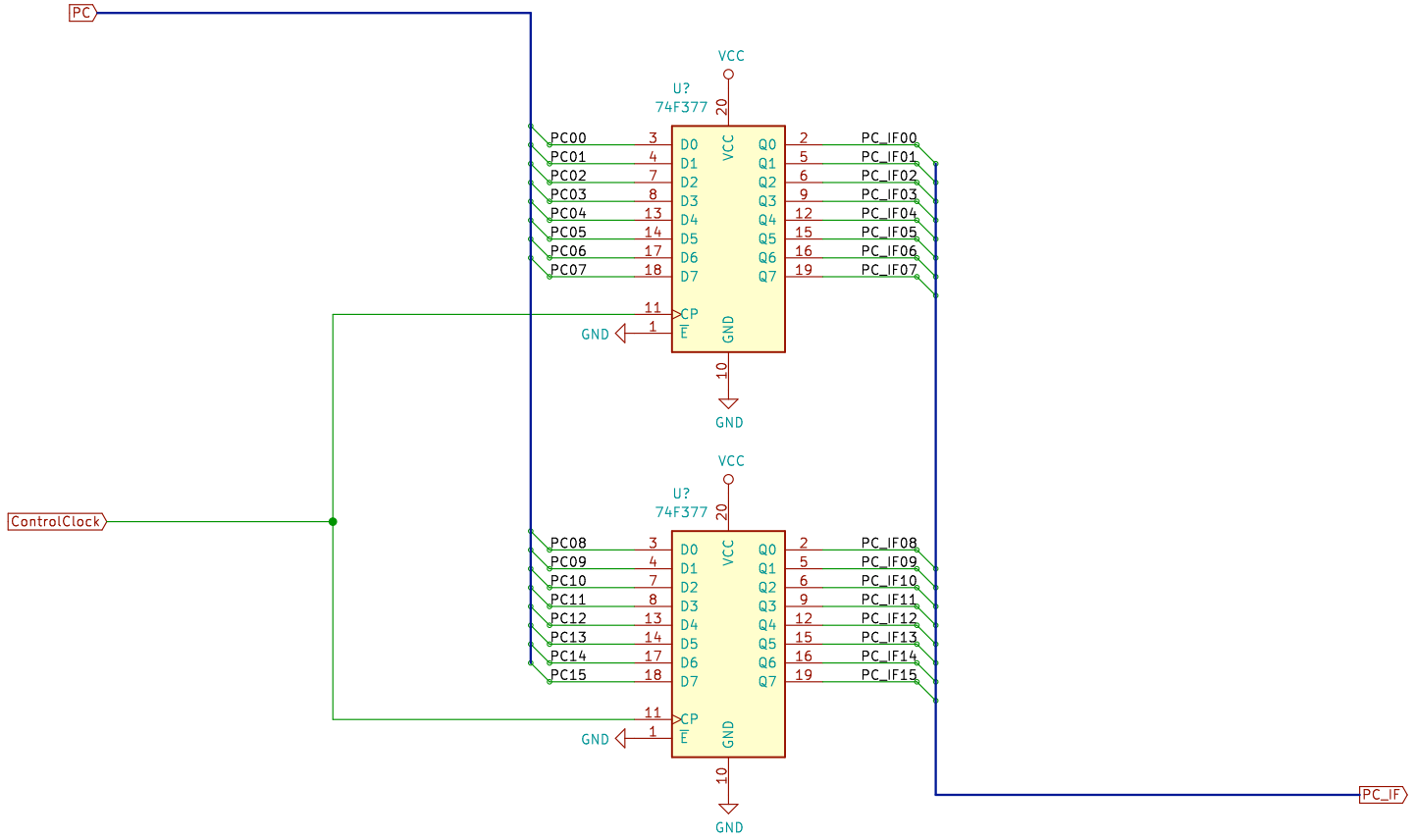
Logic for the control signals which drive Instruction RAM.

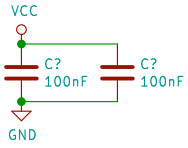
Sheet: /Instruction Fetch/Instruction RAM/Upper Instruction RAM Control/
File: Instruction RAM Control.sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 21/24


IROM
IRAM
PC
U?
74F157
VCC
GND
IROM00
IRAM00
IROM01
IRAM01
IROM02
IRAM02
IROM03
IRAM03
PC15
IROM04
IRAM04
IROM05
IRAM05
IROM06
IRAM06
IROM07
IRAM07
IROM08
IRAM08
IROM09
IRAM09
IROM10
IRAM10
IROM11
IRAM11
IROM12
IRAM12
IROM13
IRAM13
IROM14
IRAM14
IROM15
IRAM15
U?
74F574
IR
IR00
IR01
IR02
IR03
IR04
IR05
IR06
IR07
IR08
IR09
IR10
IR11
IR12
IR13
IR14
IR15
ControlClock
D?
LED
R?
220Ω
C?
100nF
The Instruction Register receives input from either Instruction ROM or Instruction RAM.
Sheet: /Instruction Fetch/Instruction Register/
File: Instruction Register.sch
Title: Instruction Register
Size: A
Date:
Rev:
KiCad E.D.A. kicad (5.1.0-0)
Id: 22/24



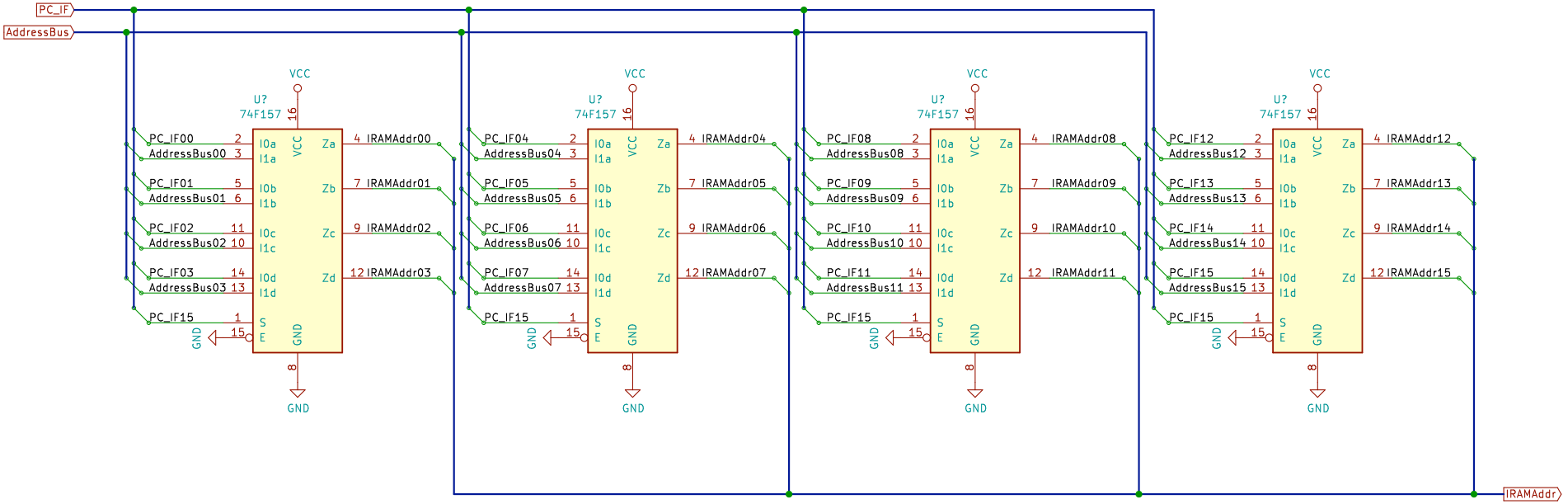


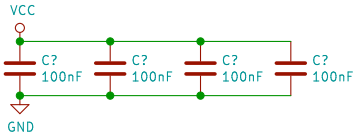
The PC/IF register sits between the program counter and the Instruction Fetch stage. This alleviates timing constraints related to the phase shift between the register clock and the control clock.

Sheet: /Instruction Fetch/PC/IF/
File: PC_IF.sch

Title:

Size: USLetter	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 23/24





Instruction RAM uses an address from either the program counter or the address bus.
If the program counter is in the upper 32KB of the address space then use the program counter as we're running in Instruction RAM.

Sheet: /Instruction Fetch/Instruction RAM Address/
File: Instruction RAM Address.sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. kicad (5.1.0-0)		Id: 24/24