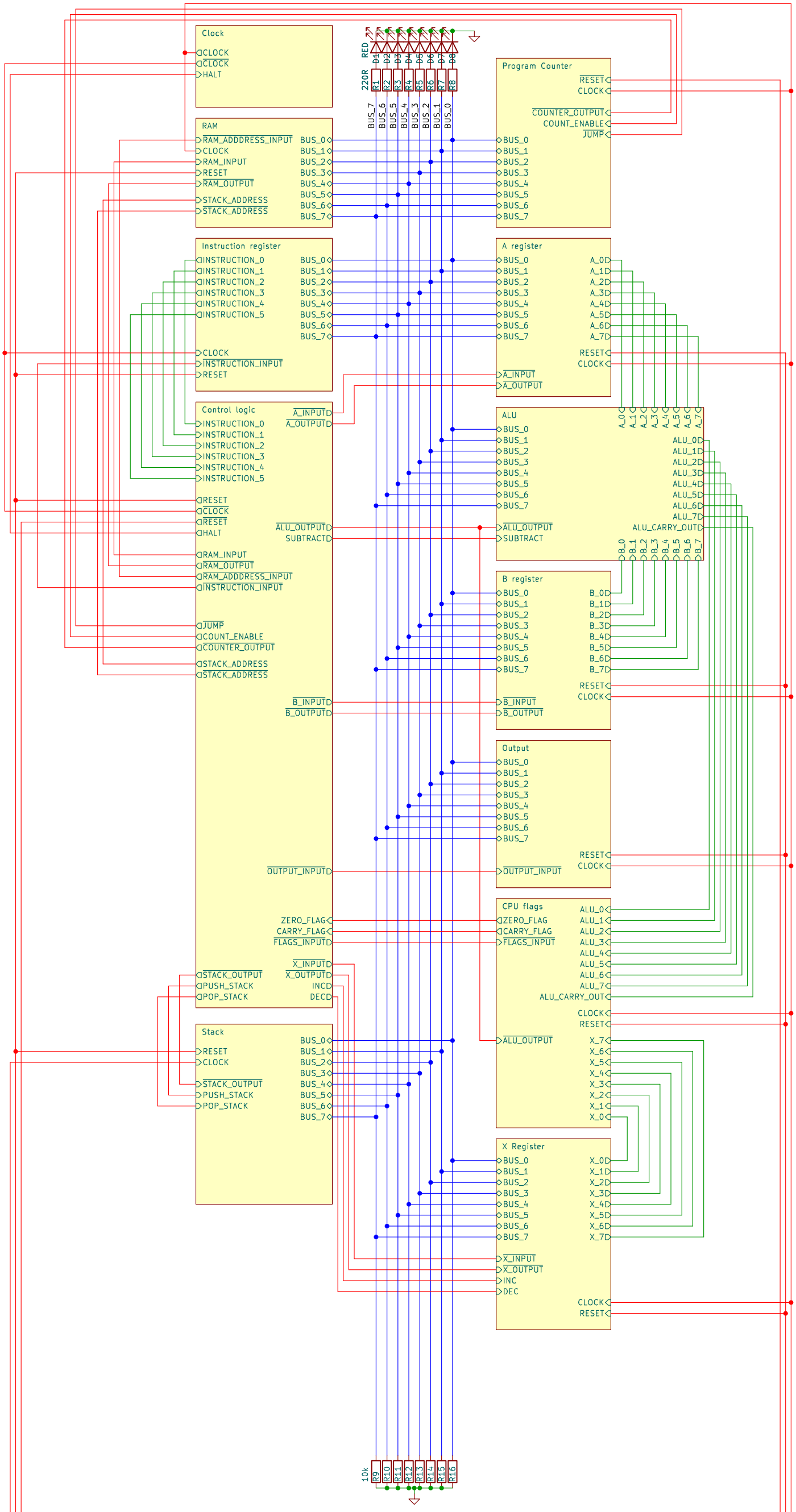
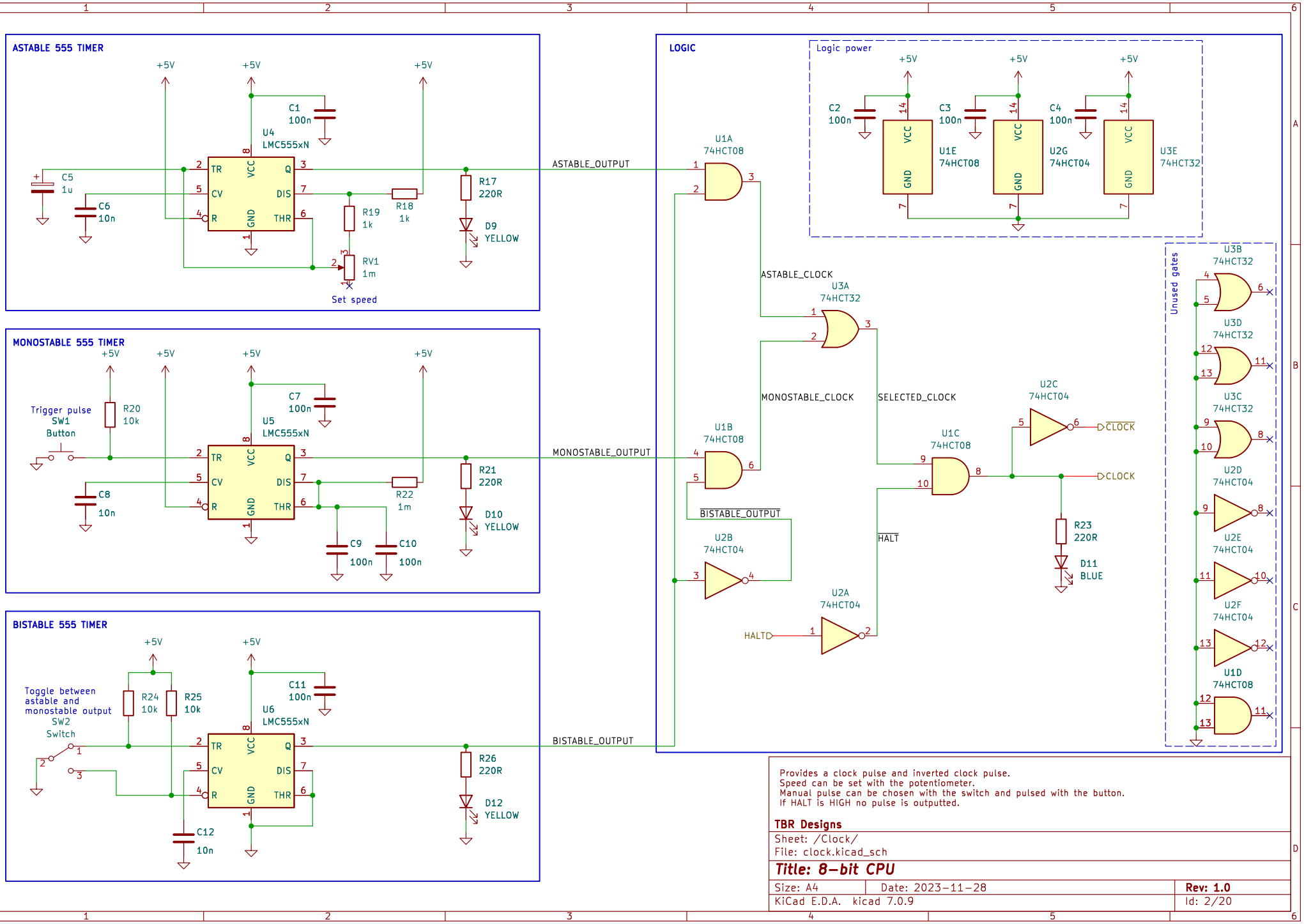
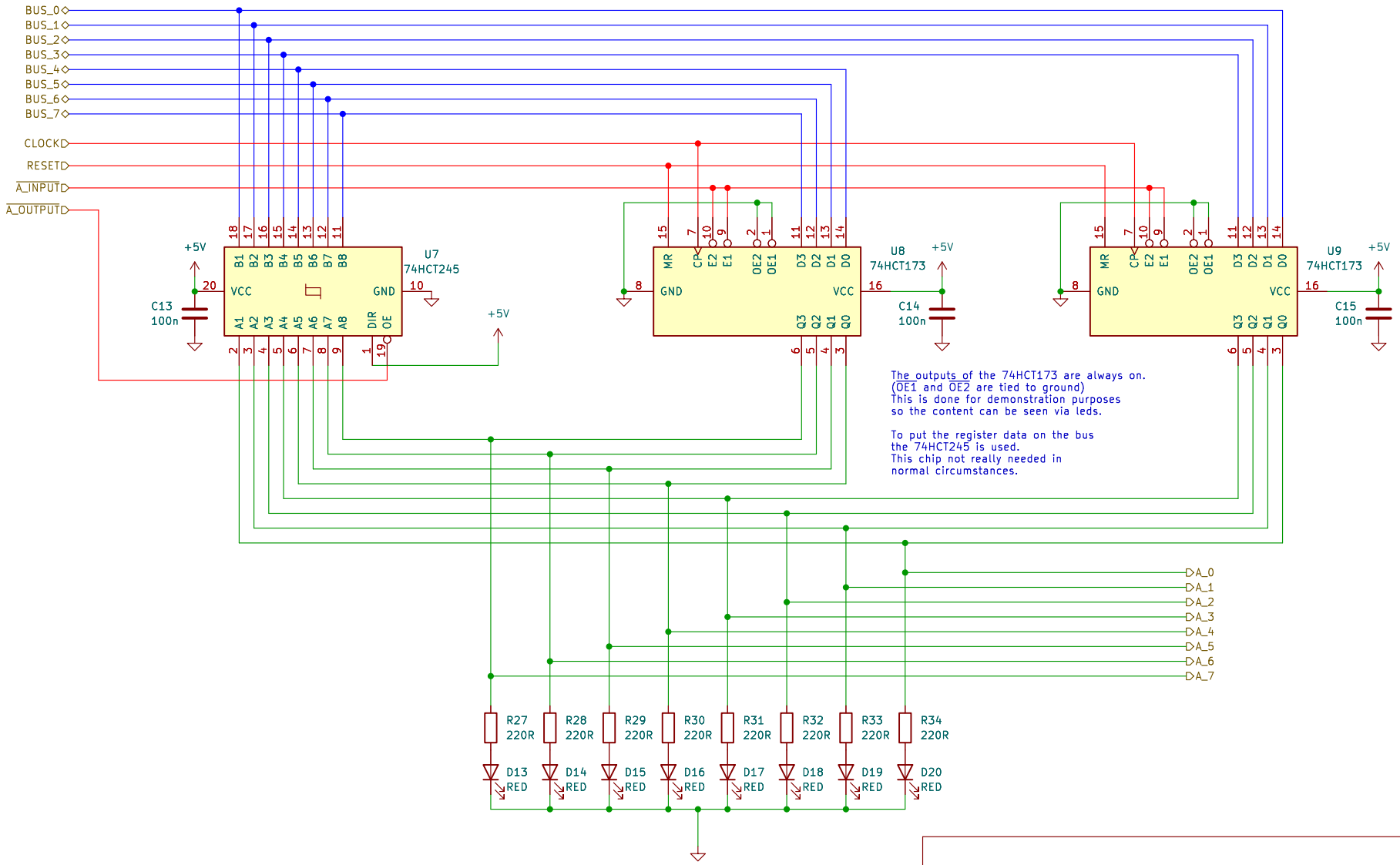


TBR Designs		
Sheet: /		
File: 8-bit-cpu.kicad_sch		
Title: 8-bit CPU		
Size: A3	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 1/20


ASTABLE 555 TIMER
+5V
C1
100n
U4
LMC555xN
C5
1u
C6
10n
R19
1k
R18
1k
R17
220R
D9
YELLOW
RV1
1m
Set speed
ASTABLE_OUTPUT
MONOSTABLE 555 TIMER
Trigger pulse
SW1
Button
R20
10k
C7
100n
U5
LMC555xN
C8
10n
R22
1m
R21
220R
D10
YELLOW
C9
100n
C10
100n
MONOSTABLE_OUTPUT
BISTABLE 555 TIMER
Toggle between
astable and
monostable output
SW2
Switch
R24
10k
R25
10k
C11
100n
U6
LMC555xN
C12
10n
R26
220R
D12
YELLOW
BISTABLE_OUTPUT
LOGIC
Logic power
C2
100n
C3
100n
C4
100n
U1E
74HCT08
U2G
74HCT04
U3E
74HCT32
U1A
74HCT08
ASTABLE_CLOCK
U3A
74HCT32
MONOSTABLE_CLOCK
U1B
74HCT08
BISTABLE_OUTPUT
U2B
74HCT04
SELECTED_CLOCK
U1C
74HCT08
U2C
74HCT04
CLOCK
CLOCK
R23
220R
D11
BLUE
HALT
U2A
74HCT04
HALT
Unused gates
U3B
74HCT32
U3D
74HCT32
U3C
74HCT32
U2D
74HCT04
U2E
74HCT04
U2F
74HCT04
U1D
74HCT08
Provides a clock pulse and inverted clock pulse.
Speed can be set with the potentiometer.
Manual pulse can be chosen with the switch and pulsed with the button.
If HALT is HIGH no pulse is outputted.
TBR Designs
Sheet: /Clock/
File: clock.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 2/20

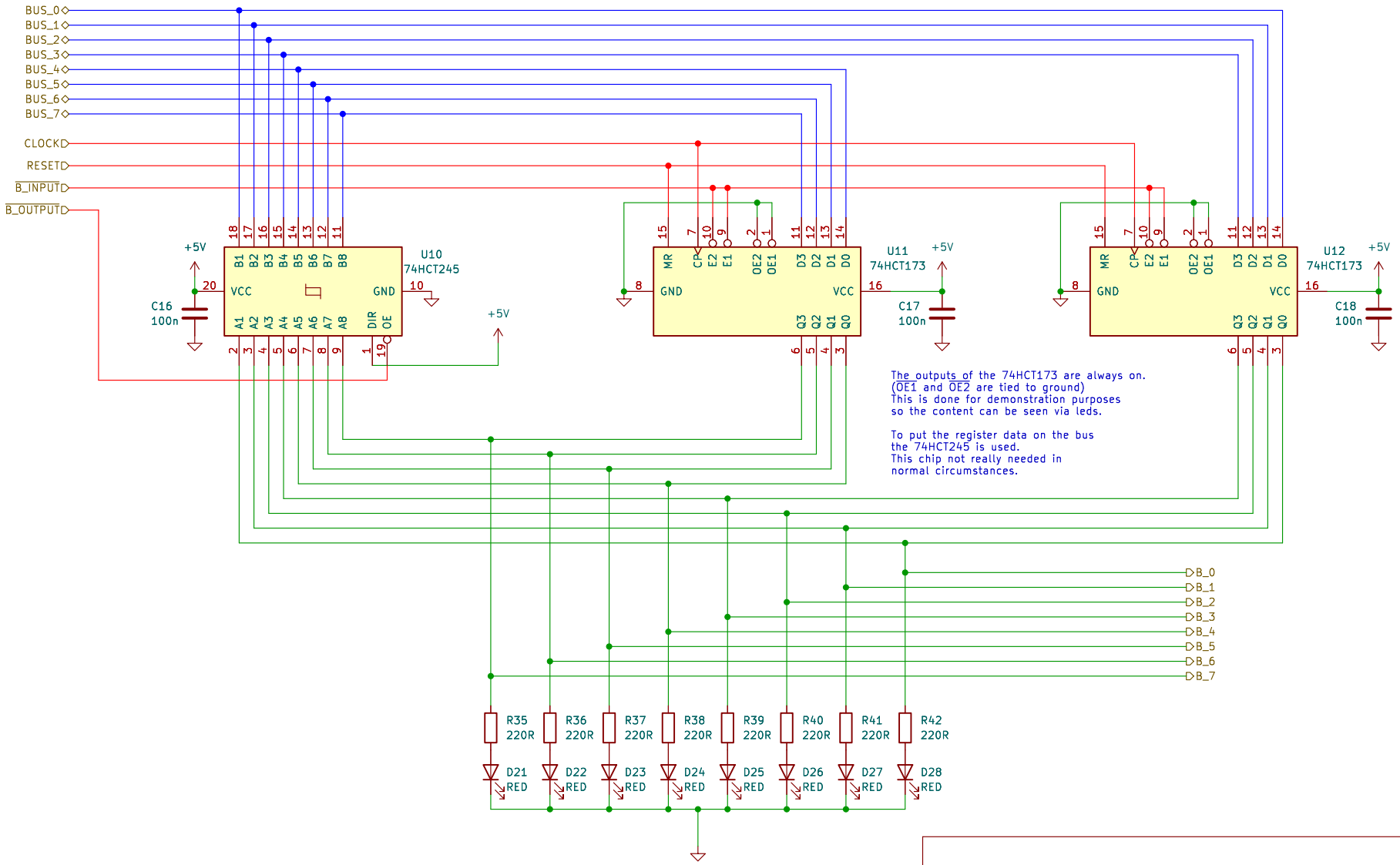


TBR Designs

Sheet: /A register/
File: a-register.kicad_sch

Title: 8-bit CPU

Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 3/20

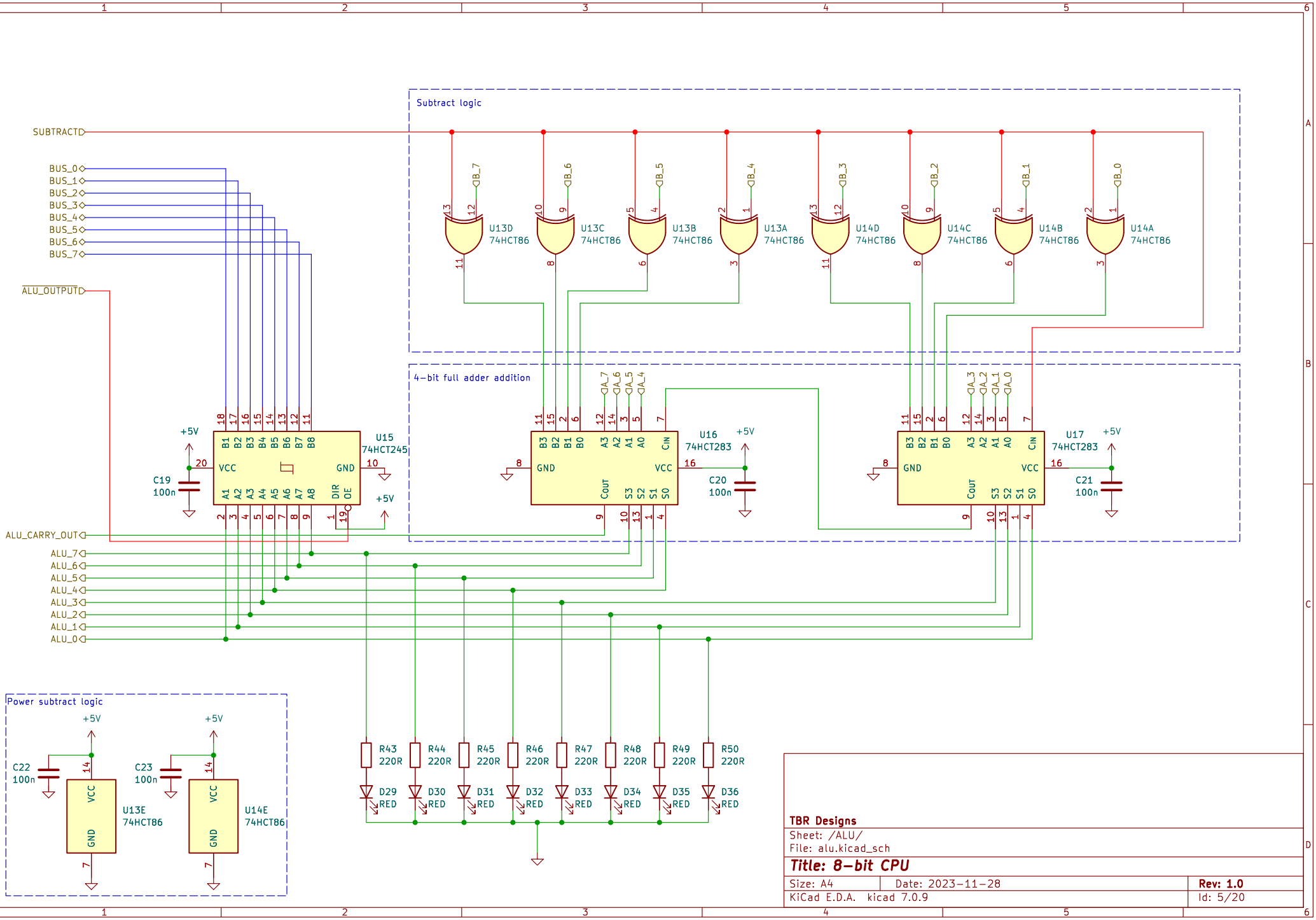


TBR Designs

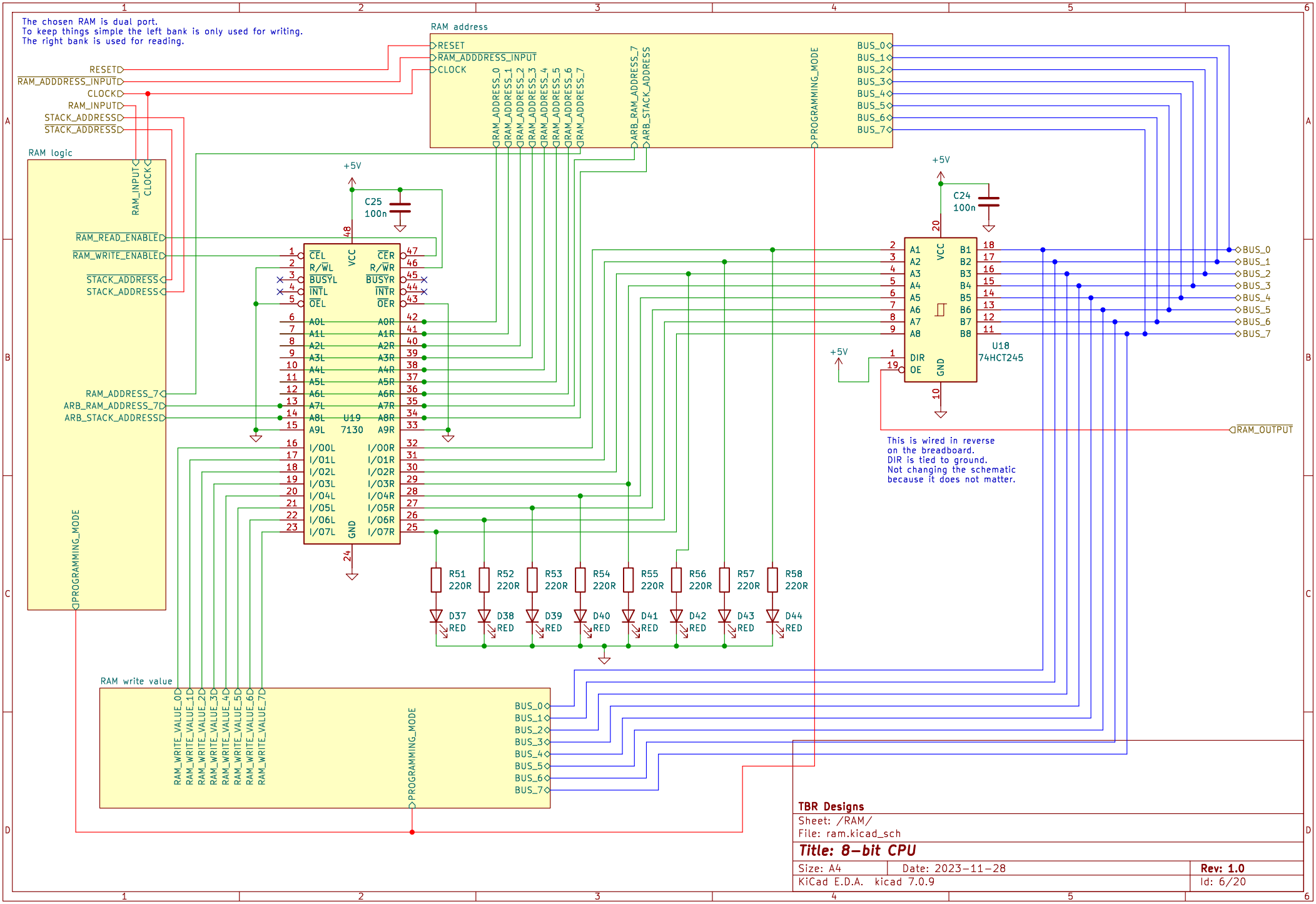
Sheet: /B register/
File: b-register.kicad_sch

Title: 8-bit CPU

Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 4/20



TBR Designs		
Sheet: /ALU/		
File: alu.kicad_sch		
Title: 8-bit CPU		
Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 5/20


The chosen RAM is dual port.
To keep things simple the left bank is only used for writing.
The right bank is used for reading.
RAM address
RESET
RAM_ADDDRESS_INPUT
CLOCK
RAM_ADDRESS_0
RAM_ADDRESS_1
RAM_ADDRESS_2
RAM_ADDRESS_3
RAM_ADDRESS_4
RAM_ADDRESS_5
RAM_ADDRESS_6
RAM_ADDRESS_7
ARB_RAM_ADDRESS_7
ARB_STACK_ADDRESS
PROGRAMMING_MODE
BUS_0
BUS_1
BUS_2
BUS_3
BUS_4
BUS_5
BUS_6
BUS_7
RAM_INPUT
STACK_ADDRESS
RAM logic
RAM_READ_ENABLE
RAM_WRITE_ENABLE
+5V
C25
100n
C24
U19
7130
U18
74HCT245
DIR
OE
GND
VCC
RAM_OUTPUT
This is wired in reverse
on the breadboard.
DIR is tied to ground.
Not changing the schematic
because it does not matter.
R51 220R
R52 220R
R53 220R
R54 220R
R55 220R
R56 220R
R57 220R
R58 220R
D37 RED
D38 RED
D39 RED
D40 RED
D41 RED
D42 RED
D43 RED
D44 RED
RAM write value
RAM_WRITE_VALUE_0
RAM_WRITE_VALUE_1
RAM_WRITE_VALUE_2
RAM_WRITE_VALUE_3
RAM_WRITE_VALUE_4
RAM_WRITE_VALUE_5
RAM_WRITE_VALUE_6
RAM_WRITE_VALUE_7
TBR Designs
Sheet: /RAM/
File: ram.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 6/20

A word on the RESET line.
On my breadboards this is giving me a lot of problems.
When there are 4 or more lines high on the RAM value
and I toggle between RAM_OUTPUT and not RAM_OUTPUT
the RAM address gets RESET for some reason.
After trying everything I could think of (pull ups, pull downs,
disconnecting the bus, replacing chips, replacing wires, ...)
I got nowhere. So I decided to disconnect the RESET line from the
flip flops and just tie them to GND.
RESET for the RAM address is not really needed given the pogram
loads the first address from the program counter which is 0.
I might hook it up again and see once all the control
signals are connected.

This is not reflected in the schematic because I do not know if
it is related to the breadboards or not.

When in programming mode the
value from the dip-switches is selected.
Otherwise the values from the flip flops are selected.

PROGRAMMING_MODE
BUS_0
BUS_1
BUS_2
BUS_3
BUS_4
BUS_5
BUS_6
BUS_7
RESET
CLOCK
RAM_ADDDRESS_INPUT

J1
Arduino_Address_Programming_Header

SW3
SW_DIP_x08

+5V

10k
R59 R60 R61 R62 R63 R64 R65 R66

U20
74HCT173
C26
100n

U21
74HCT173
C27
100n

U22
74HCT157
C28
100n

U23
74HCT157
C29
100n

RAM_ADDRESS_0
RAM_ADDRESS_1
RAM_ADDRESS_2
RAM_ADDRESS_3
RAM_ADDRESS_4
RAM_ADDRESS_5
RAM_ADDRESS_6
RAM_ADDRESS_7

ARB_RAM_ADDRESS_7
ARB_STACK_ADDRESS

R67 220R
R68 220R
R69 220R
R70 220R
R71 220R
R72 220R
R73 220R
R74 220R
R75 220R

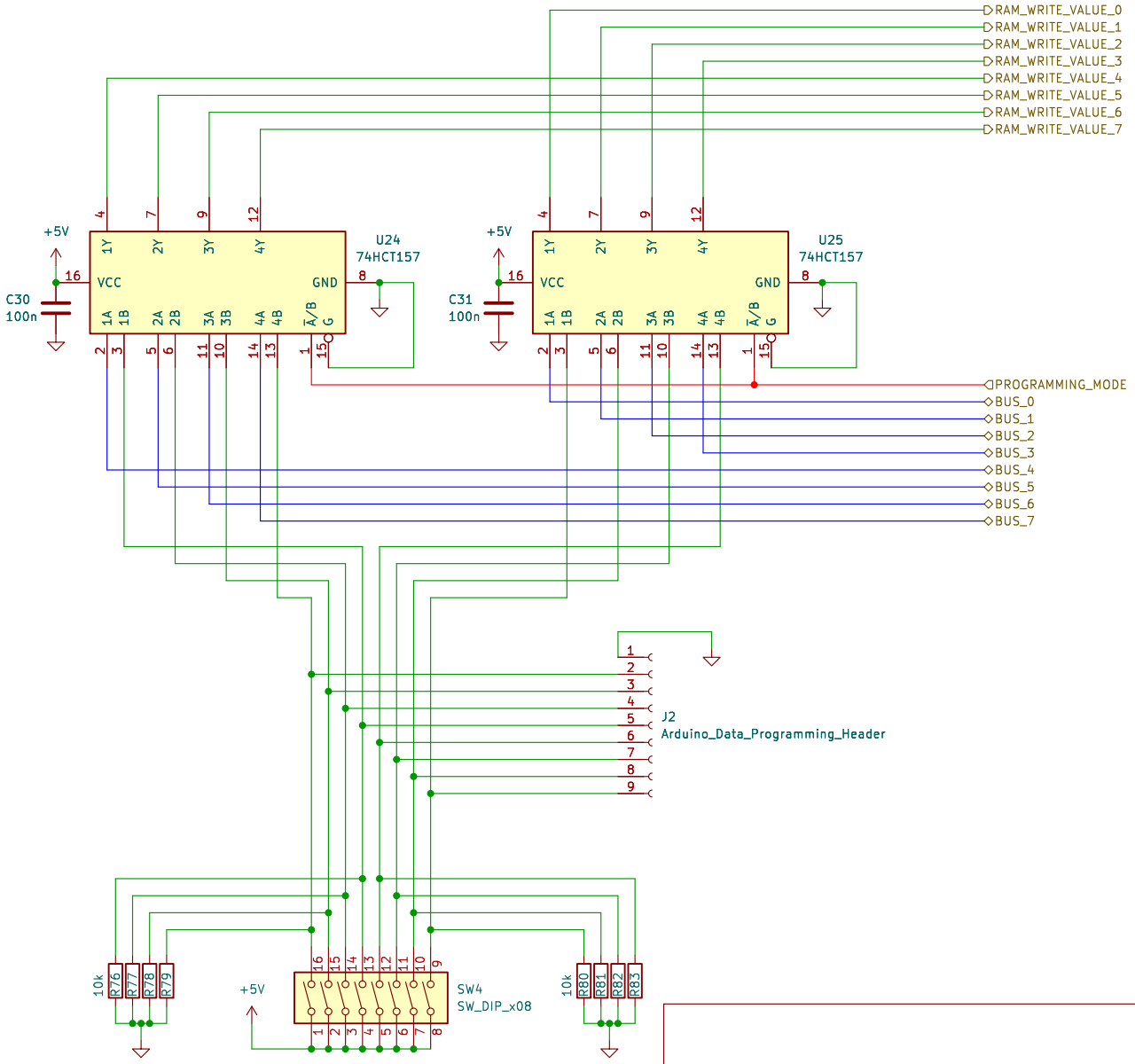
D45 YELLOW
D46 YELLOW
D47 YELLOW
D48 YELLOW
D49 YELLOW
D50 YELLOW
D51 YELLOW
D52 YELLOW
D53 YELLOW

TBR Designs
Sheet: /RAM/RAM address/
File: ram-address.kicad_sch

Title: 8-bit CPU

Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 7/20

When in programming mode the
value from the dip-switches is selected.
Otherwise the value from the bus is selected.

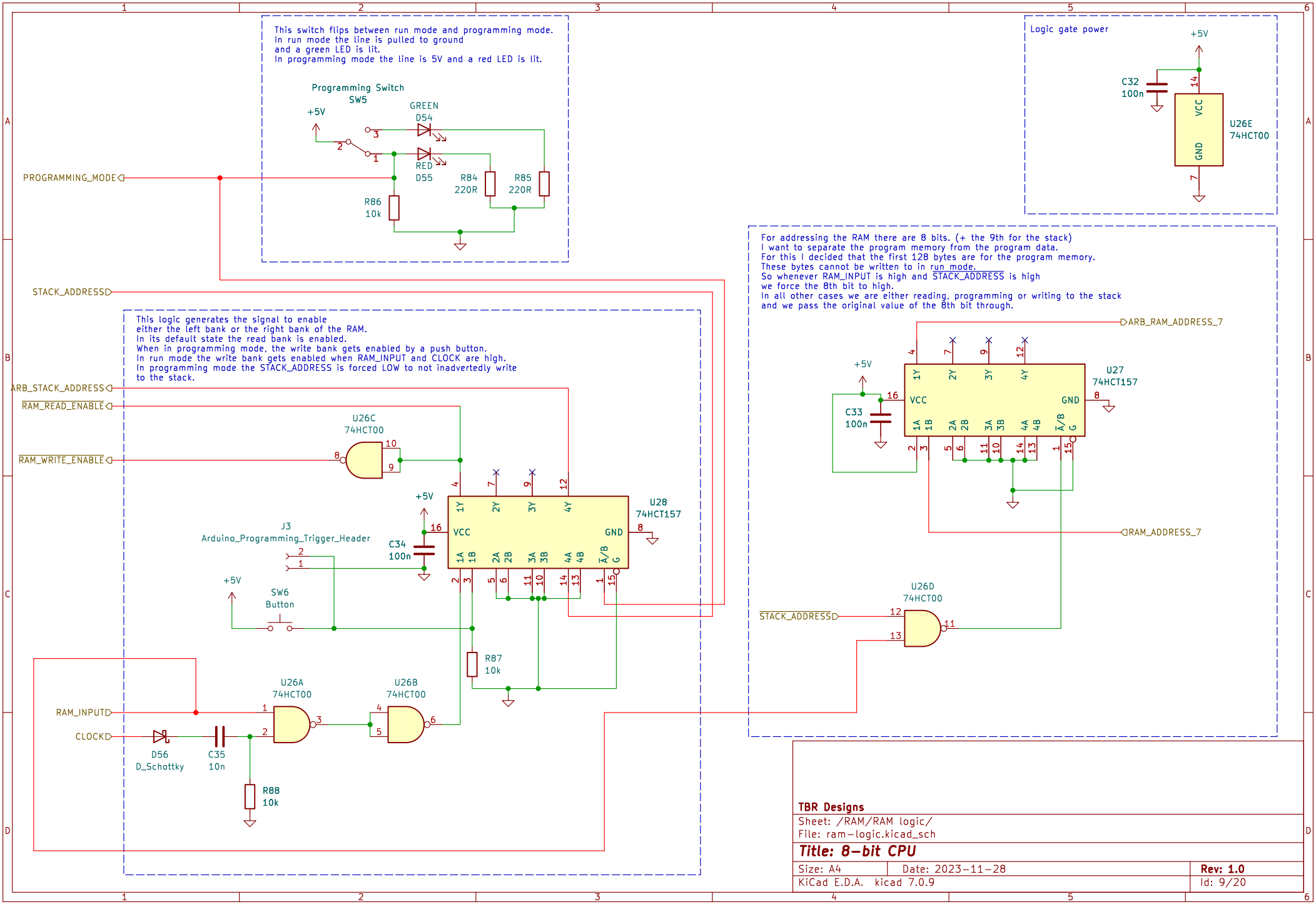


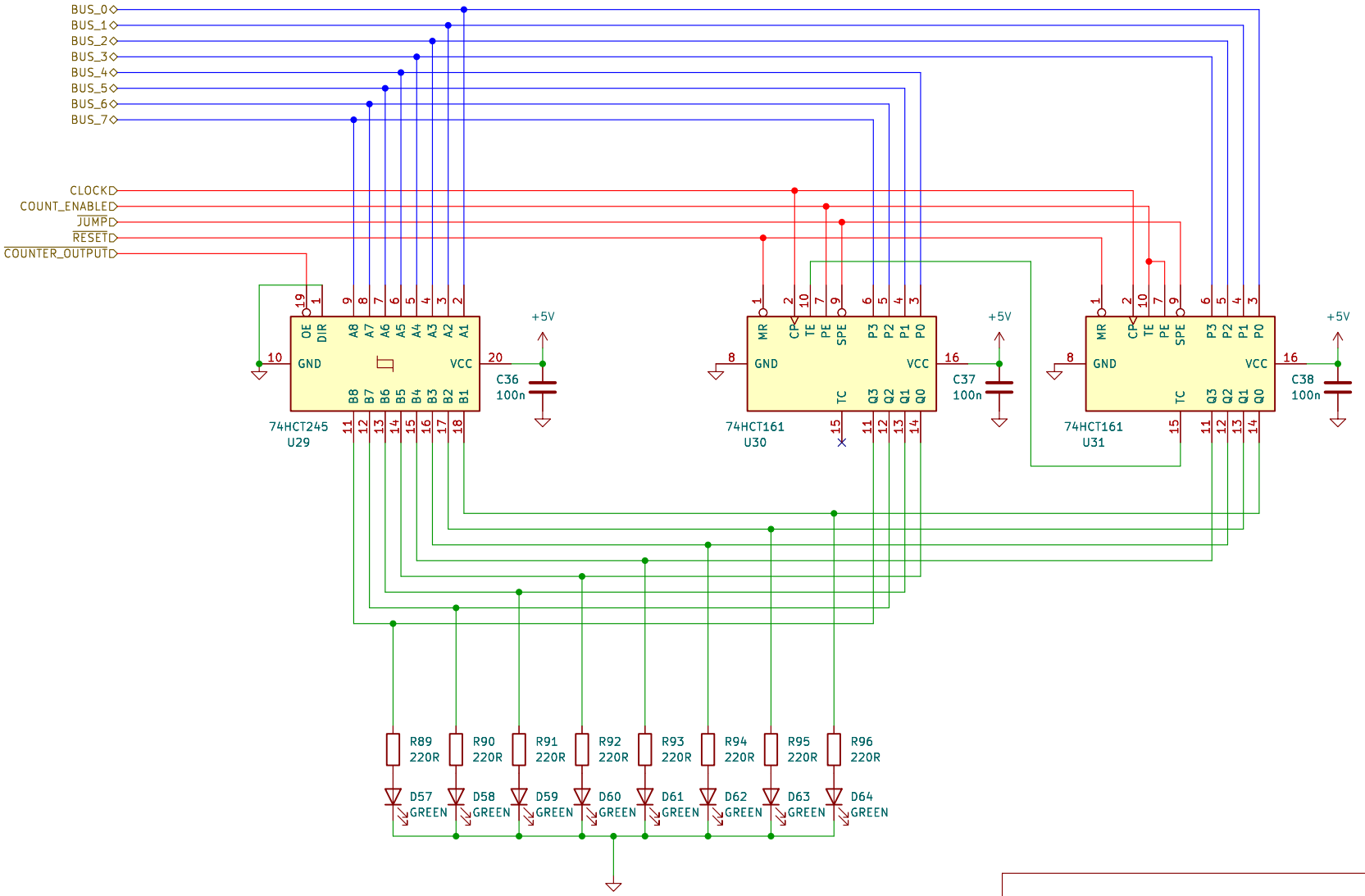
TBR Designs

Sheet: /RAM/RAM write value/
File: ram-write-value.kicad_sch

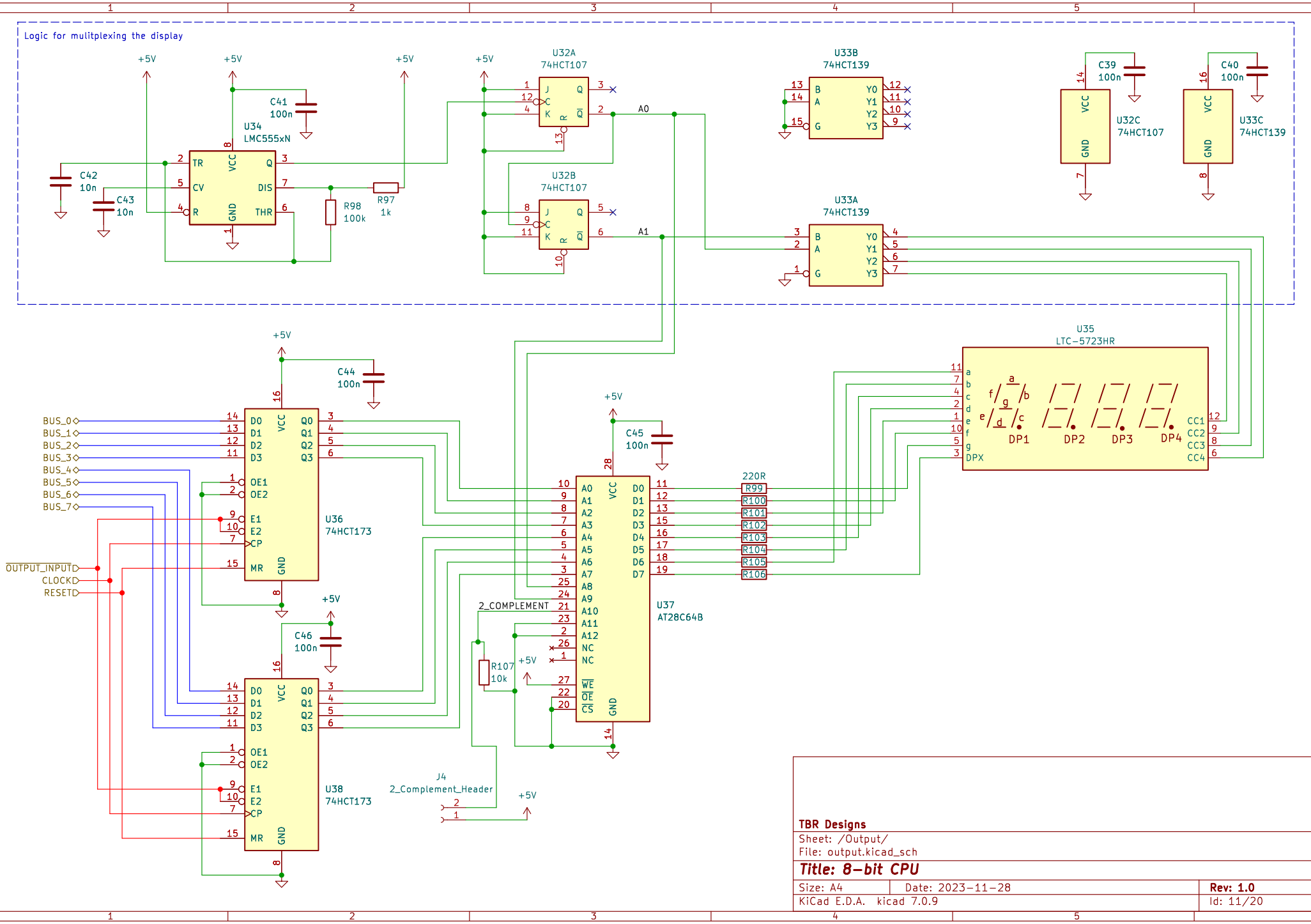
Title: 8-bit CPU

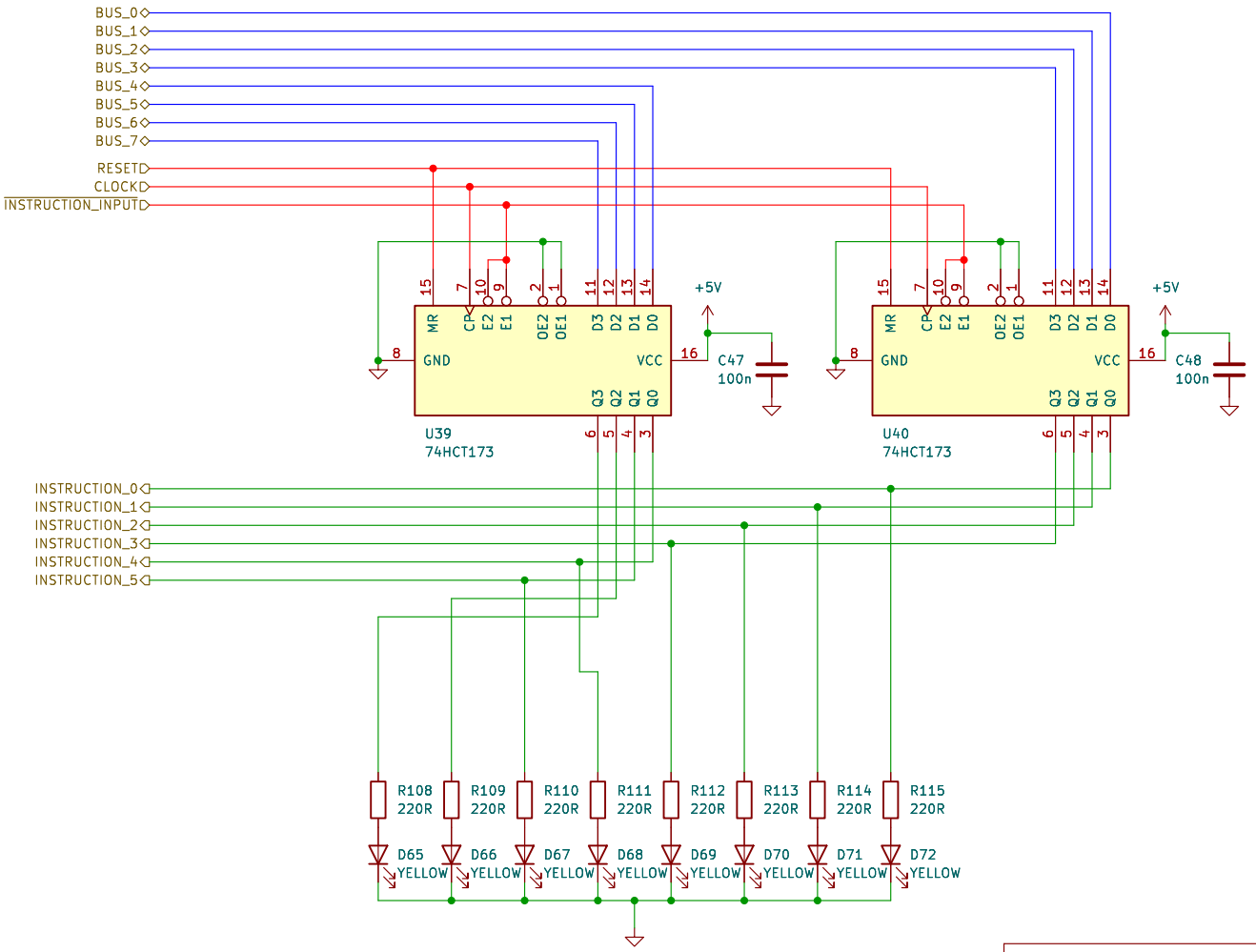
Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 8/20


This switch flips between run mode and programming mode.
In run mode the line is pulled to ground
and a green LED is lit.
In programming mode the line is 5V and a red LED is lit.
Programming Switch
SW5
+5V
GREEN
D54
RED
D55
R84
220R
R85
220R
R86
10k
PROGRAMMING_MODE
Logic gate power
+5V
C32
100n
U26E
74HCT00
VCC
GND
For addressing the RAM there are 8 bits. (+ the 9th for the stack)
I want to separate the program memory from the program data.
For this I decided that the first 128 bytes are for the program memory.
These bytes cannot be written to in run mode.
So whenever RAM_INPUT is high and STACK_ADDRESS is high
we force the 8th bit to high.
In all other cases we are either reading, programming or writing to the stack
and we pass the original value of the 8th bit through.
ARB_RAM_ADDRESS_7
U27
74HCT157
C33
100n
RAM_ADDRESS_7
U26D
74HCT00
STACK_ADDRESS
This logic generates the signal to enable
either the left bank or the right bank of the RAM.
In its default state the read bank is enabled.
When in programming mode, the write bank gets enabled by a push button.
In run mode the write bank gets enabled when RAM_INPUT and CLOCK are high.
In programming mode the STACK_ADDRESS is forced LOW to not inadvertedly write
to the stack.
ARB_STACK_ADDRESS
RAM_READ_ENABLE
RAM_WRITE_ENABLE
U26C
74HCT00
U28
74HCT157
C34
100n
J3
Arduino_Programming_Trigger_Header
SW6
Button
R87
10k
U26A
74HCT00
U26B
74HCT00
RAM_INPUT
CLOCK
D56
D_Schottky
C35
10n
R88
10k
TBR Designs
Sheet: /RAM/RAM logic/
File: ram-logic.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 9/20



TBR Designs		
Sheet: /Program Counter/		
File: program-counter.kicad_sch		
Title: 8-bit CPU		
Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 10/20


Logic for mulitplexing the display
+5V
U34
LMC555xN
C41
100n
C42
10n
C43
10n
R98
100k
R97
1k
U32A
74HCT107
U32B
74HCT107
A0
A1
U33B
74HCT139
U33A
74HCT139
C39
100n
C40
100n
U32C
74HCT107
U33C
74HCT139
U35
LTC-5723HR
DP1
DP2
DP3
DP4
C44
100n
C45
100n
C46
100n
BUS_0
BUS_1
BUS_2
BUS_3
BUS_4
BUS_5
BUS_6
BUS_7
OUTPUT_INPUT
CLOCK
RESET
U36
74HCT173
U38
74HCT173
U37
AT28C64B
2_COMPLEMENT
R107
10k
220R
R99
R100
R101
R102
R103
R104
R105
R106
J4
2_Complement_Header
TBR Designs
Sheet: /Output/
File: output.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 11/20

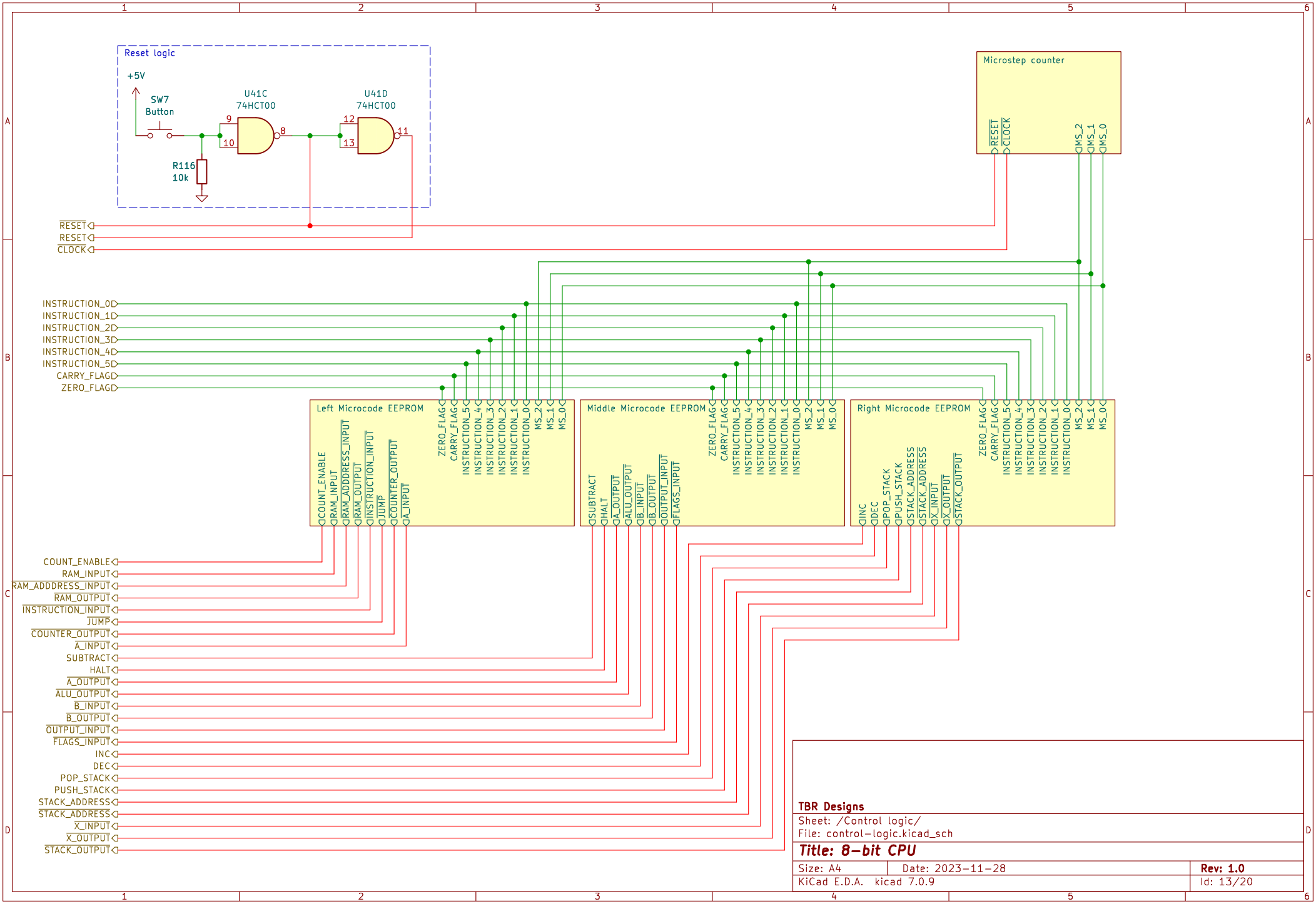


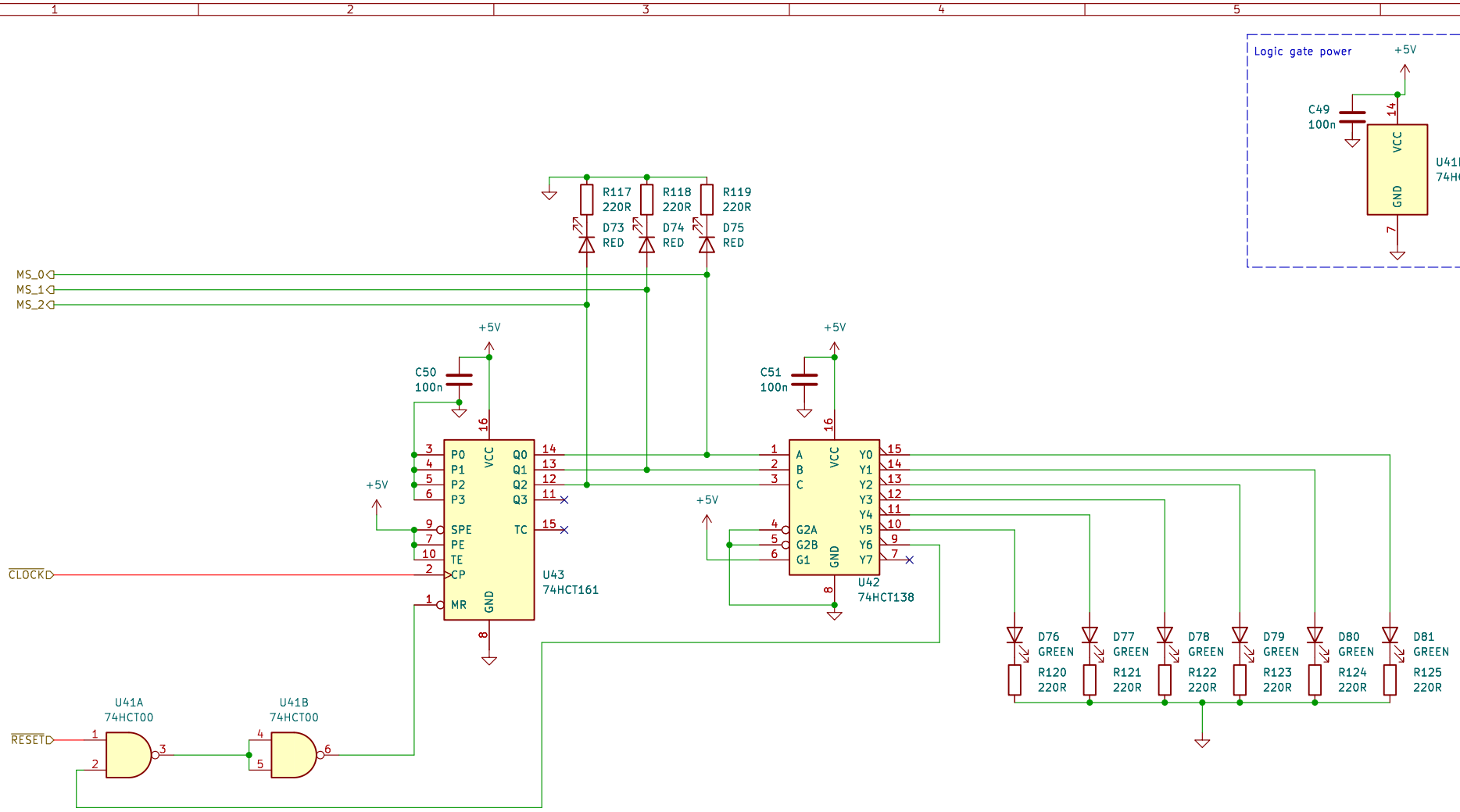
TBR Designs

Sheet: /Instruction register/
File: instruction-register.kicad_sch

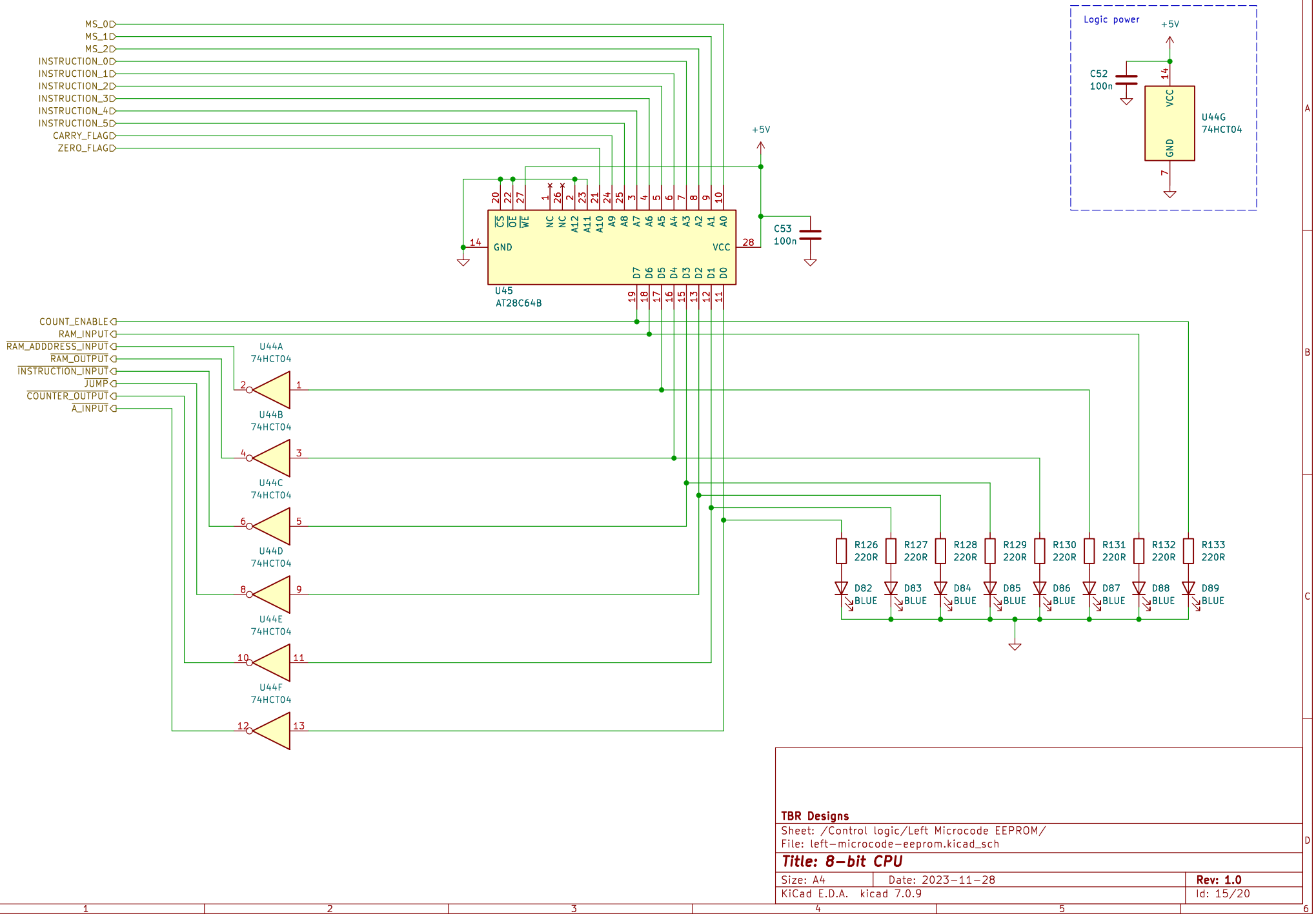
Title: 8-bit CPU

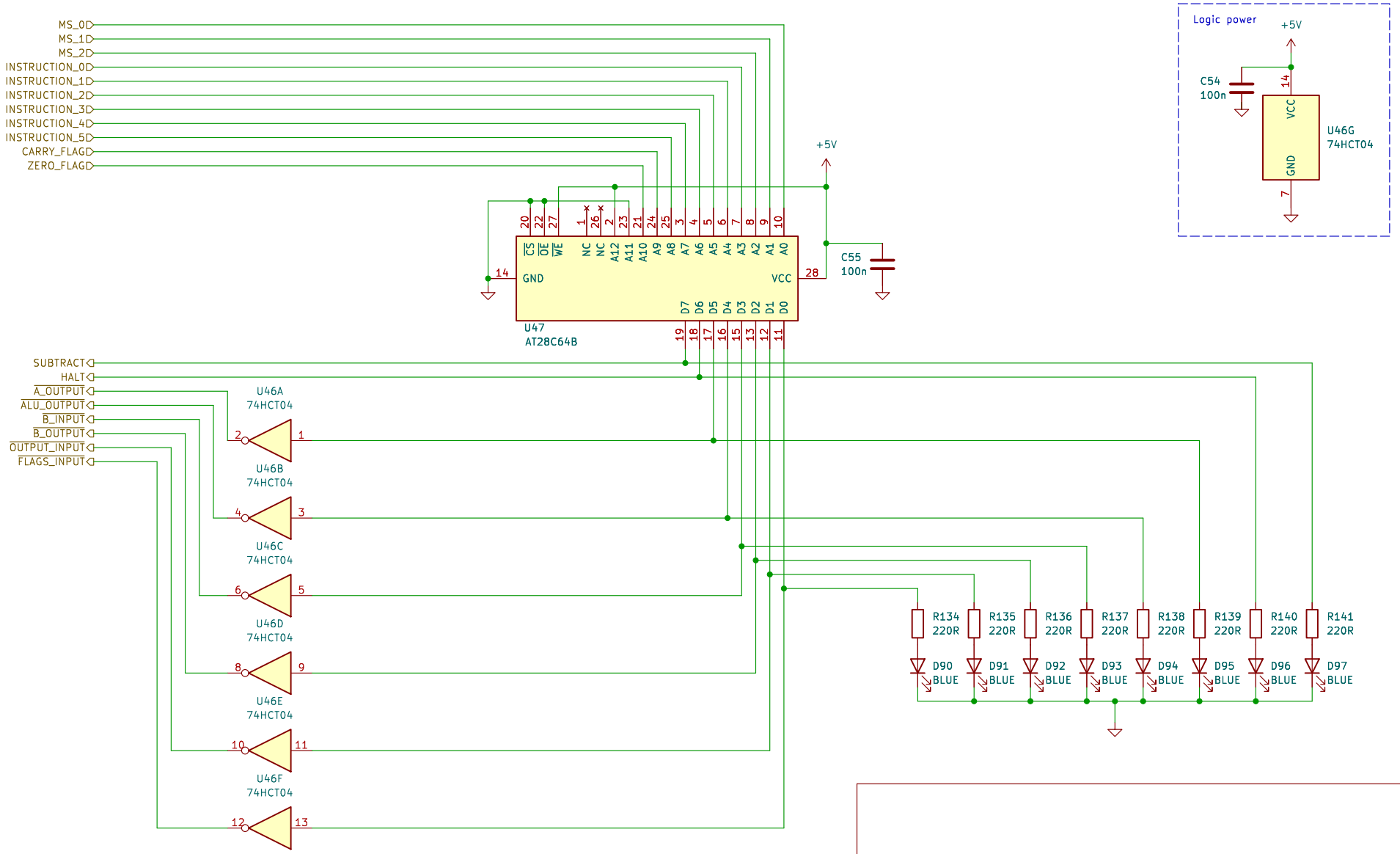
Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 12/20


Reset logic
+5V
SW7
Button
U41C
74HCT00
U41D
74HCT00
R116
10k
Microstep counter
RESET
CLOCK
MS_2
MS_1
MS_0
RESET
RESET
CLOCK
INSTRUCTION_0
INSTRUCTION_1
INSTRUCTION_2
INSTRUCTION_3
INSTRUCTION_4
INSTRUCTION_5
CARRY_FLAG
ZERO_FLAG
Left Microcode EEPROM
Middle Microcode EEPROM
Right Microcode EEPROM
COUNT_ENABLE
RAM_INPUT
RAM_ADDDRESS_INPUT
RAM_OUTPUT
INSTRUCTION_INPUT
JUMP
COUNTER_OUTPUT
A_INPUT
SUBTRACT
HALT
A_OUTPUT
ALU_OUTPUT
B_INPUT
B_OUTPUT
OUTPUT_INPUT
FLAGS_INPUT
INC
DEC
POP_STACK
PUSH_STACK
STACK_ADDRESS
STACK_ADDRESS
X_INPUT
X_OUTPUT
STACK_OUTPUT
TBR Designs
Sheet: /Control logic/
File: control-logic.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 13/20



TBR Designs		
Sheet: /Control logic/Microstep counter/		
File: microstep-counter.kicad_sch		
Title: 8-bit CPU		
Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 14/20


MS_0
MS_1
MS_2
INSTRUCTION_0
INSTRUCTION_1
INSTRUCTION_2
INSTRUCTION_3
INSTRUCTION_4
INSTRUCTION_5
CARRY_FLAG
ZERO_FLAG
Logic power
+5V
C52
100n
U44G
74HCT04
U45
AT28C64B
C53
100n
COUNT_ENABLE
RAM_INPUT
RAM_ADDDRESS_INPUT
RAM_OUTPUT
INSTRUCTION_INPUT
JUMP
COUNTER_OUTPUT
A_INPUT
U44A
U44B
U44C
U44D
U44E
U44F
R126 220R
R127 220R
R128 220R
R129 220R
R130 220R
R131 220R
R132 220R
R133 220R
D82 BLUE
D83 BLUE
D84 BLUE
D85 BLUE
D86 BLUE
D87 BLUE
D88 BLUE
D89 BLUE
TBR Designs
Sheet: /Control logic/Left Microcode EEPROM/
File: left-microcode-eeprom.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 15/20

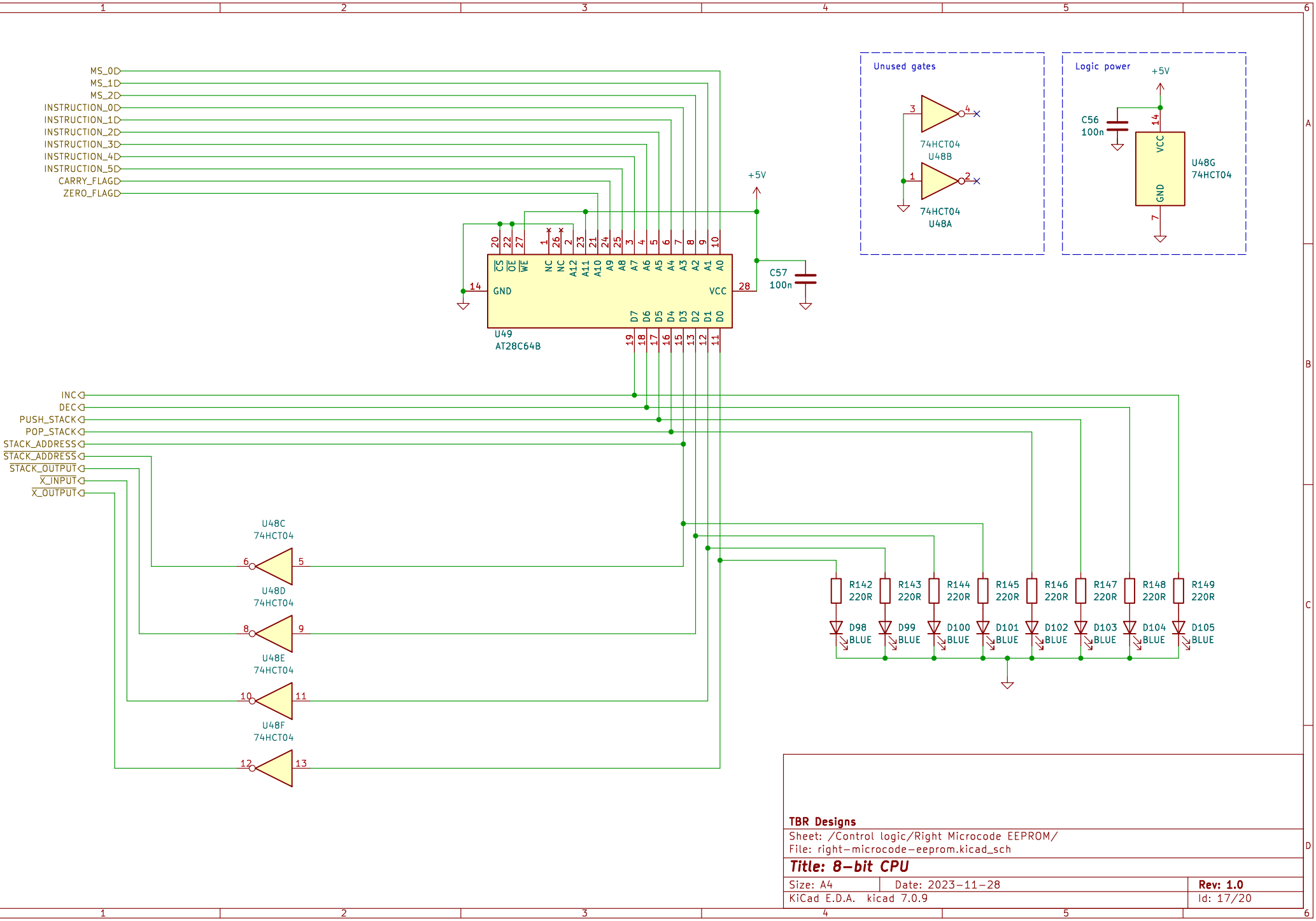


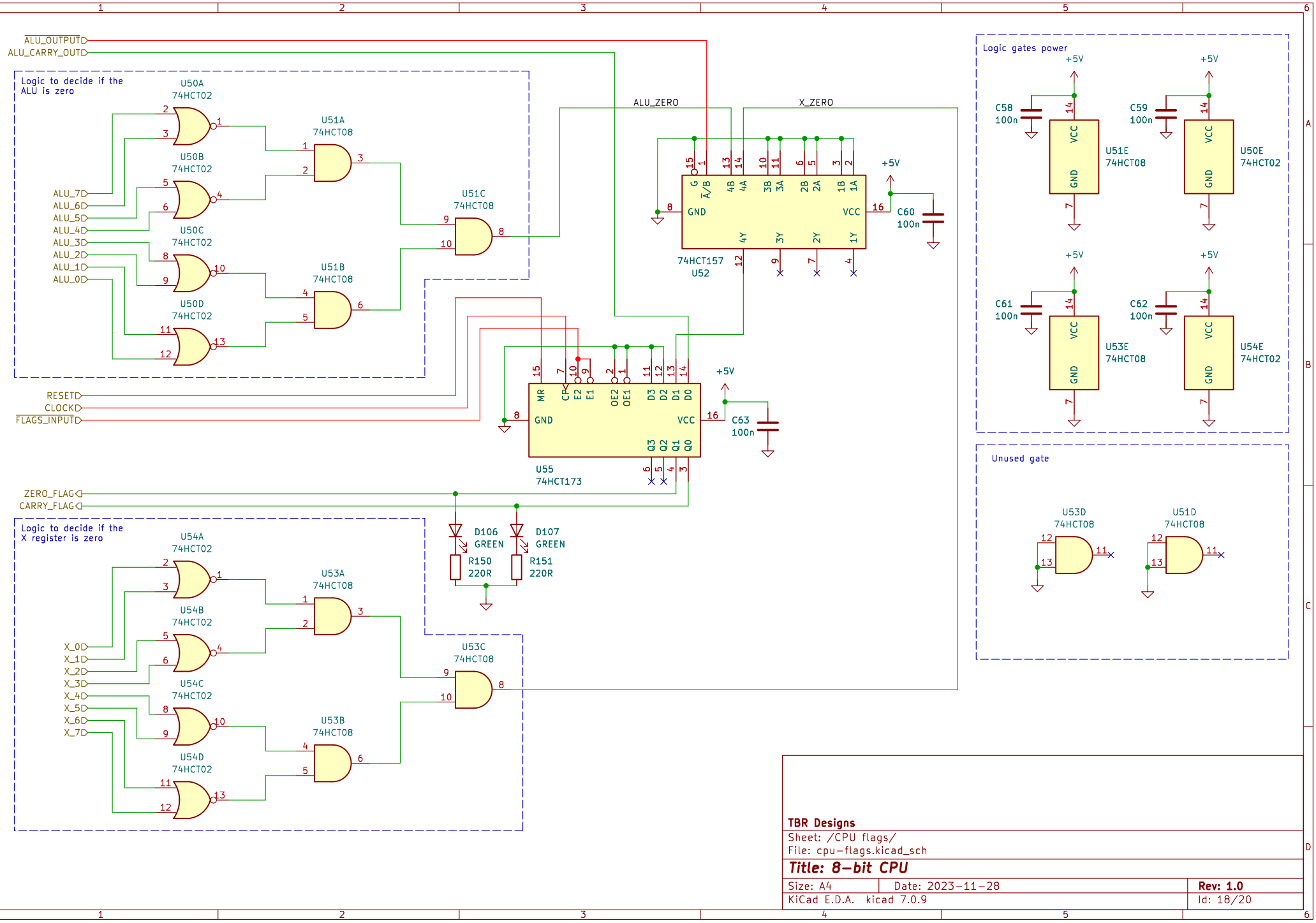
TBR Designs

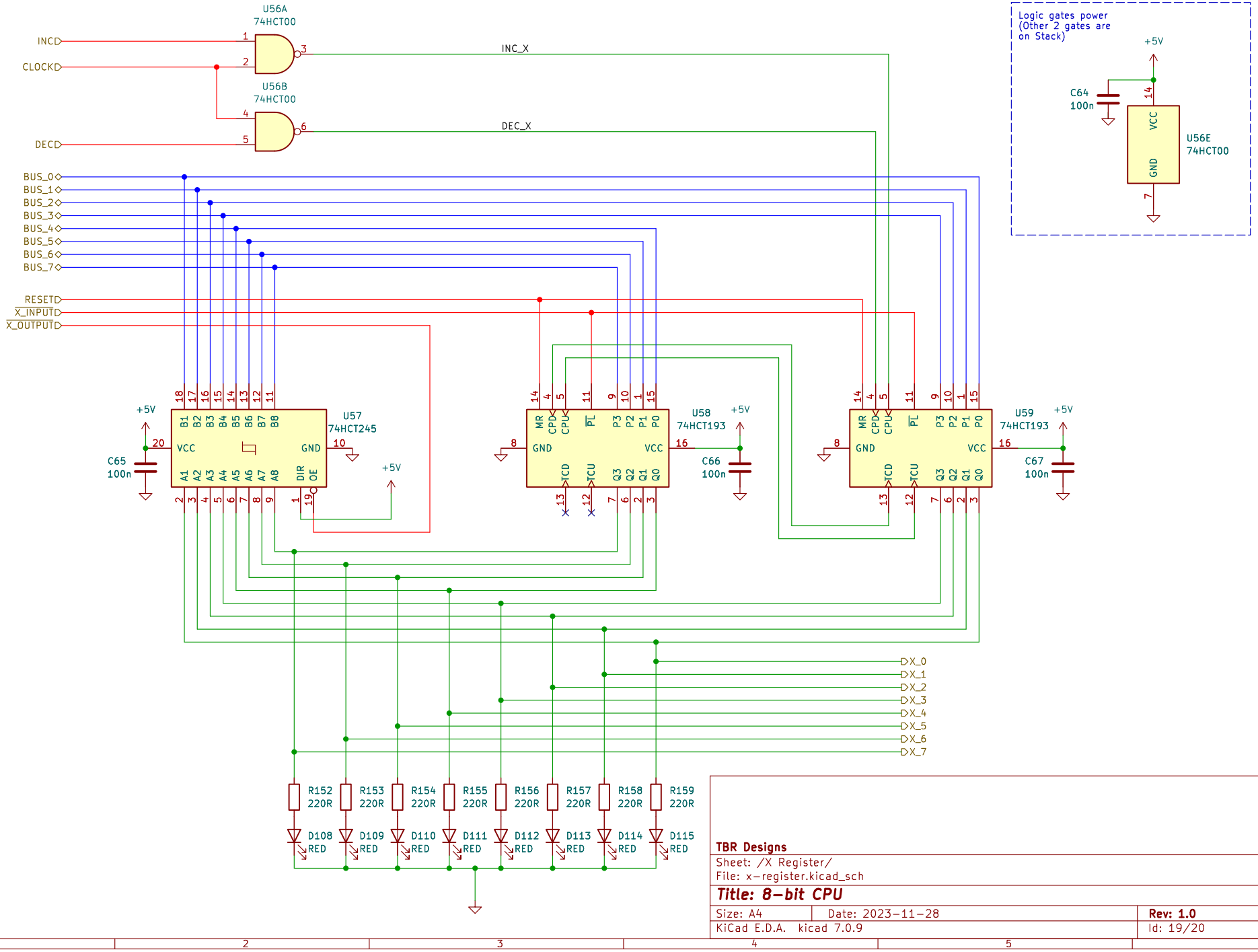
Sheet: /Control logic/Middle Microcode EEPROM/
File: middle-microcode-eeprom.kicad_sch

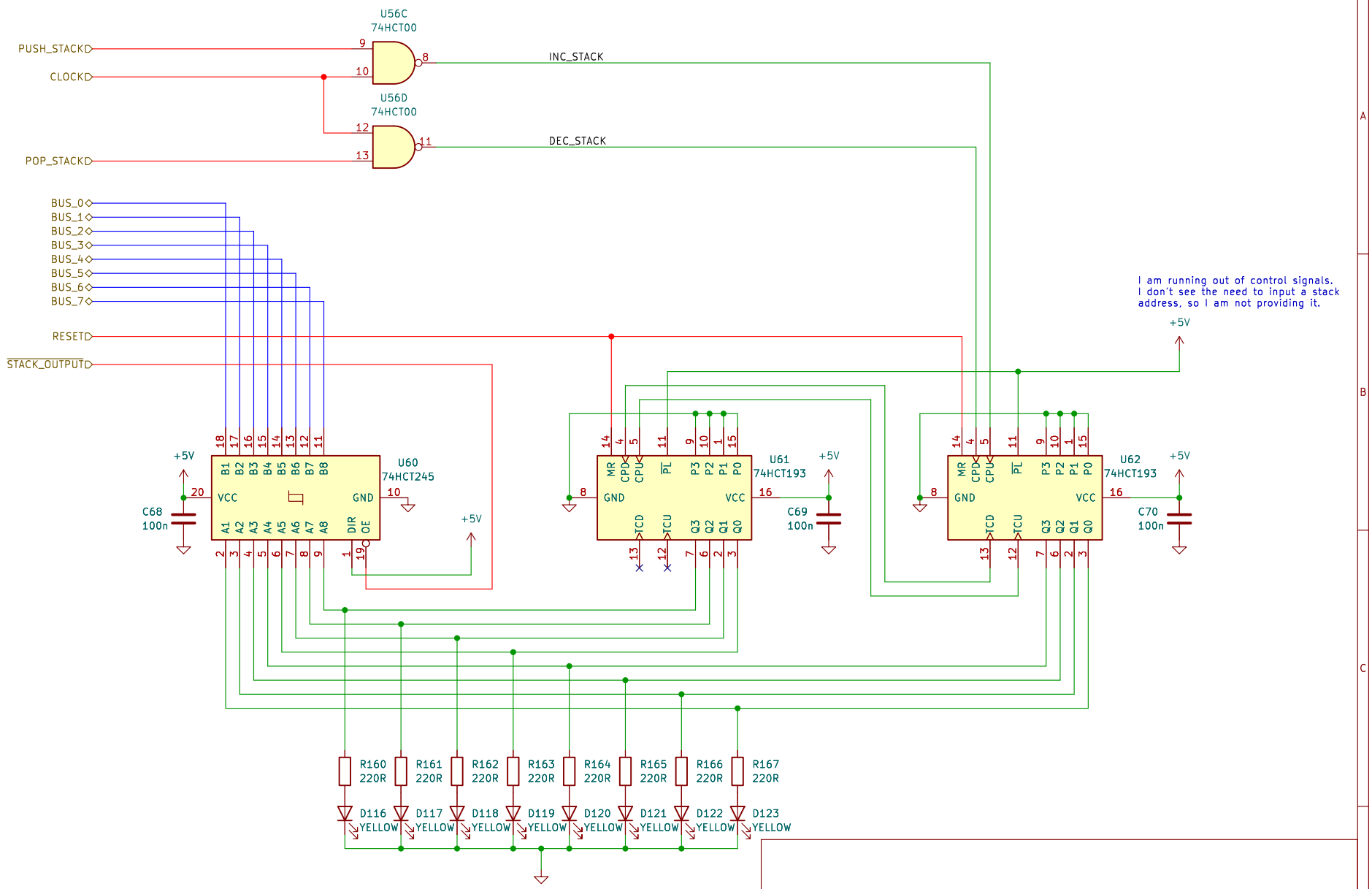
Title: 8-bit CPU

Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 16/20


MS_0
MS_1
MS_2
INSTRUCTION_0
INSTRUCTION_1
INSTRUCTION_2
INSTRUCTION_3
INSTRUCTION_4
INSTRUCTION_5
CARRY_FLAG
ZERO_FLAG
+5V
U49
AT28C64B
GND
VCC
C57
100n
INC
DEC
PUSH_STACK
POP_STACK
STACK_ADDRESS
STACK_ADDRESS
STACK_OUTPUT
X_INPUT
X_OUTPUT
U48C
74HCT04
U48D
74HCT04
U48E
74HCT04
U48F
74HCT04
Unused gates
74HCT04
U48B
74HCT04
U48A
Logic power
C56
100n
U48G
74HCT04
R142 220R
R143 220R
R144 220R
R145 220R
R146 220R
R147 220R
R148 220R
R149 220R
D98 BLUE
D99 BLUE
D100 BLUE
D101 BLUE
D102 BLUE
D103 BLUE
D104 BLUE
D105 BLUE
TBR Designs
Sheet: /Control logic/Right Microcode EEPROM/
File: right-microcode-eeprom.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 17/20


ALU_OUTPUT
ALU_CARRY_OUT
Logic to decide if the ALU is zero
U50A 74HCT02
U50B 74HCT02
U50C 74HCT02
U50D 74HCT02
U51A 74HCT08
U51B 74HCT08
U51C 74HCT08
ALU_7
ALU_6
ALU_5
ALU_4
ALU_3
ALU_2
ALU_1
ALU_0
ALU_ZERO
X_ZERO
U52 74HCT157
+5V
C60 100n
RESET
CLOCK
FLAGS_INPUT
U55 74HCT173
C63 100n
ZERO_FLAG
CARRY_FLAG
D106 GREEN
D107 GREEN
R150 220R
R151 220R
Logic to decide if the X register is zero
U54A 74HCT02
U54B 74HCT02
U54C 74HCT02
U54D 74HCT02
U53A 74HCT08
U53B 74HCT08
U53C 74HCT08
X_0
X_1
X_2
X_3
X_4
X_5
X_6
X_7
Logic gates power
C58 100n
C59 100n
C61 100n
C62 100n
U51E 74HCT08
U50E 74HCT02
U53E 74HCT08
U54E 74HCT02
Unused gate
U53D 74HCT08
U51D 74HCT08
TBR Designs
Sheet: /CPU flags/
File: cpu-flags.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 18/20


U56A
74HCT00
INC
CLOCK
INC_X
U56B
74HCT00
DEC
DEC_X
Logic gates power
(Other 2 gates are
on Stack)
+5V
C64
100n
U56E
74HCT00
BUS_0
BUS_1
BUS_2
BUS_3
BUS_4
BUS_5
BUS_6
BUS_7
RESET
X_INPUT
X_OUTPUT
U57
74HCT245
C65
100n
U58
74HCT193
C66
100n
U59
74HCT193
C67
100n
X_0
X_1
X_2
X_3
X_4
X_5
X_6
X_7
R152 220R
R153 220R
R154 220R
R155 220R
R156 220R
R157 220R
R158 220R
R159 220R
D108 RED
D109 RED
D110 RED
D111 RED
D112 RED
D113 RED
D114 RED
D115 RED
TBR Designs
Sheet: /X Register/
File: x-register.kicad_sch
Title: 8-bit CPU
Size: A4
Date: 2023-11-28
Rev: 1.0
KiCad E.D.A. kicad 7.0.9
Id: 19/20



TBR Designs

Sheet: /Stack/
File: stack.kicad_sch

Title: 8-bit CPU

Size: A4	Date: 2023-11-28	**Rev: 1.0**
KiCad E.D.A. kicad 7.0.9		Id: 20/20