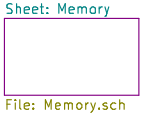
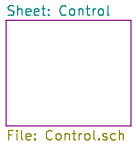


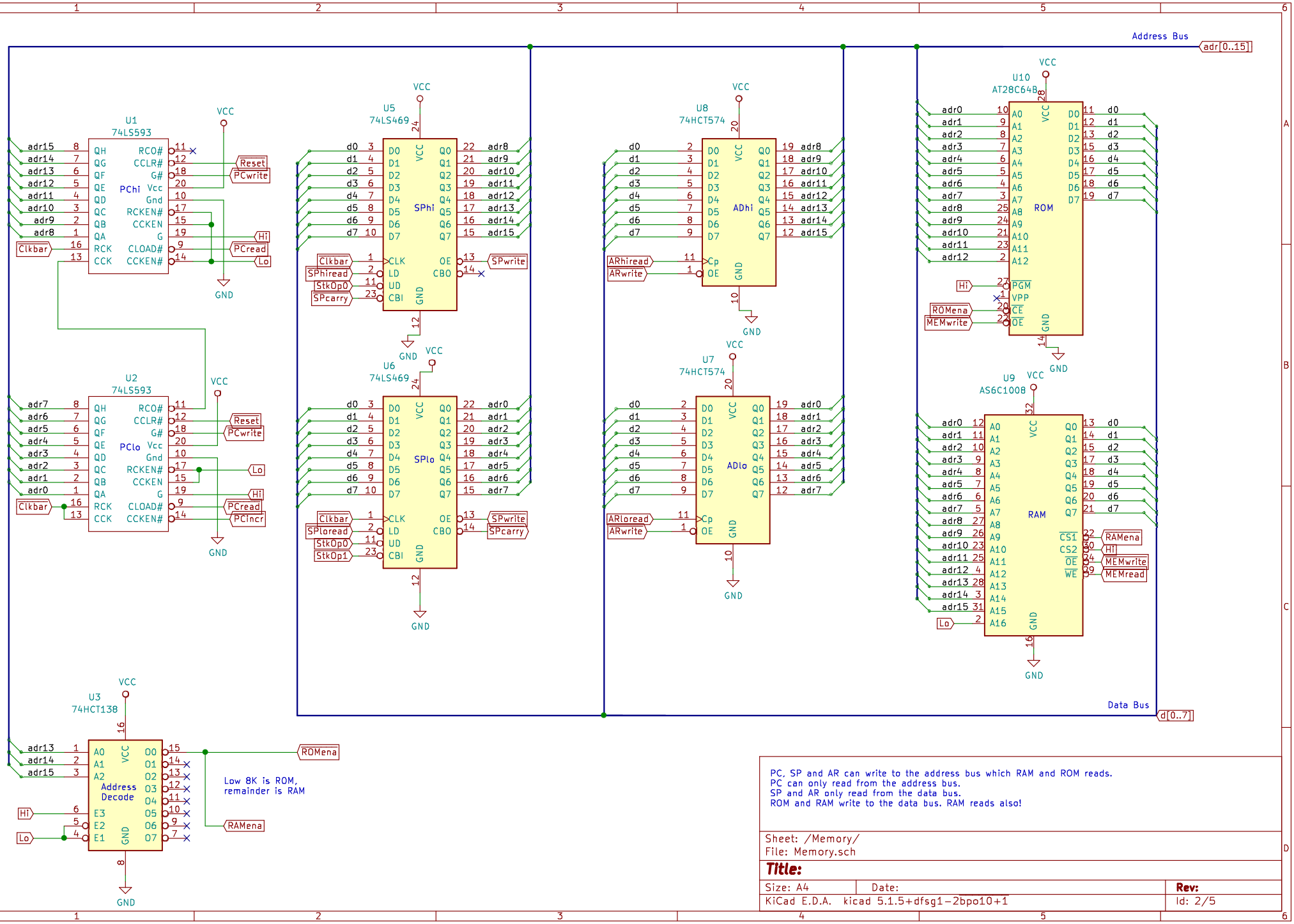


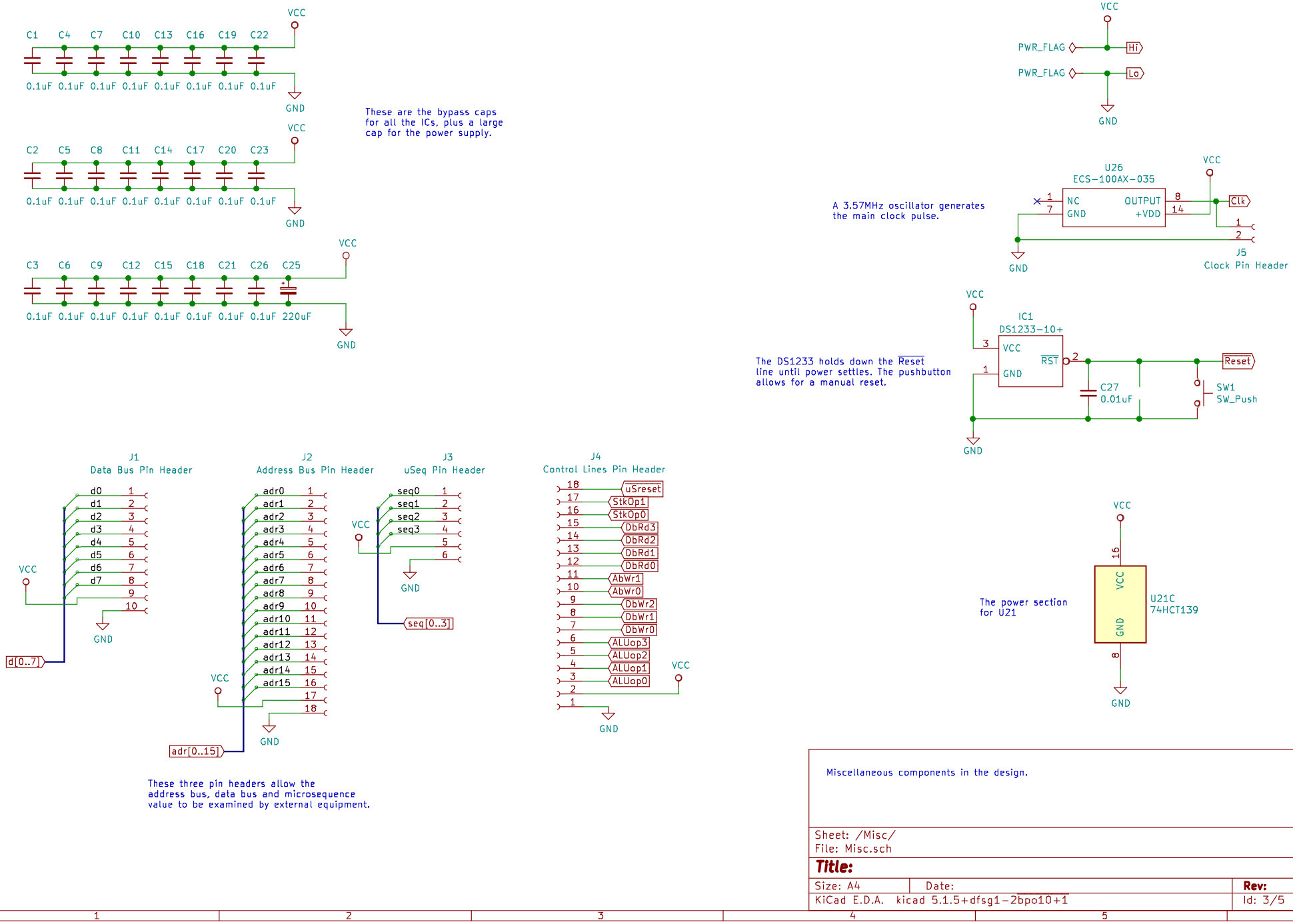
This is the FISC TTL CPU. Each one of the four sheets on the left holds the main four sections of the design.

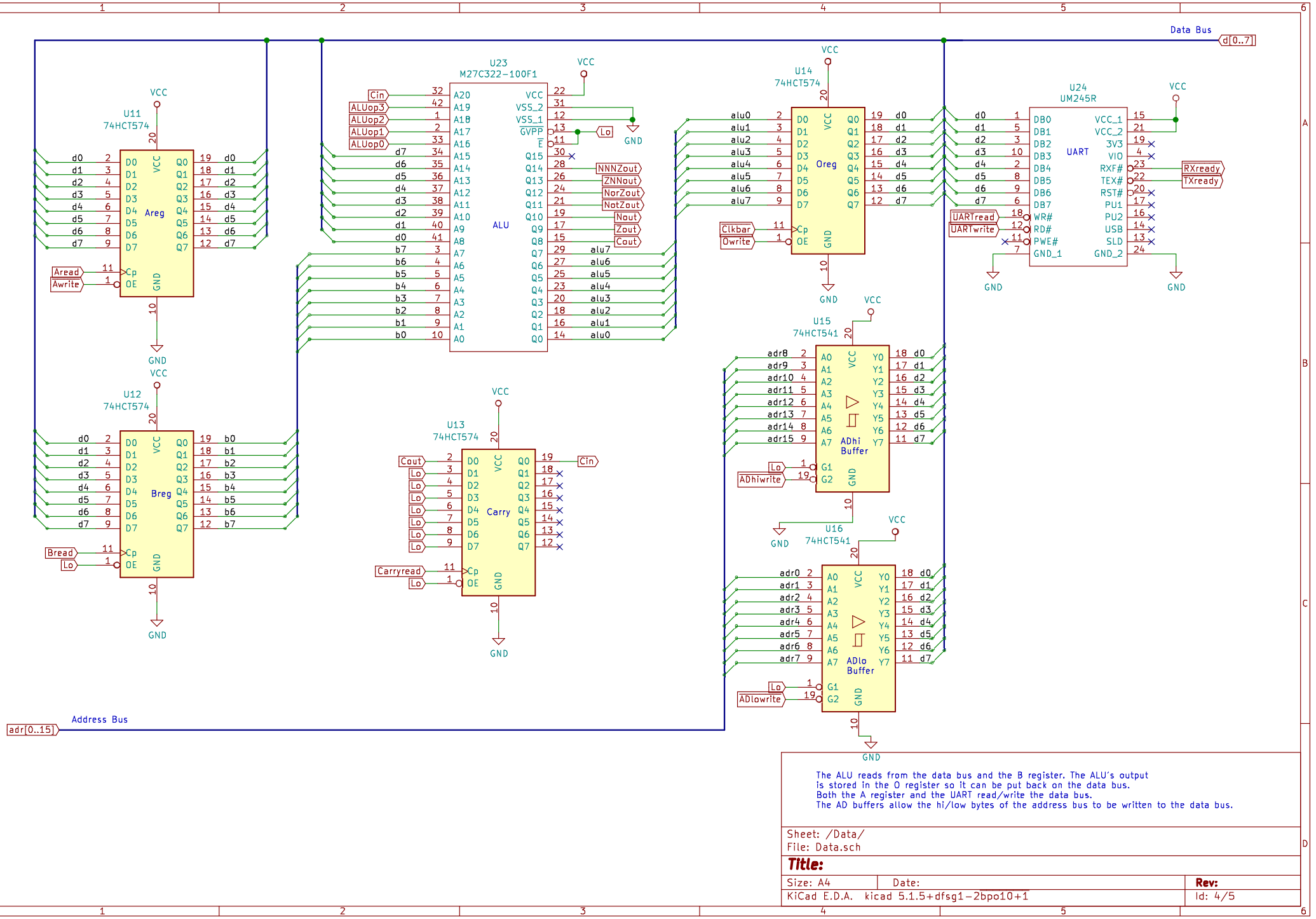
Sheet: /
File: FISC.sch

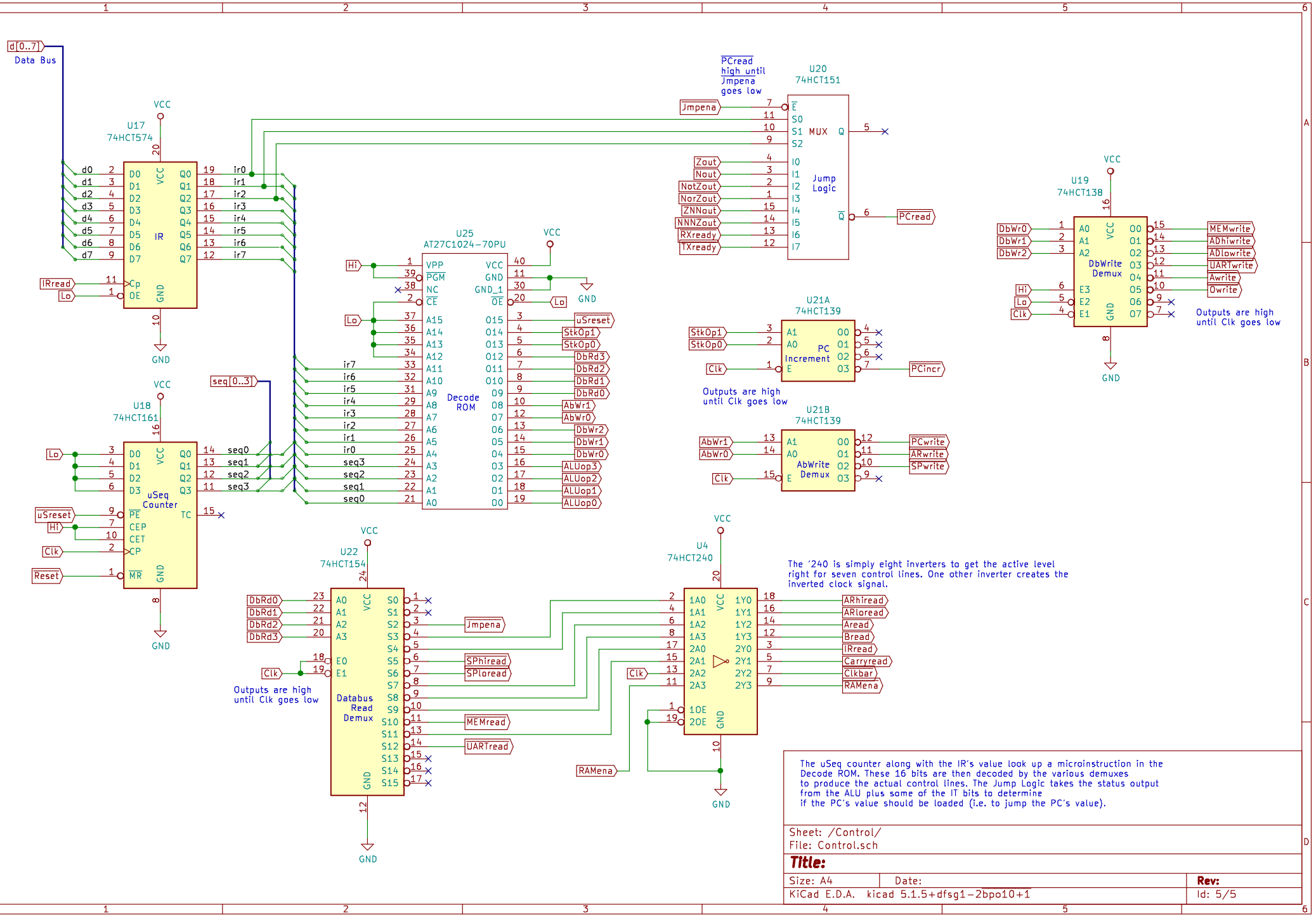
Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. kicad 5.1.5+dfsg1-2bpo10+1		Id: 1/5


Address Bus
adr[0..15]
Data Bus
d[0..7]
U1
74LS593
PChi
U2
74LS593
PClo
U3
74HCT138
Address Decode
U5
74LS469
SPhi
U6
74LS469
SPlo
U7
74HCT574
ADlo
U8
74HCT574
ADhi
U9
AS6C1008
RAM
U10
AT28C64B
ROM
VCC
GND
Reset
PCwrite
PCread
PCincr
Hi
Lo
Clkbar
SPhiread
SPloread
SPwrite
SPcarry
StkOp0
StkOp1
ARhiread
ARloread
ARwrite
ROMena
RAMena
MEMwrite
MEMread
Low 8K is ROM,
remainder is RAM
PC, SP and AR can write to the address bus which RAM and ROM reads.
PC can only read from the address bus.
SP and AR only read from the data bus.
ROM and RAM write to the data bus. RAM reads also!
Sheet: /Memory/
File: Memory.sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. kicad 5.1.5+dfsg1-2bpo10+1
Id: 2/5


C1 C4 C7 C10 C13 C16 C19 C22
0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF
VCC
GND
C2 C5 C8 C11 C14 C17 C20 C23
0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF
VCC
GND
C3 C6 C9 C12 C15 C18 C21 C26 C25
0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 0.1uF 220uF
VCC
GND
These are the bypass caps for all the ICs, plus a large cap for the power supply.
PWR_FLAG
Hi
Lo
VCC
GND
U26
ECS-100AX-035
NC
GND
OUTPUT
+VDD
Clk
J5
Clock Pin Header
A 3.57MHz oscillator generates the main clock pulse.
IC1
DS1233-10+
VCC
GND
RST
Reset
C27
0.01uF
SW1
SW_Push
The DS1233 holds down the Reset line until power settles. The pushbutton allows for a manual reset.
J1
Data Bus Pin Header
d0 d1 d2 d3 d4 d5 d6 d7
d[0..7]
J2
Address Bus Pin Header
adr0 adr1 adr2 adr3 adr4 adr5 adr6 adr7 adr8 adr9 adr10 adr11 adr12 adr13 adr14 adr15
adr[0..15]
J3
uSeq Pin Header
seq0 seq1 seq2 seq3
seq[0..3]
These three pin headers allow the address bus, data bus and microsequence value to be examined by external equipment.
J4
Control Lines Pin Header
uSreset
StkOp1
StkOp0
DbRd3
DbRd2
DbRd1
DbRd0
AbWr1
AbWr0
DbWr2
DbWr1
DbWr0
ALUop3
ALUop2
ALUop1
ALUop0
The power section for U21
U21C
74HCT139
Miscellaneous components in the design.
Sheet: /Misc/
File: Misc.sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. kicad 5.1.5+dfsg1-2bpo10+1
Id: 3/5


Data Bus
d[0..7]
Address Bus
adr[0..15]
U11
74HCT574
Areg
U12
74HCT574
Breg
U13
74HCT574
Carry
U23
M27C322-100F1
ALU
U14
74HCT574
Oreg
U24
UM245R
UART
U15
74HCT541
ADhi Buffer
U16
74HCT541
ADlo Buffer
VCC
GND
Lo
Aread
Awrite
Bread
Carryread
Clkbar
Owrite
ADhiwrite
ADlowrite
UARTread
UARTwrite
RXready
TXready
Cin
Cout
ALUop3
ALUop2
ALUop1
ALUop0
NNNZout
ZNNout
NorZout
NotZout
Nout
Zout
Cout
The ALU reads from the data bus and the B register. The ALU's output
is stored in the O register 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:
Size: A4
Date:
Rev:
KiCad E.D.A. kicad 5.1.5+dfsg1-2bpo10+1
Id: 4/5


d[0..7]
Data Bus
U17
74HCT574
IR
IRread
Lo
U18
74HCT161
uSeq
Counter
uSreset
Hi
Clk
Reset
seq[0..3]
U25
AT27C1024-70PU
Decode
ROM
U20
74HCT151
Jump
Logic
PCread
high until
Jmpena
goes low
Zout
Nout
NotZout
NorZout
ZNNout
NNNZout
RXready
TXready
U21A
74HCT139
PC
Increment
StkOp1
StkOp0
PCincr
Outputs are high
until Clk goes low
U21B
74HCT139
AbWrite
Demux
AbWr1
AbWr0
PCwrite
ARwrite
SPwrite
U19
74HCT138
DbWrite
Demux
DbWr0
DbWr1
DbWr2
MEMwrite
ADhiwrite
ADlowrite
UARTwrite
Awrite
Owrite
Outputs are high
until Clk goes low
U22
74HCT154
Databus
Read
Demux
Jmpena
SPhiread
SPloread
MEMread
UARTread
Outputs are high
until Clk goes low
U4
74HCT240
ARhiread
ARloread
Aread
Bread
IRread
Carryread
Clkbar
RAMena
The '240 is simply eight inverters to get the active level
right for seven control lines. One other inverter creates the
inverted clock signal.
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. The Jump Logic takes the status output
from the ALU plus some of the IT 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:
Size: A4
Date:
Rev:
KiCad E.D.A. kicad 5.1.5+dfsg1-2bpo10+1
Id: 5/5