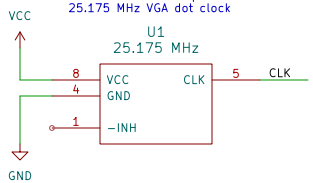
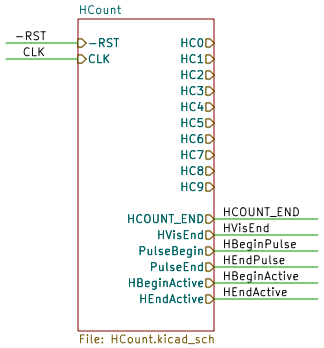
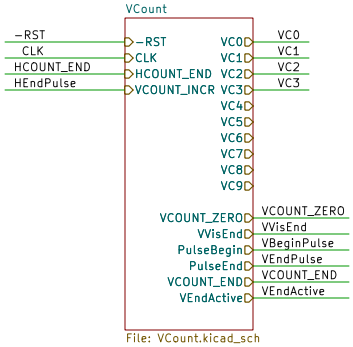
Oscillator module to produce
25.175 MHz VGA dot clock



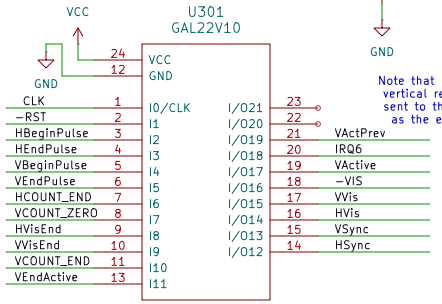
Horizontal count

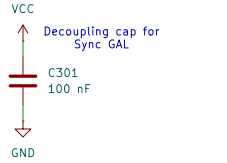


Vertical count



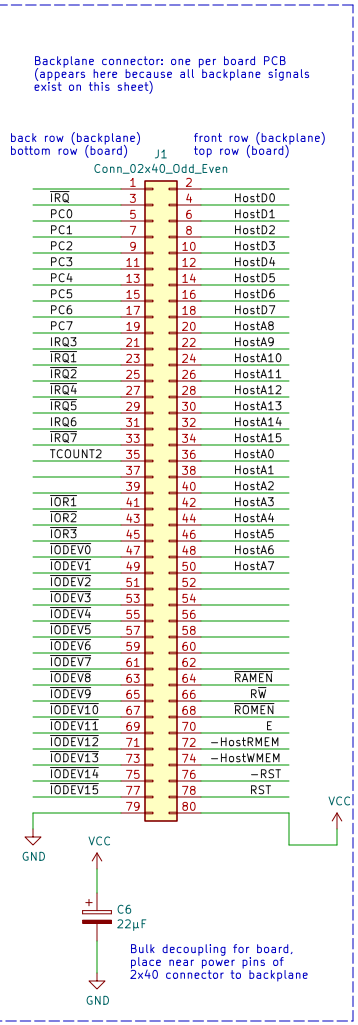
Sync gen and visibility signals:
Logic is in Sync.pld



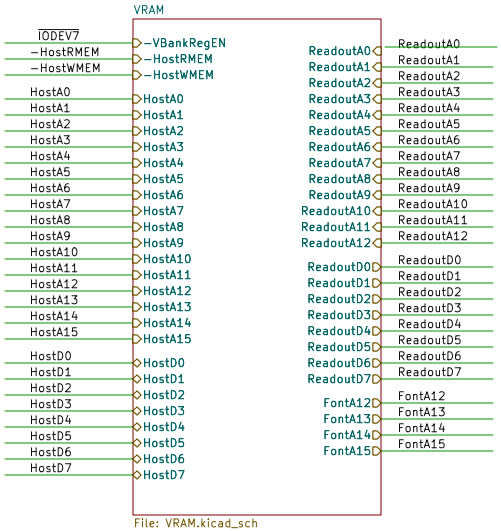


Note that pin 20 is a one-cycle vertical refresh pulse, which is sent to the interrupt controller as the edge-triggered IRQ6.

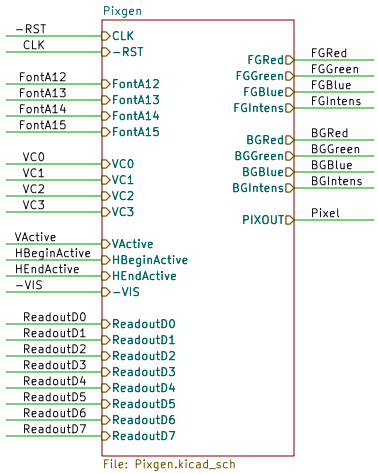
Backplane connector: one per board PCB
(appears here because all backplane signals exist on this sheet)



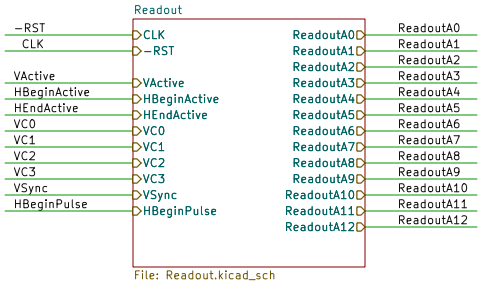
VRAM (video memory for
character and attribute data)



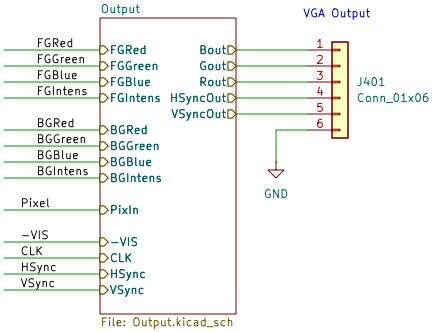
Pixel generation (using data from VRAM,
generate sequence of pixel fg/bg values
along with fg/bg color information
to send to Output module for display)



Readout module (generate addresses
for reading character and attribute
data from VRAM)



Video output
(to VGA monitor)

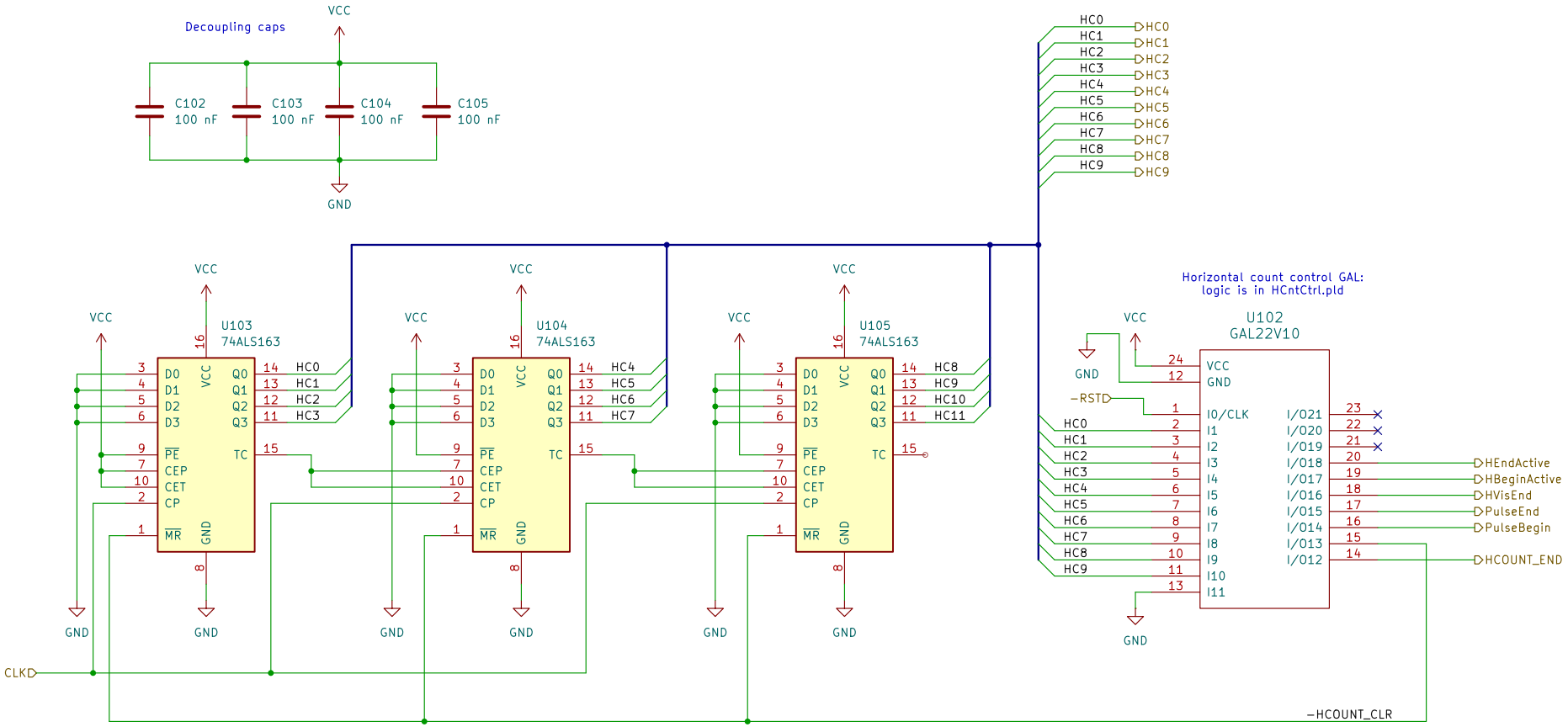


daveho hacks

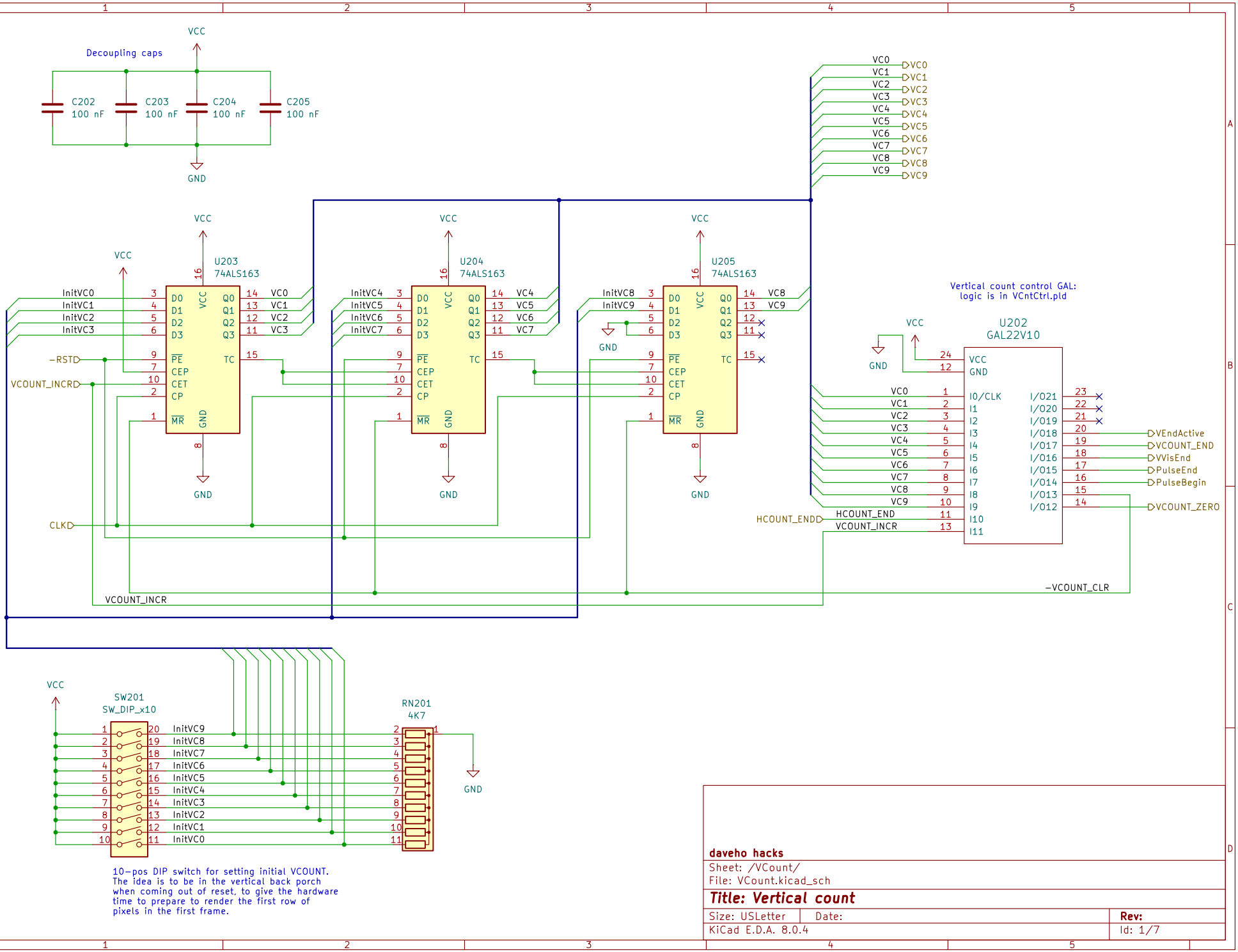
Sheet: /
File: HW_VGA.kicad_sch

Title: Hardware VGA

Size: User	Date:	Rev:
KiCad E.D.A. 8.0.4		Id: 1/7



daveho hacks	
Sheet: /HCount/	
File: HCount.kicad_sch	
Title: Horizontal count	
Size: USLetter | Date:	**Rev:**
KiCad E.D.A. 8.0.4	Id: 1/7



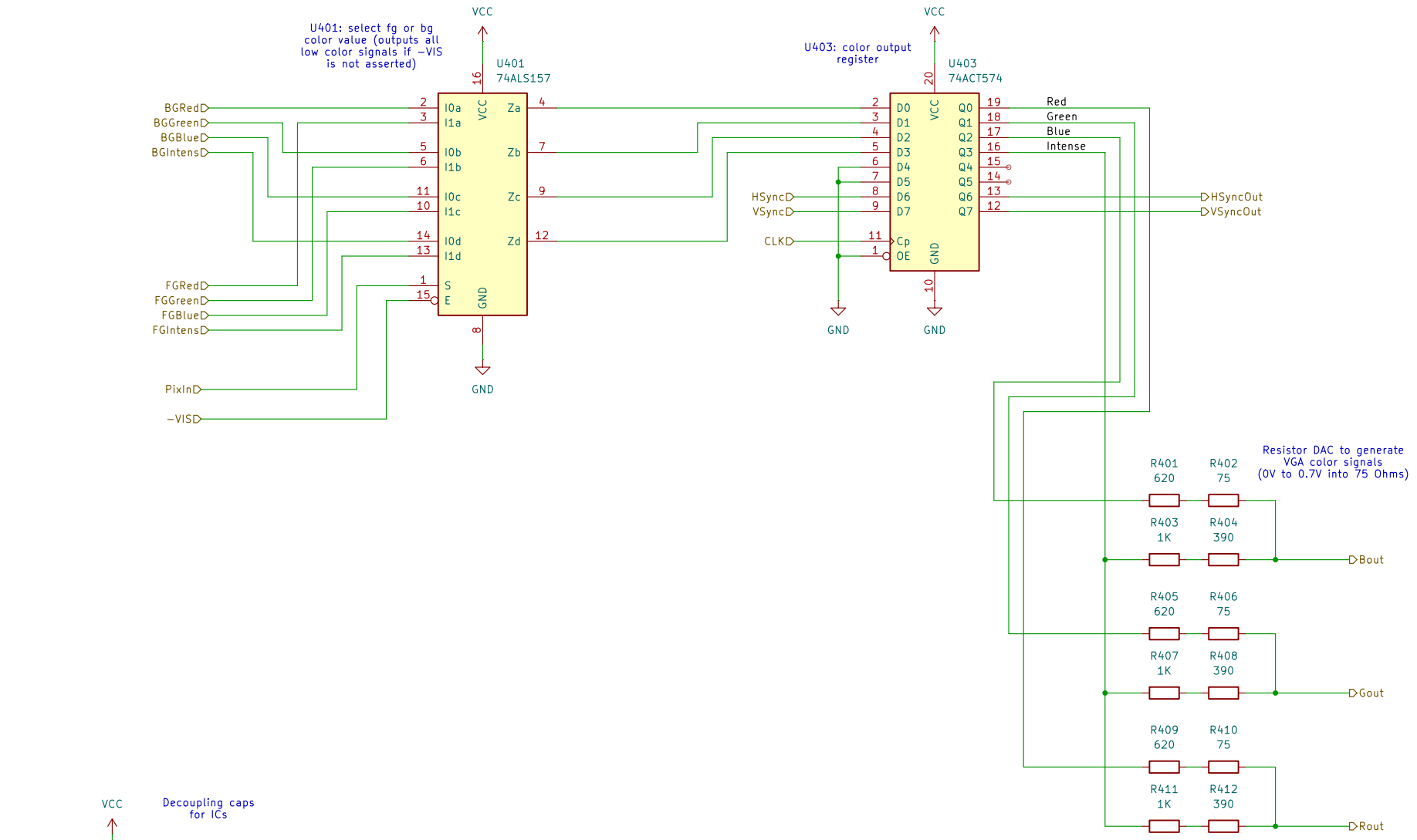
daveho hacks

Sheet: /VCount/

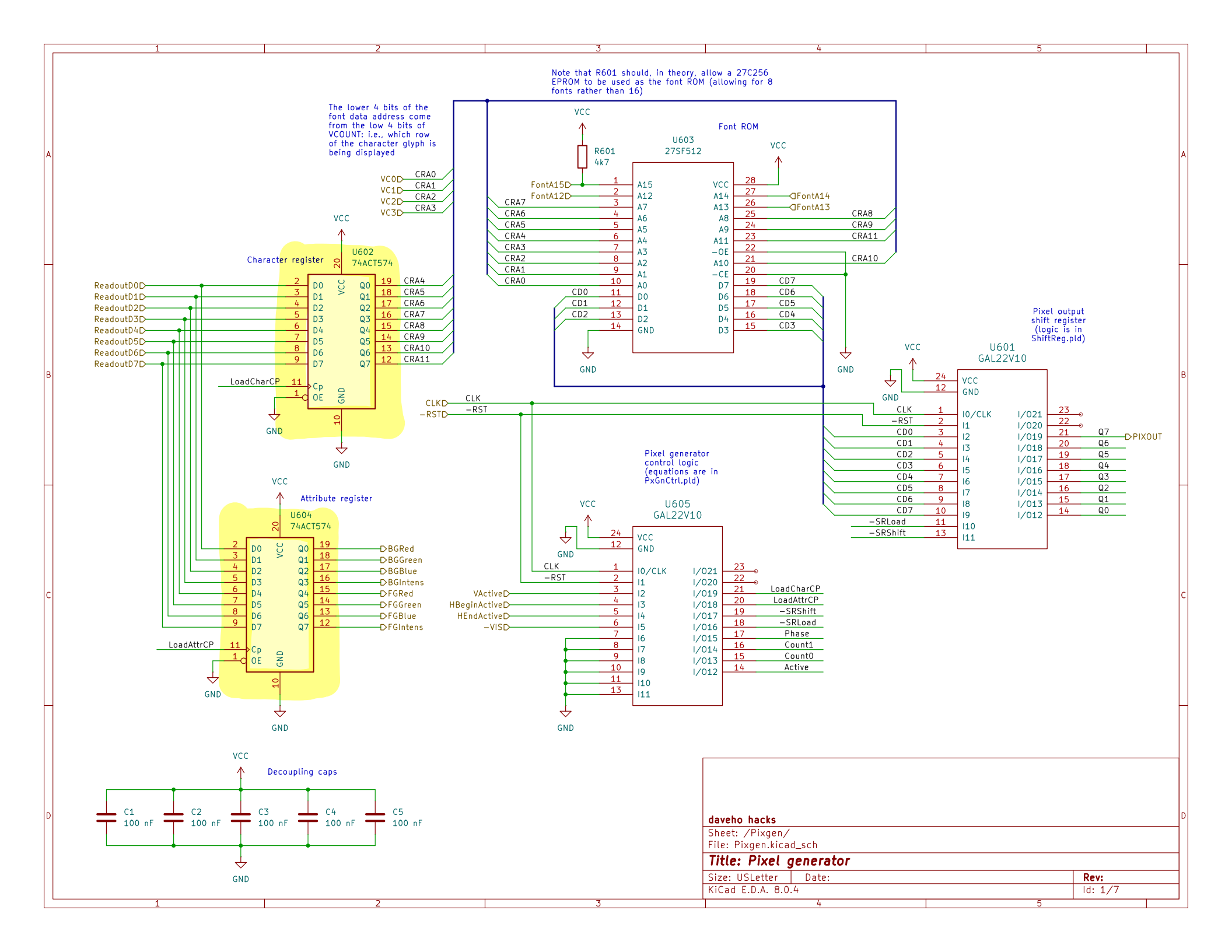
File: VCount.kicad_sch

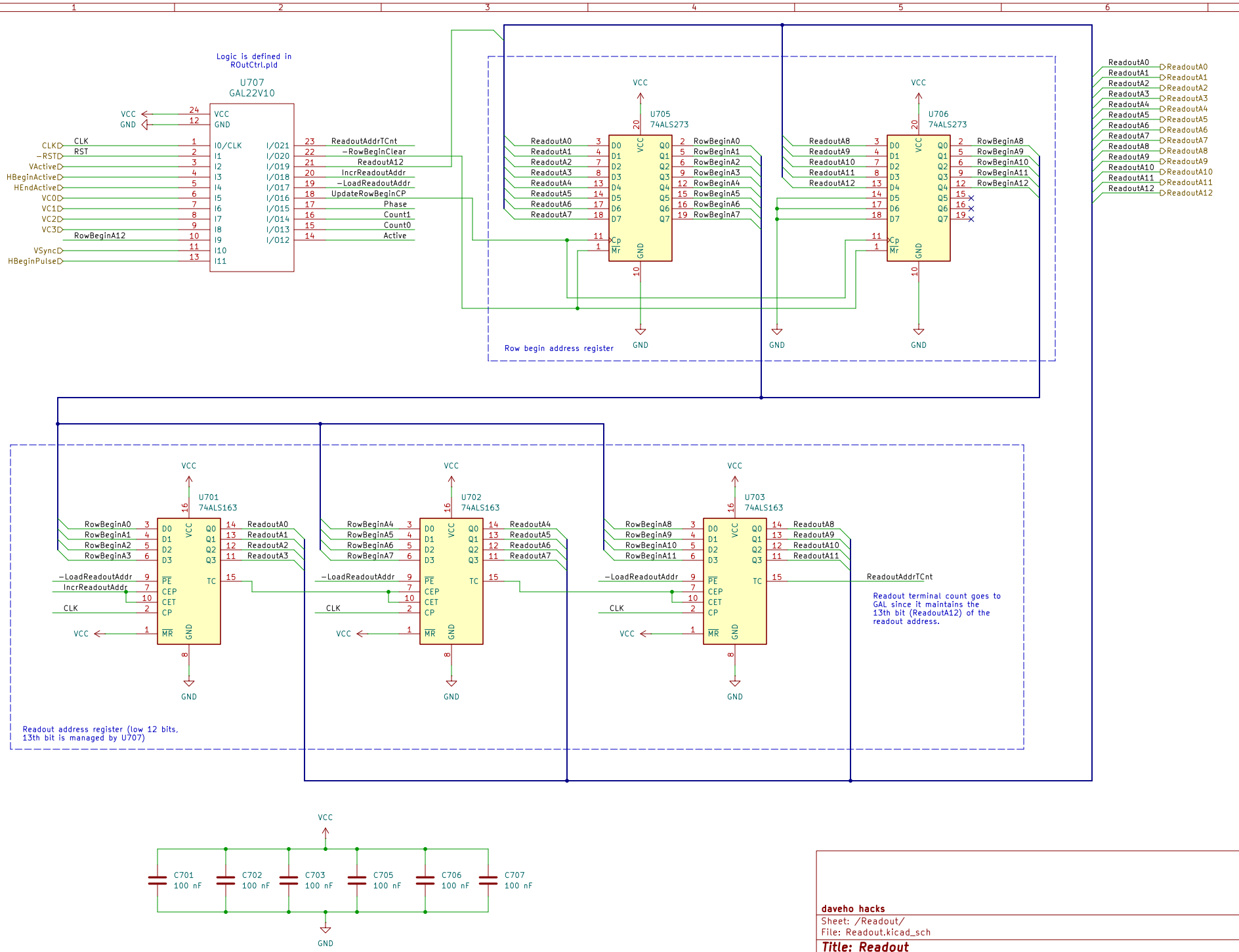
Title: Vertical count

Size: USLetter	Date:	**Rev:**
KiCad E.D.A. 8.0.4		Id: 1/7

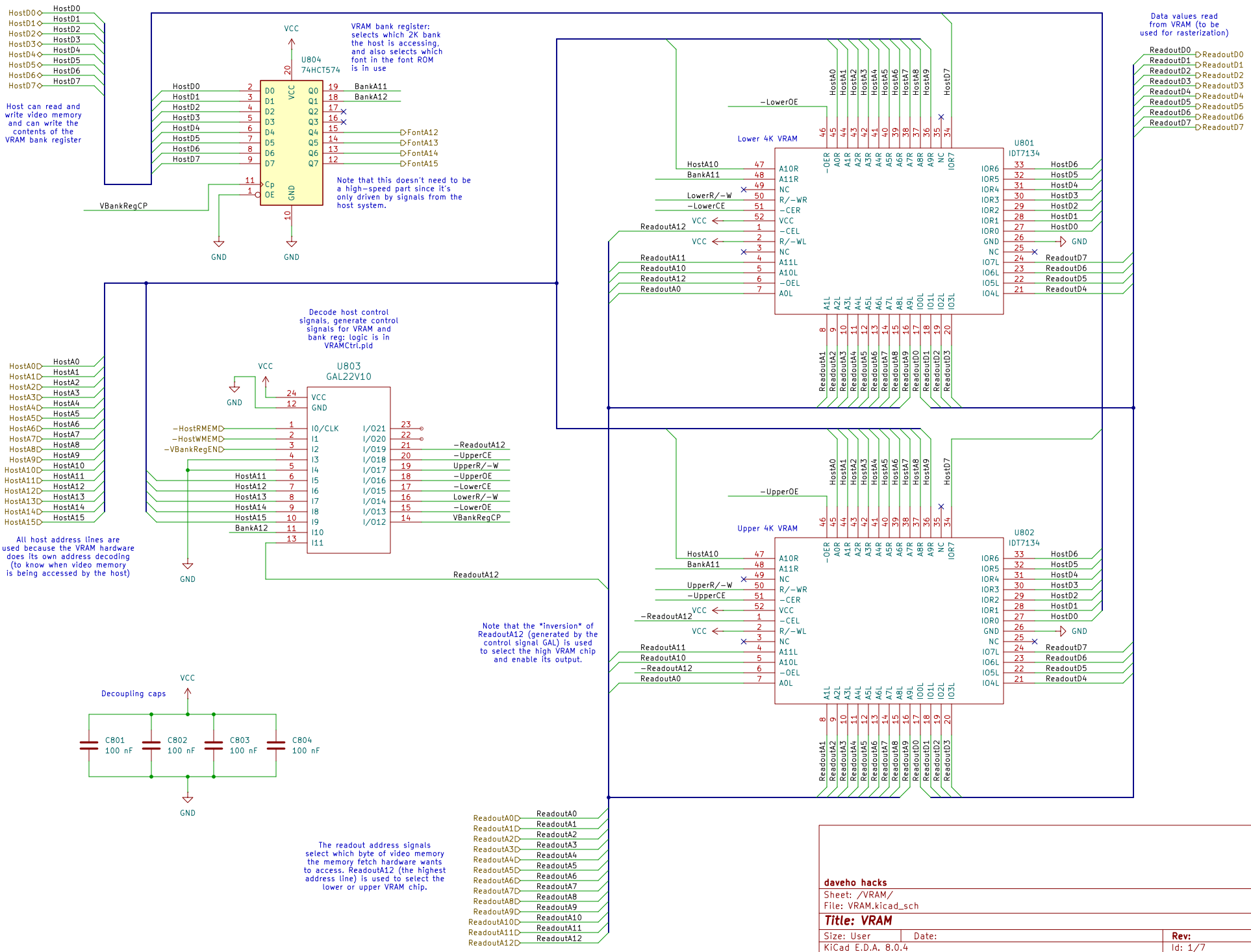


daveho hacks	
Sheet: /Output/	
File: Output.kicad_sch	
Title: Output	
Size: USLetter | Date:	**Rev:**
KiCad E.D.A. 8.0.4	Id: 1/7


Note that R601 should, in theory, allow a 27C256 EPROM to be used as the font ROM (allowing for 8 fonts rather than 16)
The lower 4 bits of the font data address come from the low 4 bits of VCOUNT: i.e., which row of the character glyph is being displayed
Font ROM
U603
27SF512
R601
4k7
VC0
VC1
VC2
VC3
CRA0
CRA1
CRA2
CRA3
FontA15
FontA12
FontA14
FontA13
CRA4
CRA5
CRA6
CRA7
CRA8
CRA9
CRA10
CRA11
CD0
CD1
CD2
CD3
CD4
CD5
CD6
CD7
VCC
GND
Character register
U602
74ACT574
ReadoutD0
ReadoutD1
ReadoutD2
ReadoutD3
ReadoutD4
ReadoutD5
ReadoutD6
ReadoutD7
LoadCharCP
Pixel output shift register (logic is in ShiftReg.pld)
U601
GAL22V10
CLK
−RST
PIXOUT
Pixel generator control logic (equations are in PxGnCtrl.pld)
U605
GAL22V10
VActive
HBeginActive
HEndActive
−VIS
LoadCharCP
LoadAttrCP
−SRShift
−SRLoad
Phase
Count1
Count0
Active
Attribute register
U604
74ACT574
BGRed
BGGreen
BGBlue
BGIntens
FGRed
FGGreen
FGBlue
FGIntens
LoadAttrCP
Decoupling caps
C1
C2
C3
C4
C5
100 nF
daveho hacks
Sheet: /Pixgen/
File: Pixgen.kicad_sch
Title: Pixel generator
Size: USLetter
Date:
Rev:
KiCad E.D.A. 8.0.4
Id: 1/7



daveho hacks	
Sheet: /Readout/	
File: Readout.kicad_sch	
Title: Readout	
Size: User	Date:
KiCad E.D.A. 8.0.4	Rev:
	Id: 1/7


Host can read and write video memory and can write the contents of the VRAM bank register
VRAM bank register: selects which 2K bank the host is accessing, and also selects which font in the font ROM is in use
U804 74HCT574
Note that this doesn't need to be a high-speed part since it's only driven by signals from the host system.
Data values read from VRAM (to be used for rasterization)
Lower 4K VRAM
U801 IDT7134
Upper 4K VRAM
U802 IDT7134
Decode host control signals, generate control signals for VRAM and bank reg: logic is in VRAMCtrl.pld
U803 GAL22V10
All host address lines are used because the VRAM hardware does its own address decoding (to know when video memory is being accessed by the host)
Note that the *inversion* of ReadoutA12 (generated by the control signal GAL) is used to select the high VRAM chip and enable its output.
Decoupling caps
C801 100 nF
C802 100 nF
C803 100 nF
C804 100 nF
The readout address signals select which byte of video memory the memory fetch hardware wants to access. ReadoutA12 (the highest address line) is used to select the lower or upper VRAM chip.
daveho hacks
Sheet: /VRAM/
File: VRAM.kicad_sch
Title: VRAM
Size: User
Date:
Rev:
KiCad E.D.A. 8.0.4
Id: 1/7