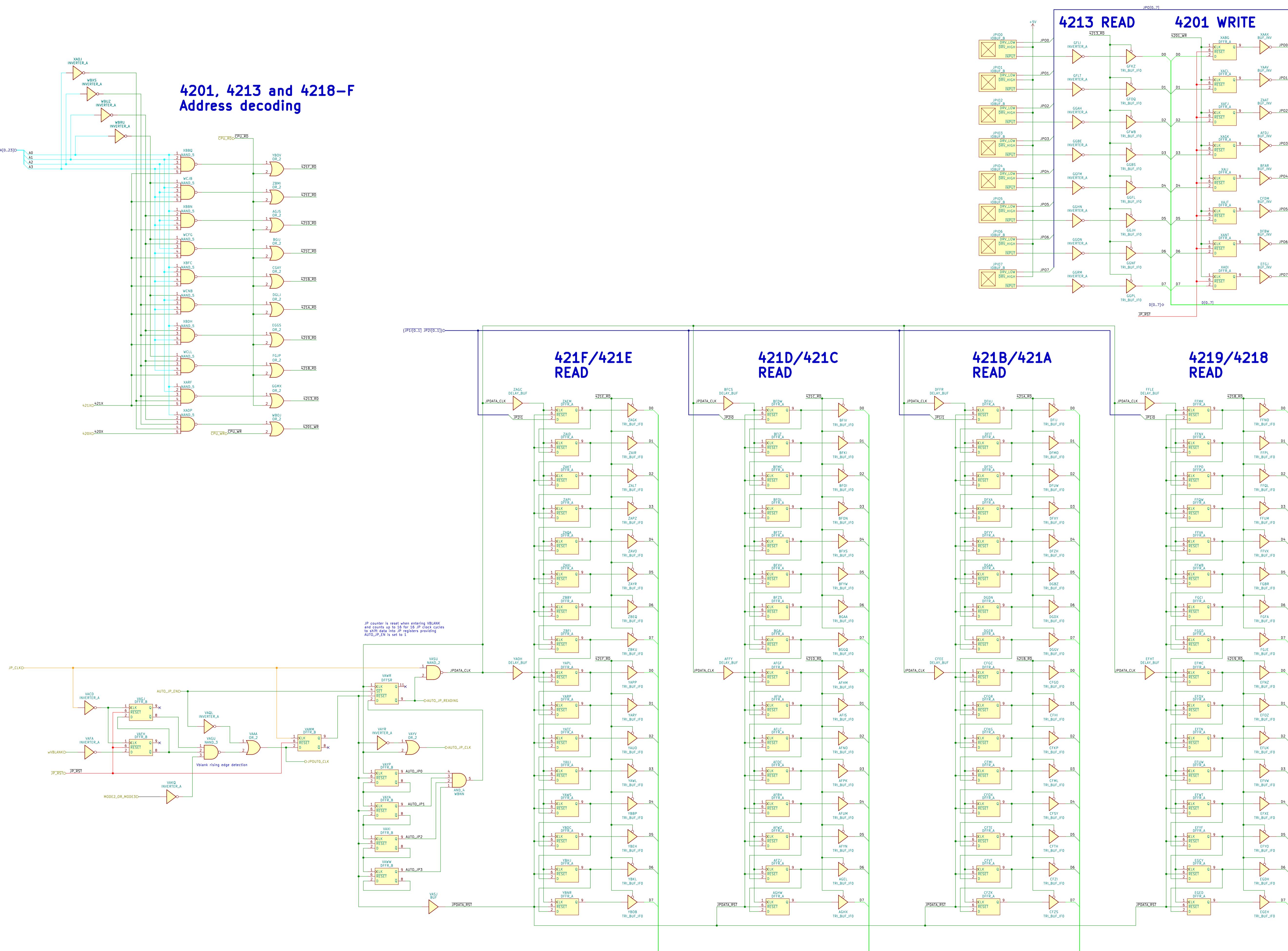
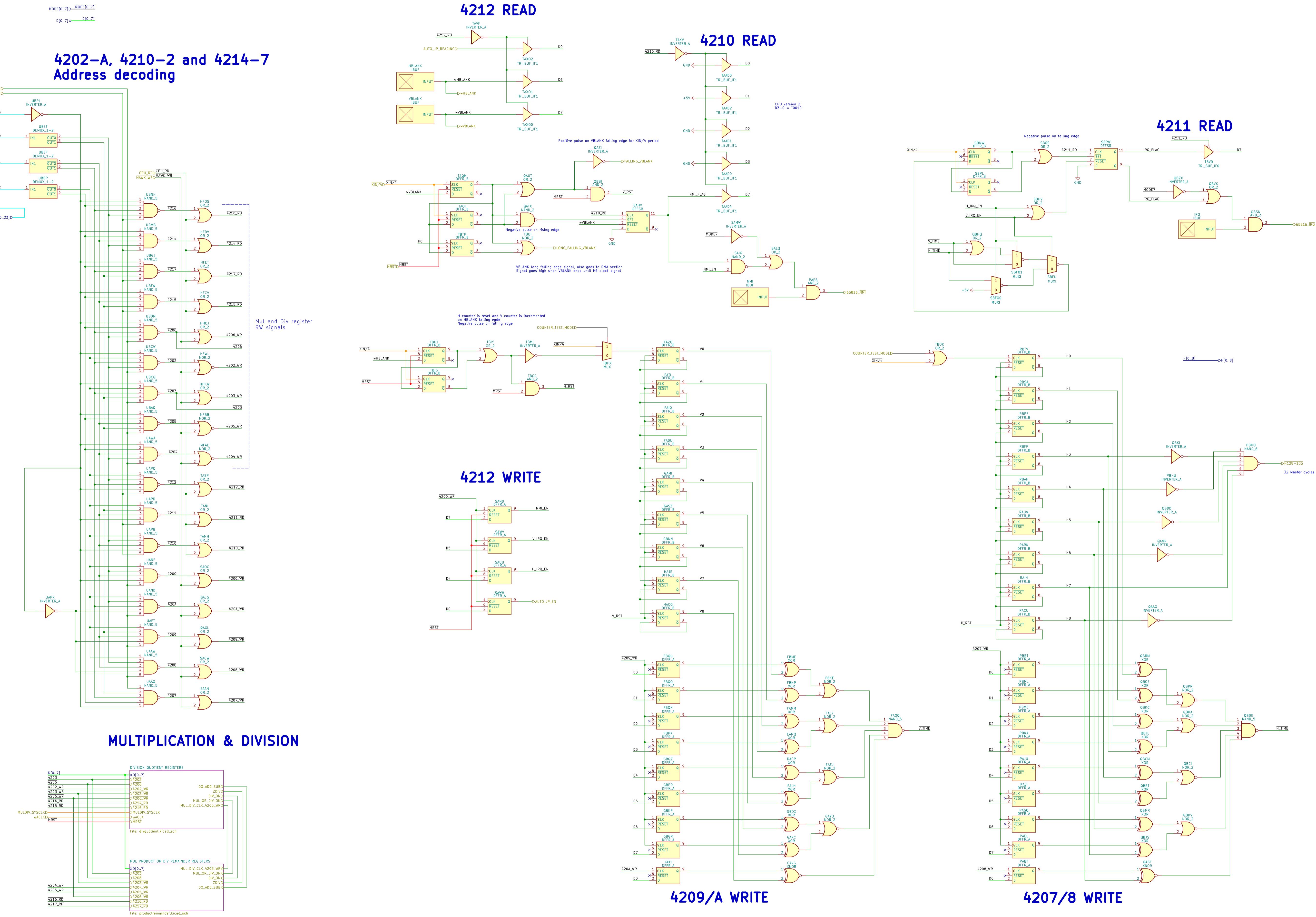


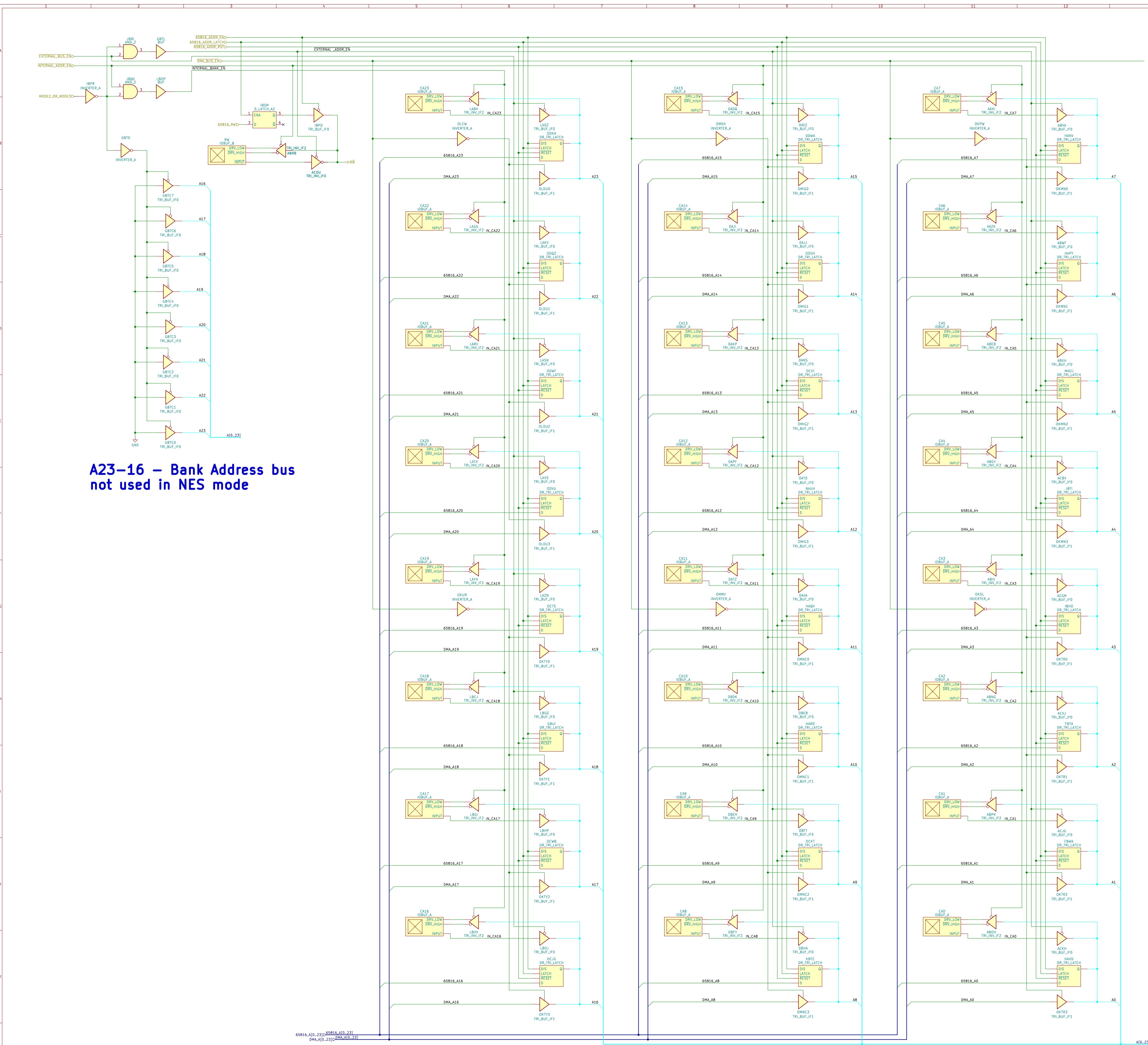
Author: Regis Galland  
Sheet: /  
File: S-CPU.kicad\_sch

Title: Reverse Engineered SNES S-CPU Schematics

Size: A2	Date: 2024-12-24	Rev: 0.2
KiCad E.D.A. 8.0.7		16: 1/20

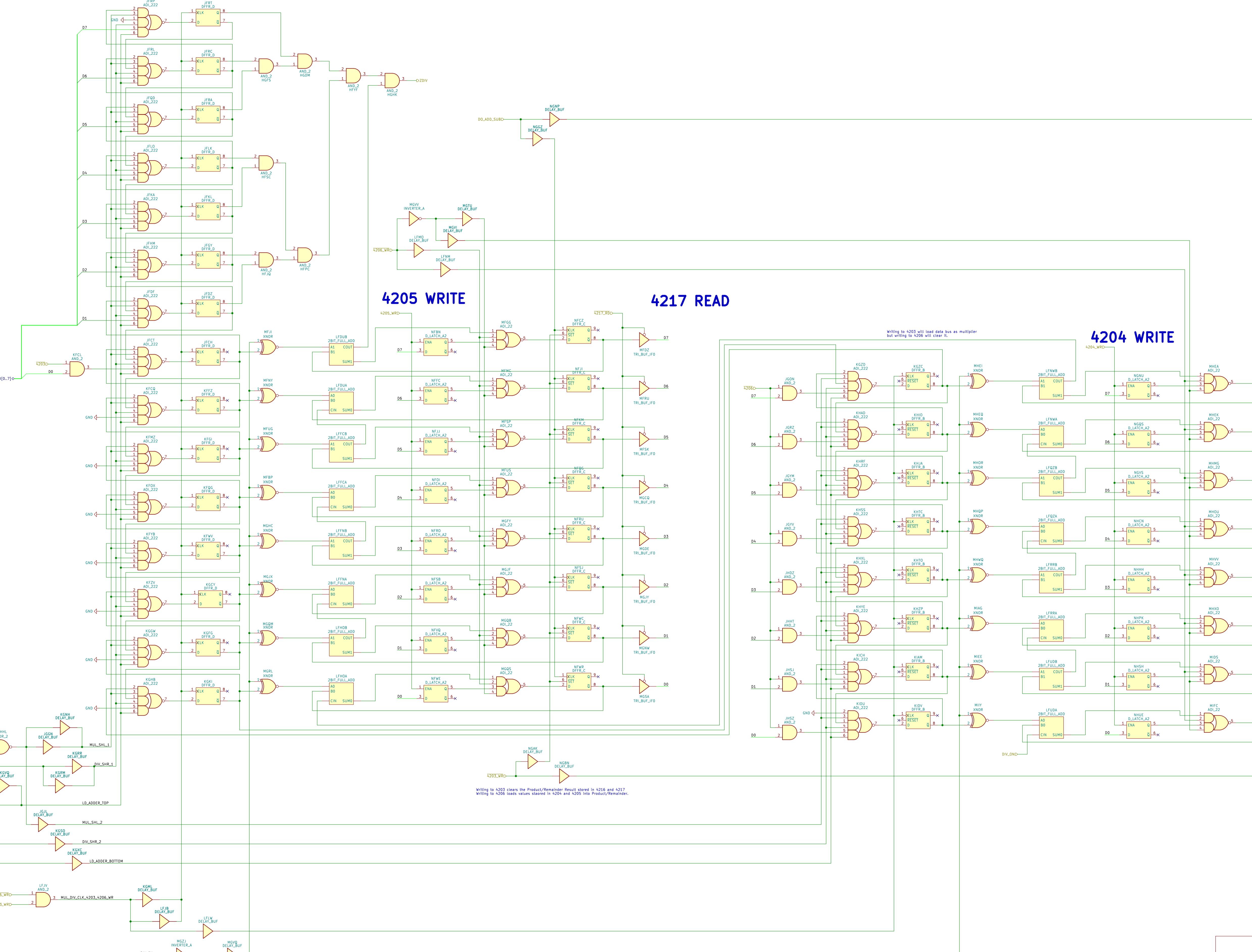


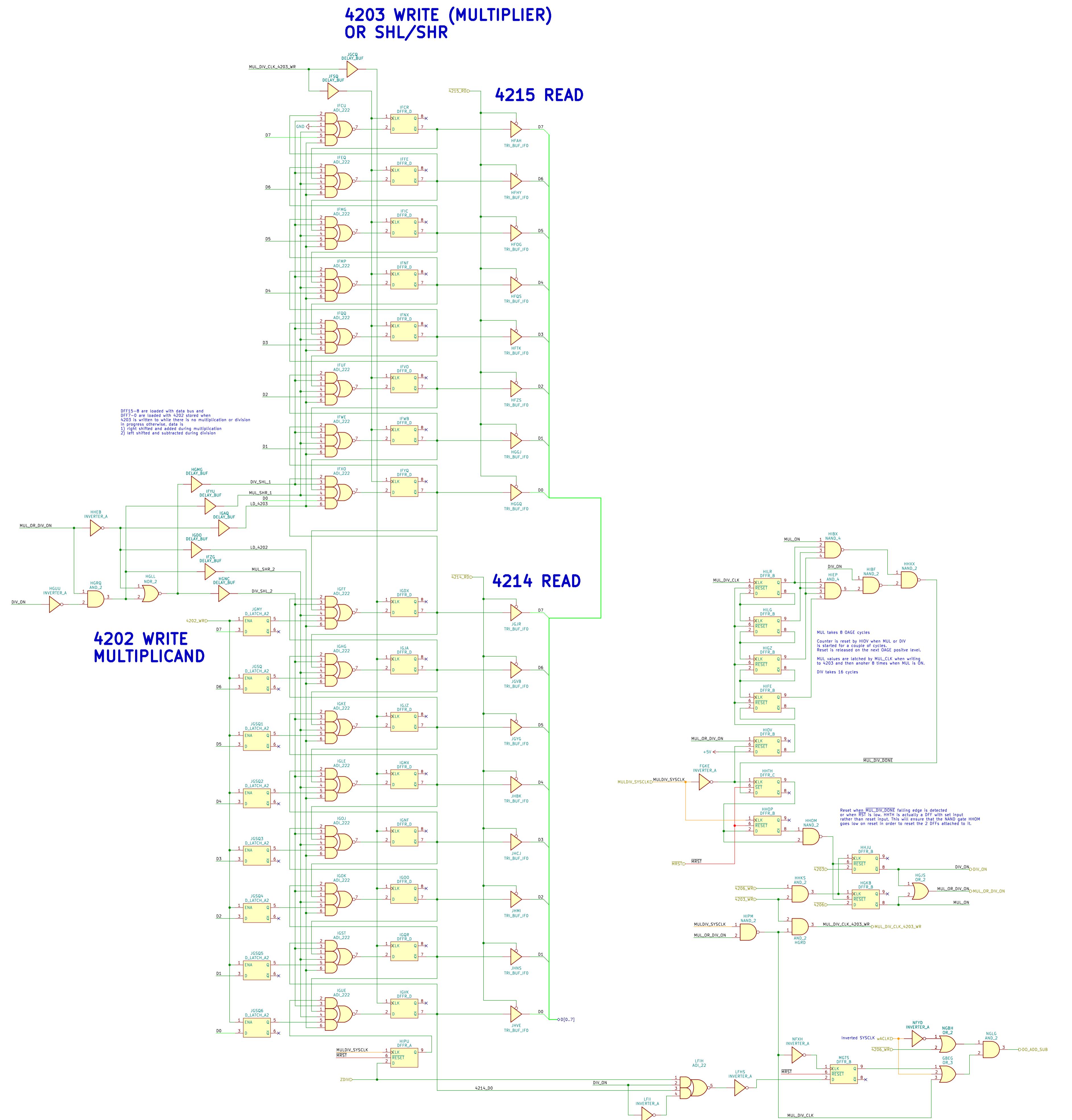


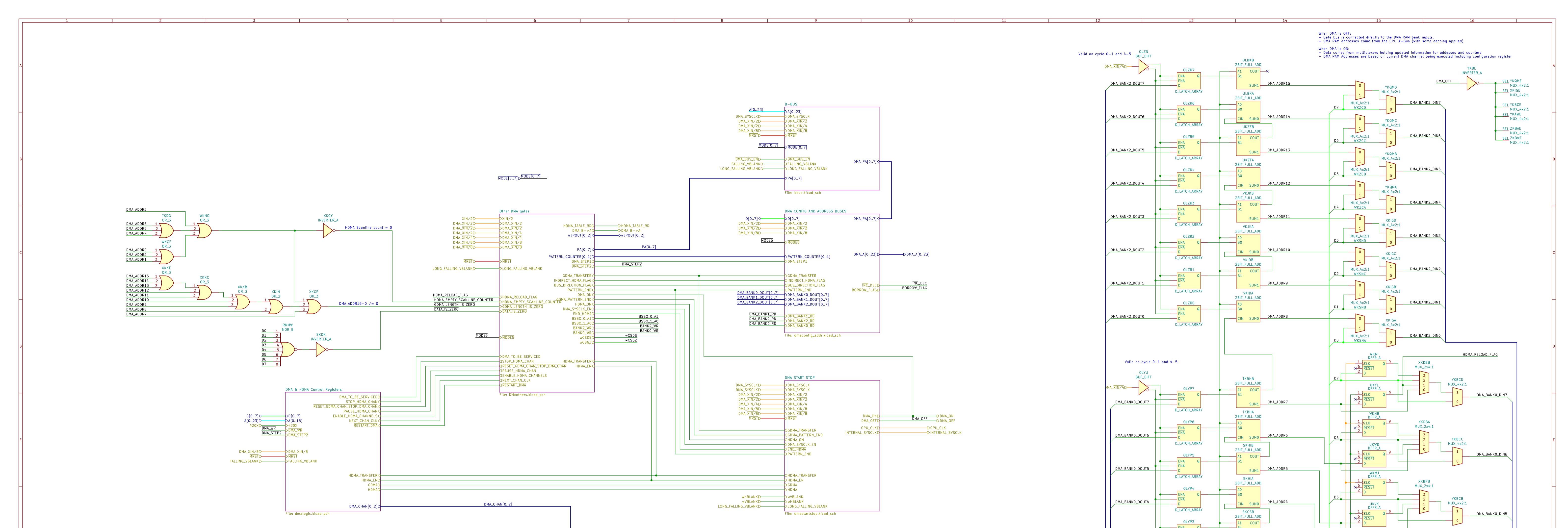


**A-BUS**

# 4203 WRITE (MULTIPLIER) OR 4206 WRITE OR SHL/SHR





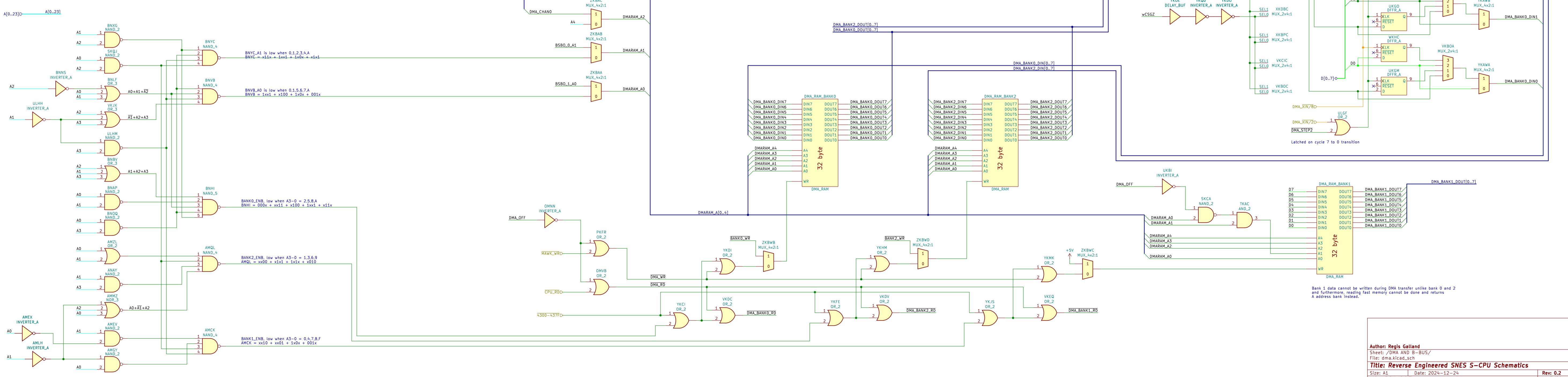


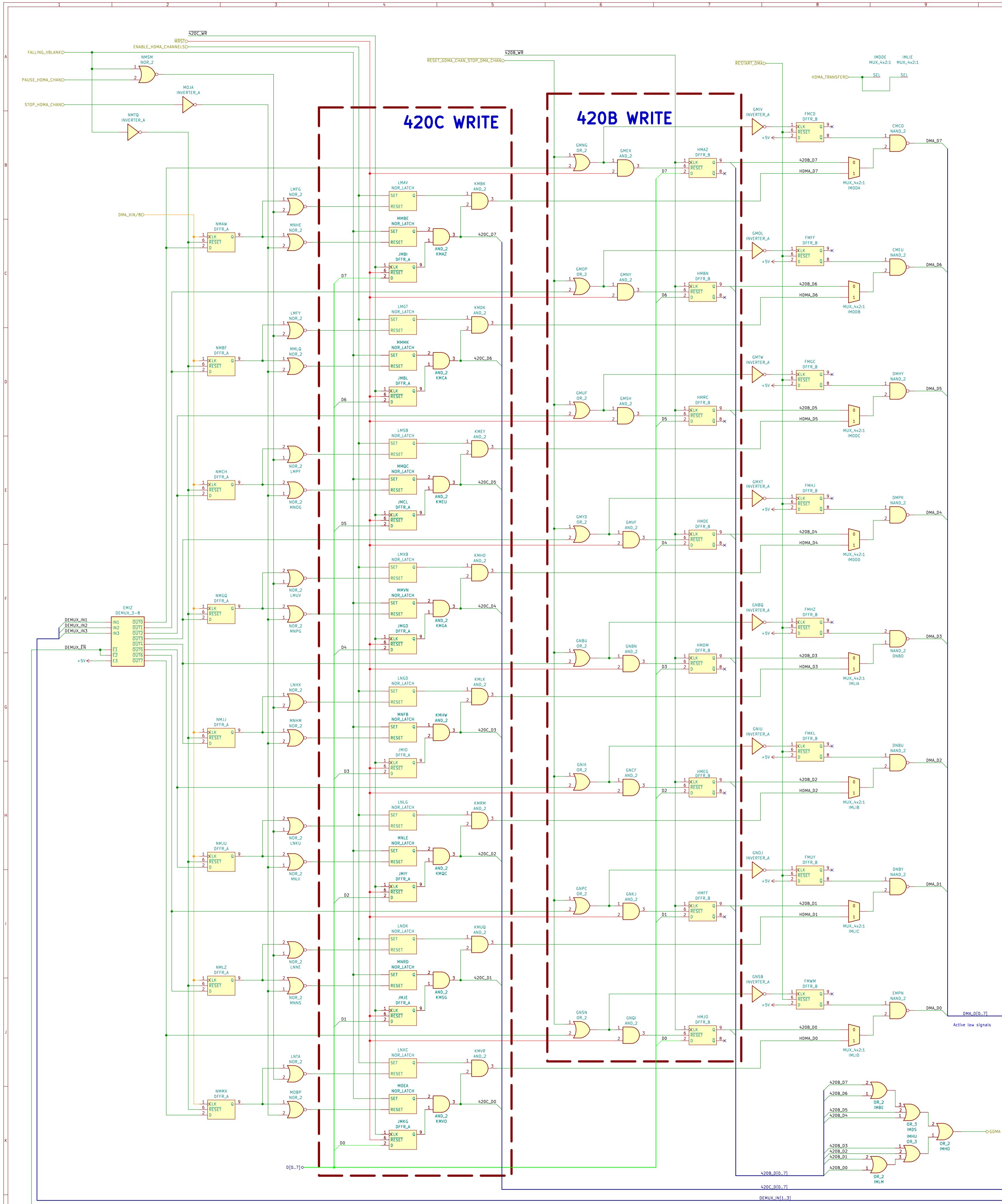
DMA RAM control signals decoding (Registers 43x0–43xF)

A3	A2	A1	A0	BNHI BO_ENB	AMCK B1_ENB	AMQL B2_ENB	BNYC A1	BNVB A0	DMA Registers
0	0	0	0	1	0	1	0	0	Parameters for DMA Transfer
0	0	0	1	1	1	0	0	0	B Address
0	0	1	0	0	1	1	0	1	A Address (Low Byte)
0	0	1	1	1	1	0	0	1	A Address (High Byte)
0	1	0	0	1	0	1	0	1	A Address Bank
0	1	0	1	0	1	1	1	0	Number Bytes to Transfer (Low Byte) (DMA)
0	1	1	0	1	1	0	1	0	Number Bytes to Transfer (High Byte) (DMA)
0	1	1	1	1	0	1	1	0	Data Bank (HDMA)
1	0	0	0	0	1	1	1	1	A2 Table Address (Low Byte)
1	0	0	1	1	1	0	1	1	A2 Table Address (High Byte)
1	0	1	0	0	1	1	0	0	Number of Lines to Transfer (HDMA)
1	0	1	1	1	0	1	1	1	Fast Access memory
1	1	0	0	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x
1	1	1	1	1	0	1	1	1	Fast Access memory mirror

**g DMA, A1-0 =**  
" will read DMA parameters from bank 1. B-bus address from bank 2 and Number of lines to transfer from bank 0 (only during HDMA).  
" will read A-BUS address (16 bits) from bank 0 and 2 and A-BUS bank from bank 1. Bank does not get incremented when 16-bit address spills over.  
" will read number of GDMA bytes to be transferred (16-bit) from bank 0 and 2 and HDMA data bank from bank 1  
" will read HDMA A2 table address from bank 0 and 2 and Fast access memory from bank 1 (but not used for anything)

## Address Decoding

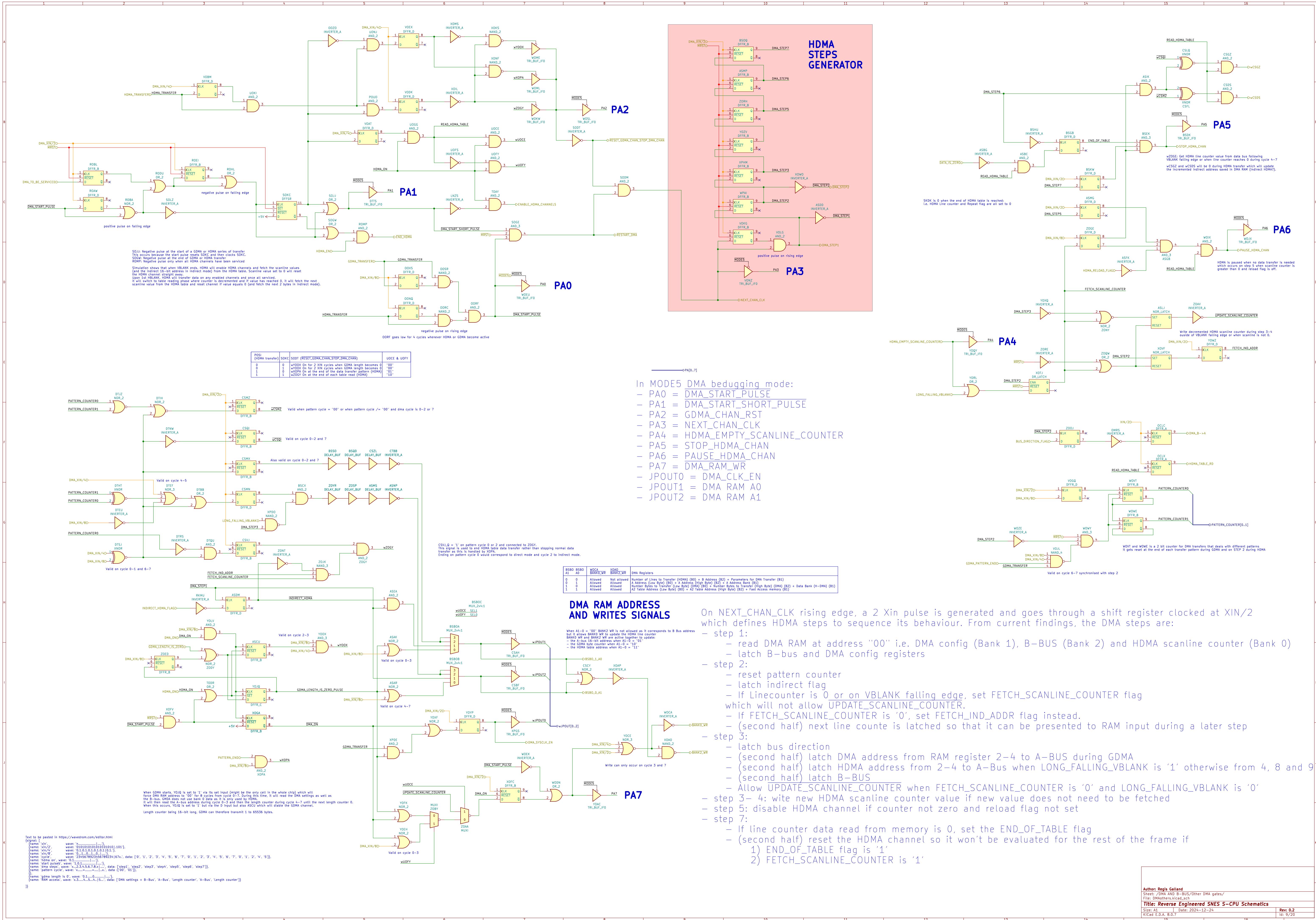




### ADDRESS DECODING

### 420C WRITE

### 8x3 Priority Encoder

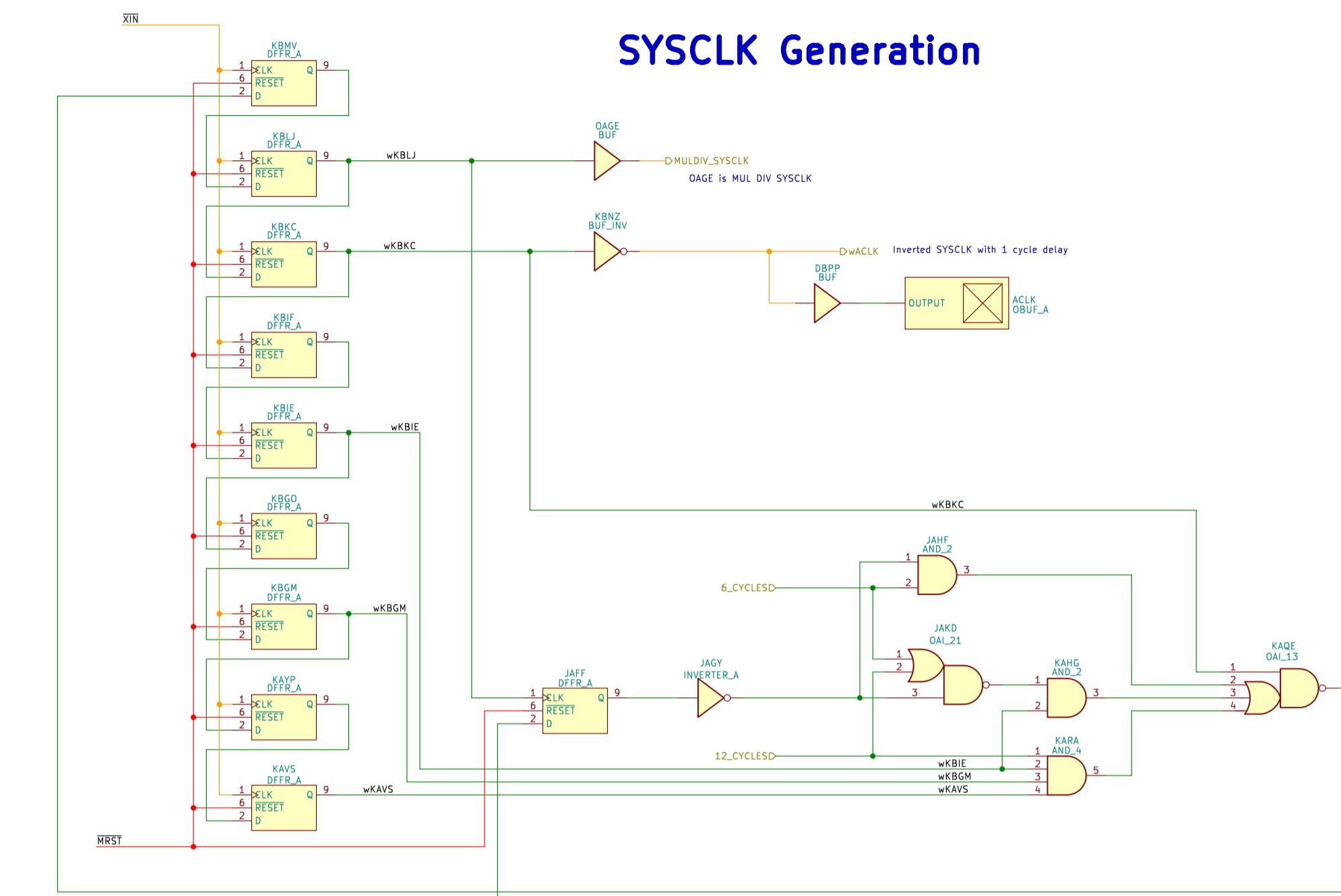


In MODE5 DMA debugging mode:

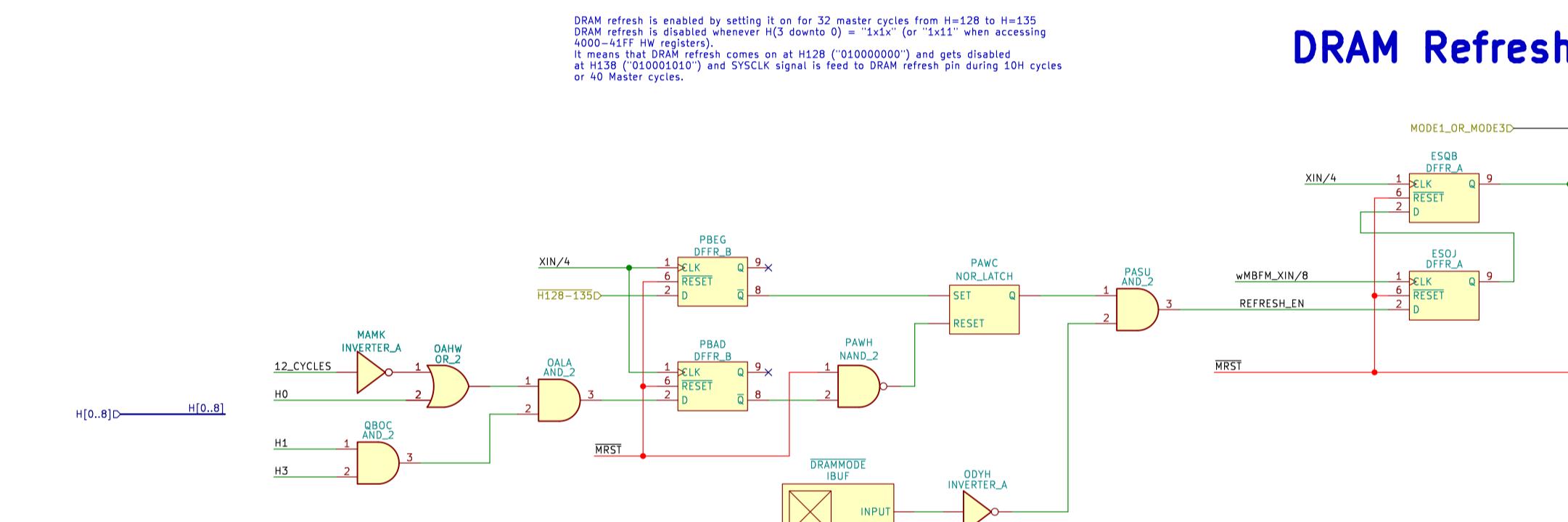
- PA0 = DMA\_START\_PULSE
- PA1 = DMA\_START\_SHORT\_PULSE
- PA2 = GDMA\_CHAN\_RST
- PA3 = NEXT\_CHAN\_CLK
- PA4 = HDMA\_EMPTY\_SCANLINE\_COUNTER
- PA5 = STOP\_HDMA\_CHAN
- PA6 = PAUSE\_HDMA\_CHAN
- PA7 = DMA\_RAM\_WR
- JPOUT0 = DMA\_CLK\_EN
- JPOUT1 = DMA RAM AO
- JPOUT2 = DMA RAM A1

On NEXT\_CHAN\_CLK rising edge, a 2 Xin pulse is generated and goes through a shift register clocked at XIN/2 which defines HDMA steps to sequence its behaviour. From current findings, the DMA steps are:

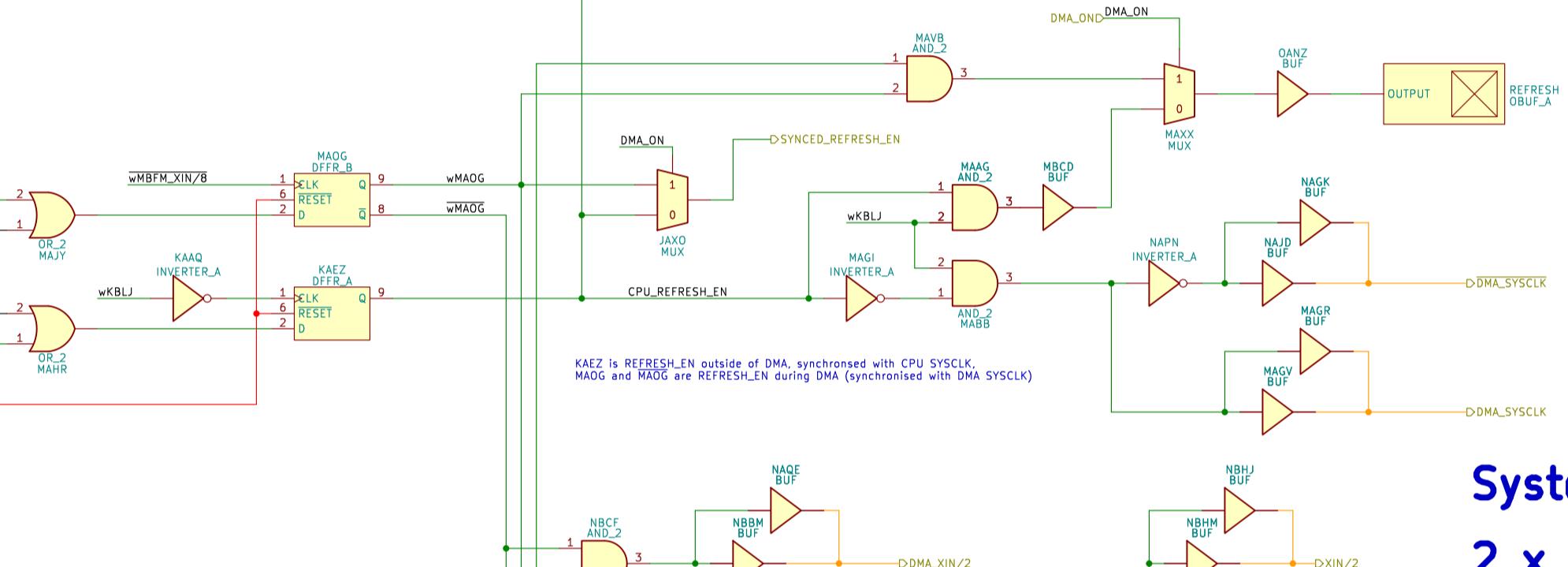
- step 1:
  - read DMA RAM at address "00" i.e. DMA config (Bank 1), B-BUS (Bank 2) and HDMA scanline counter (Bank 0)
  - latch B-bus and DMA config registers
- step 2:
  - reset pattern counter
  - latch indirect flag
  - If Linecounter is 0 or on VBLANK falling edge, set FETCH\_SCANLINE\_COUNTER flag which will not allow UPDATE\_SCANLINE\_COUNTER.
  - If FETCH\_SCANLINE\_COUNTER is '0', set FETCH\_IND\_ADDR flag instead.
  - (second half) next line count is latched so that it can be presented to RAM input during a later step
- step 3:
  - latch bus direction
  - (second half) latch DMA address from RAM register 2-4 to A-Bus during GDMA
  - (second half) latch HDMA address from 2-4 to A-Bus when LONG\_FALLING\_VBLANK is '1' otherwise from 4, 8 and 9
  - (second half) latch B-BUS
  - Allow UPDATE\_SCANLINE\_COUNTER when FETCH\_SCANLINE\_COUNTER is '0' and LONG\_FALLING\_VBLANK is '0'
- step 3-4: write new HDMA scanline counter value if new value does not need to be fetched
- step 5: disable HDMA channel if counter not zero and reload flag not set
- step 7:
  - If line counter data read from memory is 0, set the END\_OF\_TABLE flag
  - (second half) reset the HDMA channel so it won't be evaluated for the rest of the frame if
    - 1) END\_OF\_TABLE flag is '1'
    - 2) FETCH\_SCANLINE\_COUNTER is '1'



DRAM refresh is enabled by setting it on for 32 master cycles from H=128 to H=0. DRAM refresh is disabled whenever H(3 downto 0) = "1x1x" (or "1x11" when acc=4000-41FF HW registers). It means that DRAM refresh comes on at H128 ("01000000") and gets disabled at H138 ("01000101") and SYSLCK signal is feed to DRAM refresh pin during 10 or 40 Master cycles.



# DRAM Refresh



# SYSCLK and SYSCLK signals used in the DMA region

# System clocks

## 2 x XIN/2

**1 x XIN/2**

**2 x XIN/4**

$$2 \times \overline{XIN/4}$$

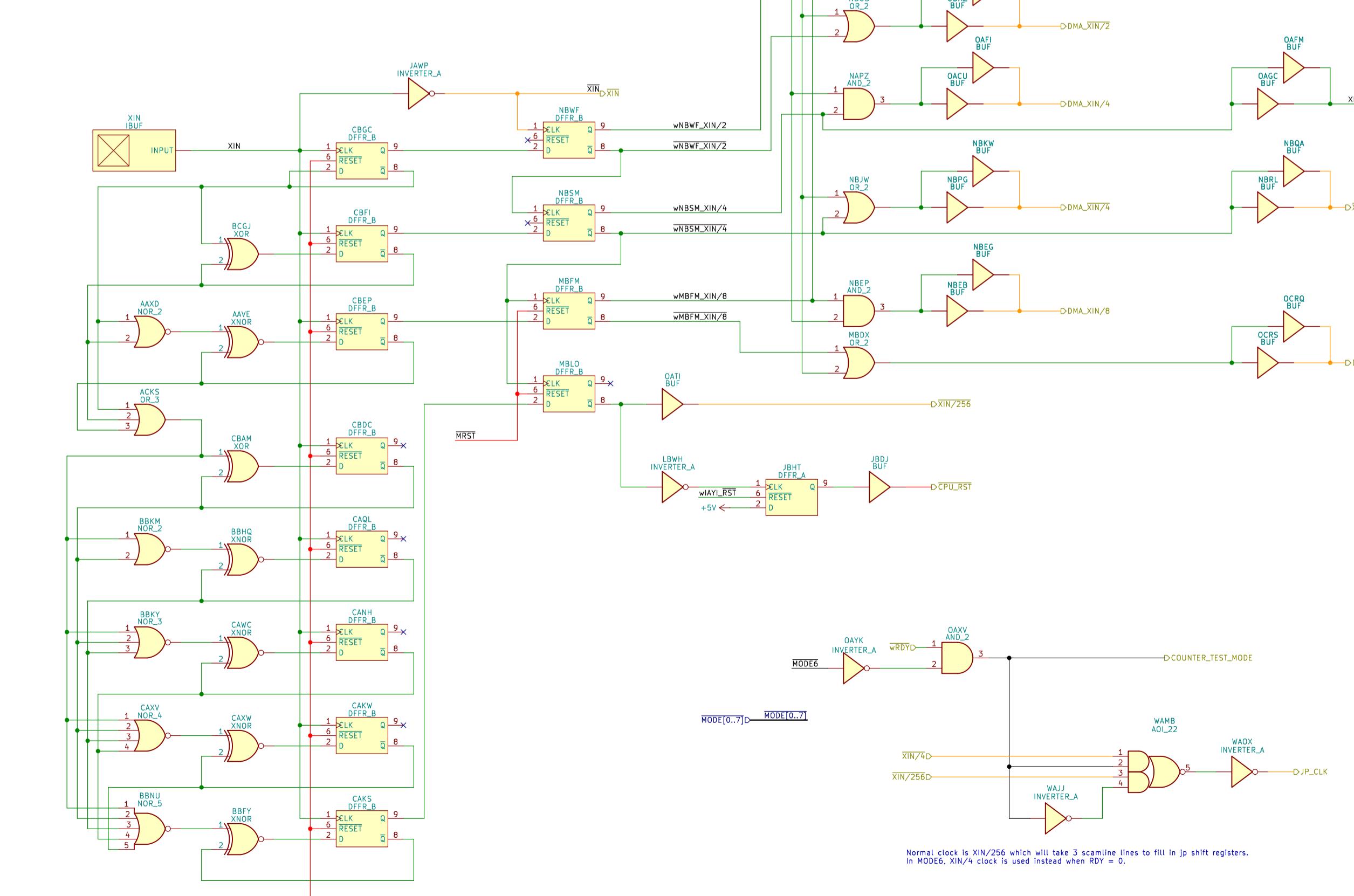
1 x XIN/8

1 x XIN/8

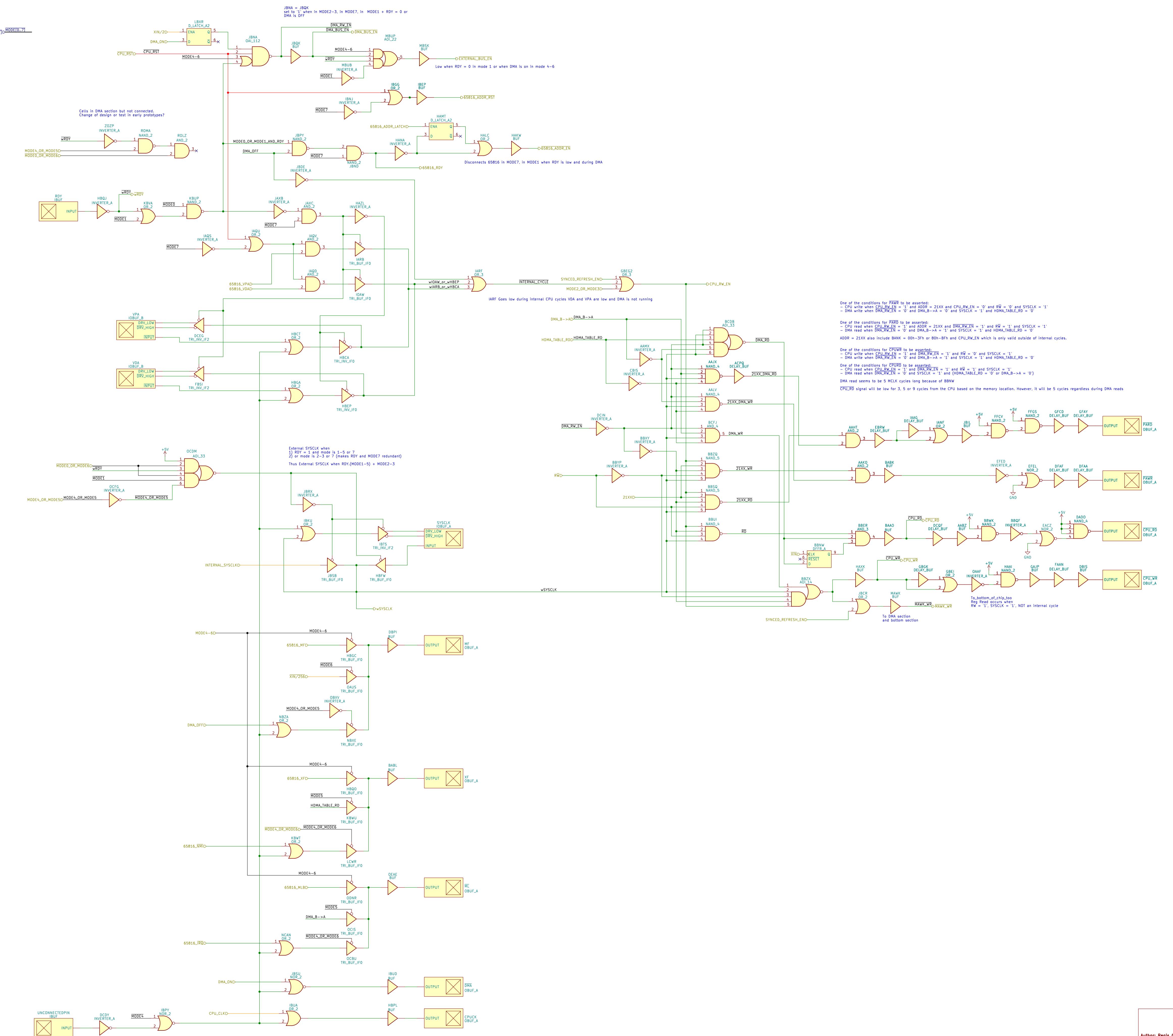
**1 x XIN/25**

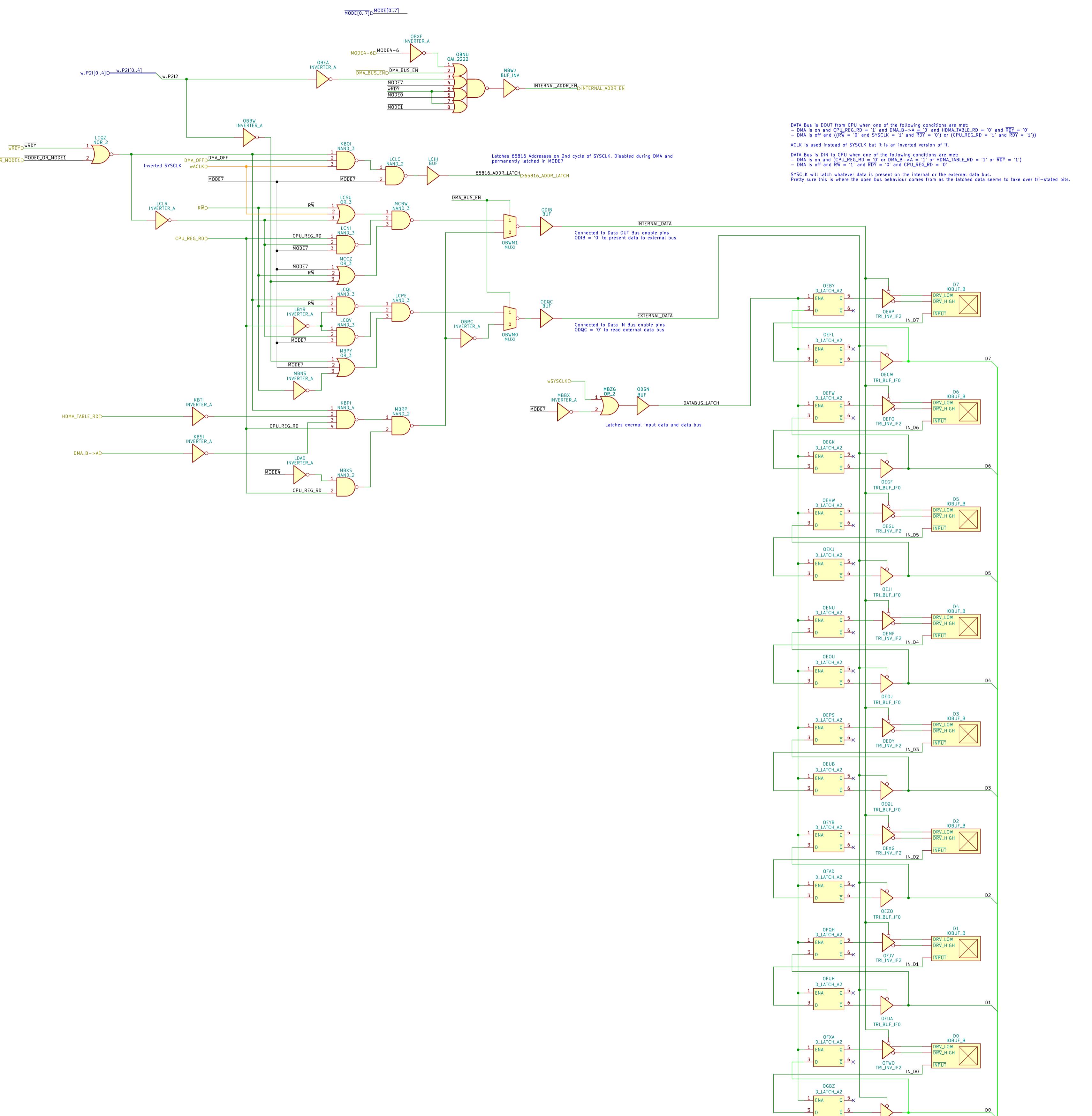
The diagram illustrates the RESET logic for several components:

- RES (BUF)**: A buffer with an input labeled **INPUT** and an output labeled **wRES**.
- XIN**: An input signal connected to the **RESET** pin of three D flip-flops.
- IBAL\_DFFRA**: A D flip-flop with **RESET** pin 2 connected to **wRES**. Its output **Q** is connected to the **MASS BUF** and to the **JP regs reset** output.
- IAYI\_DFFRA**: A D flip-flop with **RESET** pin 2 connected to **wRES**. Its output **Q** is connected to the **wIAYI\_RST** output.
- IAUB\_INVERTER\_A**: An inverter with an input from **wRES** and an output connected to the **NOR\_2** gate.
- IARQ\_NOR\_2**: A NOR gate with two inputs: one from the **IAUB\_INVERTER\_A** output and another from the **IBAL\_DFFRA** **RESET** pin. Its output is connected to the **IAUO\_DFFRA** **RESET** pin.
- IAUO\_DFFRA**: A D flip-flop with **RESET** pin 2 connected to the output of the **IARQ\_NOR\_2** gate. Its output **Q** is connected to the **DBJW BUF** and to the **MRST** output.
- HASI\_BUF**: A buffer with an input from the **IAUO\_DFFRA** **Q** pin and an output connected to the **MRST** output.
- JP regs reset**, **wIAYI\_RST**, and **MRST** are the final outputs of the system.

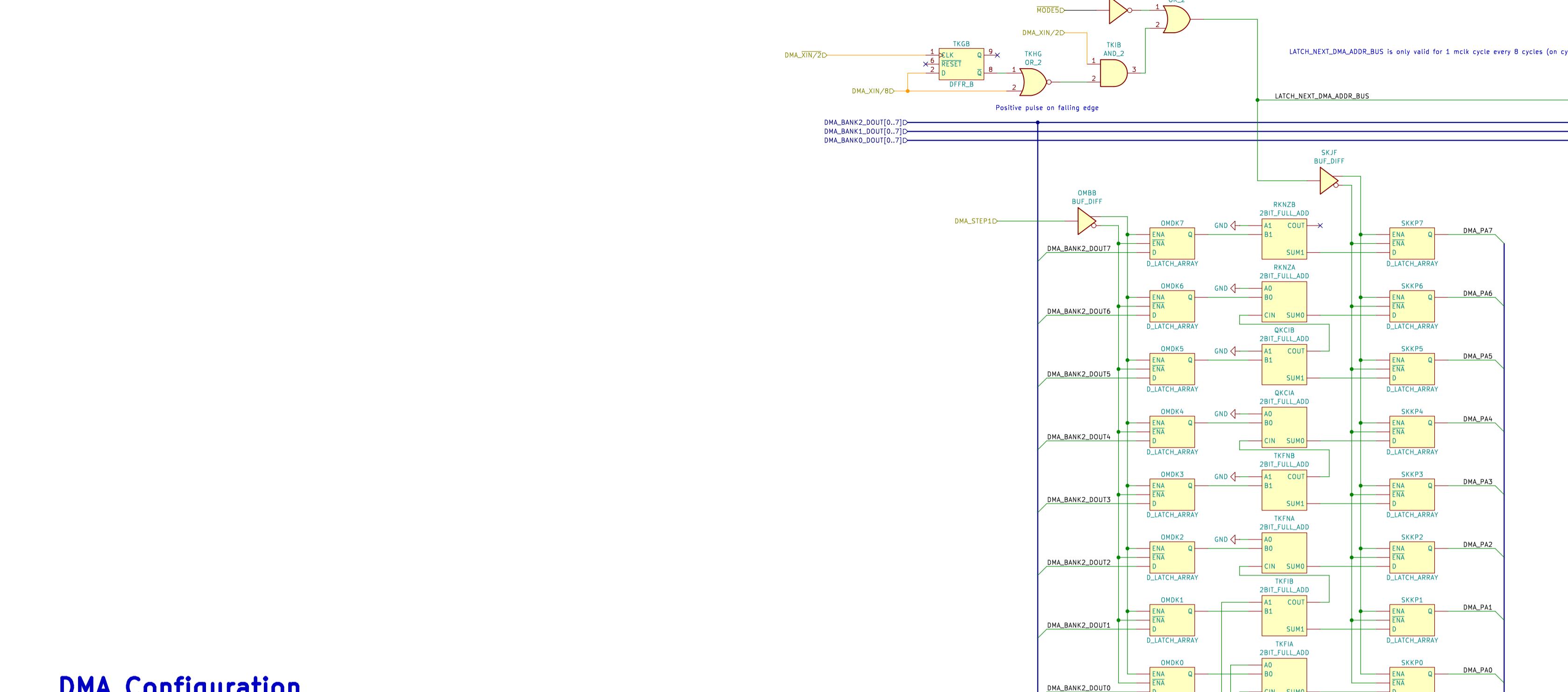


Normal clock is XIN/256 which will take 3 scamline lines to fill in jp shift regis.  
In MODE6, XIN/4 clock is used instead when RDY = 0.

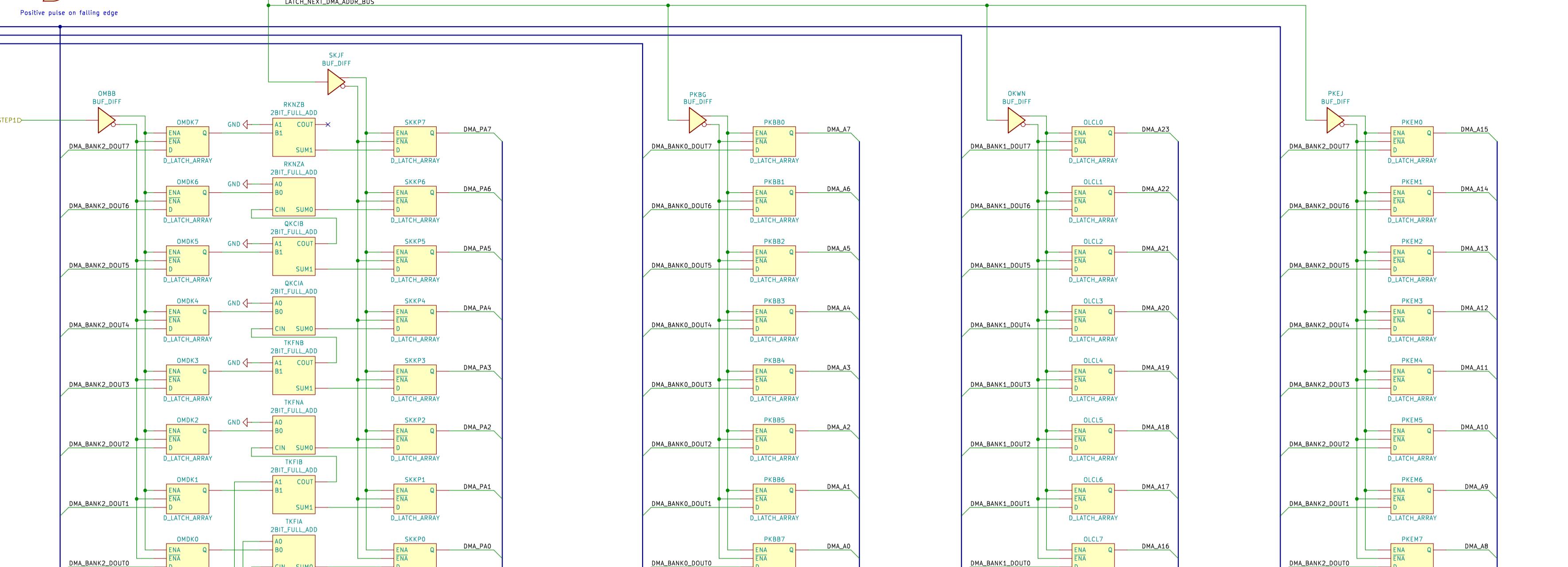




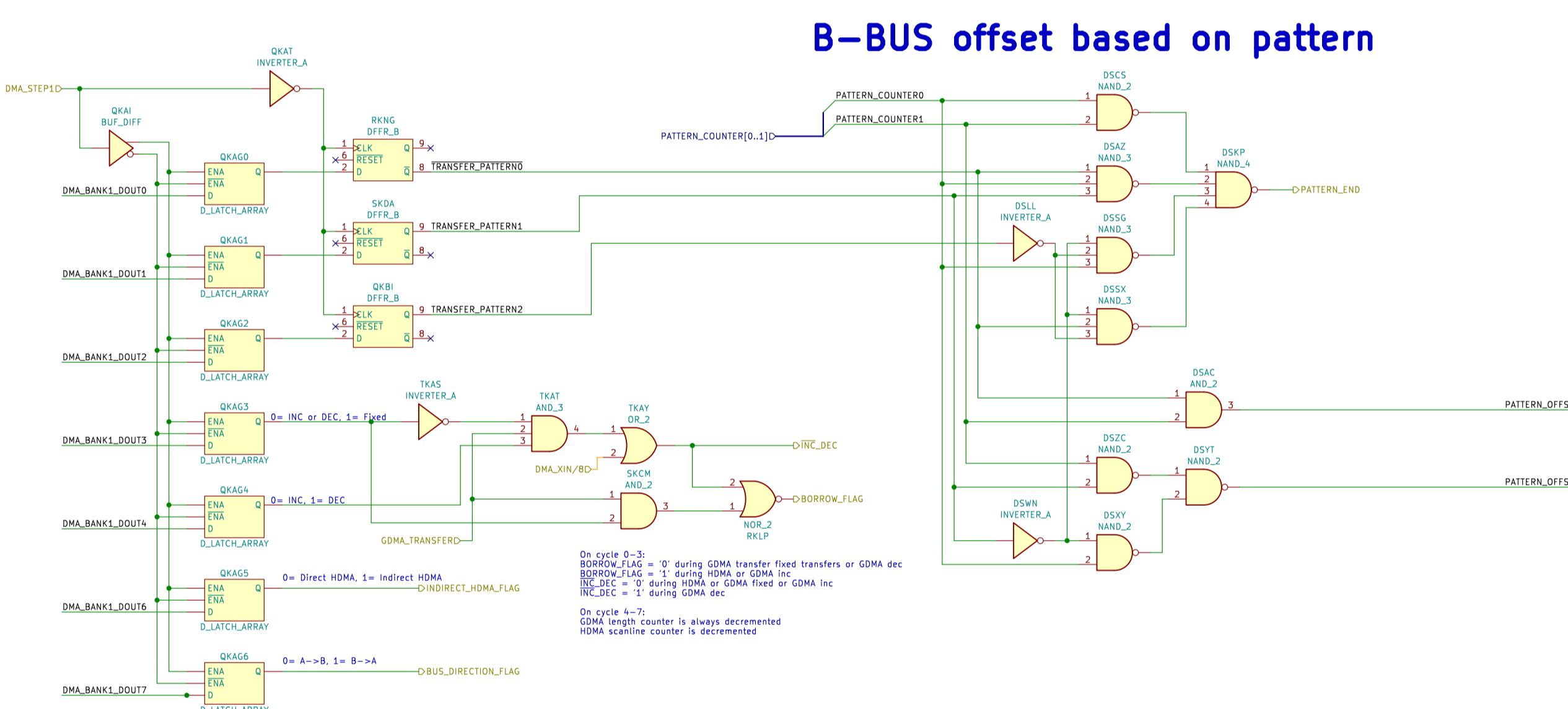
## DMA Configuration



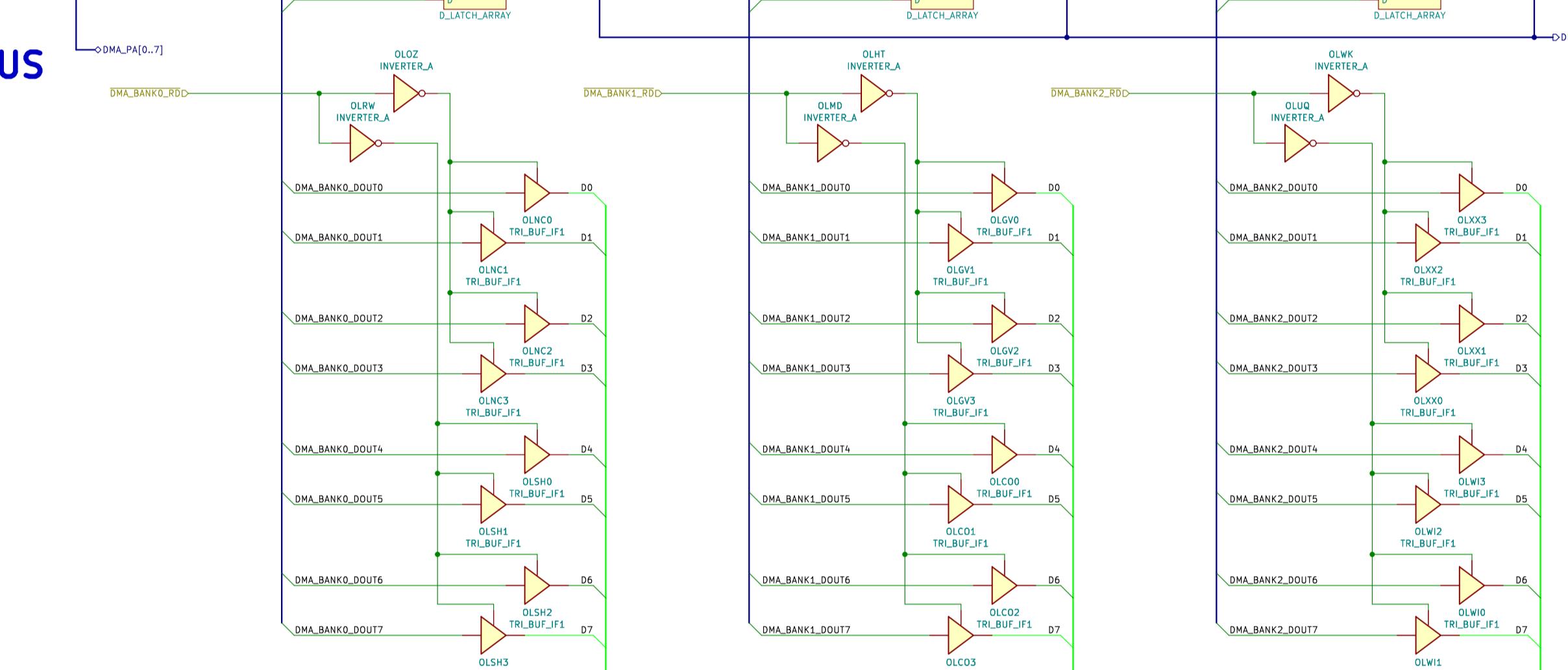
## DMA A-BUS



## B-BUS offset based on pattern

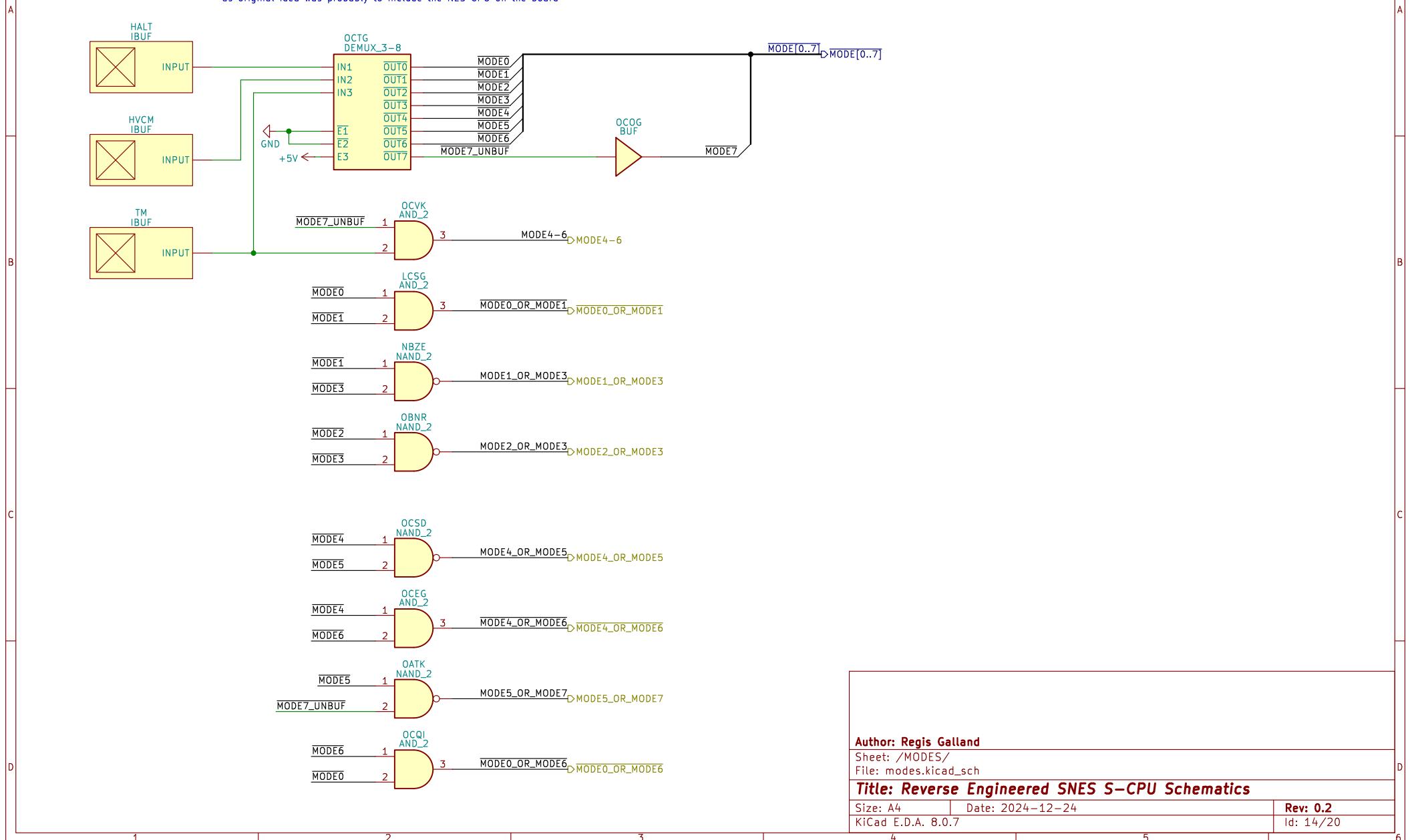


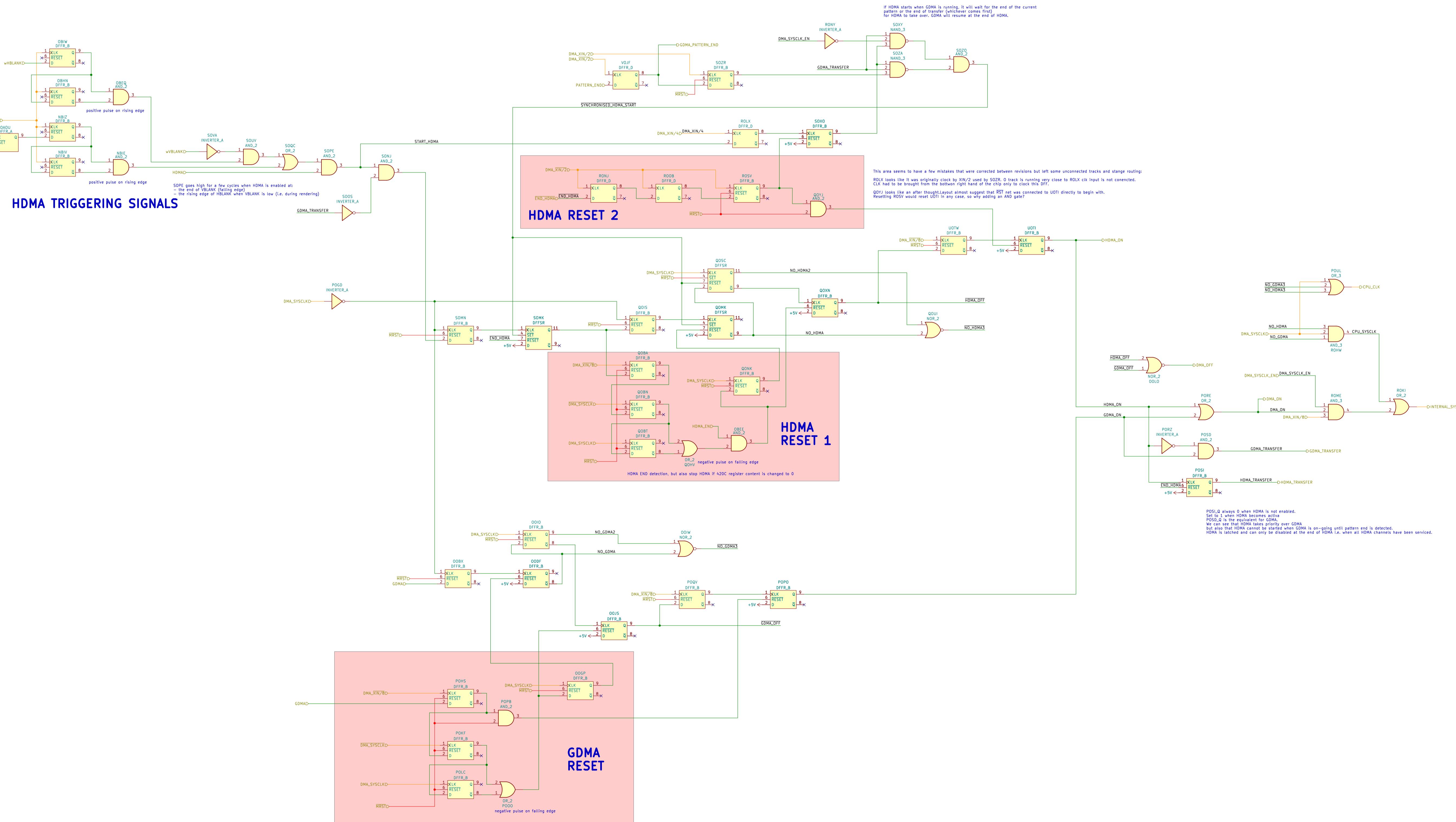
## DMA B-BUS

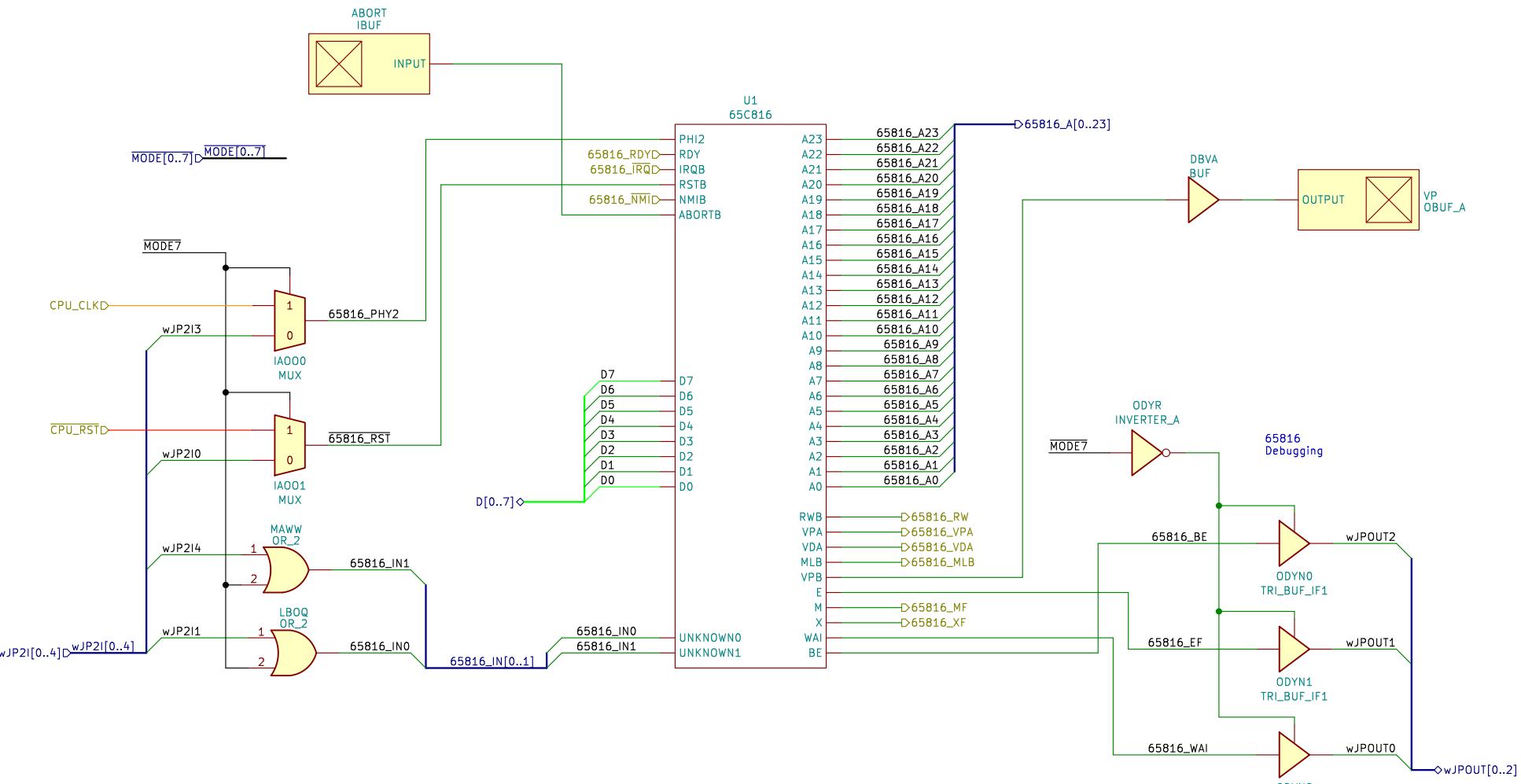


Modes are active low – inputs seems to be active low  
 MODE7 = Test mode with external CPU connections bypassing 65816  
 MODE4–6 = DMA test modes

MODE0 = SNES mode  
 MODE1 = SNES in halt mode where RDY is used to pause the 65816  
 MODE2–3 = NES modes where 4200–43FF registers are disabled and CPU connections are external  
 as original idea was probably to include the NES CPU on the board







There 3 outputs which I guess must be:  
 - Wait for interrupt output (multiplexed with RDY pin in W65C816 chip)  
 - Emulation flag  
 - Bus Enable (not needed as full 24-address bus is presented at all times)

Author: Regis Galland

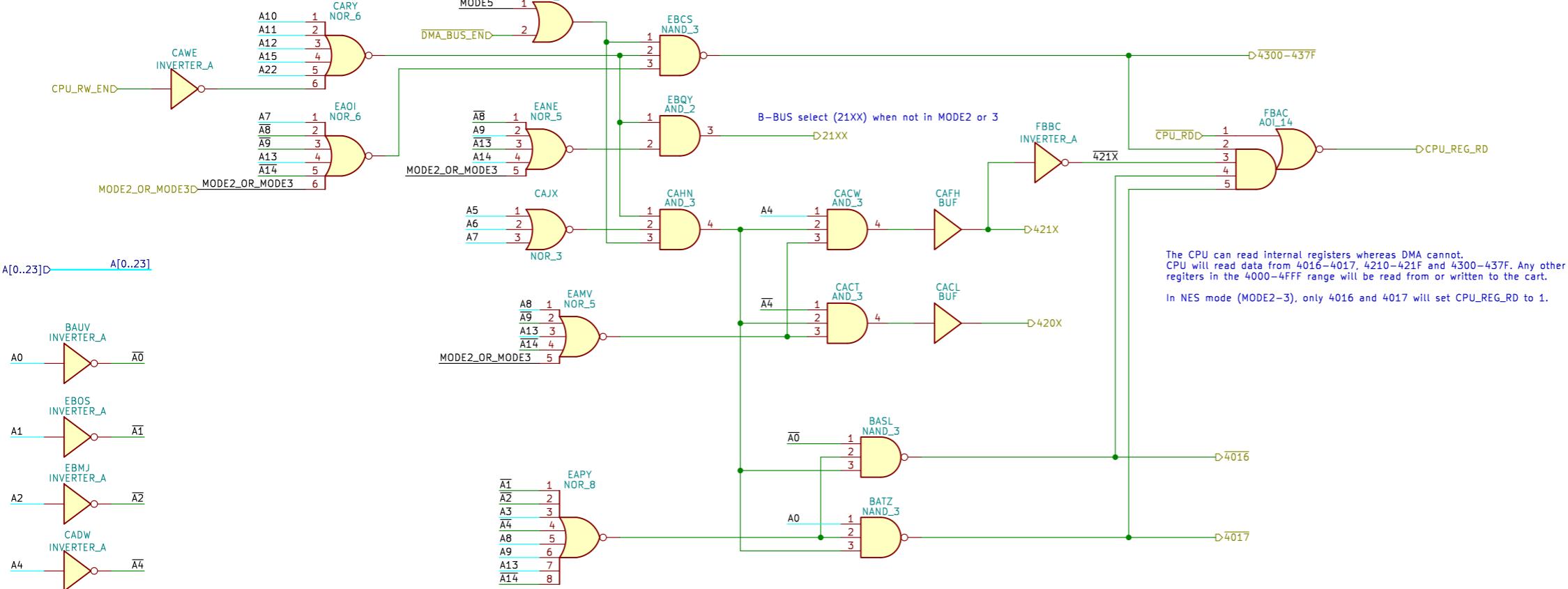
Sheet: /65C816 CPU/  
File: 65c816.kicad\_sch

Title: Reverse Engineered SNES S-CPU Schematics

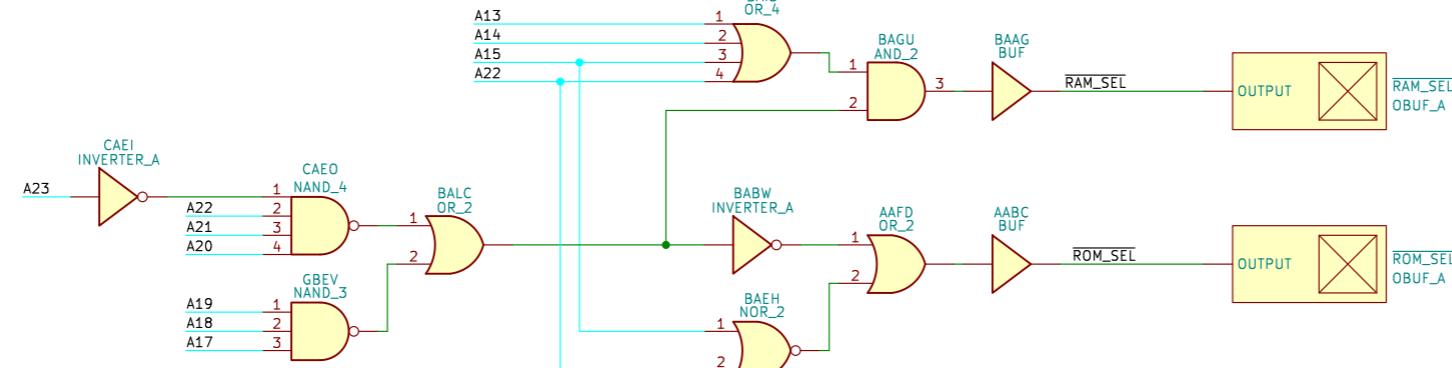
Size: A4 Date: 2024-12-24  
KiCad E.D.A. 8.0.7

Rev: 0.2  
Id: 17/20

## Address decoding



## RAM and ROM Selection



ROMSEL =  $\overline{A23} \cdot A22 \cdot A21 \cdot A20 \cdot A19 \cdot A18 \cdot A17 + A22 \cdot A15$   
Consistent with access to:  
– 00-3F/80-BF:8000-FFFF or  
– 40-7D/C0-FF:0000-FFFF

RAMSEL =  $\overline{(A23 \cdot A22 \cdot A21 \cdot A20 \cdot A19 \cdot A18 \cdot A17 + A22 \cdot A15 \cdot A14 \cdot A13)}$   
Consistent with access to:  
– 00-3F/80-BF:0000-1FFF or  
– 7E-7F:0000-FFFF

Author: Regis Galland  
Sheet: /ADDRESS REGION DECODING/  
File: addrdecod.kicad\_sch

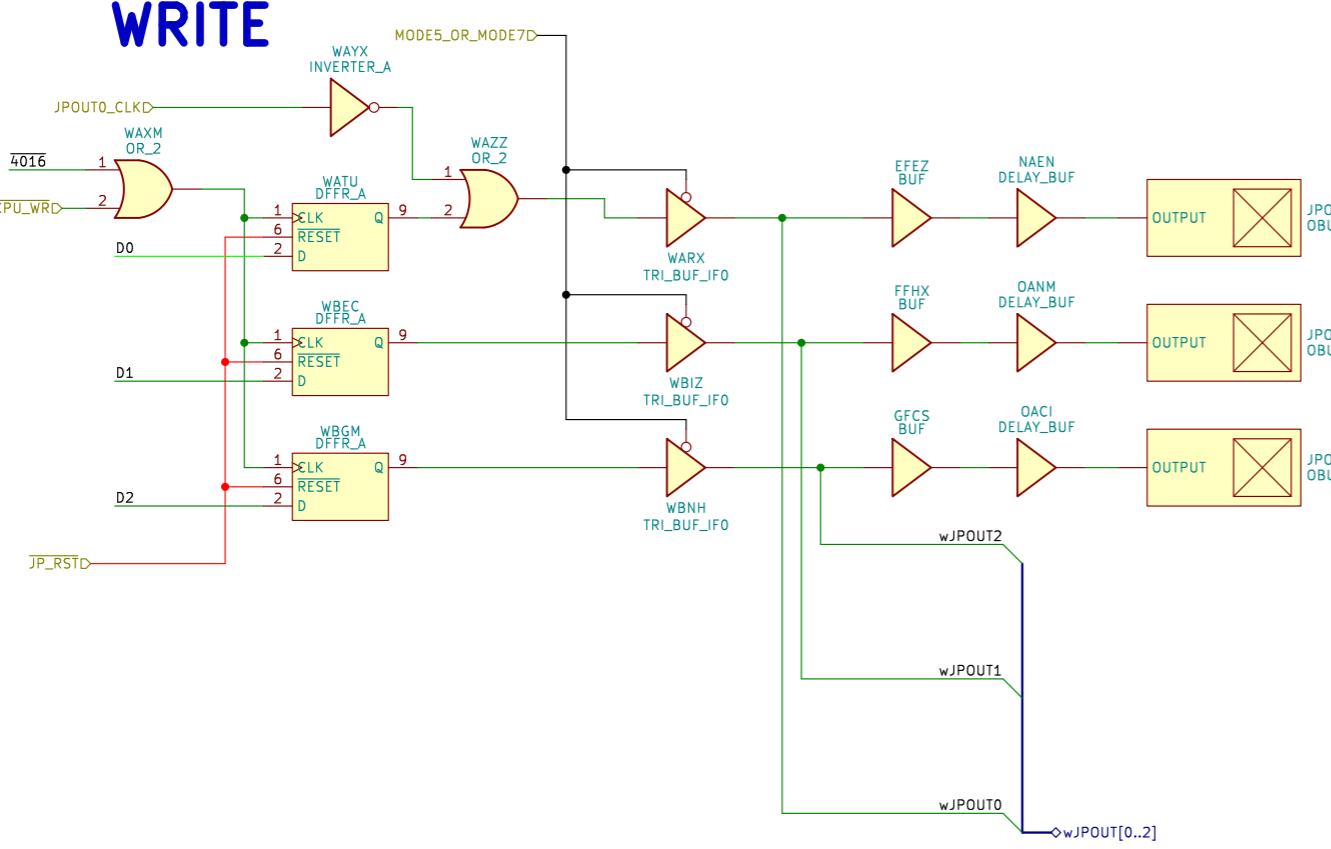
Title: Reverse Engineered SNES S-CPU Schematics

Size: A3 | Date: 2024-12-24  
KiCad E.D.A. 8.0.7

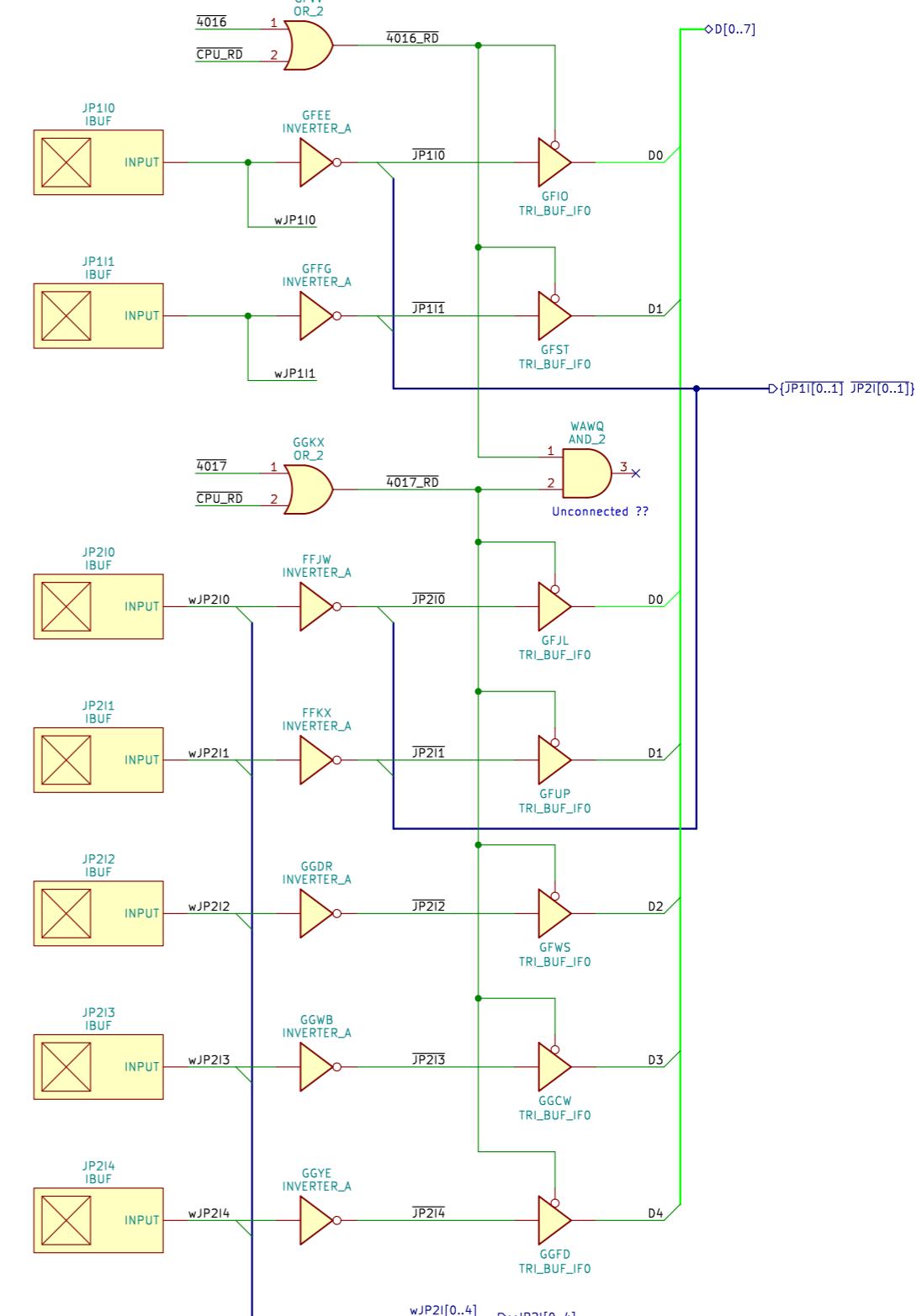
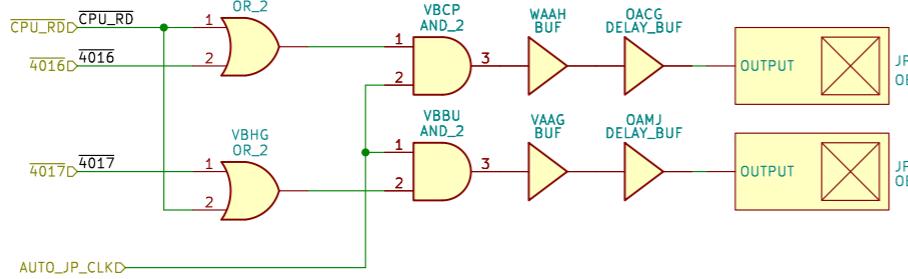
Rev: 0.2  
Id: 18/20

# 4016/4017 READ

## 4016 WRITE



## JPCLK: 4016/4017 READ OR SNES AUTO JP

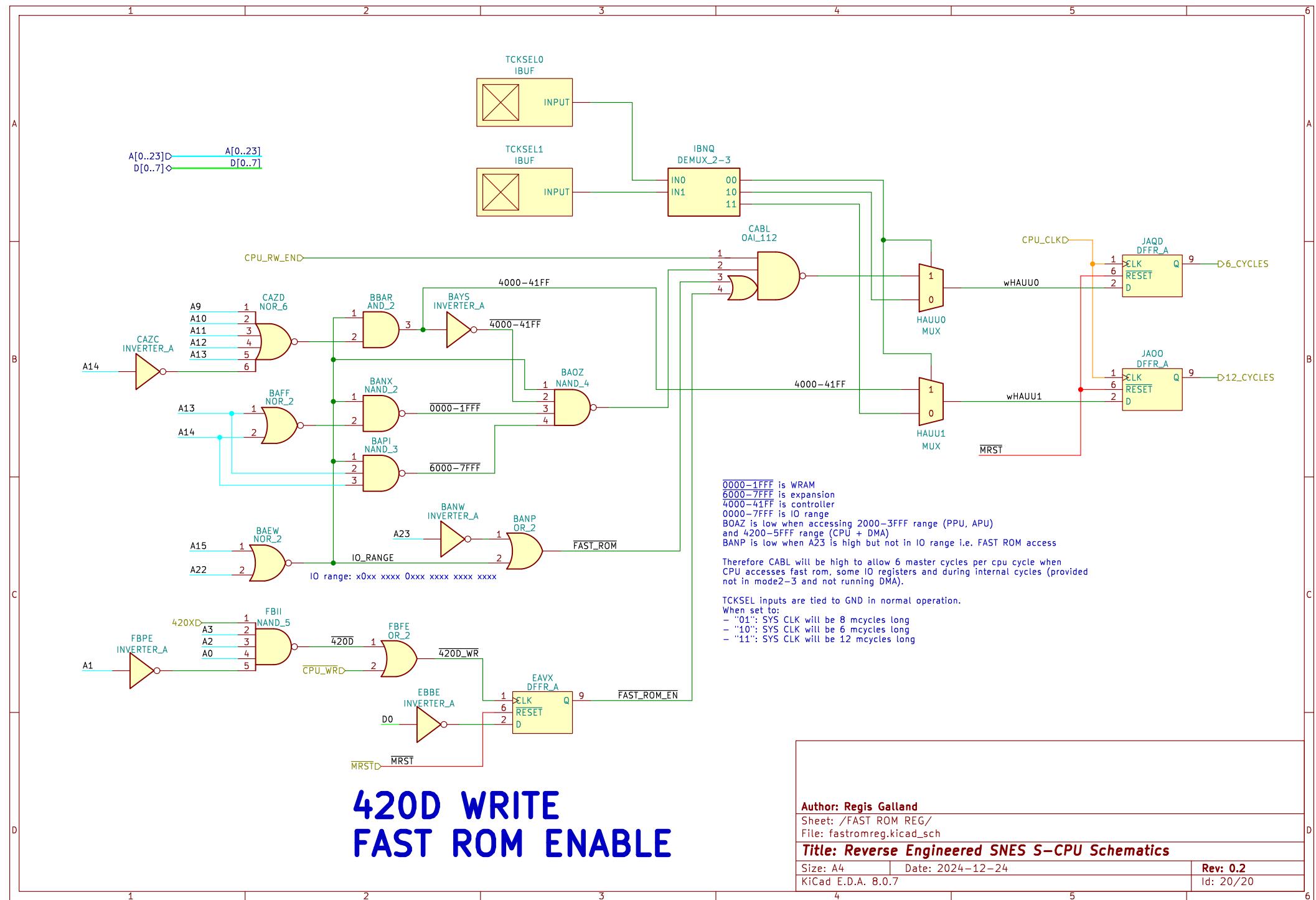


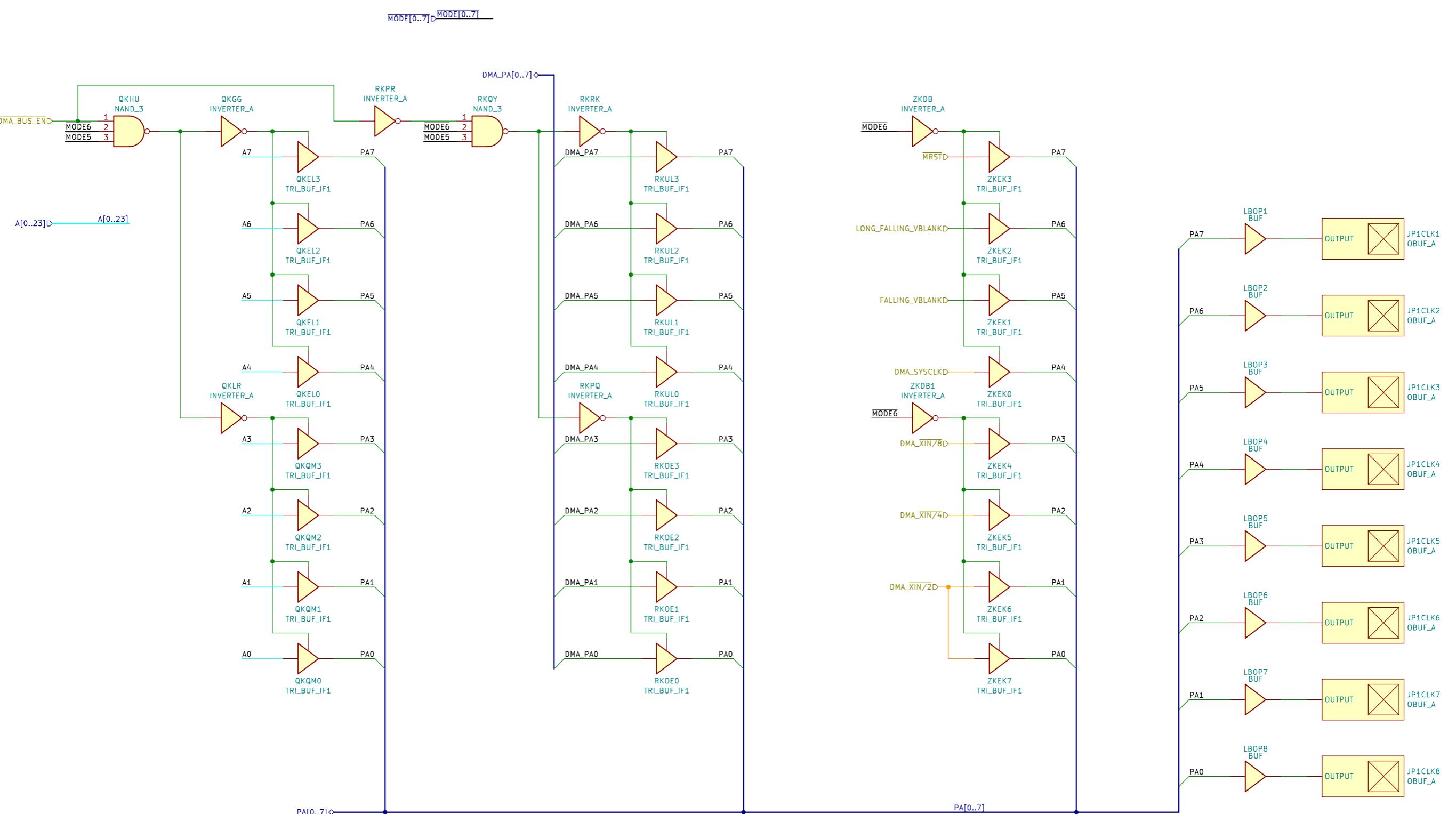
Author: Regis Galland  
Sheet: /LEGACY JP REGISTERS/  
File: legacyjpregs.kicad\_sch

Title: Reverse Engineered SNES S-CPU Schematics

Size: A3 | Date: 2024-12-24  
KiCad E.D.A. 8.0.7

Rev: 0.2  
Id: 19/20





## B-BUS and debug mode signals

Author: Regis Galland  
Sheet: /DMA AND B-BUS/B-BUS/  
File: bbus.kicad\_sch

Title: Reverse Engineered SNES S-CPU Schematics  
Size: A3 Date: 2024-12-24  
KiCad E.D.A. 8.0.7

Rev: 0.2  
Id: 21/20