

D785D: CMS CSC DDU, InCtrl Logic (file 0ddu_in) 2-2-2005_14:03

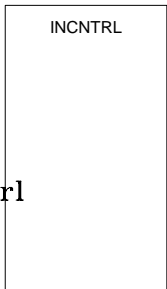
Process Rocket I/O Data from DMB, format output to DDU FIFOs DF012A01 Version 12

this DDU: v4: use RHL to hold LWEN
 RXER words are skipped, FILLER added as needed v5: All InUnits are tied to Rx1 (top) or Rx5 (bottom) for loaded rate test
 - Use code "C" with FILL flag set (b34 & b16) v6: Move JTAGreg18 to 19, Reg18 now MinMemAvail to track FIFO usage, add RPM for RHL
 - Use code "8" instead? **Mult 1-4** v7-8: ExtFIFO_PAF stops READ and prevents START of next event, tuned in v8
laptop; v9: Revert to standard Fiber 1-1 Rx inputs (removed internal 1-4 Rx fanout)
 v10: copy from laptop, mod AFUL logic for FMM WARN, try ISE6.2, add LOC for InUnit TBUFs,
 make InUnit MemError flag permanent, prepare InUnit for DMB feedback
 v11: use FPAF to stop data xmit & timeouts, use FPAE or MemCtrl_Free<=3 to set WARN,
 stop DMBs if MemCtrl_NoneFree and set BUSY
v12: Fixed CLR_FDONE on rd_ctrl.5

PART=XC2VP20-6-FG676

All I/O is 2.5V

AVOID=Y21, E23, C22, E21
 (INIT, BUSY, WRITE, CS)
 NC_XCV400_FG676=B13, AF13
 NC_XCV400E_FG676=D13, Y13



RST_1=Asynchronous Reset for FPGA1 and ALL FIFOs

BXR Pulse: Dump Data Mode (no L1A needed)

Mode 1 Switch Block (NOT labelled in reverse order)

- 1: Mode Bit 0
- 2: Mode Bit 1
- 3: Mode Bit 2
- 4: Mode Bit 3
- 7: Set Fake_L1A (data passthrough)
- 8: Set all IO bits HIGH, ~FPGA version on LEDs

PROM=2*XC18V04-VQ44 (PARALLEL)

D785D\DDUproj\in4ctrl

To Do List:

- Drive more LA ports, LEDs and TPs
- Disable REN/OWEN when Ext FIFO goes FULL
- Put something useful on HDR/TR[35,34,17,16]

VME Broadcast Addresses:

- 24=OSU-TCB "Test Control Board"
- 25=DMB
- 26=TMB
- 27=Both DMB and TMB
- 28=DDU
- 29=DCC

Replace EmptyIN/FIFO_EMPTY PUs?

ELECTRONICS LAB
 PHYSICS DEPARTMENT
 THE OHIO STATE UNIVERSITY
 174 WEST 18TH AVE
 COLUMBUS OHIO 43210

- To Do:
- COMPARE LINUM & BXN (DMB/TMB too)
add BX_offset constants to SRAM?
 - use DDU-DMB fiber to STOP DMB on FULL, check
 - Monitor DMB: Active-DAV mismatch warn,
MOVLP bad evt, BXN match err
 - Watch for TRG buff overflows
 - On rd_ctrl.5: add 3 flags to FMM path?

H1: 0x5T/NN.NNNN/XXX/1.11/VK DDU WordCount (64-bit words) for "No Data" event: 0x006.
 H2: 0x/8000/0001/8000/8000 DDU WordCount for one DMB (only one CFEB): 0D2h = 210 dec, 1680 Bytes
 H3/T-1: 0x/SSSS.SSSS/ZZZZ/000Y DDU WC, 1 DMB with 2 CFEB (8 samples each): 19Ah = 410 dec, 3280 Bytes
 T-2: 0x/8000/FFF/8000/8000 DDU WC, 2 DMB with 1 CFEB (nCFEB=2): 19Eh = 414 dec, 3312 Bytes
 TR: 0x/A/?/WWW/RRRR/UITK DDU WC, 2 DMB with 2 CFEB (nCFEB=4): 32Eh = 814 dec, 6512 Bytes
DDU_WordCount = (6 + 25*Nts*nCFEB + 4*nDMB) < 30070; 240560 Bytes

DDU WC, 3 DMB with 1 CFEB (nCFEB=3): 26Ah = 618 dec, 4944 Bytes
 DDU WC, 4 DMB with 1 CFEB (nCFEB=4): 336h = 822 dec, 6576 Bytes
 DDU WC, 7 DMB with 1 CFEB (nCFEB=7): 59Ah = 1434 dec, 11472 Bytes
 DDU WC, 8 DMB with 1 CFEB (nCFEB=8): 666h = 1638 dec, 13104 Bytes

DDU WC, 11 DMB with 1 CFEB (nCFEB=11): 8CAh = 2250 dec, 18000 Bytes
 DDU WC, 12 DMB with 1 CFEB (nCFEB=12): 996h = 2454 dec, 19632 Bytes
 DDU WC, 15 DMB with 1 CFEB (nCFEB=15): BFAh = 3066 dec, 24528 Bytes

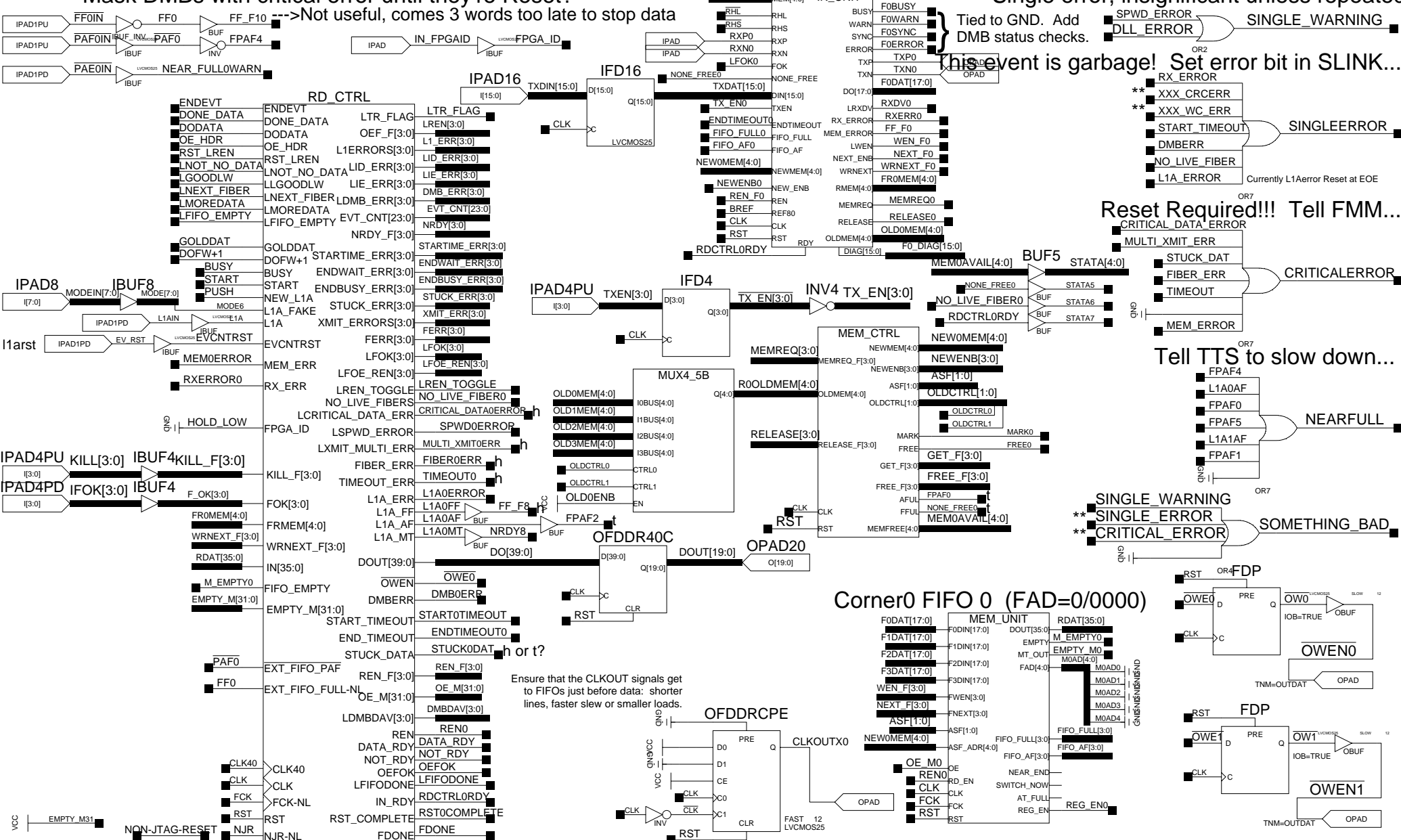
Mask DMBs with critical error until they're Reset?

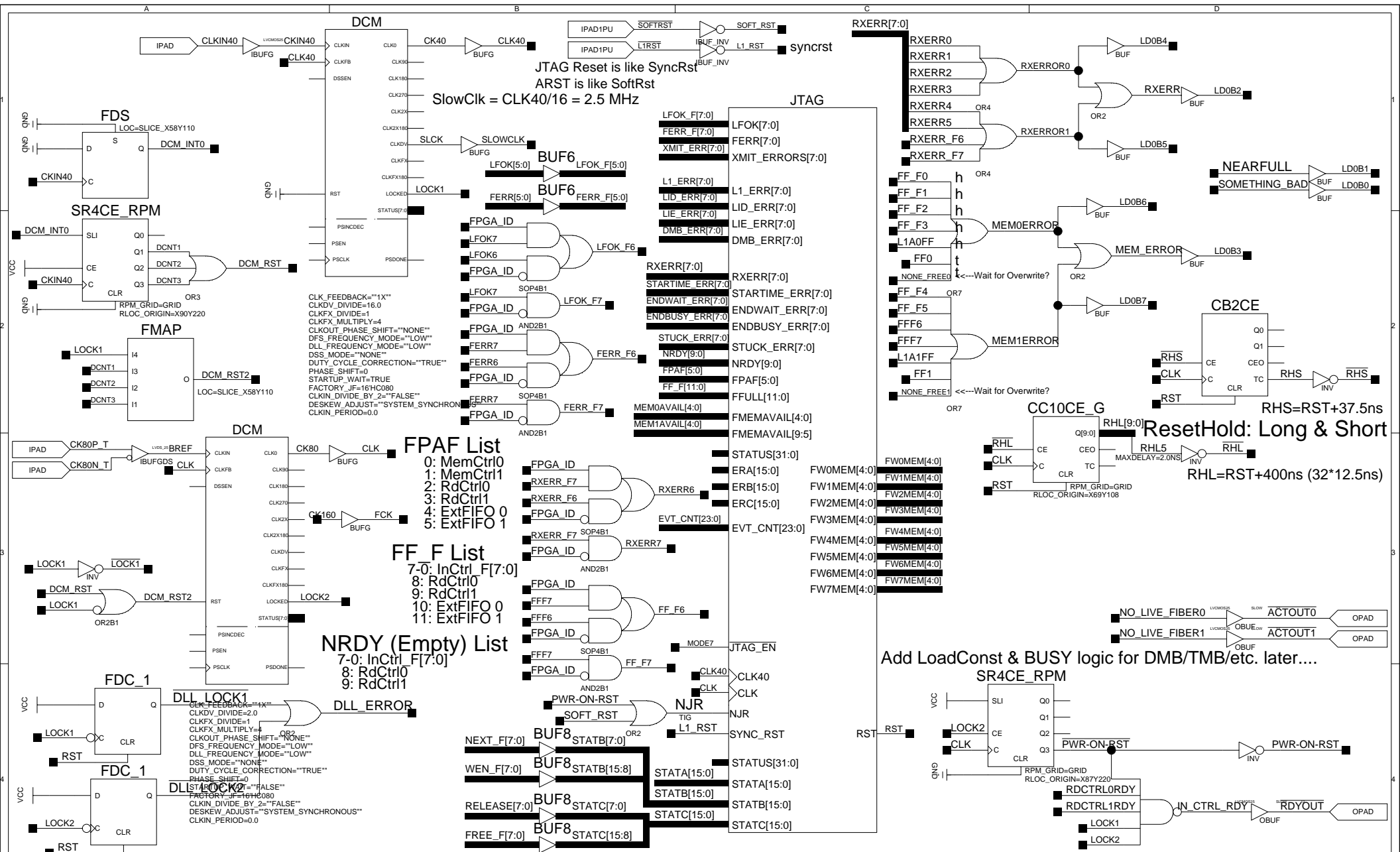
--->Not useful, comes 3 words too late to stop data

Single error, insignificant unless repeated

Tied to GND. Add DMB status checks.

This event is garbage! Set error bit in SLINK...





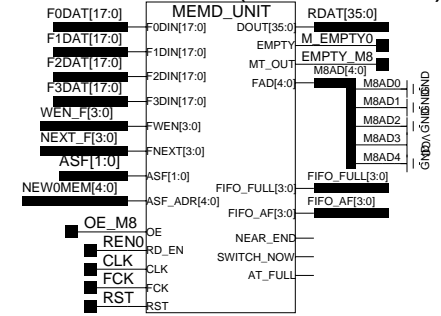
JTAG Reset is like SyncRst
 ARST is like SoftRst
 SlowClk = CLK40/16 = 2.5 MHz

- FPAF List**
- 0: MemCtrl0
 - 1: MemCtrl1
 - 2: RdCtrl0
 - 3: RdCtrl1
 - 4: ExtFIFO 0
 - 5: ExtFIFO 1

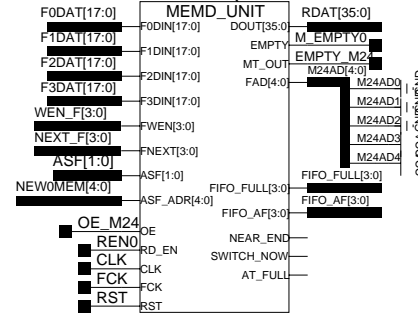
- FF F List**
- 7-0: InCtrl_F[7:0]
 - 8: RdCtrl0
 - 9: RdCtrl1
 - 10: ExtFIFO 0
 - 11: ExtFIFO 1

- NRDY (Empty) List**
- 7-0: InCtrl_F[7:0]
 - 8: RdCtrl0
 - 9: RdCtrl1

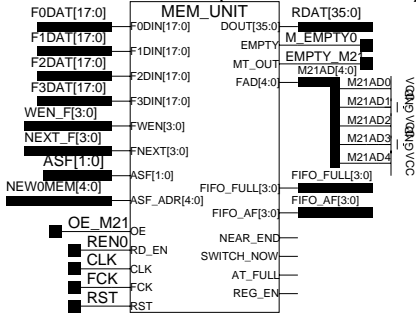
Corner0 FIFO 8* (FAD=0/1000)



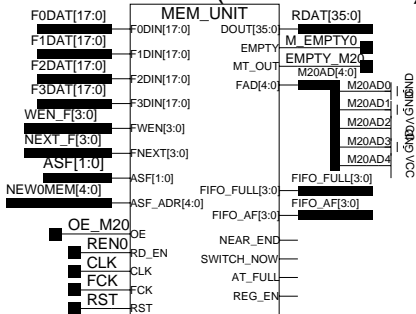
Corner1 FIFO 8* (FAD=1/1000)



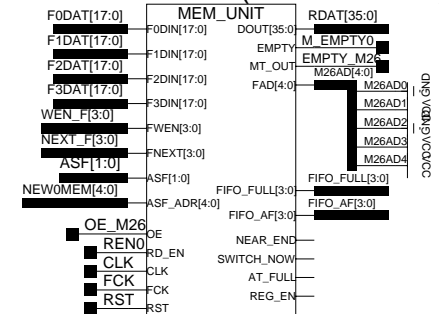
Corner1 FIFO 5 (FAD=1/0101)



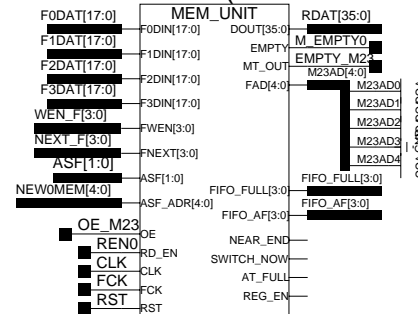
Corner1 FIFO 4 (FAD=1/0100)



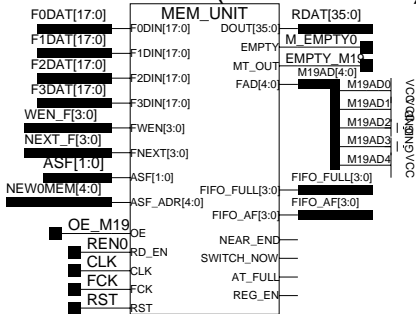
Corner1 FIFO 10 (FAD=1/1011)



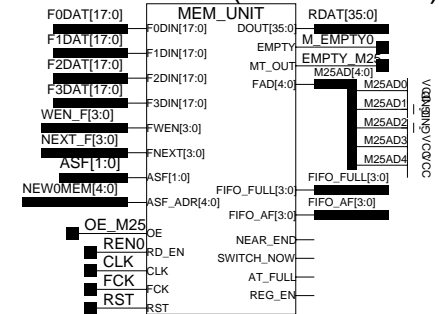
Corner1 FIFO 7 (FAD=1/0111)



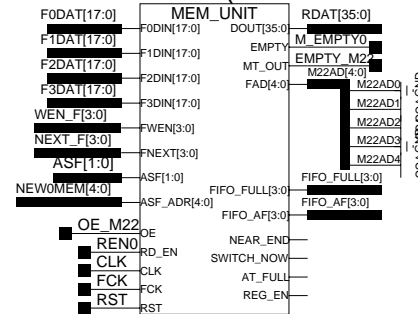
Corner1 FIFO 3 (FAD=1/0011)



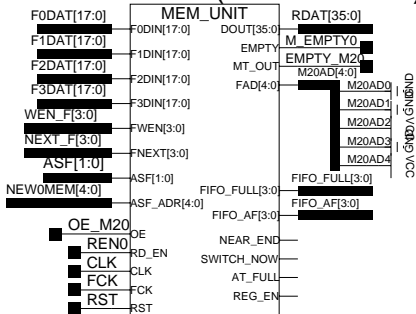
Corner1 FIFO 9 (FAD=1/1001)



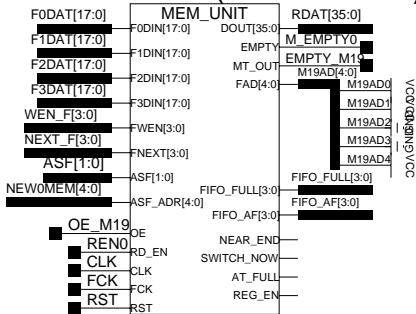
Corner1 FIFO 6 (FAD=1/0110)



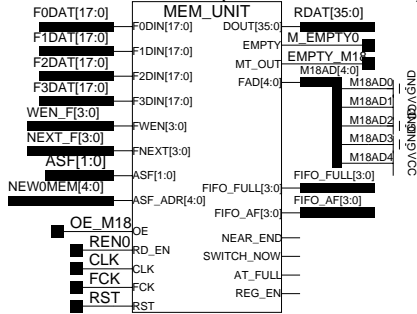
Corner1 FIFO 1 (FAD=1/0001)



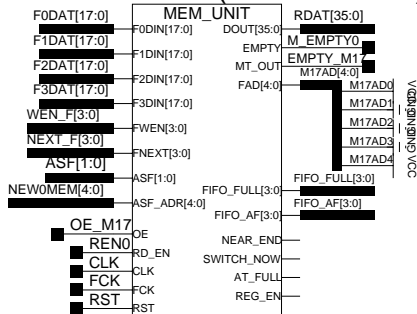
Corner1 FIFO 0 (FAD=1/0000)



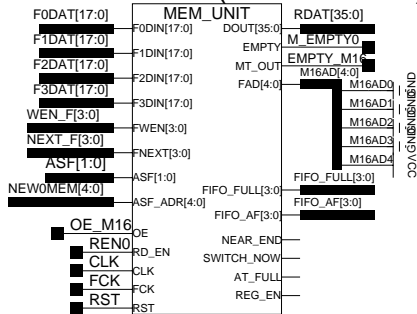
Corner1 FIFO 2 (FAD=1/0010)

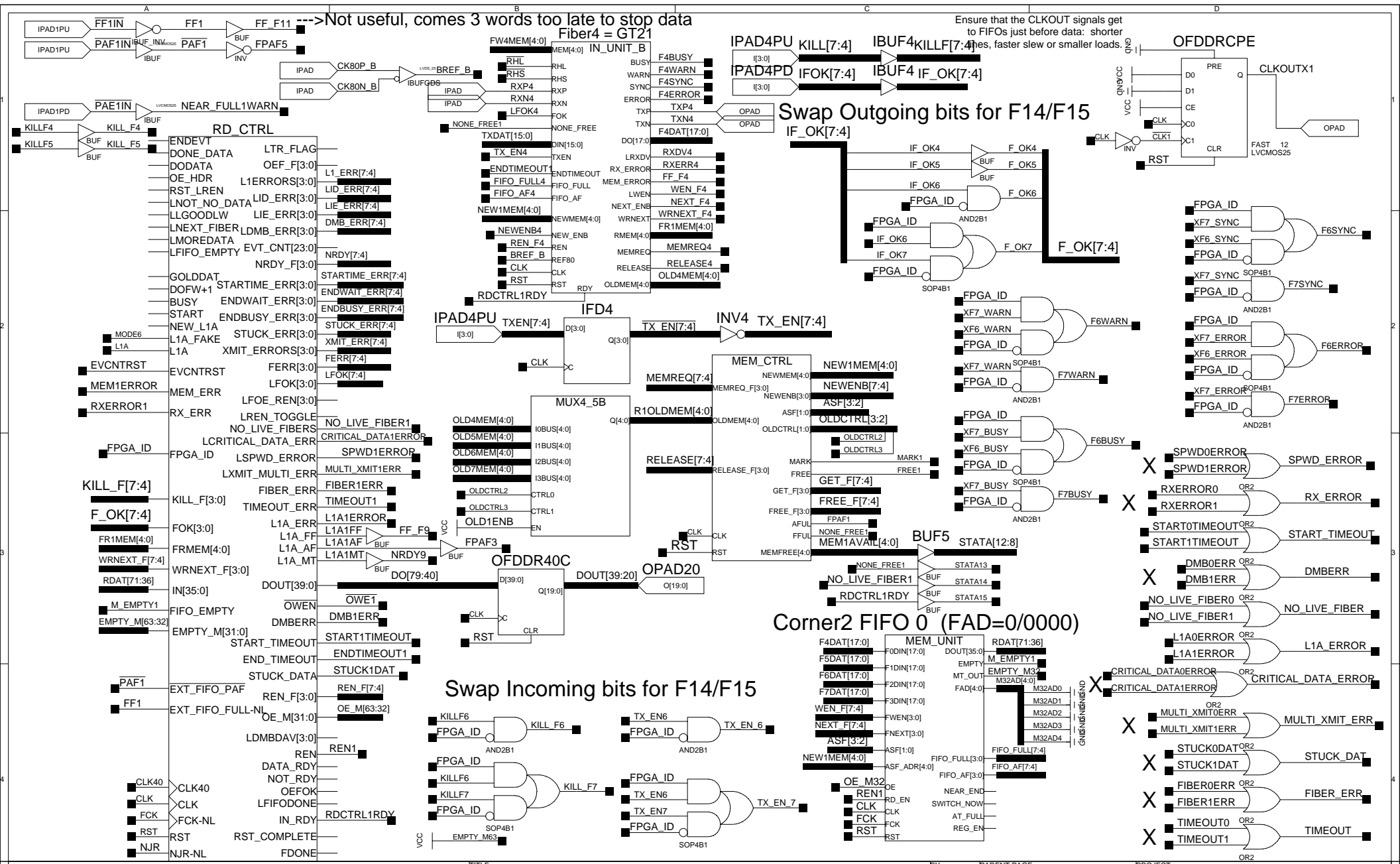


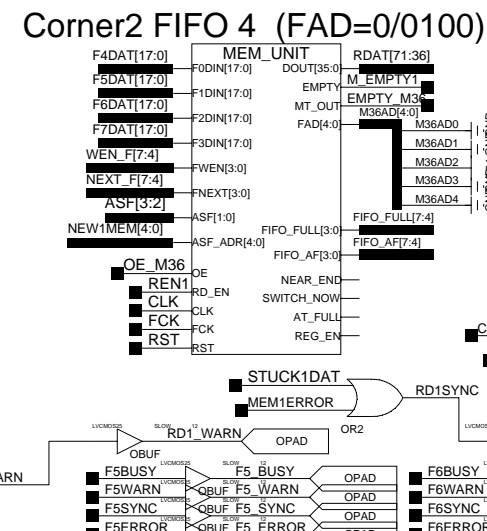
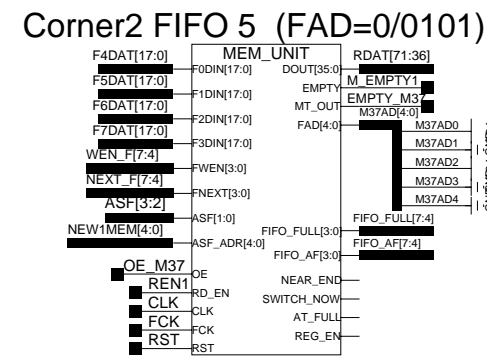
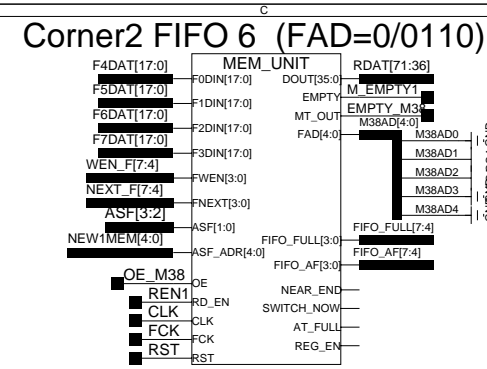
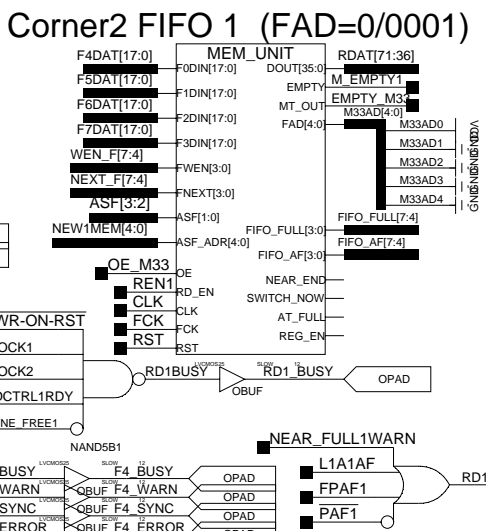
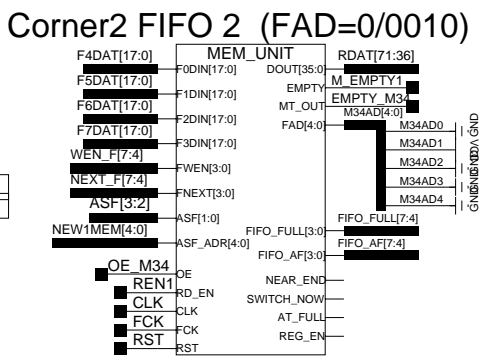
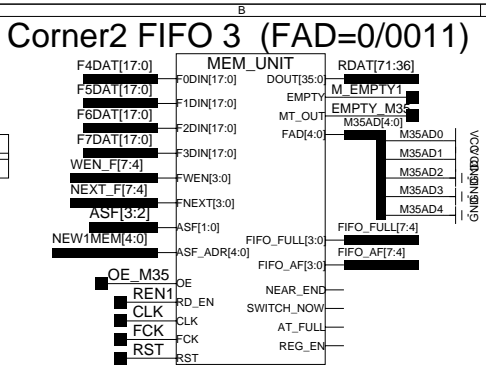
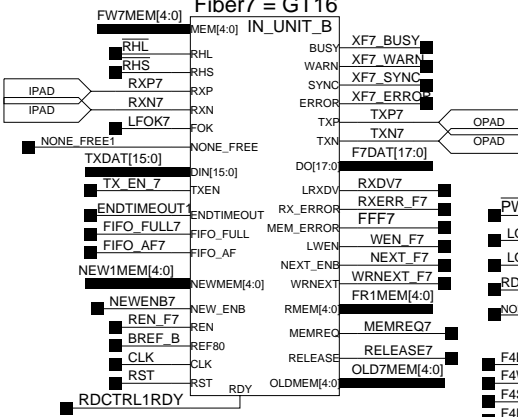
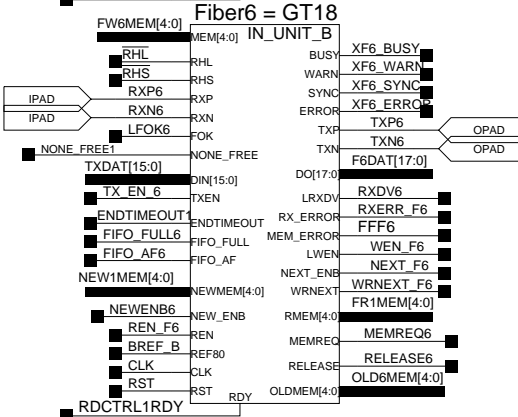
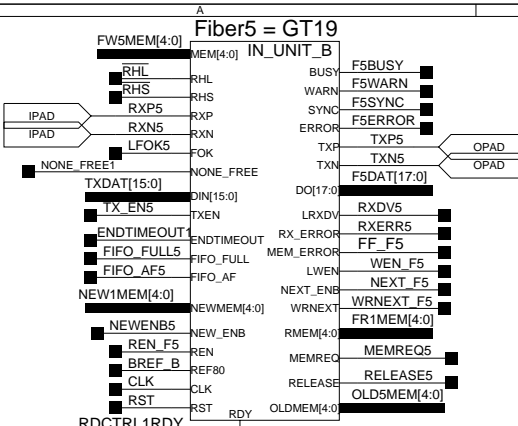
Corner1 FIFO 5 (FAD=1/0101)



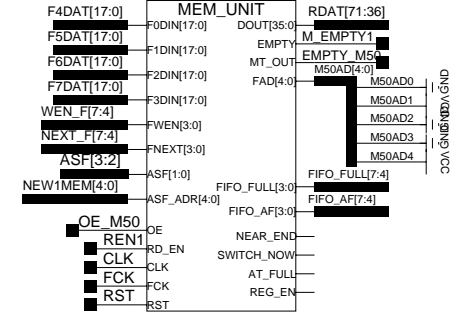
Corner1 FIFO 8* (FAD=1/1000)



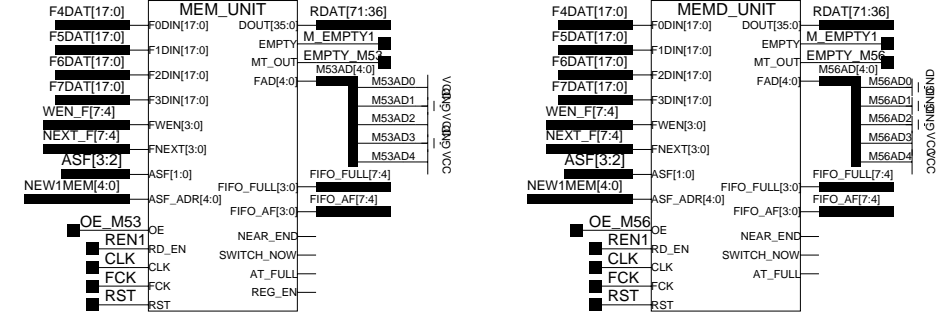




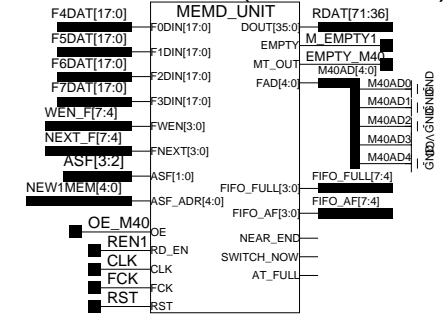
Corner3 FIFO 2 (FAD=1/0010)



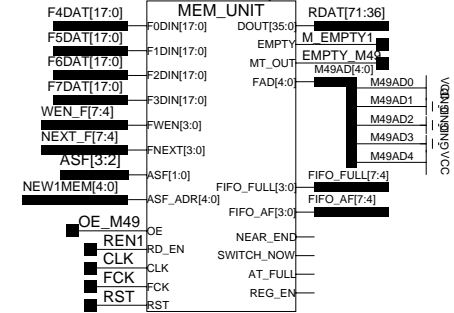
Corner3 FIFO 5 (FAD=1/0101) Corner3 FIFO 8* (FAD=1/1000)



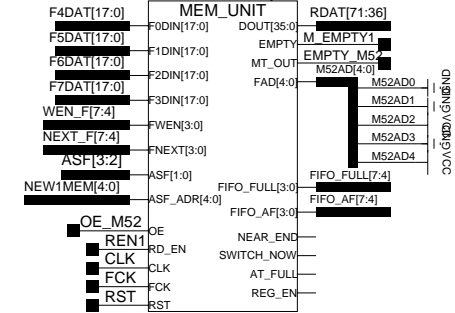
Corner2 FIFO 8* (FAD=1/1000)



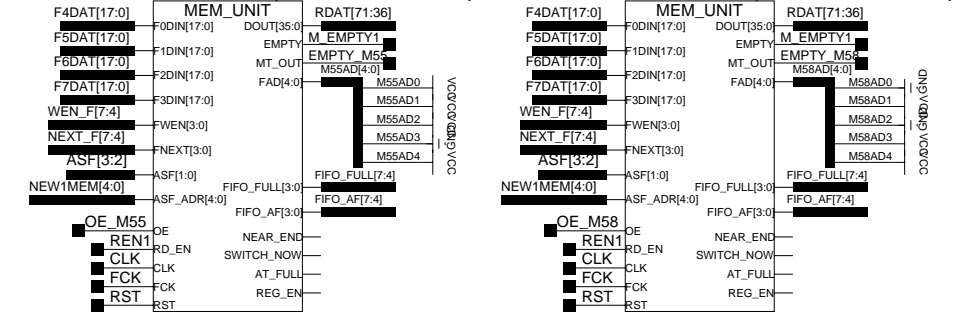
Corner3 FIFO 1 (FAD=1/0001)



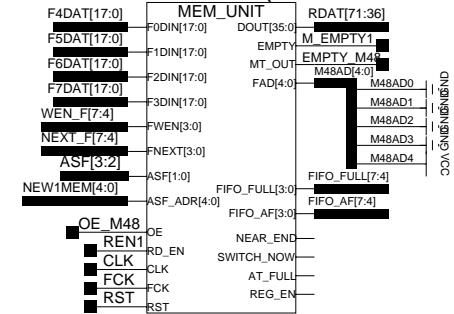
Corner3 FIFO 4 (FAD=1/0100)



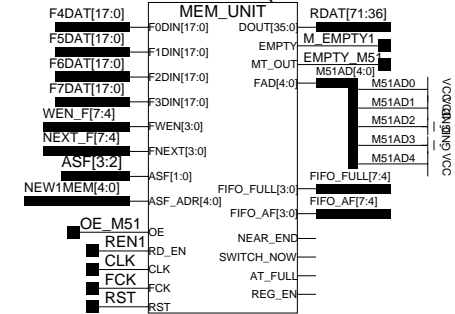
Corner3 FIFO 7 (FAD=1/0111) Corner3 FIFO 10 (FAD=1/1010)



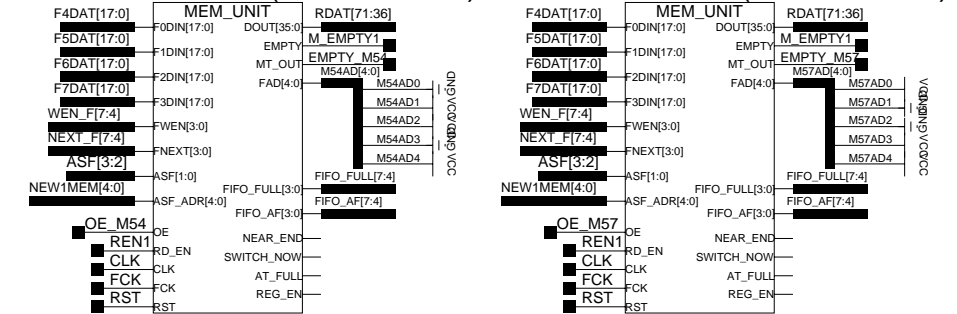
Corner3 FIFO 0 (FAD=1/0000)

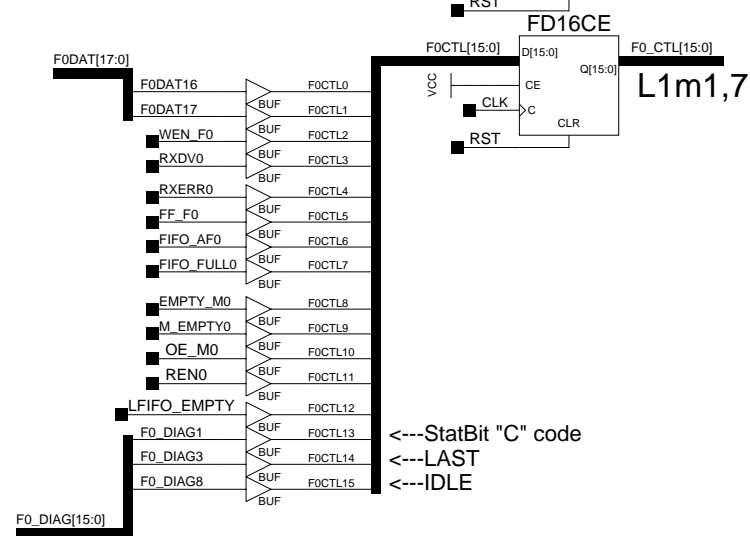
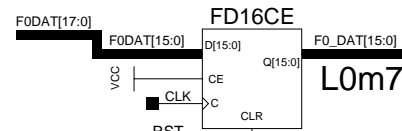
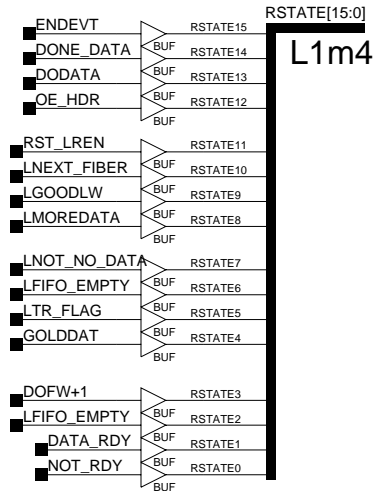
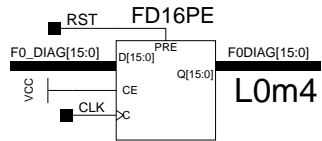
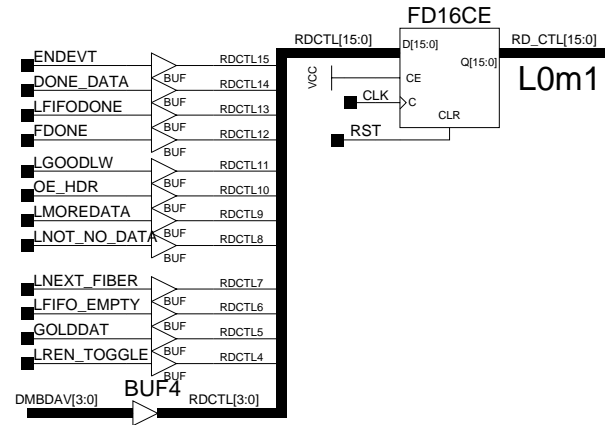
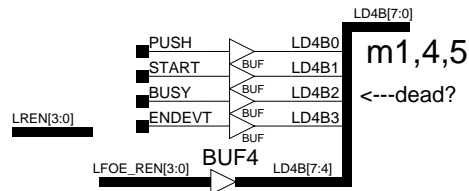
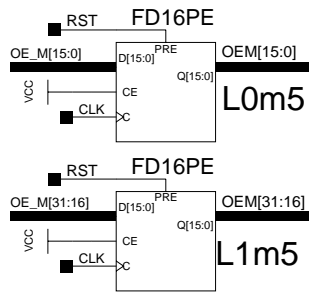


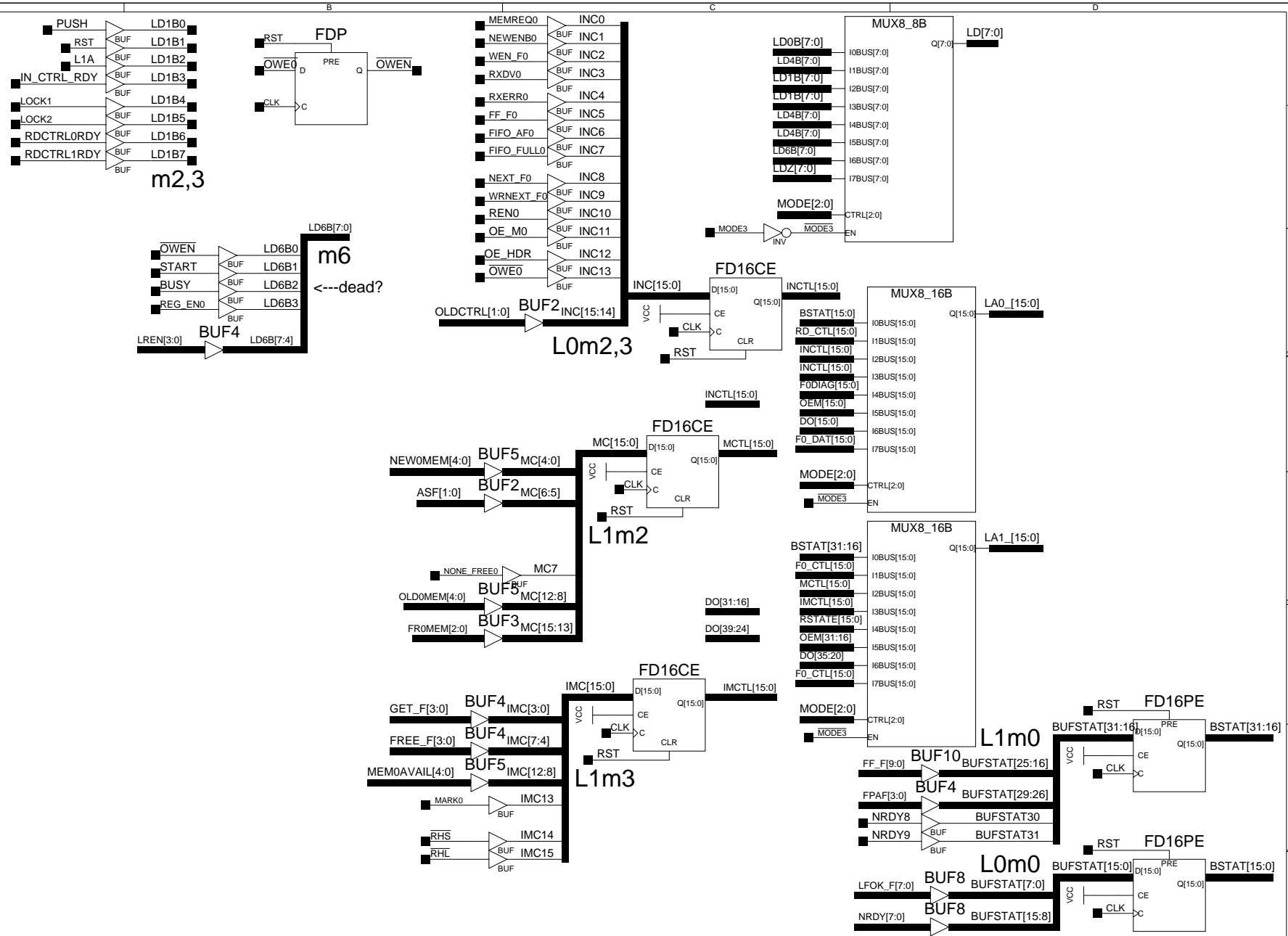
Corner3 FIFO 3 (FAD=1/0011)



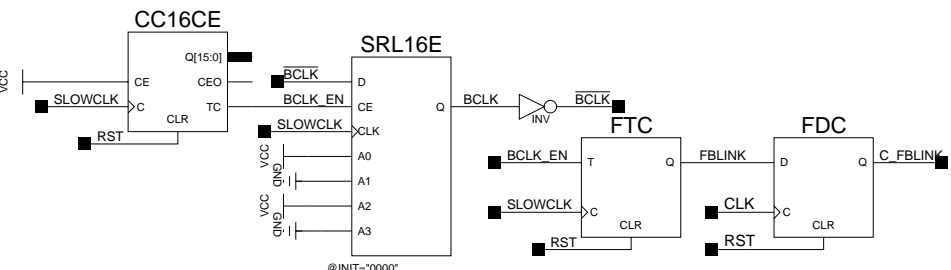
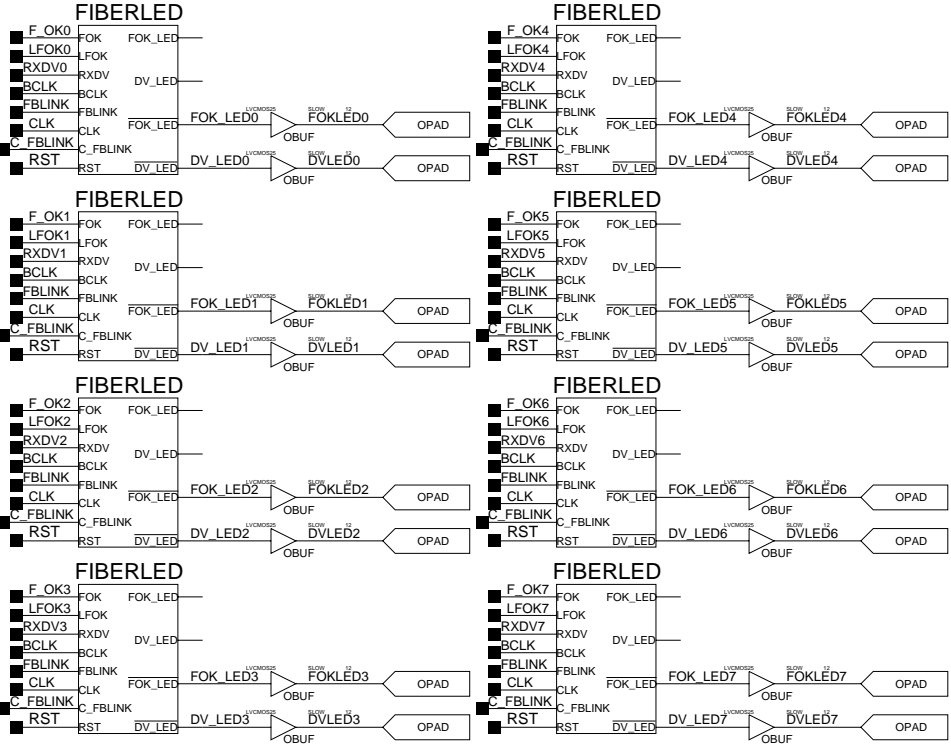
Corner3 FIFO 6 (FAD=1/0110) Corner3 FIFO 9 (FAD=1/1001)



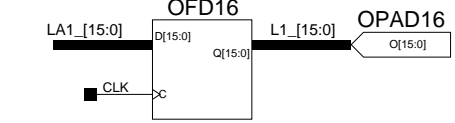
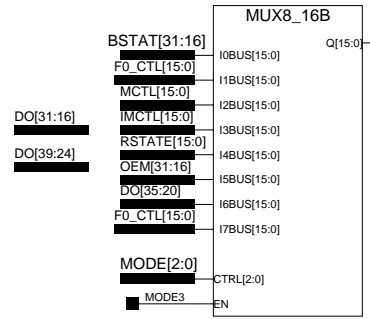
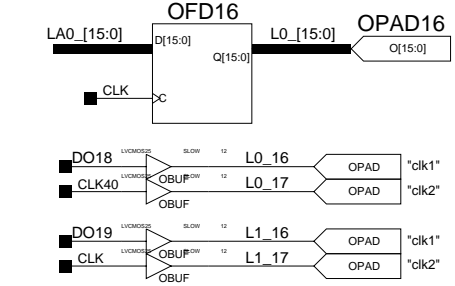
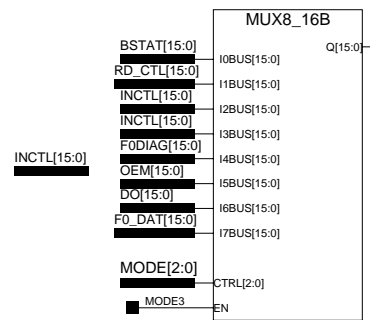
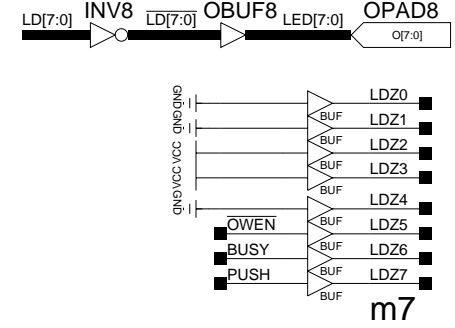
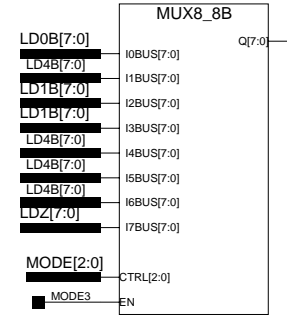




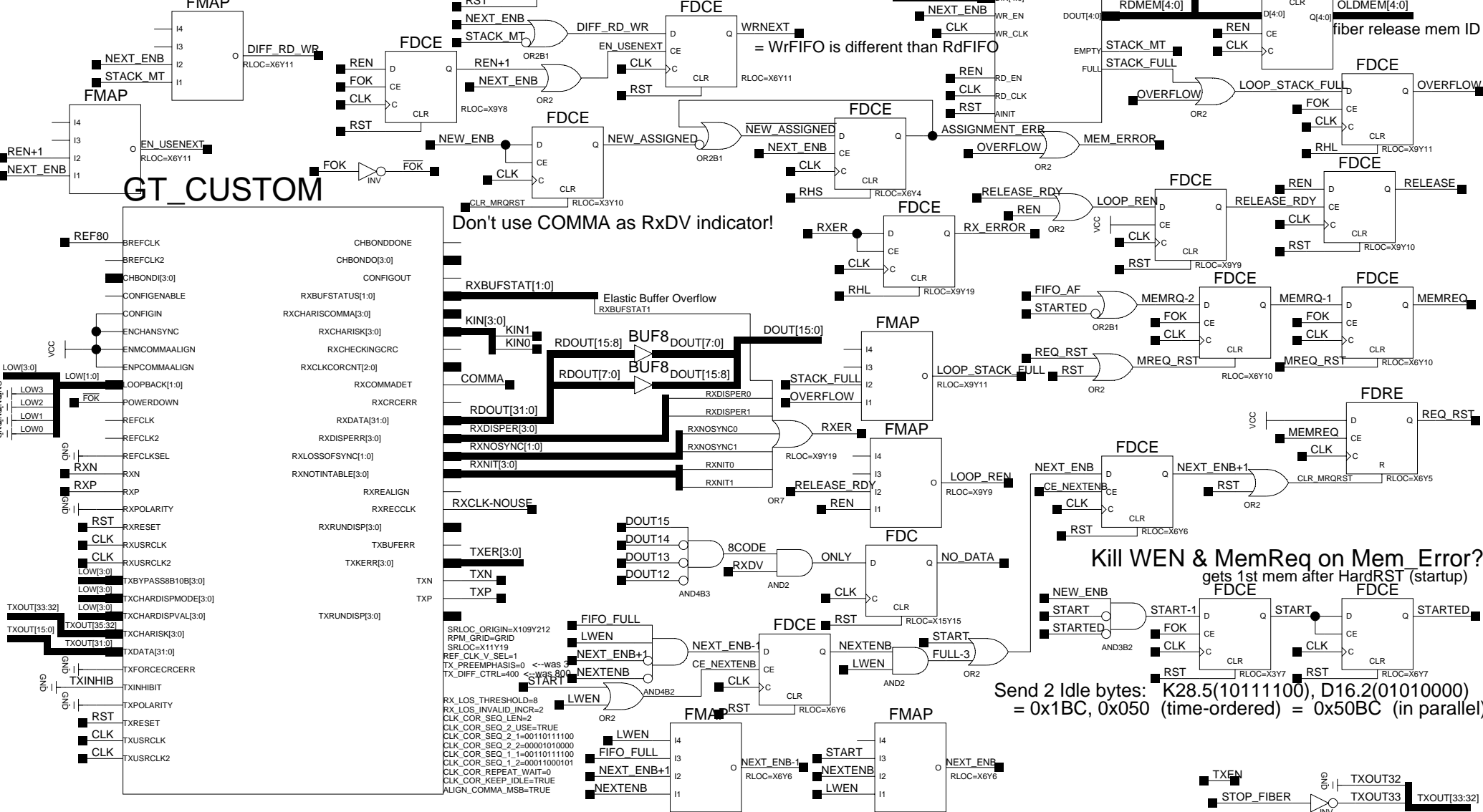
END



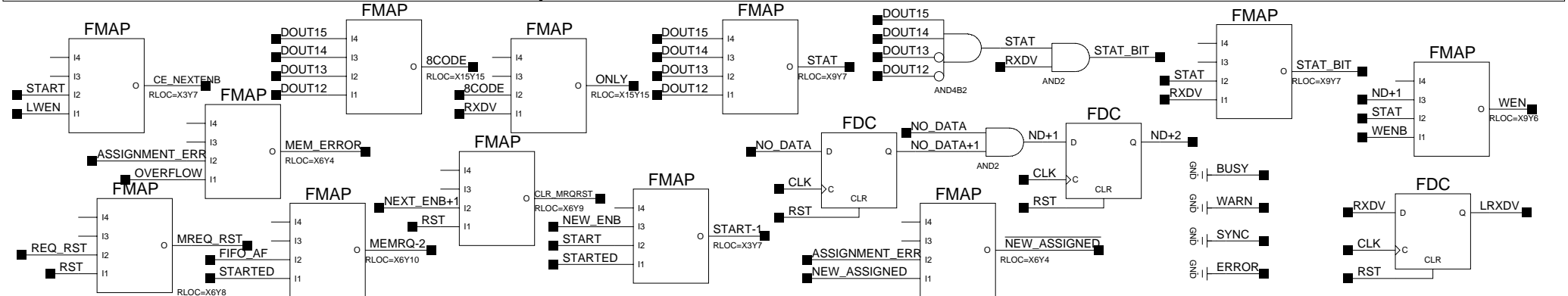
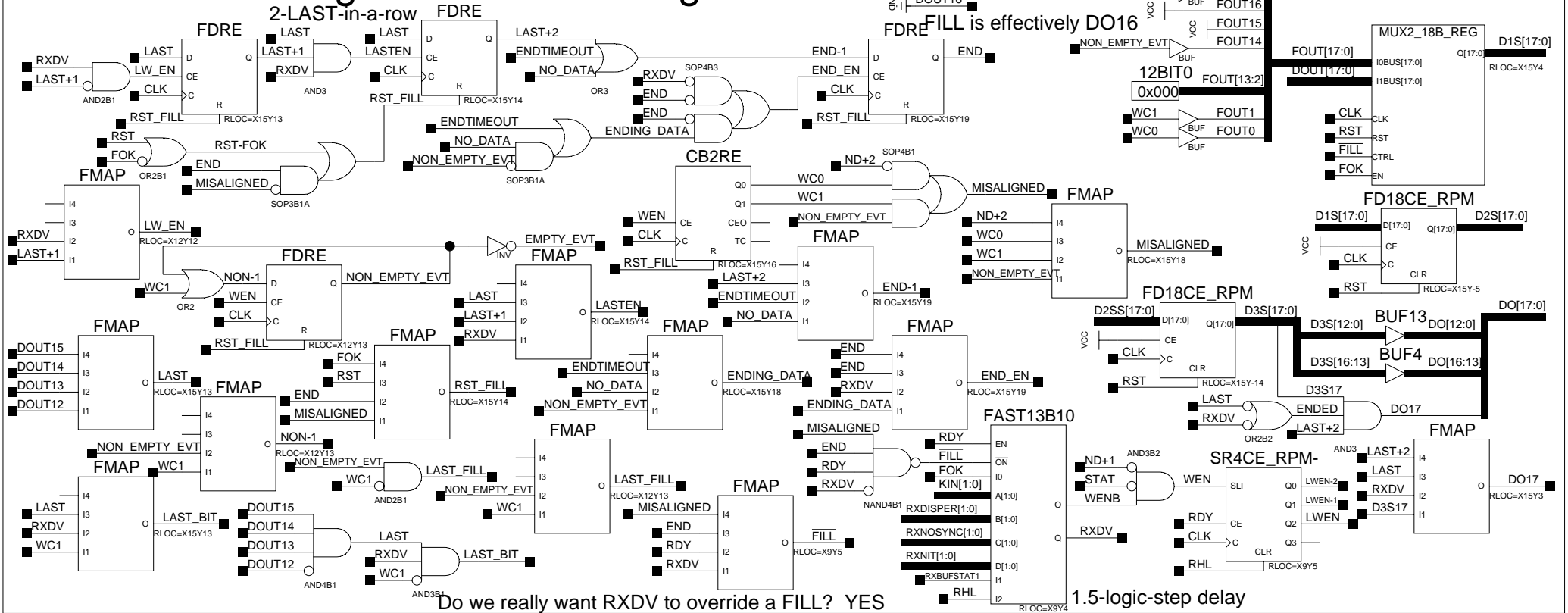
SlowCLK=2.5MHz, BCLK_EN=38 Hz. Shift 13 will give ~1.5Hz BCLK, Shift 5 gives ~4Hz.

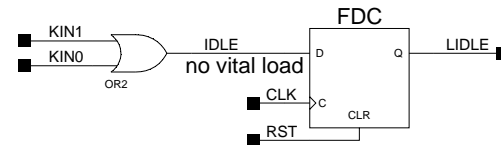
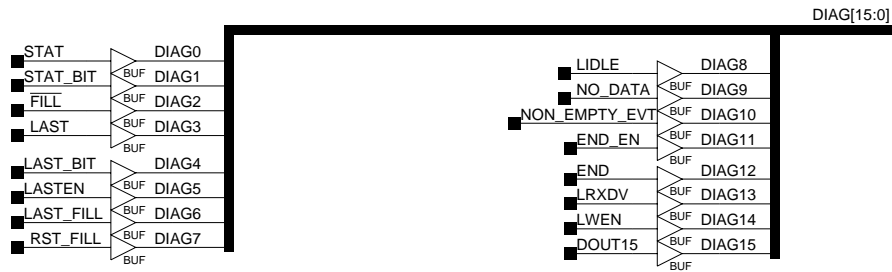


FIFO_AF & FIFO_FULL are driven by Tbufs from 22 mem's.
 NEW_ENB & NEW_MEM are driven by MemCtrl.
 TX_EN & DIN are driven by DDU_Ctrl FPGA.
 REN is driven by RD_CTRL.
 IDLEOUT needs local control logic.



FILL Logic for 2-4 word alignment





LAST Flag is insensitive to single-bit errors

LAST Flag Logic tables, checks for E-Codes in DMB Data (L means LAST Flag will be set for that word)

BRAM FIFO Output

Assume 4 E-Codes at DMB End are sequential with no gaps

Normal event sync:	X	X
	X	X
	<u>E_L</u>	<u>E</u>
	E	E

Lost 1 word OR 3 extra:	X	X
	E	X
	<u>E_L</u>	<u>E_L</u>
	Fill	E

Lost 2 words OR 2 extra:	X	X
	E	E
	<u>E_L</u>	<u>E_L</u>
	Fill	Fill

Lost 3 words OR 1 extra:	E	X
	E	E
	<u>Fill_L</u>	<u>E_L</u>
	Fill	Fill

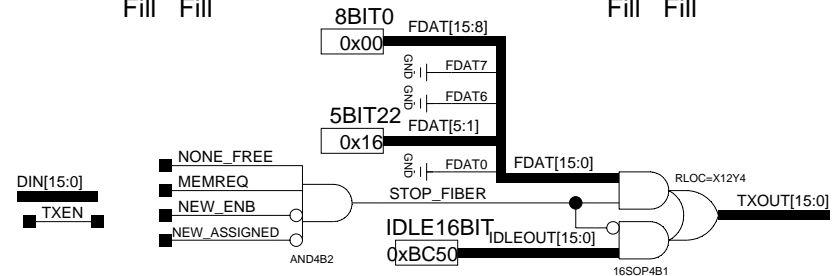
Normal, but lose 1st E-Word: (or lose last E-Word)	X	X
	X	X
	<u>E_L</u>	<u>E_L</u>
	Fill	E

Normal, but lose 2nd E-Word: (or lose 3rd E-Word)	X	X
	X	X
	<u>E_L</u>	<u>E</u>
	Fill	E

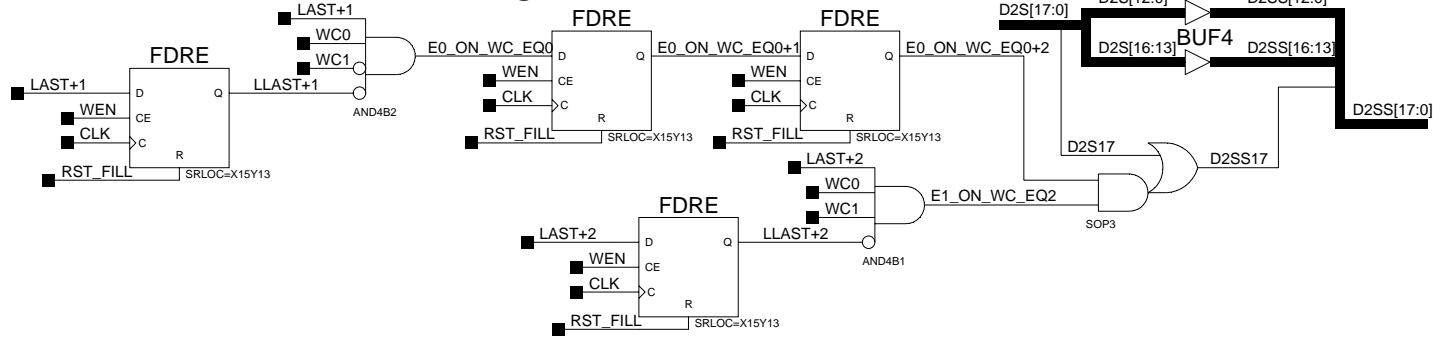
Normal, but BAD 1st E-Code:	X	X
	X	X
	<u>E_L</u>	X
	E	E

Normal, but BAD 2nd E-Code:	X	X
	X	X
	<u>X_L</u>	<u>E</u>
	E	E

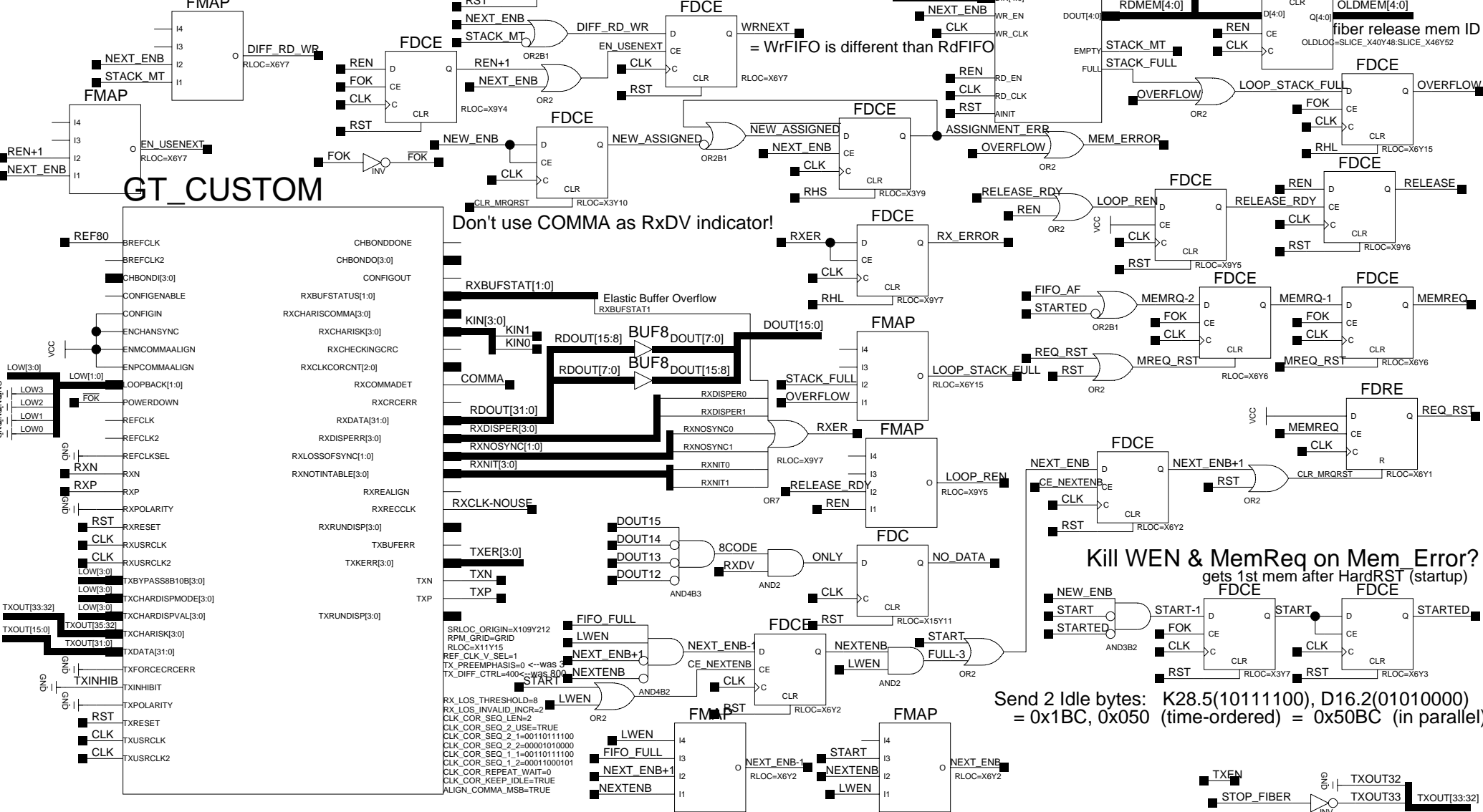
BAD 3rd or 4th E-Code is no problem



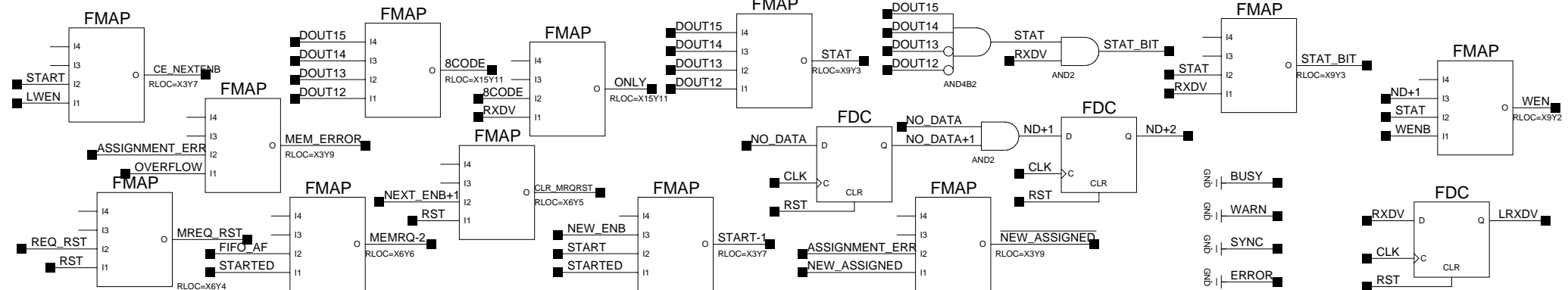
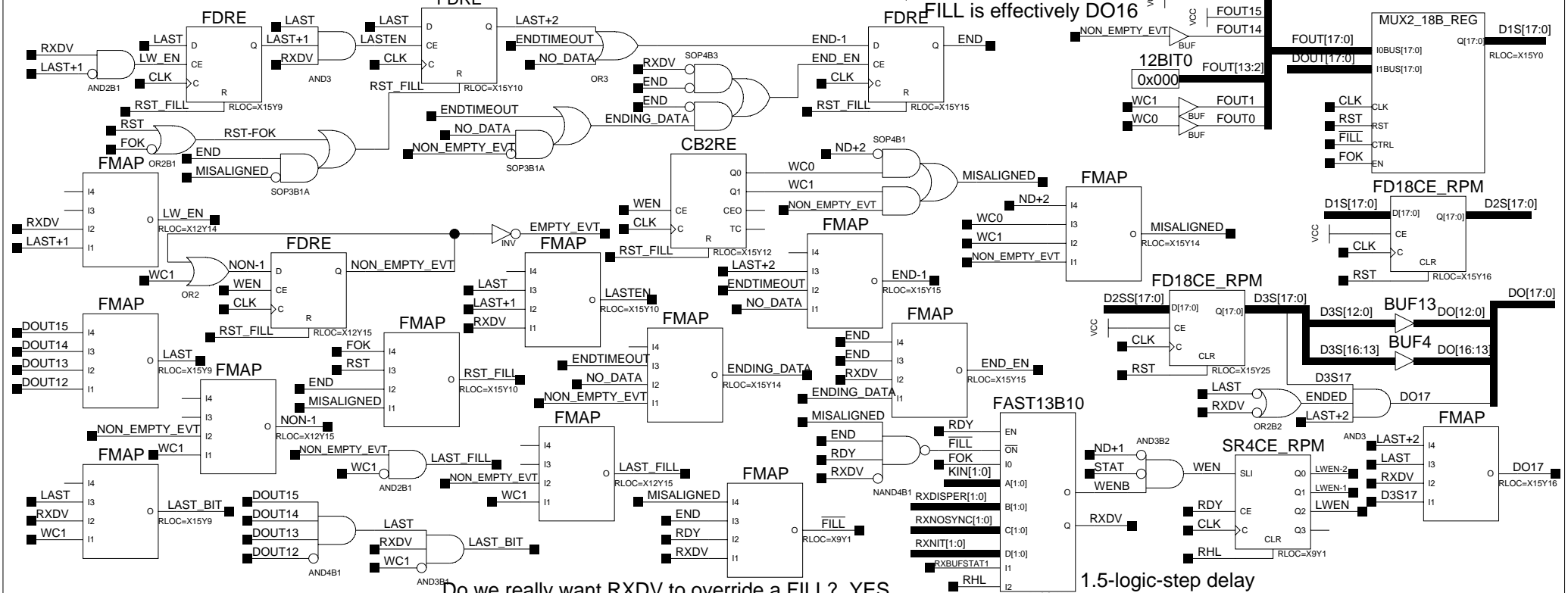
Fix to set LAST Flag for Bad 2nd E-Code case

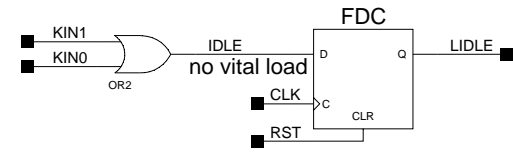


FIFO_AF & FIFO_FULL are driven by Tbufs from 22 mem's.
 NEW_ENB & NEW_MEM are driven by MemCtrl.
 TX_EN & DIN are driven by DDU_Ctrl FPGA.
 REN is driven by RD_CTRL.
 IDLEOUT needs local control logic.



FILL Logic for 2-4 word alignment





LAST Flag is insensitive to single-bit errors

LAST Flag Logic tables, checks for E-Codes in DMB Data (L means LAST Flag will be set for that word)

BRAM FIFO Output

Assume 4 E-Codes at DMB End are sequential with no gaps

Normal event sync:	X	X
	X	X
	<u>E_L</u>	<u>E</u>
	E	E

Lost 1 word OR 3 extra:	X	X
	E	X
	<u>E_L</u>	<u>E_L</u>
	Fill	E

Lost 2 words OR 2 extra:	X	X
	E	E
	<u>E_L</u>	<u>E_L</u>
	Fill	Fill

Lost 3 words OR 1 extra:	E	X
	E	E
	<u>Fill_L</u>	<u>E_L</u>
	Fill	Fill

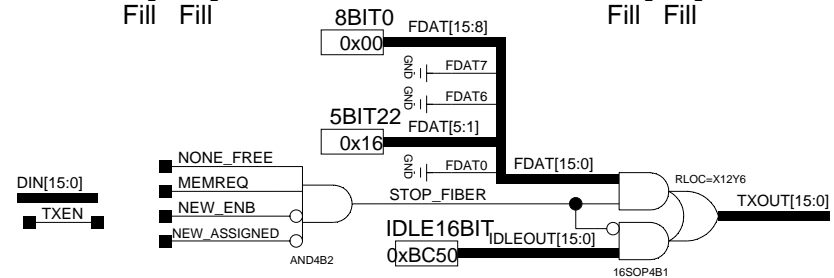
Normal, but lose 1st E-Word:	X	X
(or lose last E-Word)	X	X
	<u>E_L</u>	<u>E_L</u>
	Fill	E

Normal, but lose 2nd E-Word:	X	X
(or lose 3rd E-Word)	X	X
	<u>E_L</u>	<u>E</u>
	Fill	E

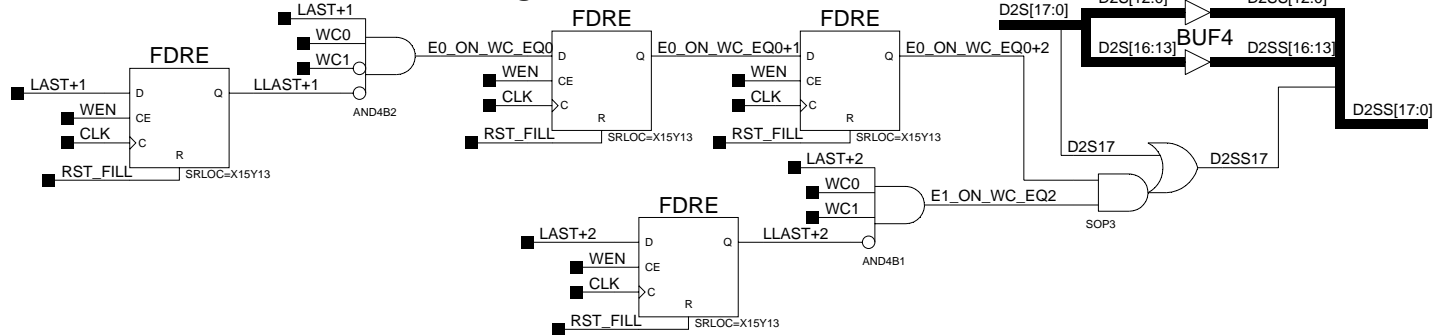
Normal, but BAD 1st E-Code:	X	X
	X	X
	<u>E_L</u>	X
	E	E

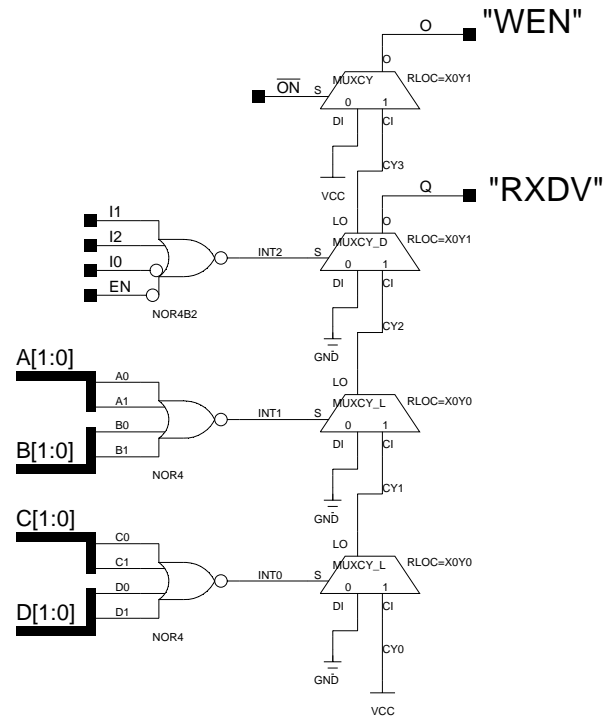
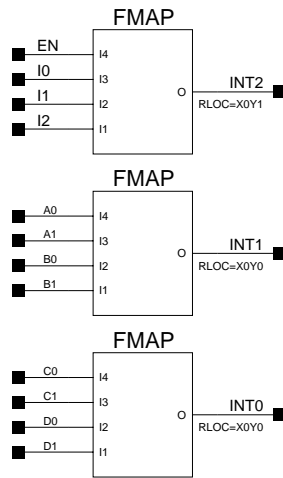
Normal, but BAD 2nd E-Code:	X	X
	X	X
	<u>X_L</u>	<u>E</u>
	E	E

BAD 3rd or 4th E-Code is no problem



Fix to set LAST Flag for Bad 2nd E-Code case



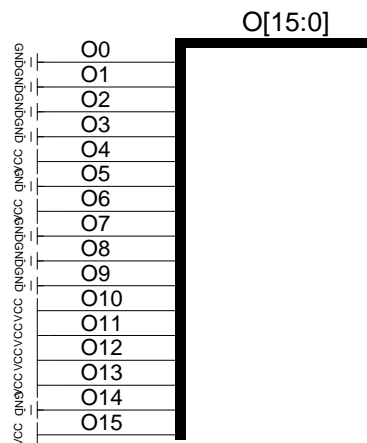


JRG

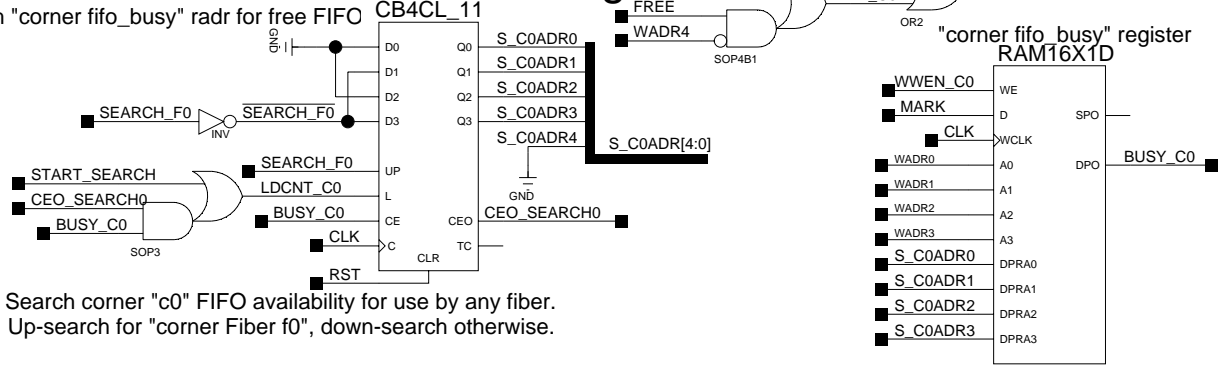
Title:	FAST13B10	
Comments:	Custom Logic for DDU similar to: AND12B10 with an OR2 (allows ON to override)	
Date:	19th December 2003	Ver: 1
Sheet Size:	B	Rev: A

Send 2 Idle bytes:

$K28.5(10111100)+D16.2(01010000)$
= 0x1BC + 0x050 (time-ordered)
= 0xBC50 (in parallel)



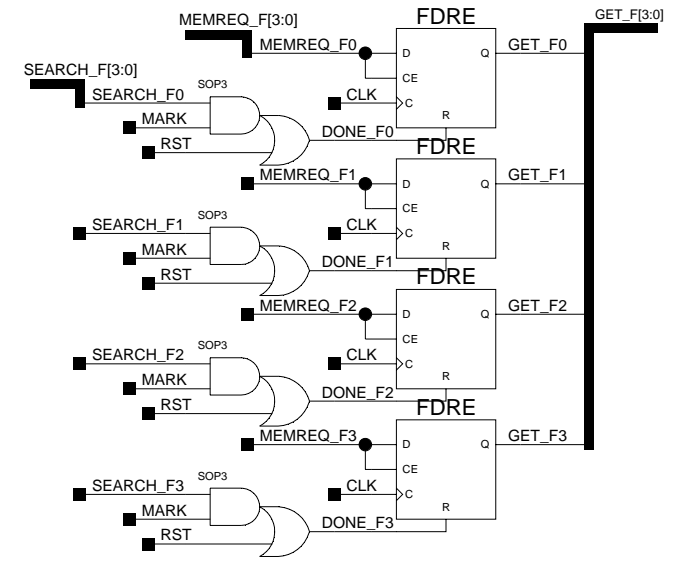
Corner 0 FIFO usage



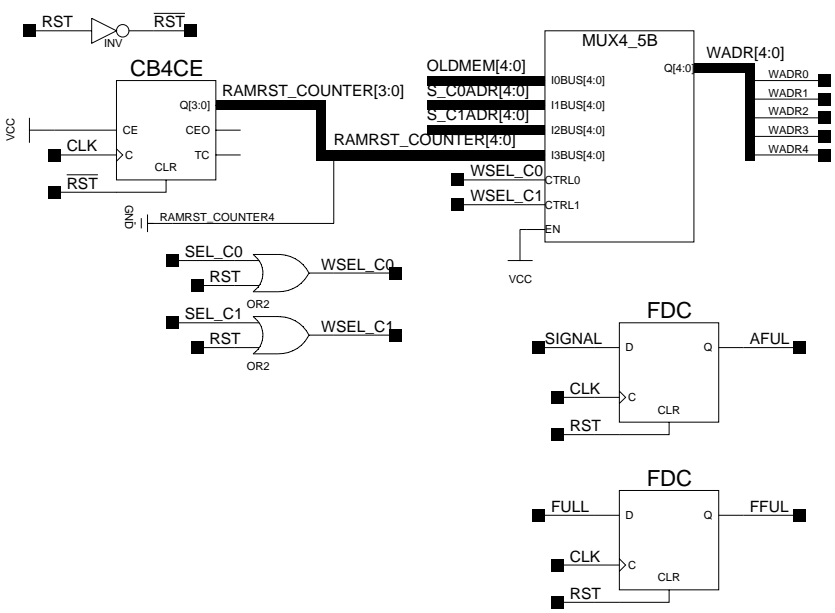
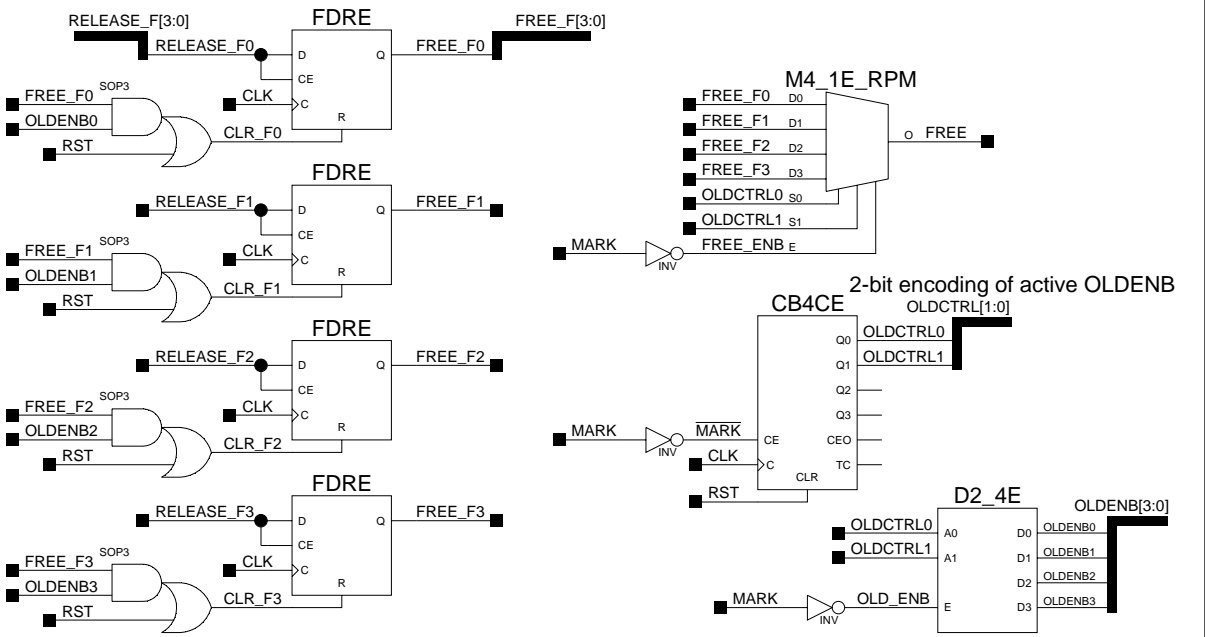
Search corner "c0" FIFO availability for use by any fiber.
Up-search for "corner Fiber f0", down-search otherwise.

write "0" into adr "oldmem" to mark FIFO available

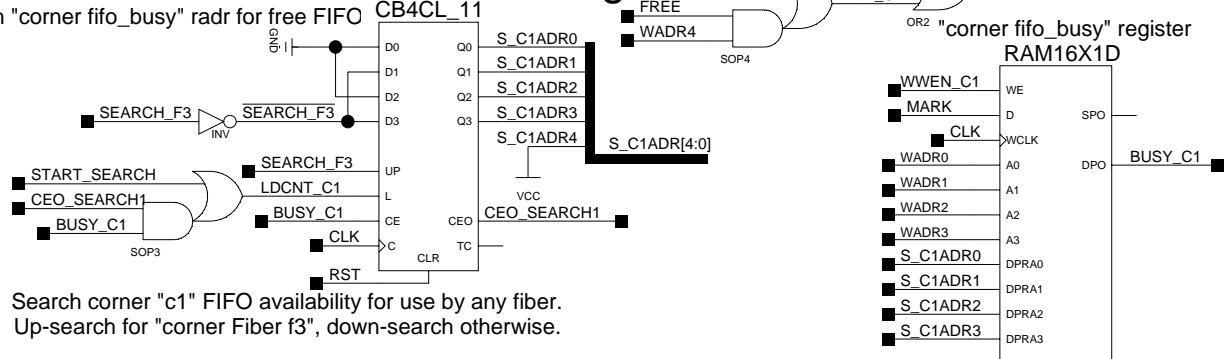
Which Fibers need a new FIFO?



FIFO Release Control



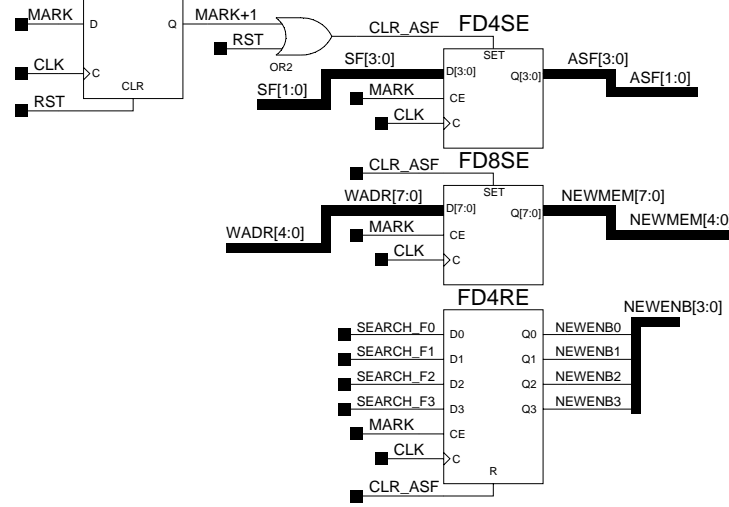
Corner 1 FIFO usage



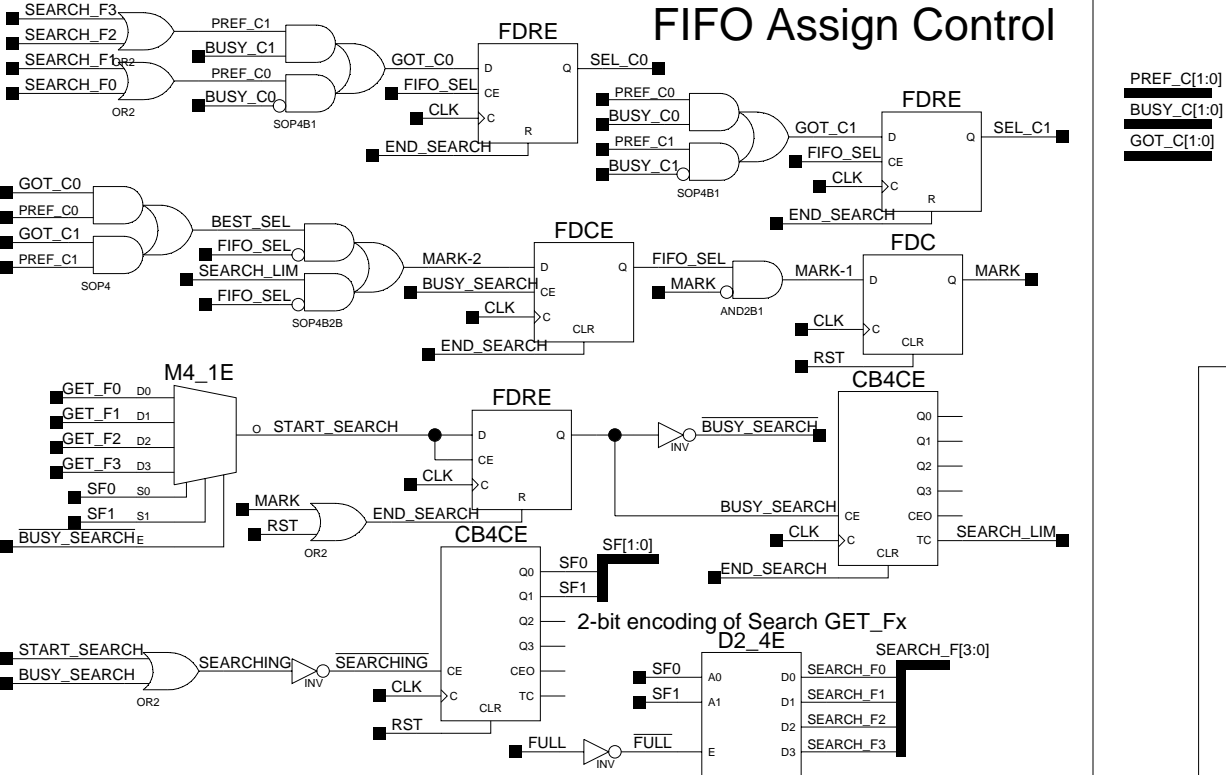
Search corner "c1" FIFO availability for use by any fiber.
Up-search for "corner Fiber f3", down-search otherwise.

write "0" into adr "oldmem" to mark FIFO available

Assign Fiber to FIFO & Assign FIFO to Fiber

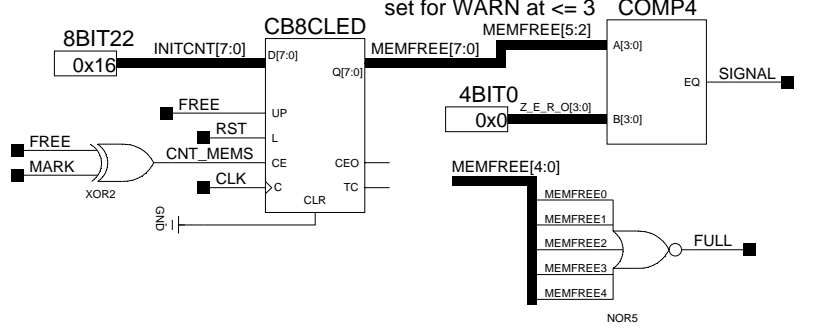


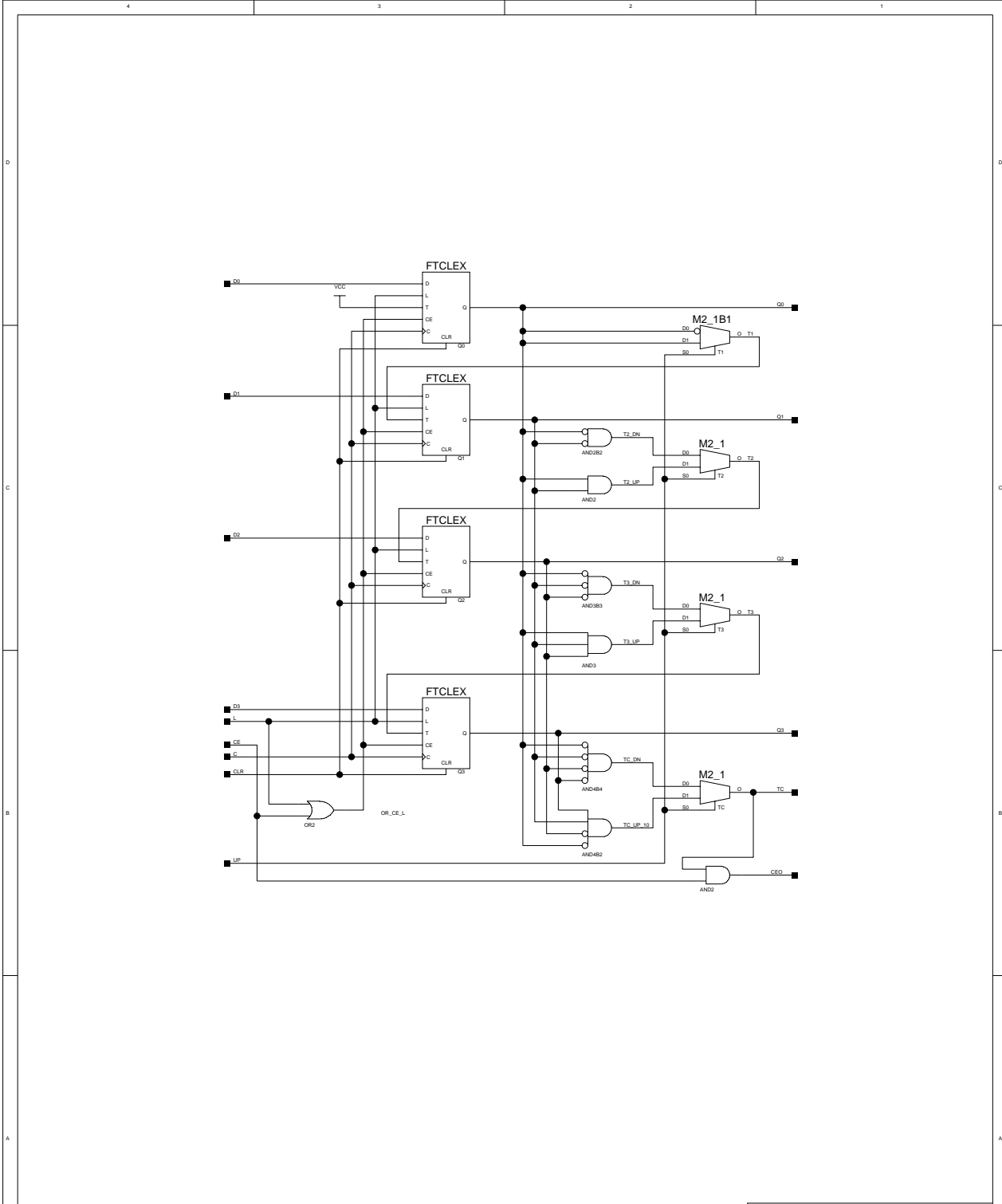
FIFO Assign Control



PREF_C[1:0]
BUSY_C[1:0]
GOT_C[1:0]

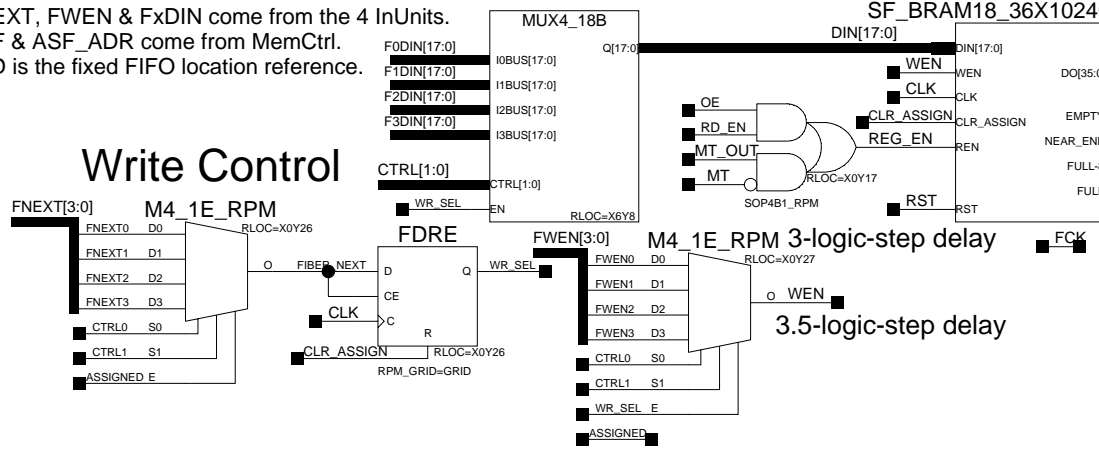
How many of 22 FIFOs are free?





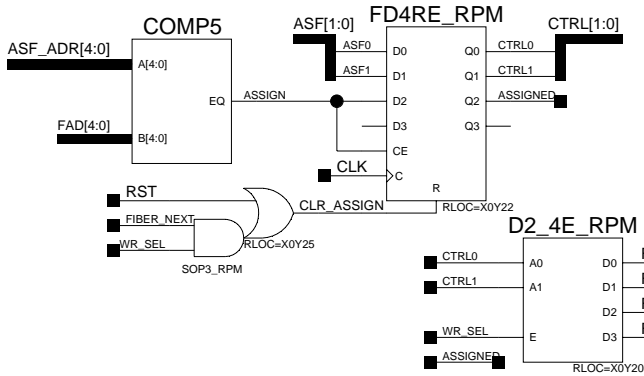
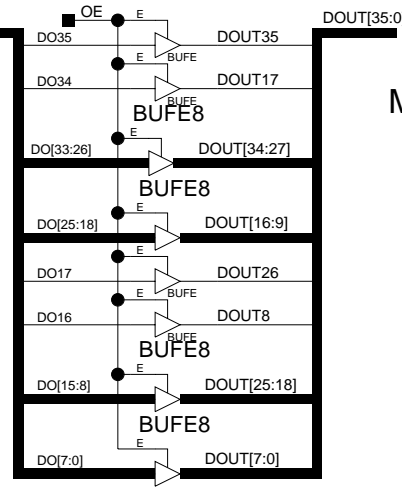
FNEXT, FWEN & FxDIN come from the 4 InUnits.
 ASF & ASF_ADR come from MemCtrl.
 FAD is the fixed FIFO location reference.

Write Control

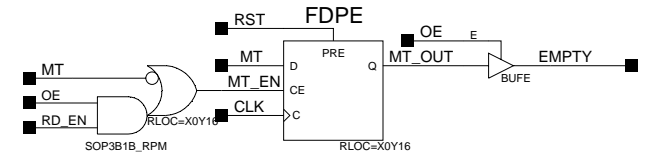
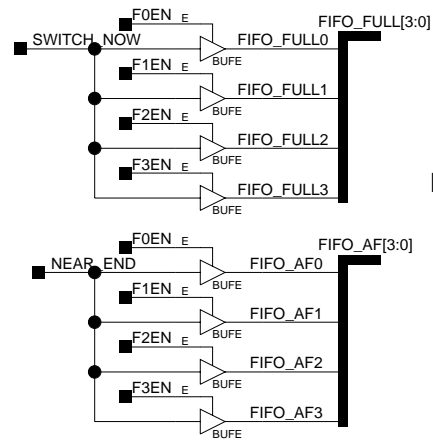


MemIn-MemOut Mapping:

- | | |
|---------|-----------|
| DO35 | Dout35 |
| DO34 | Dout17 |
| DO33-26 | Dout34-27 |
| DO25-18 | Dout16-9 |
| DO17 | Dout26 |
| DO16 | Dout8 |
| DO15-8 | Dout25-18 |
| Din7-0 | Dout7-0 |



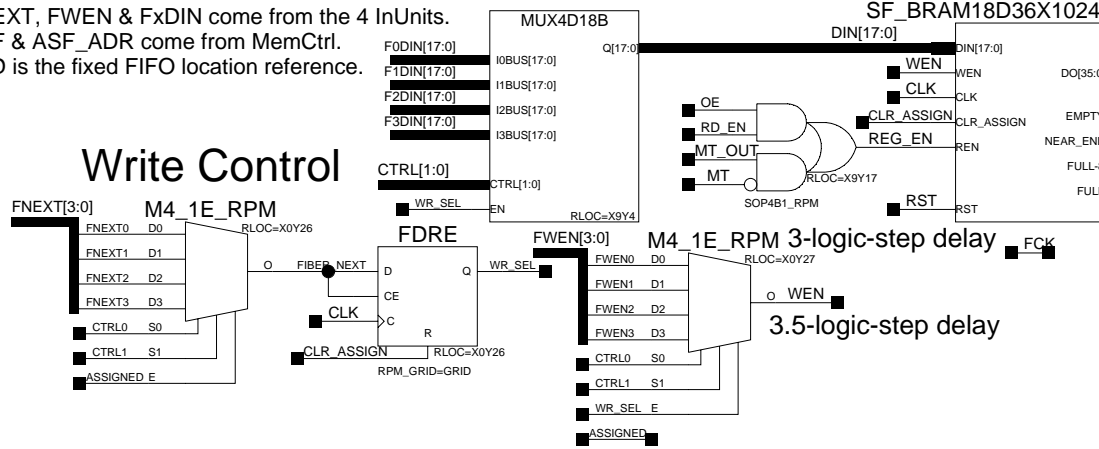
First Word Fall Through logic:



Not Used: REN = OE * RD_EN

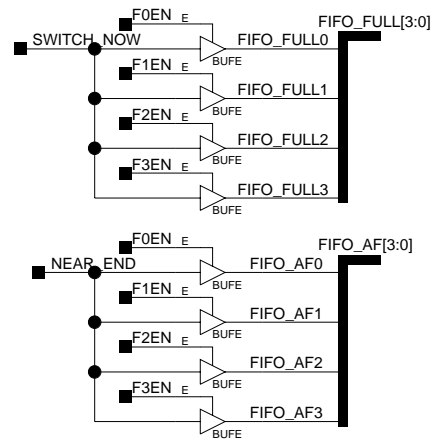
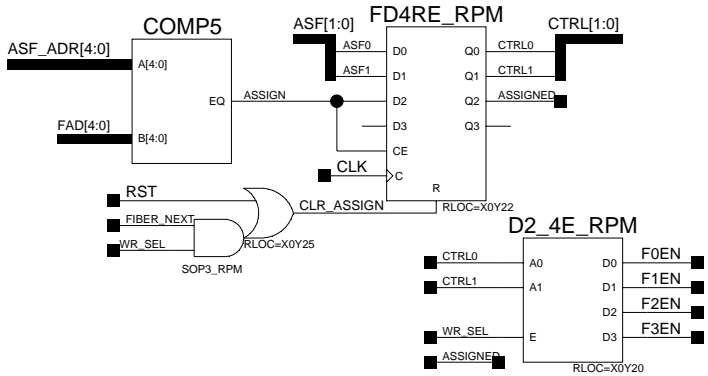
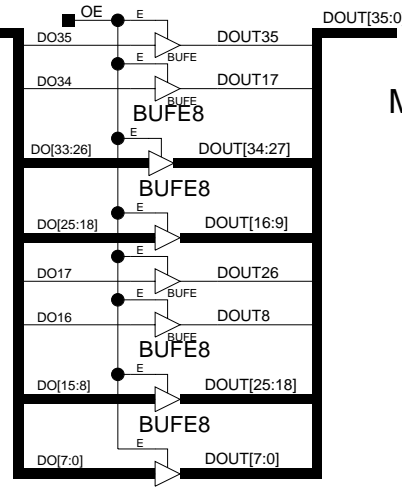
FNEXT, FWEN & FxDIN come from the 4 InUnits.
 ASF & ASF_ADR come from MemCtrl.
 FAD is the fixed FIFO location reference.

Write Control

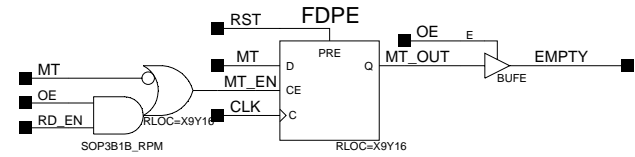


MemIn-MemOut Mapping:

DO35	Dout35
DO34	Dout17
DO33-26	Dout34-27
DO25-18	Dout16-9
DO17	Dout26
DO16	Dout8
DO15-8	Dout25-18
Din7-0	Dout7-0

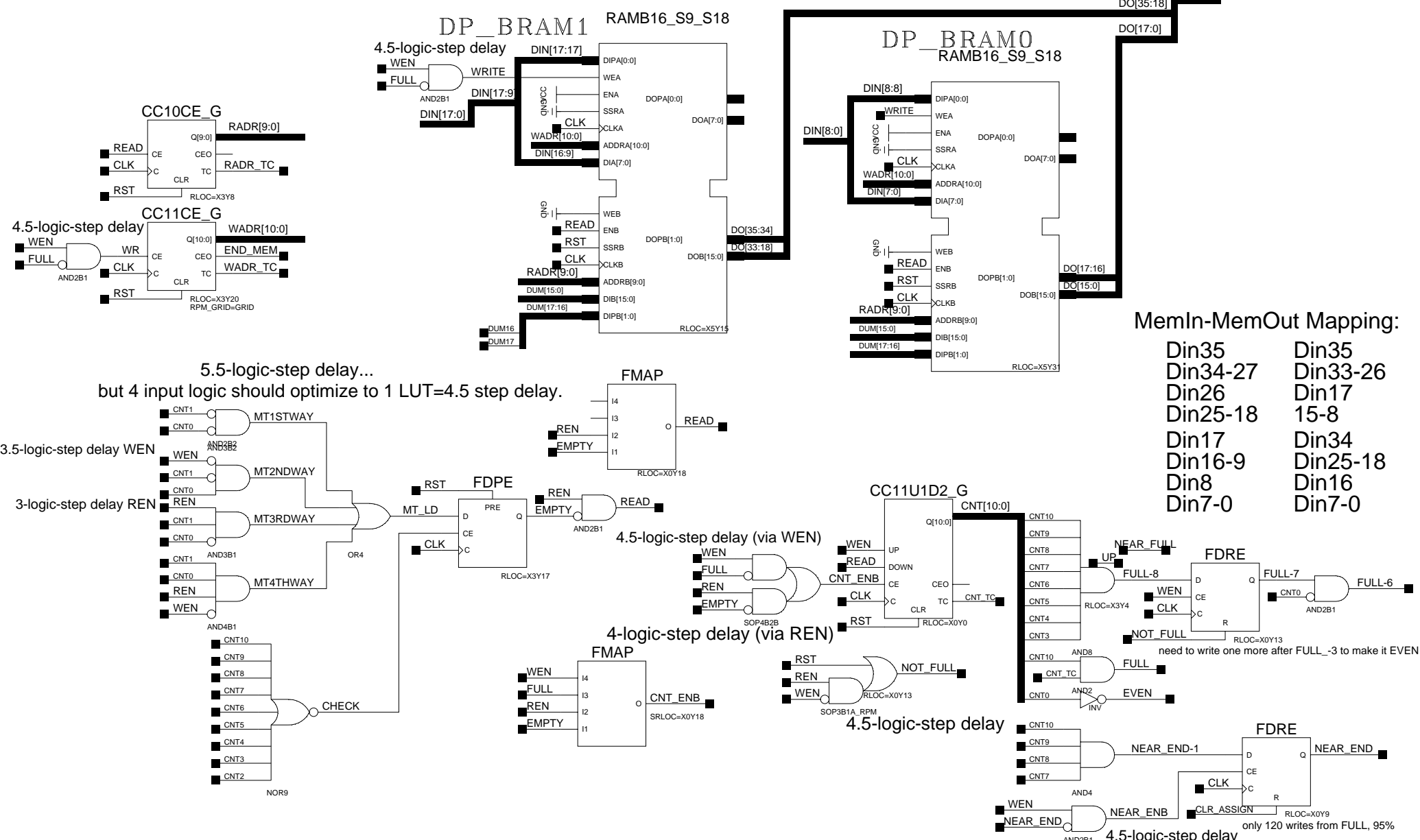


First Word Fall Through logic:



Not Used: REN = OE * RD_EN

sfifo18_36x1024



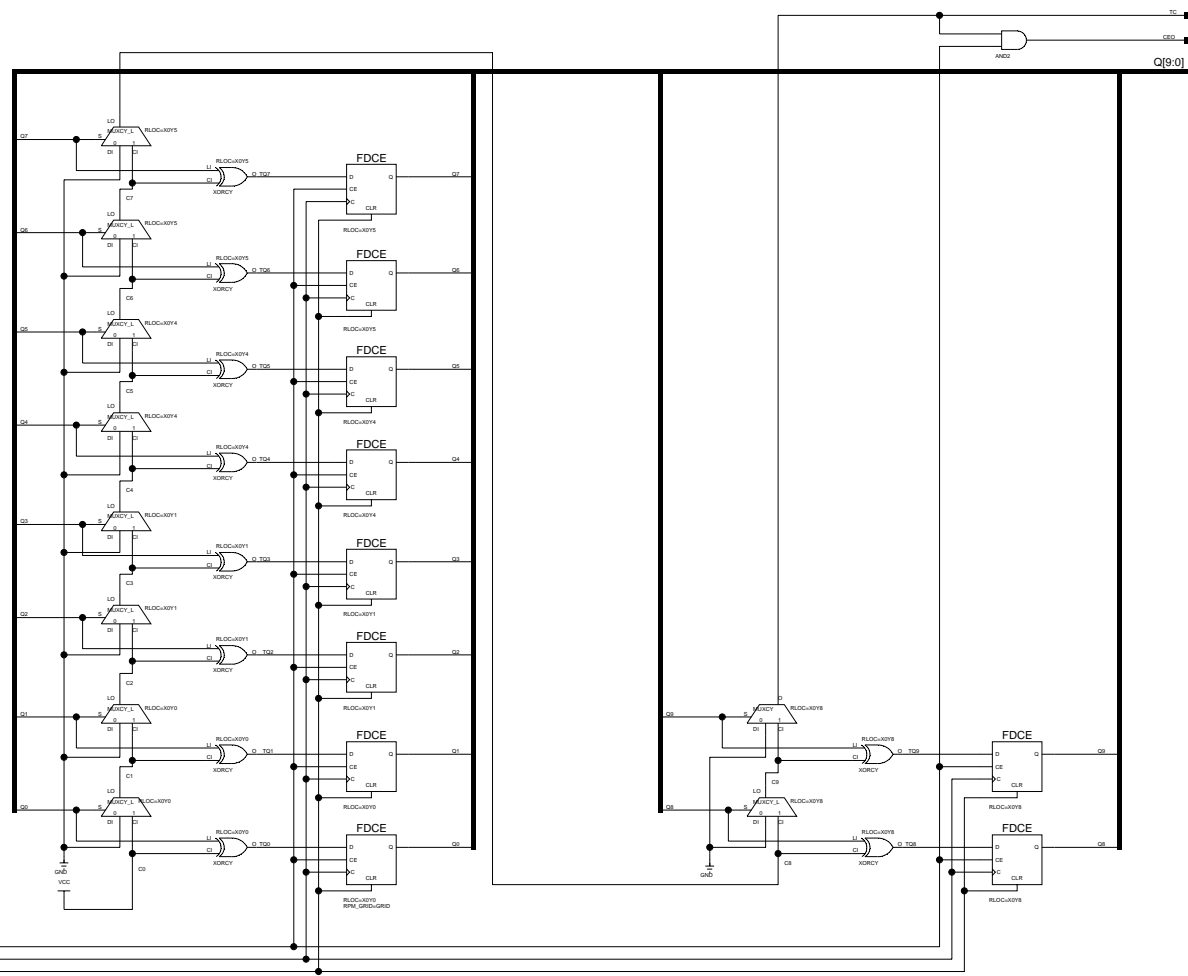
5.5-logic-step delay...
but 4 input logic should optimize to 1 LUT=4.5 step delay.

4.5-logic-step delay (via WEN)

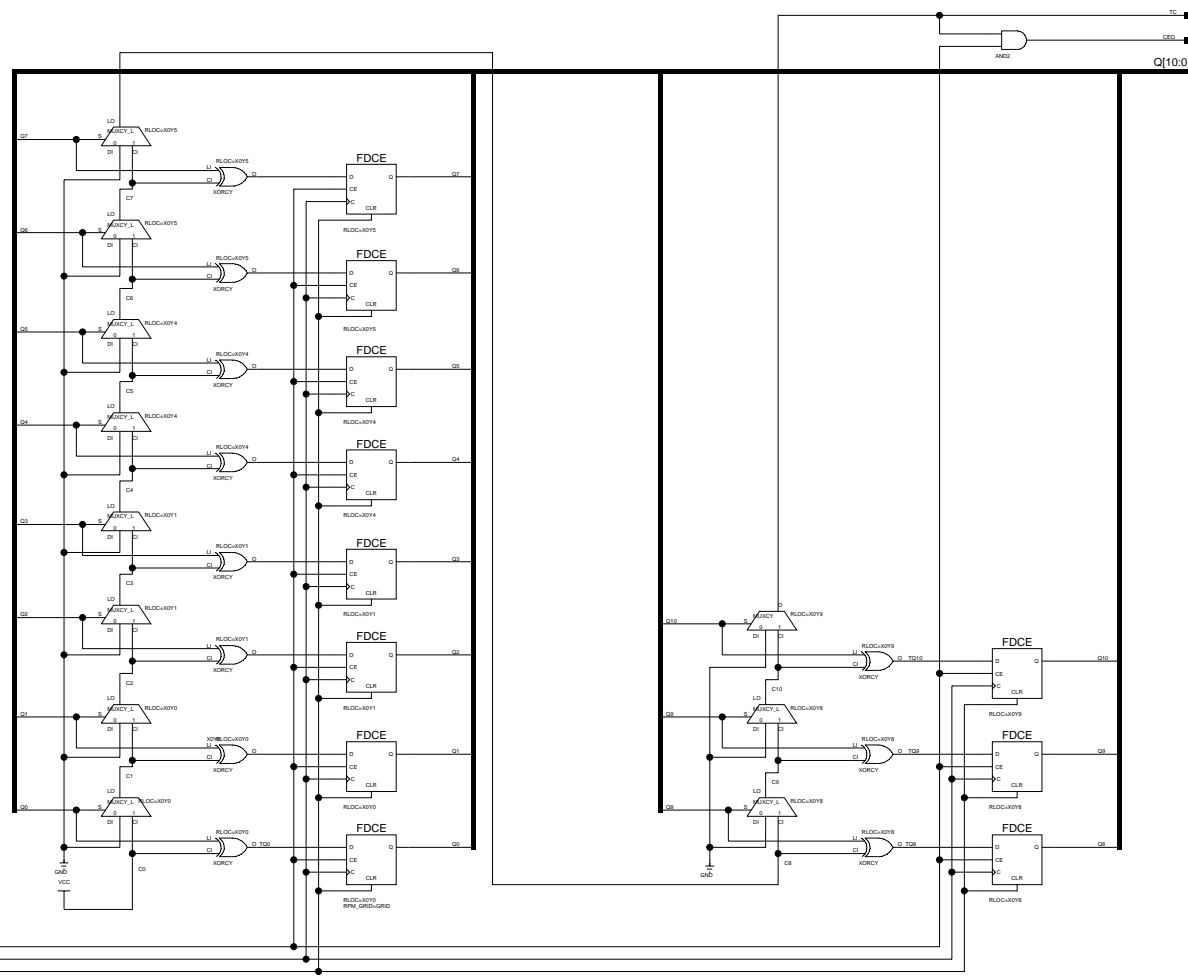
4-logic-step delay (via REN)

4.5-logic-step delay

4.5-logic-step delay

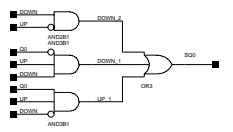
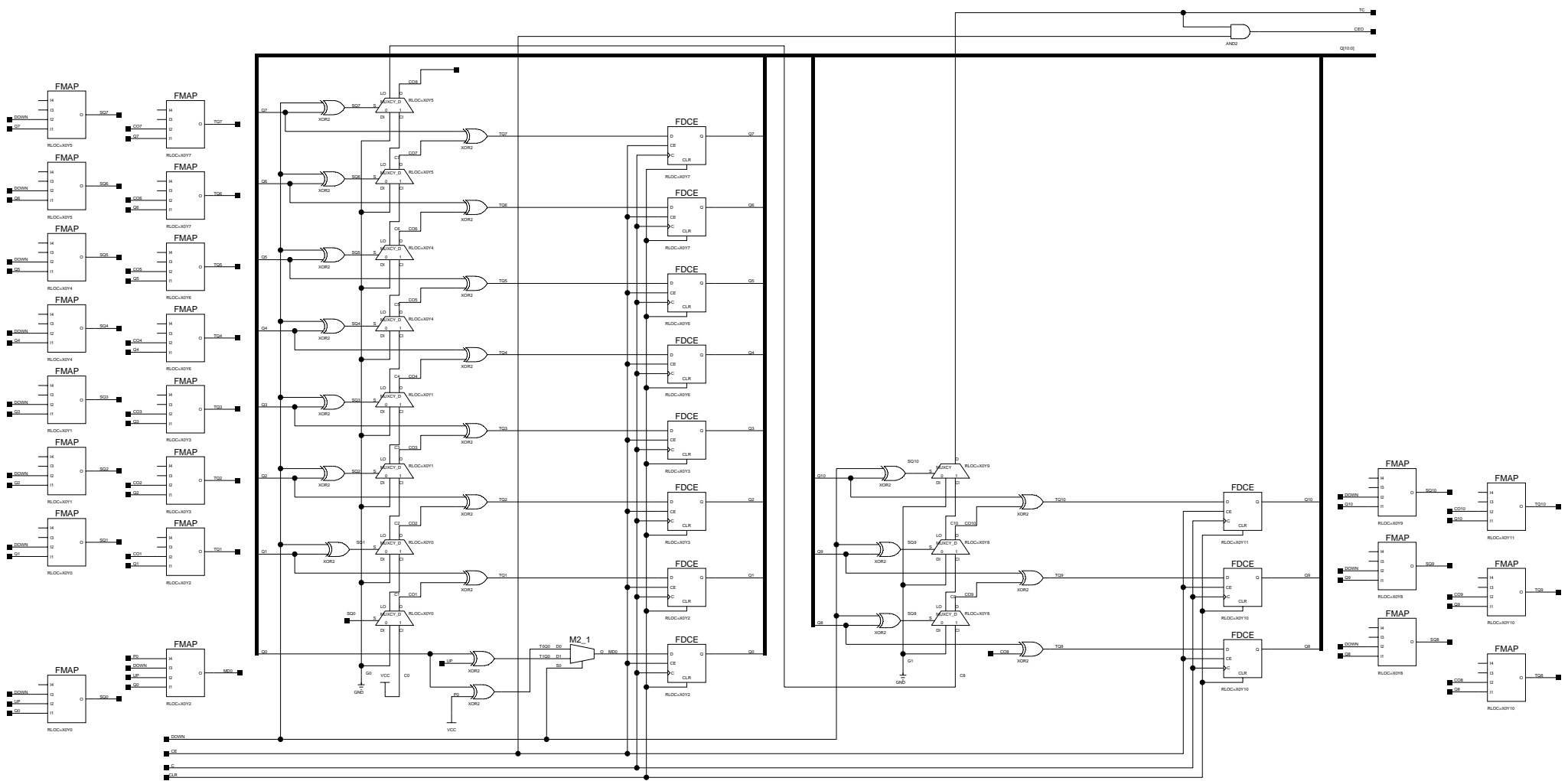


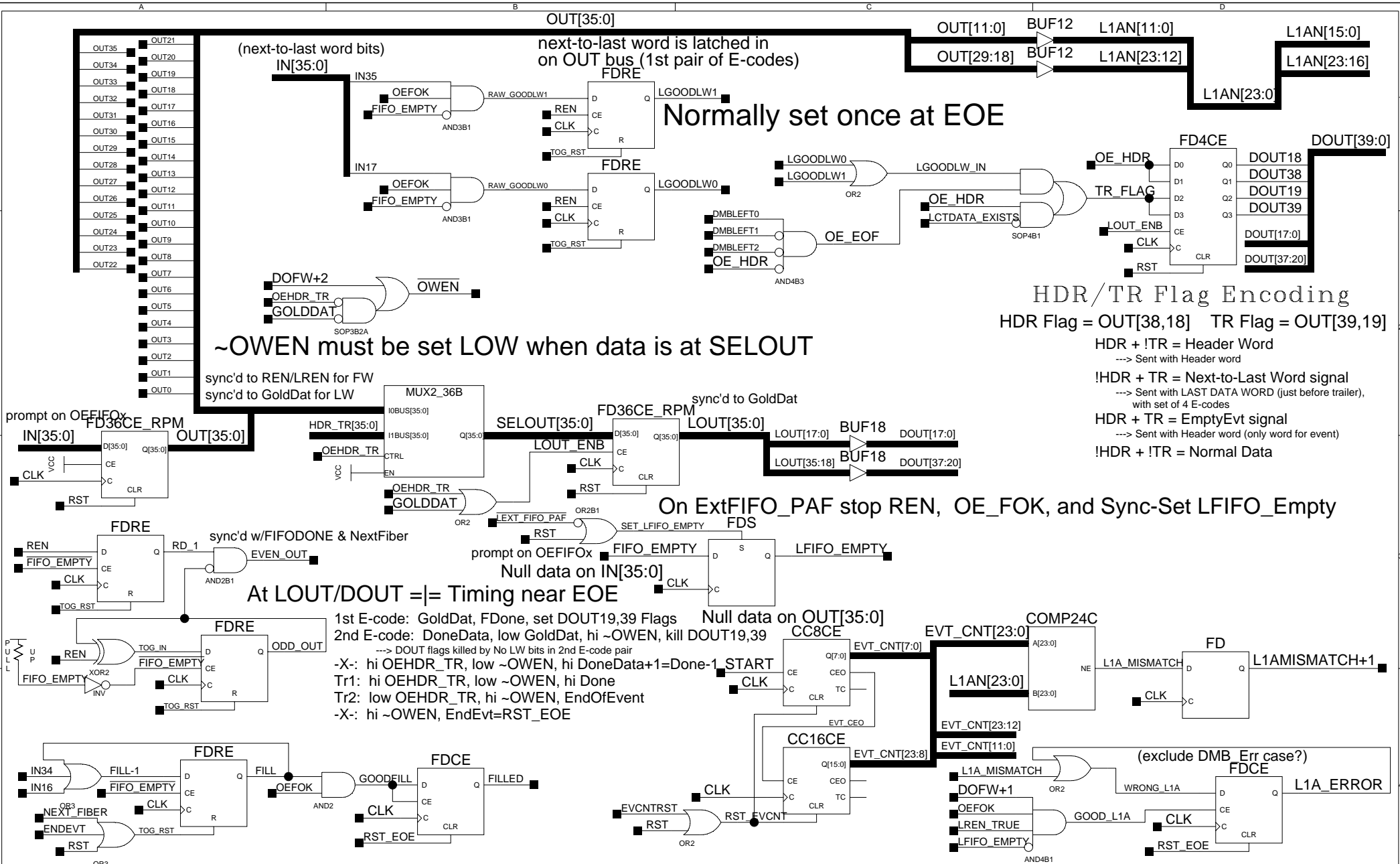
CLR
 D
 Q[9:0]



■ CE
 ■ C
 ■ CLR

UP = Increment 1
 UP.DOWN = Decrement 1
 DOWN = Decrement 2
 Assume that CE includes DOWN+UP





(next-to-last word bits)
 next-to-last word is latched in on OUT bus (1st pair of E-codes)

Normally set once at EOE

HDR/TR Flag Encoding

- HDR Flag = OUT[38,18] TR Flag = OUT[39,19]
- HDR + !TR = Header Word
 ---> Sent with Header word
- !HDR + TR = Next-to-Last Word signal
 ---> Sent with LAST DATA WORD (just before trailer), with set of 4 E-codes
- HDR + TR = EmptyEvt signal
 ---> Sent with Header word (only word for event)
- !HDR + !TR = Normal Data

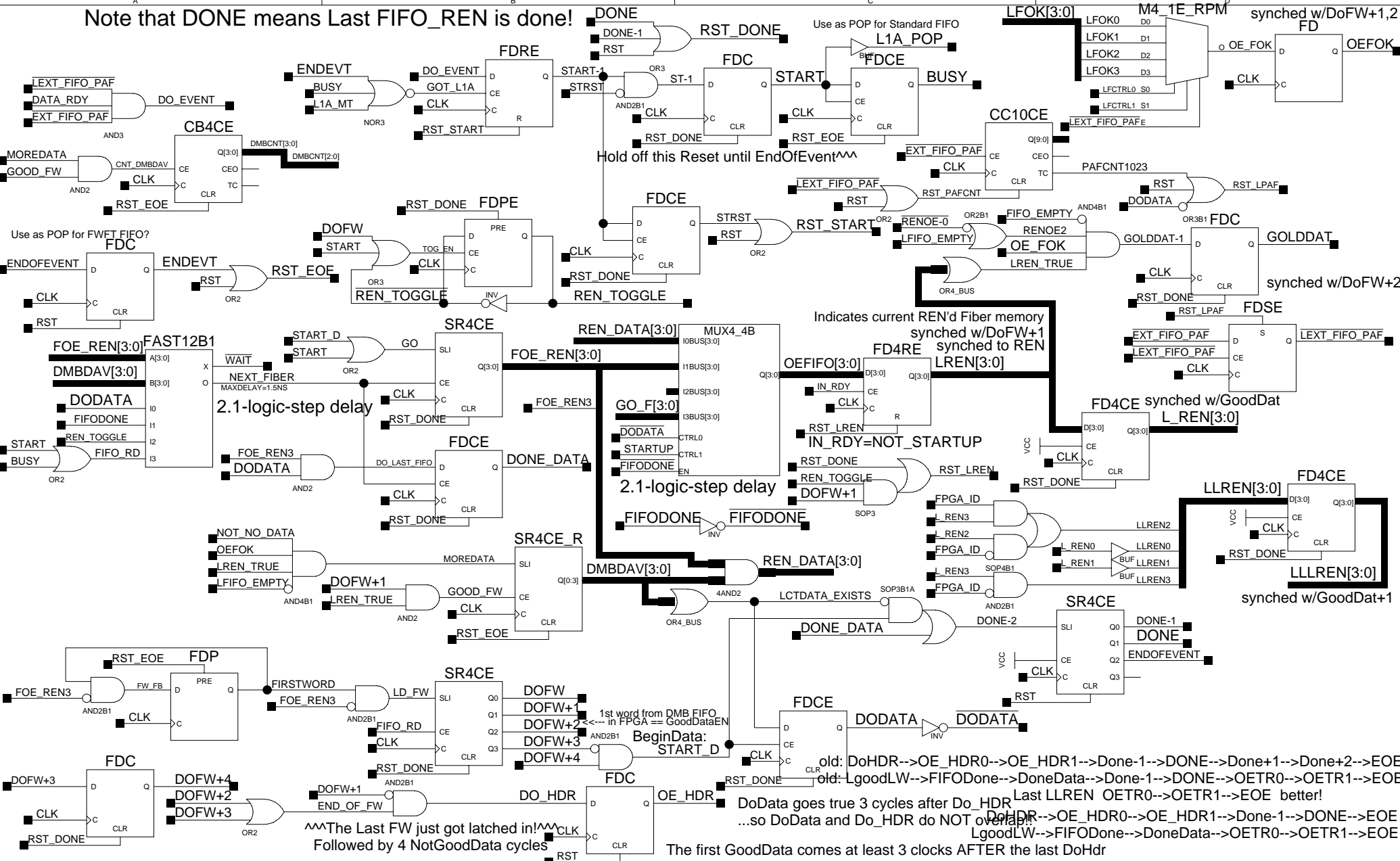
~OWEN must be set LOW when data is at SELOUT

On ExtFIFO_PAF stop REN, OE_FOK, and Sync-Set LFIFO_Empty

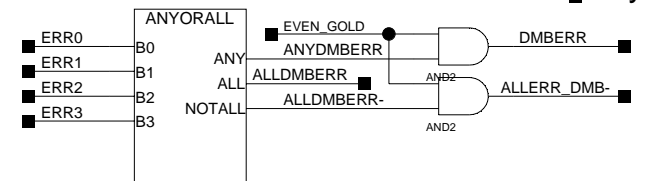
At LOUT/DOUT != Timing near EOE

- 1st E-code: GoldDat, FDone, set DOUT19,39 Flags Null data on OUT[35:0]
- 2nd E-code: DoneData, low GoldDat, hi ~OWEN, kill DOUT19,39
 ---> DOUT flags killed by No LW bits in 2nd E-code pair
- X-: hi OEHDR_TR, low ~OWEN, hi DoneData+1=Done-1 START
- Tr1: hi OEHDR_TR, low ~OWEN, hi Done
- Tr2: low OEHDR_TR, hi ~OWEN, EndOfEvent
- X-: hi ~OWEN, EndEvt=RST_EOE

3 more words to FIFO after FF is set; use LEXT_FIFO_PAF to stop REN/OWEN!



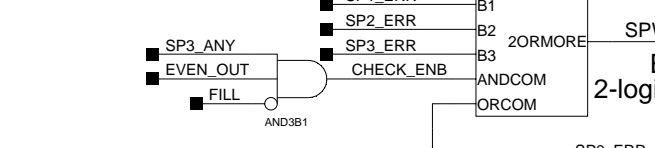
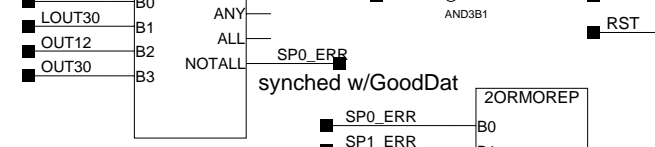
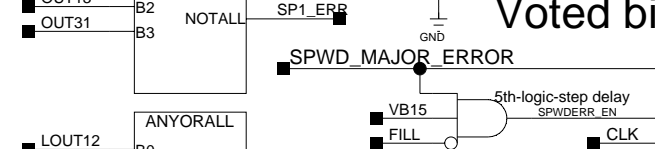
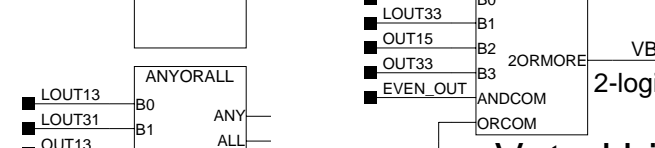
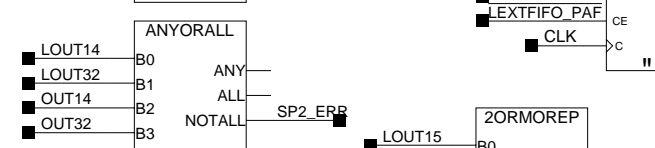
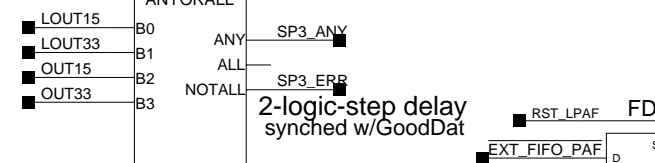
Check for DMB Error Word and consistency



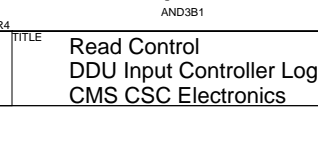
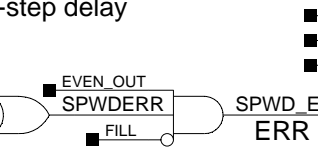
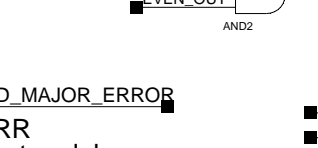
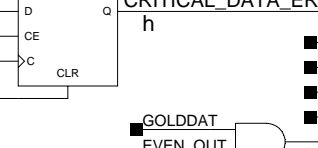
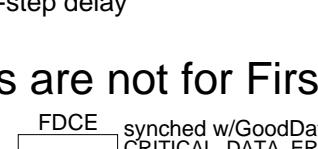
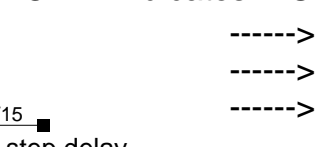
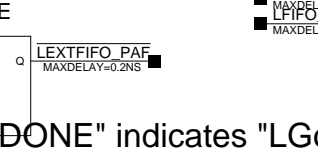
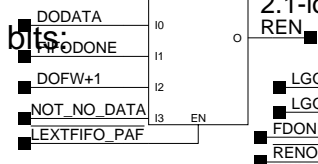
Out to LEDs?
DMBERR_WORD
ALLERR_DMB

3 more words to FIFO after FF is set; use PAF to stop!

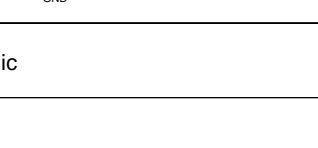
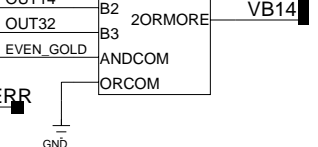
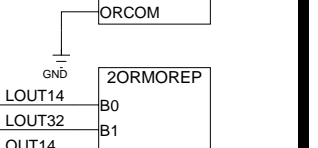
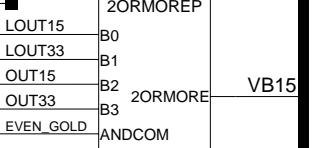
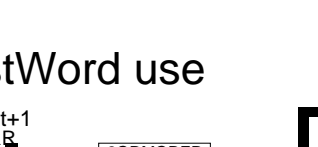
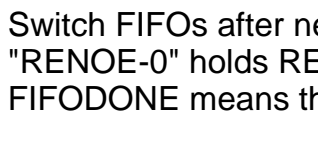
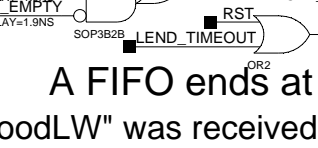
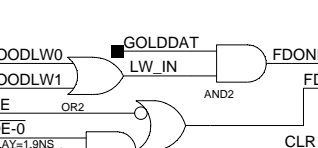
Check consistency of the four "Special Word" bits



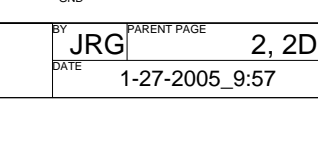
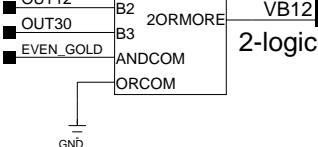
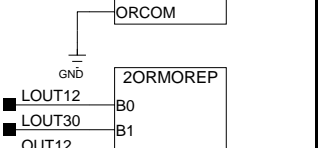
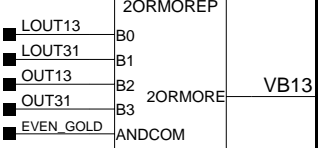
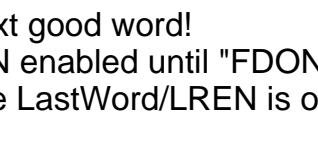
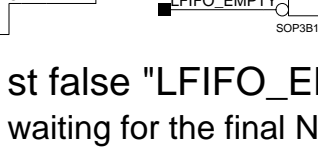
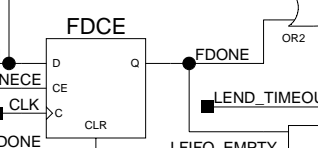
LREN[3:0] FAST8B2EN



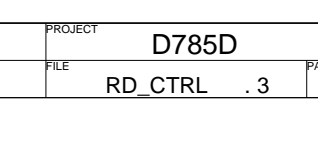
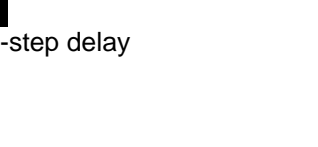
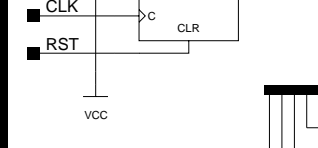
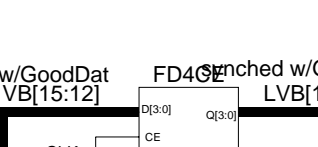
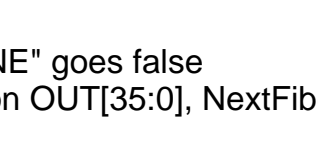
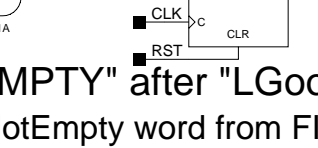
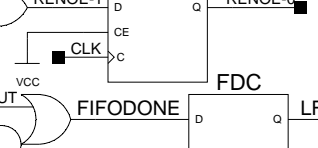
REN's the current OE'd memory
 2.1-logic-step delay



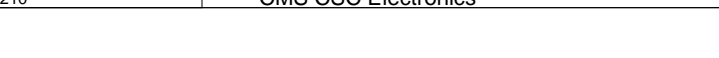
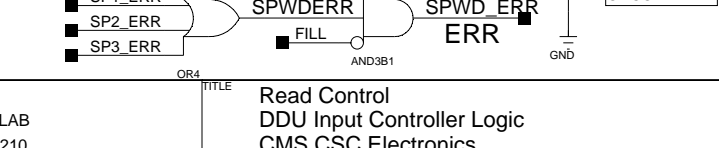
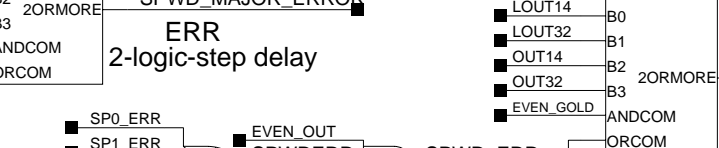
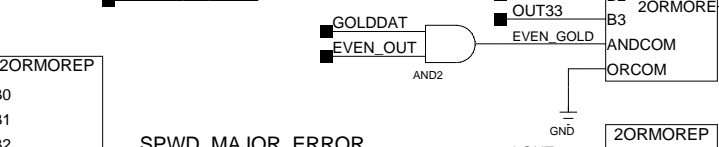
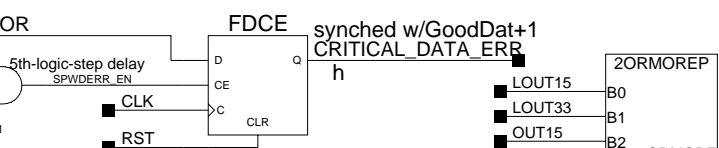
A FIFO ends at 1st false "LFIFO_EMPTY" after "LGoodLW"

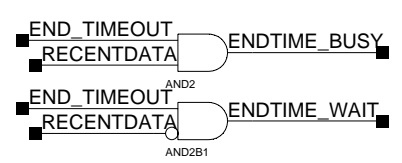
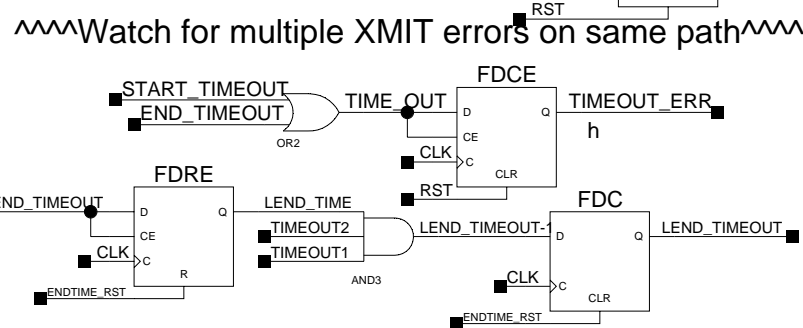
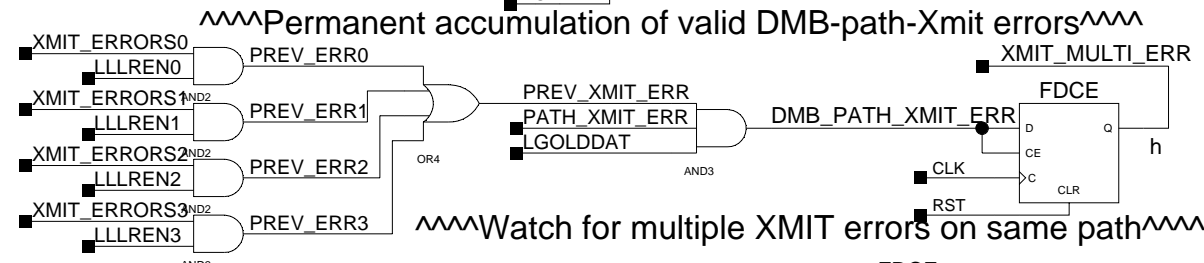
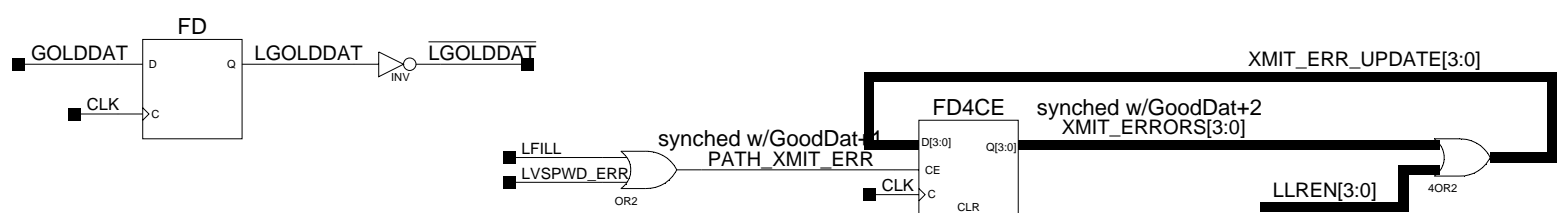


Switch FIFOs after next good word!
 "RENOE-0" holds REN enabled until "FDONE" goes false
 FIFODONE means the LastWord/LREN is on OUT[35:0], NextFiber is set

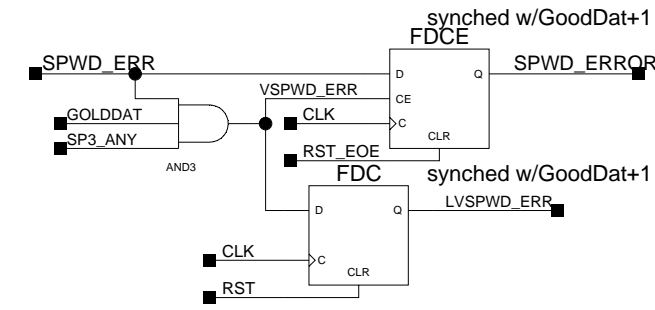
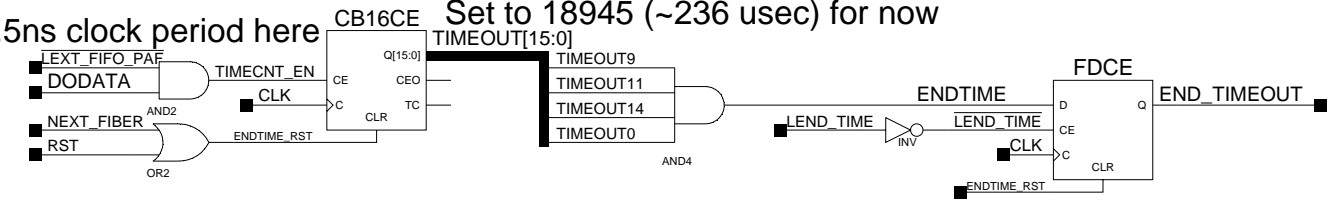


Voted bits are not for FirstWord use





FIFO Done Timeout: count to 10752, 132 usec=10625 is the worst case possible per FIFO?
 Set to 18945 (~236 usec) for now

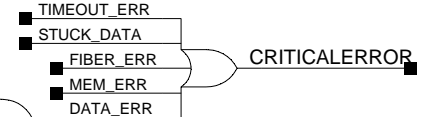
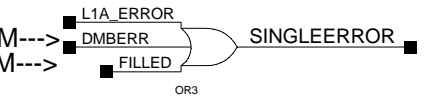
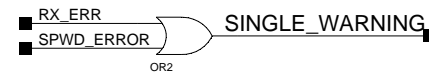


Status Monitor

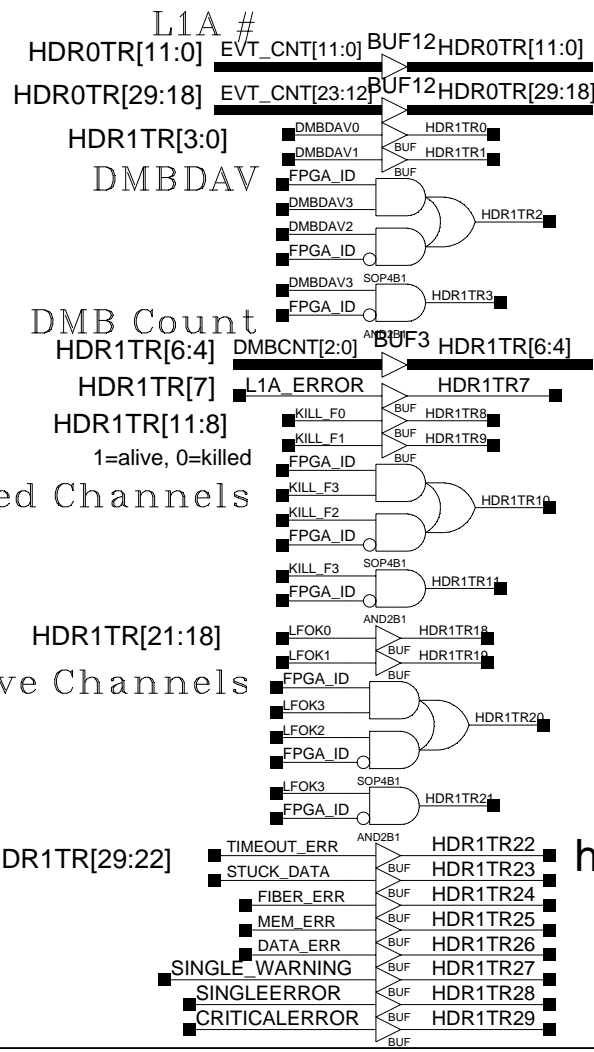
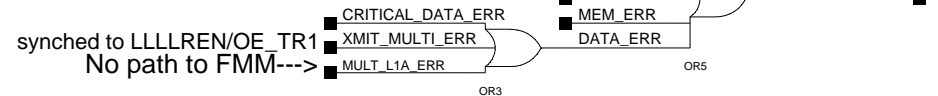
Pick 8 Status signals, include in DAQ HDR/TR

-create logic for OEHDR/TR and their format

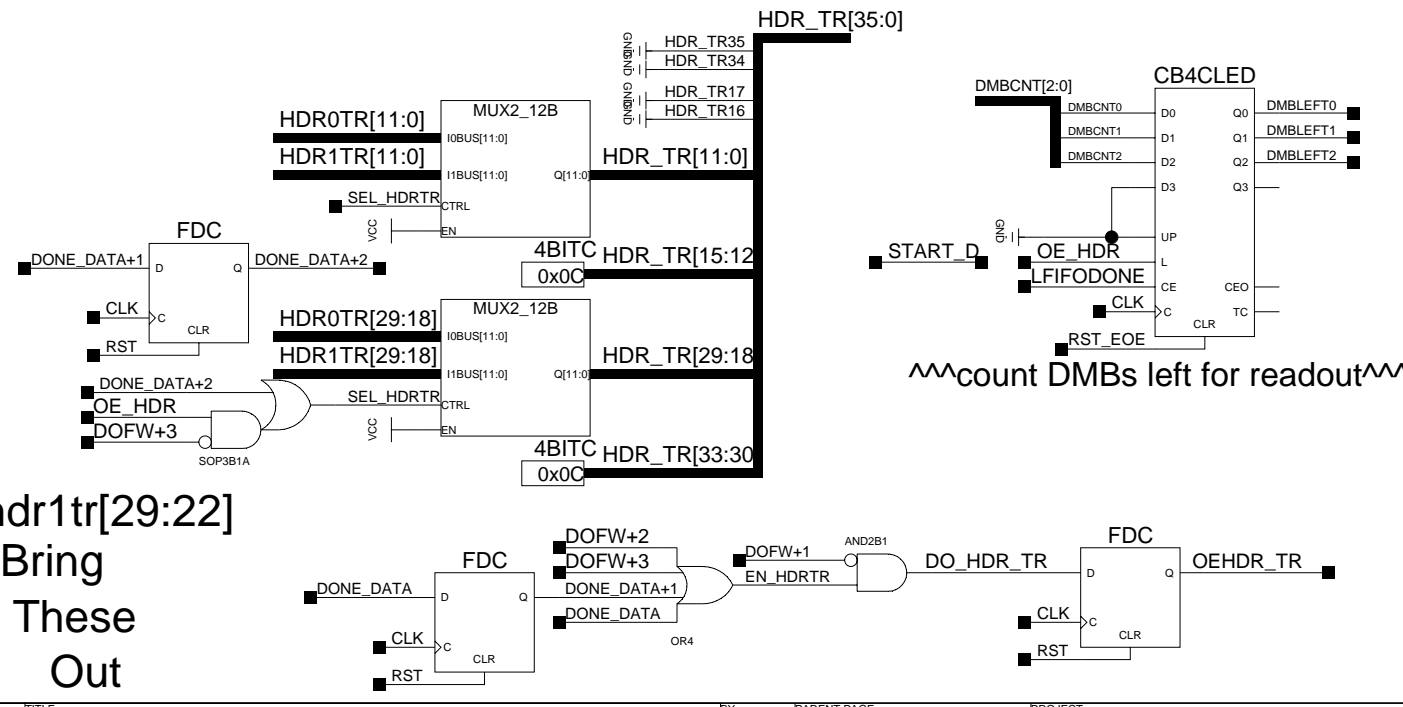
Also set 4 FMM bits for each input channel



In Hdr/Tr, removed from data: No path to FMM---> No path to FMM--->



Note: all HDR_TR[35:34,17:16] bits are free
 also for HDR0TR & HDR1TR
 - use for Event Type/Pasthrough flag?
 - use for additional error reporting?



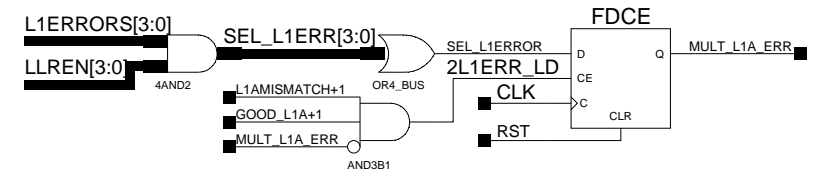
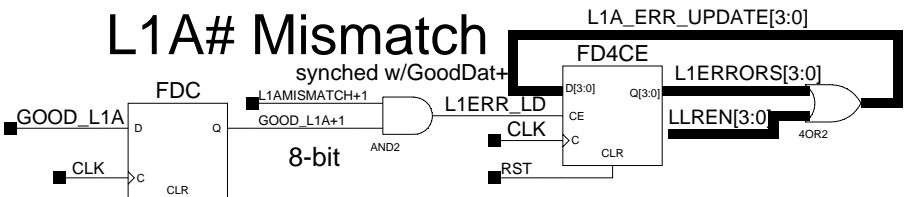
Killed Channels

Live Channels

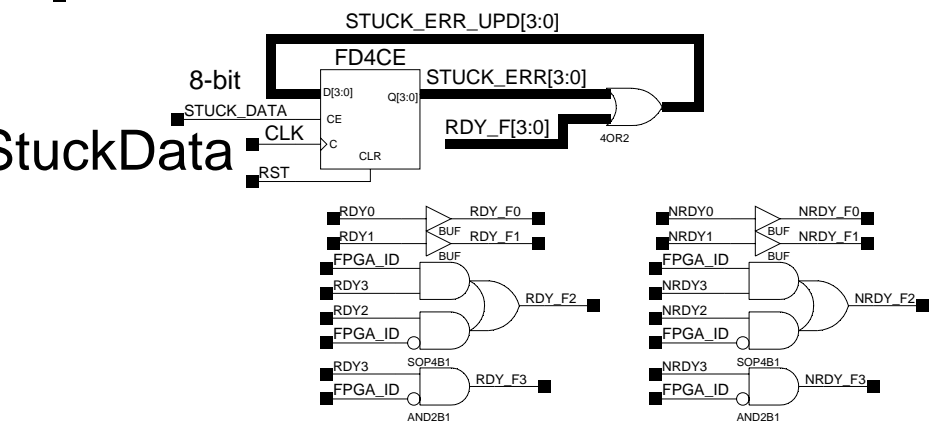
hdr1tr[29:22]
 Bring These Out

^^count DMBs left for readout^^

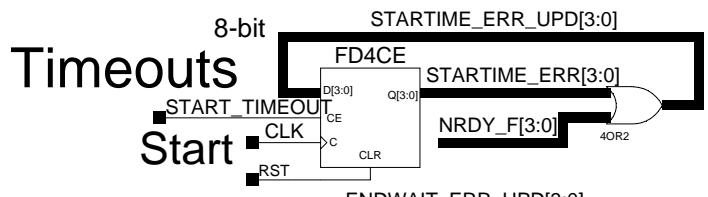
L1A# Mismatch



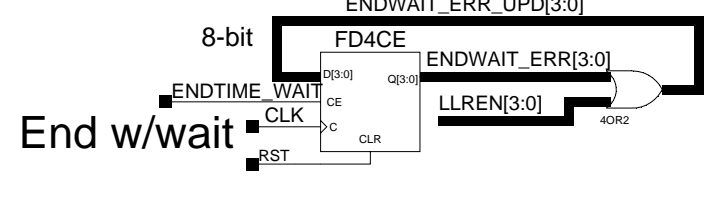
StuckData



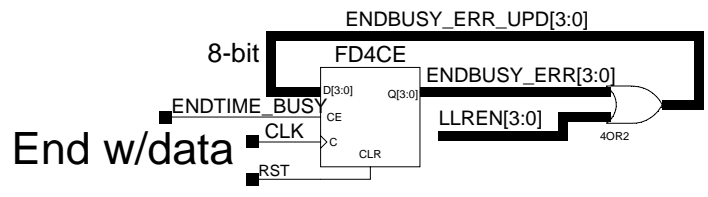
Timeouts



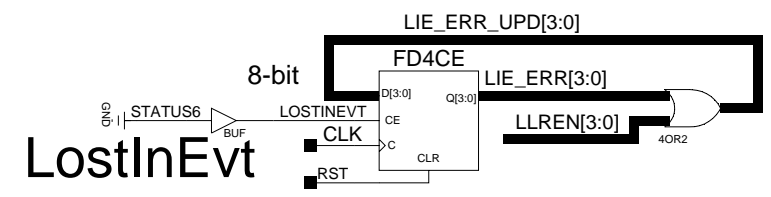
End w/wait



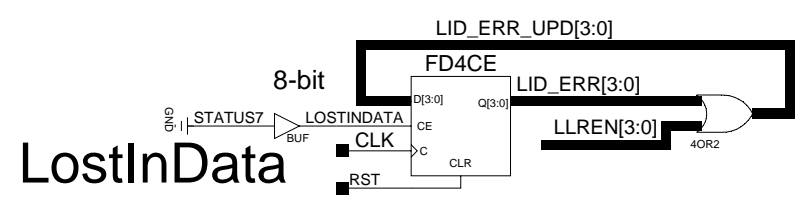
End w/data



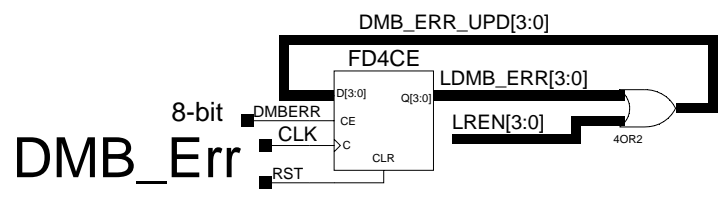
LostInEvt



LostInData



DMB_Err



A B C D

THE OHIO STATE UNIVERSITY
PHYSICS DEPARTMENT ELECTRONICS LAB
174 WEST 18TH AVE, COLUMBUS OH 43210

TITLE

Read Control
DDU Input Controller Logic
CMS CSC Electronics

BY

JRG

PARENT PAGE

2, 2D

PROJECT

D785D

DATE

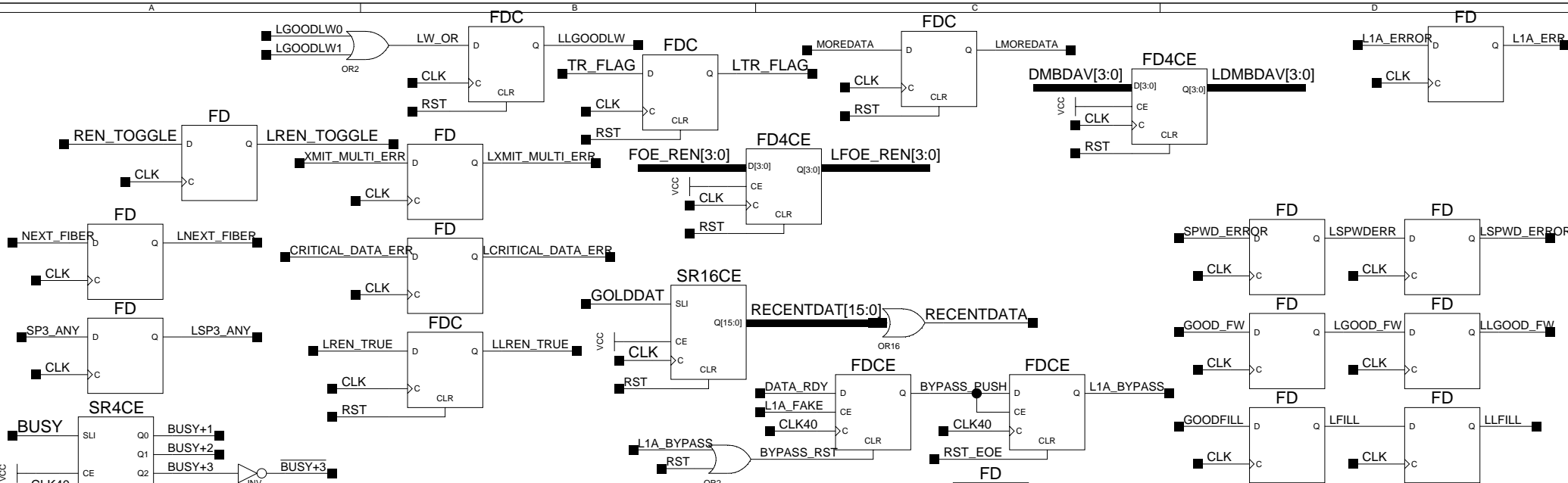
1-27-2005_9:58

FILE

RD_CTRL . 7

PAGE

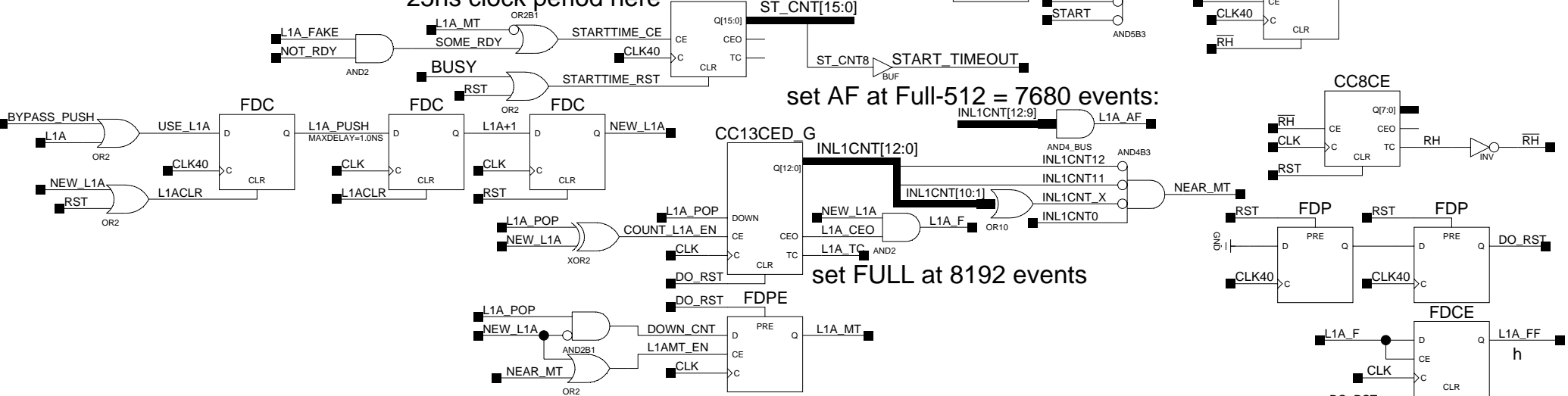
16F



Event START timeout 3.2usec: $128=1000/0000=0x80$.

Calibration START timeout 6.4usec: $256=0x0100$.

25ns clock period here

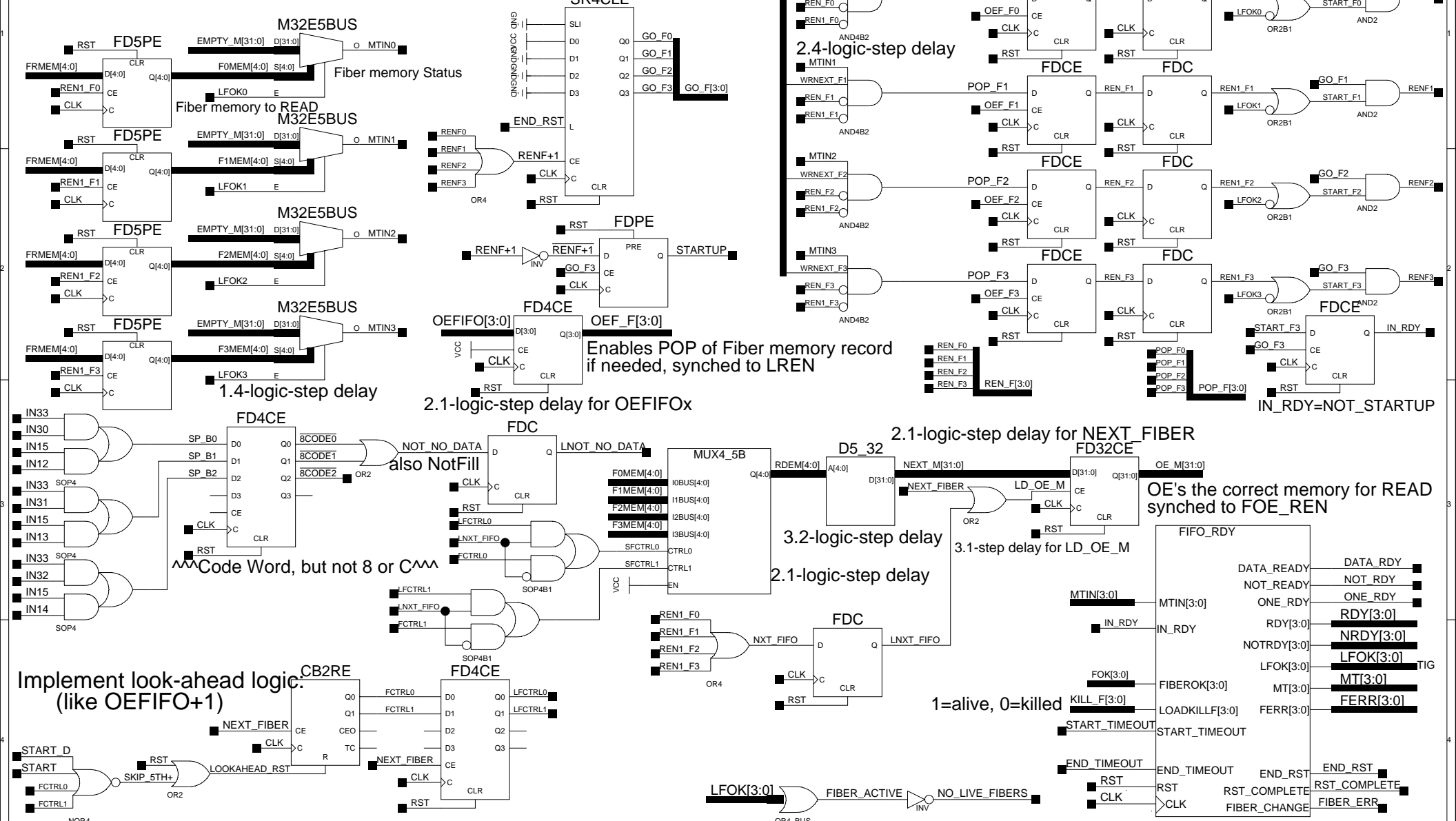


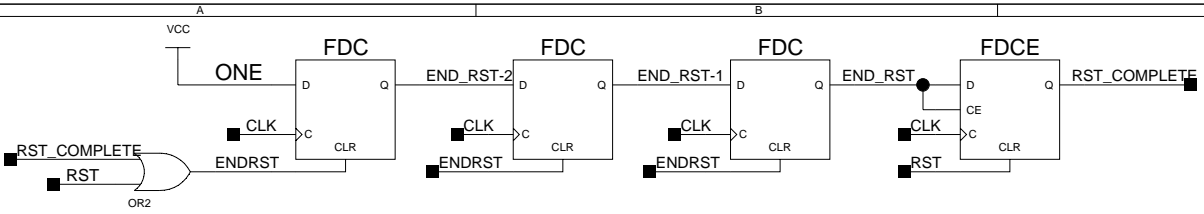
set AF at Full-512 = 7680 events:

set FULL at 8192 events

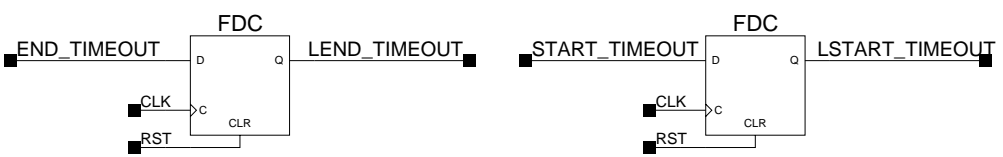
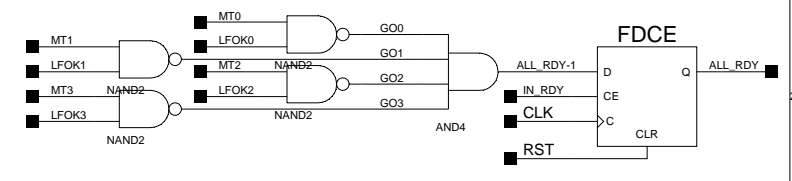
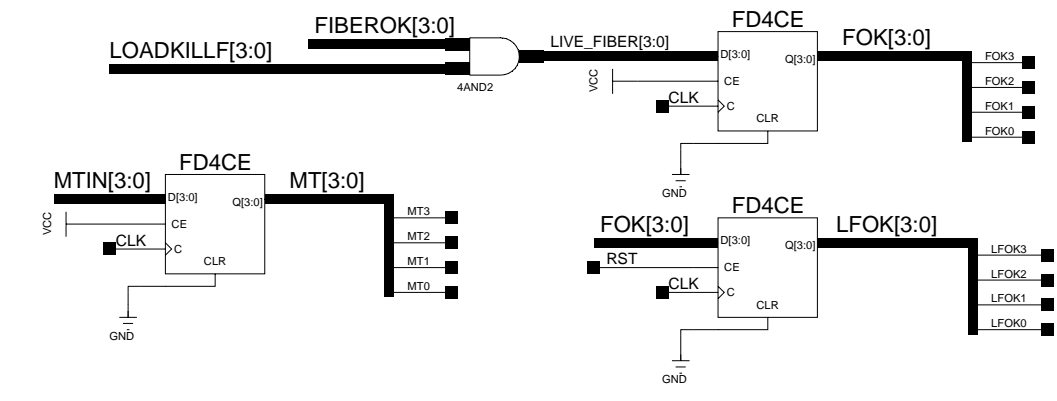
SPS BXN cycle from 0 to 923: CLR after 923=0x39B.

END

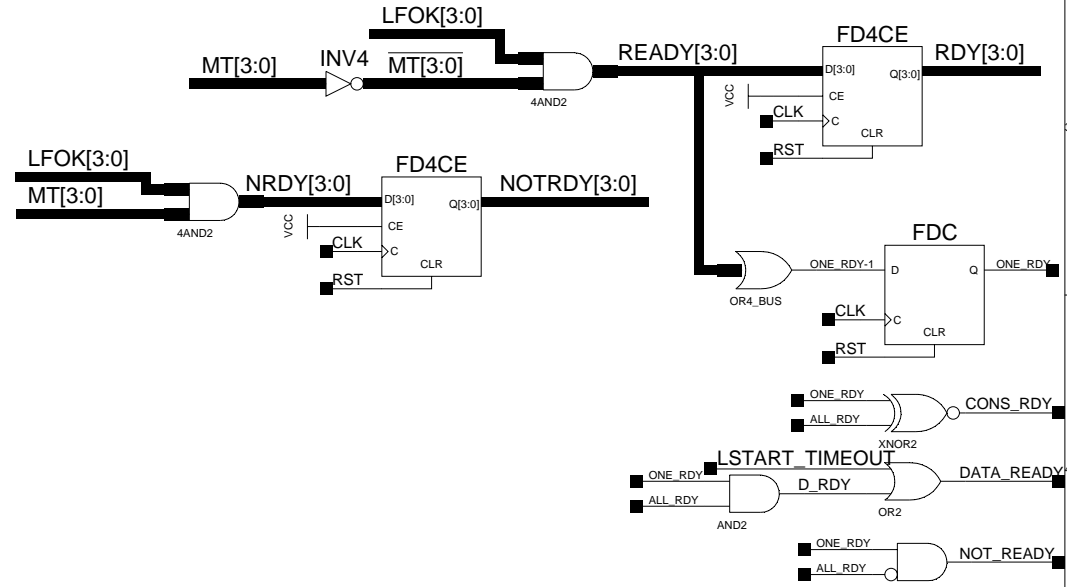


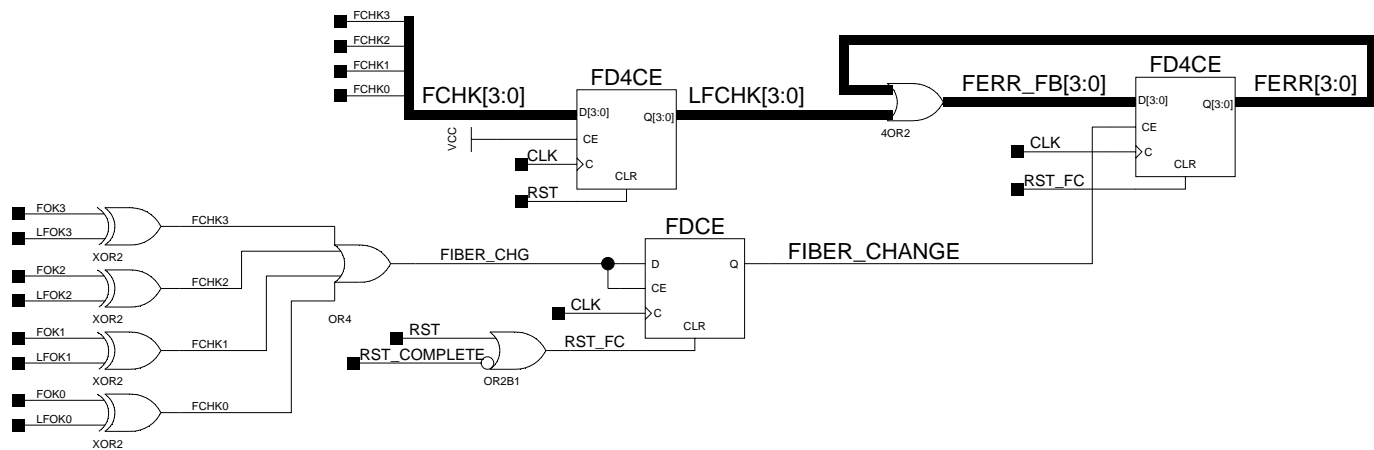


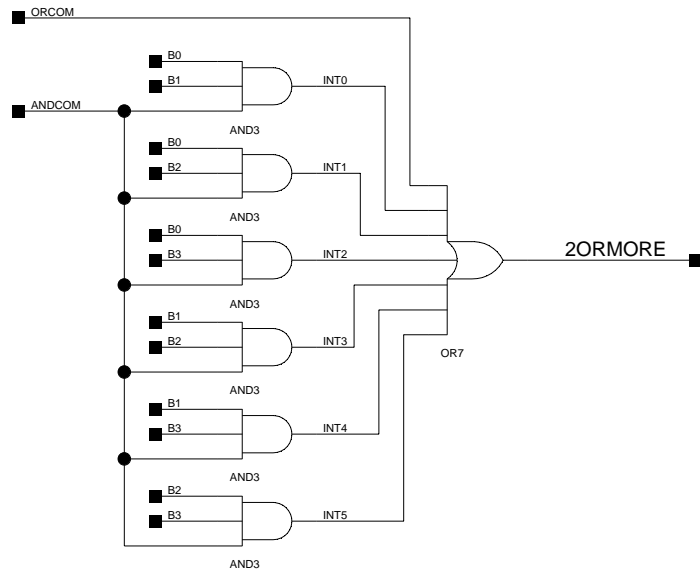
Access FIBEROK from JTAG as a fiber check.
 ---> RESET required after fiber status change for now...
 Change in FIBEROK is BAD! Set error code.
 ...notify FMM and maybe set STATUS bit, but
 ...data is OK until FIFO read time-out occurs.
 ...but how to know WHEN the bad-data comes out
 ---> timeout will probably occur for that event

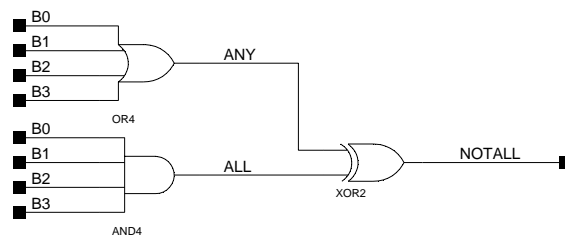


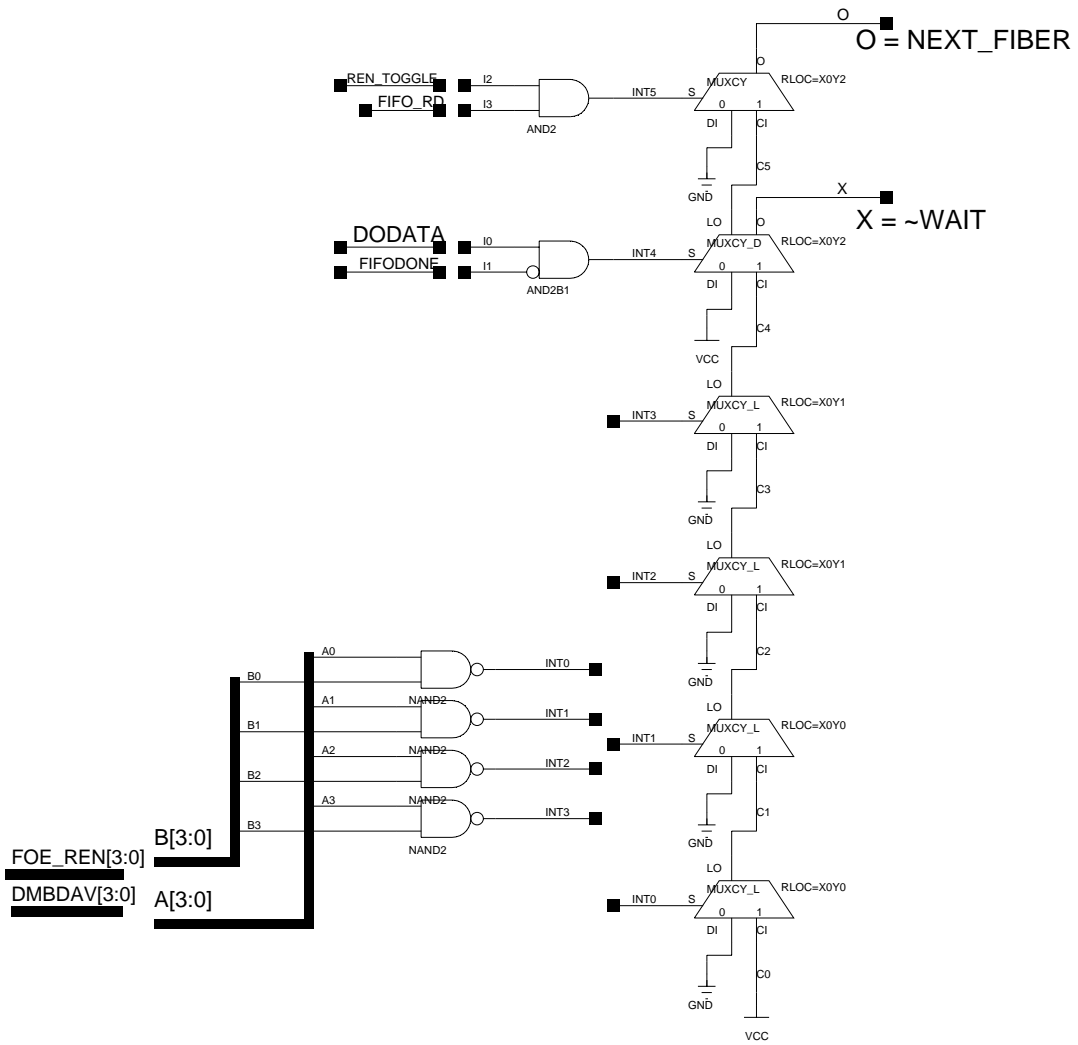
MASK fiber on Start/End TIMEOUT? Kill it in LFOK...



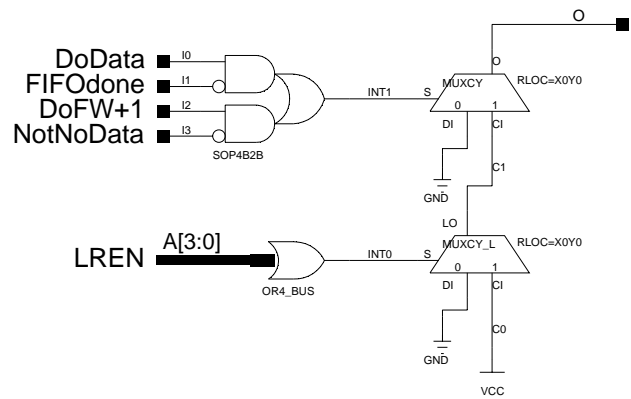








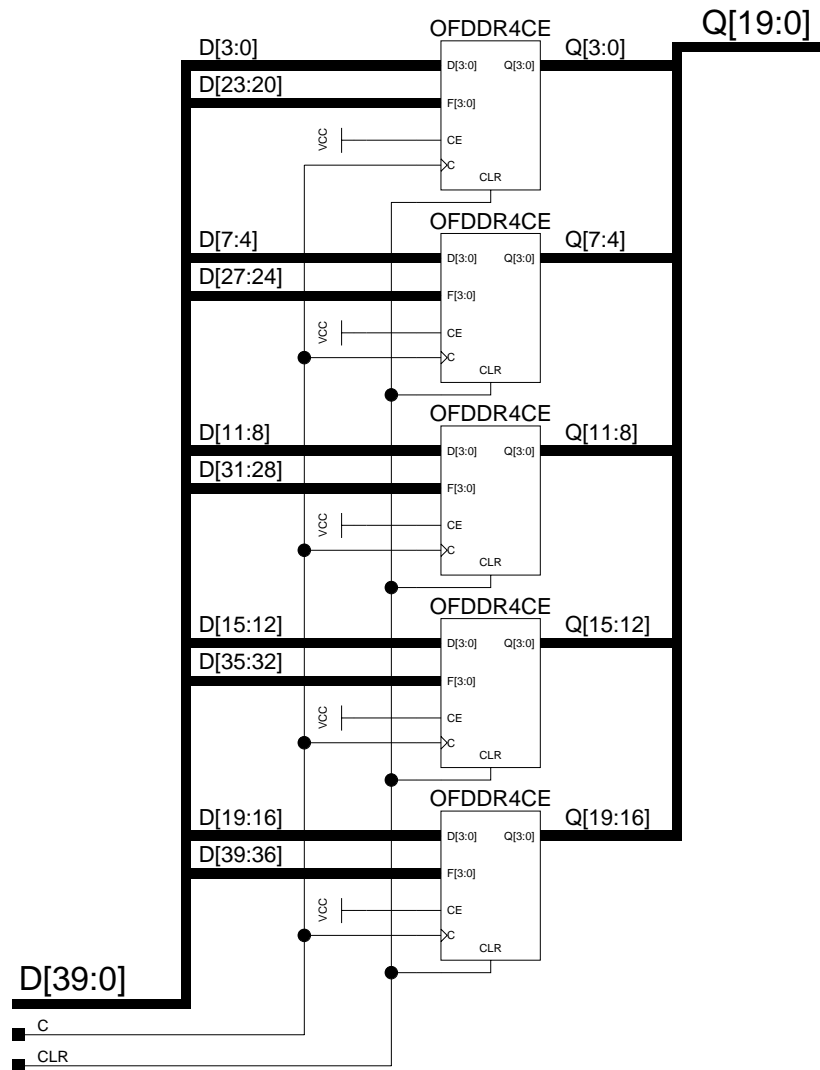
		JRG
Title:	FAST12B1	
Comments:	Custom Fast, Complex Logic for DDU, use 4 MUXCY as OR, 2 as AND similar to: OR of 4 AND2 ,AND, AND2B1 ,AND, AND2	
Date:	15th October 2003	Ver: 1
Sheet Size:	B	Rev: A



JRG

Title:	FAST8B2	Ver:	1
Comments:	Custom Fast, Complex Logic for DDU, use 2 MUXCY as AND similar to: OR4 AND SOP4B2b		
Date:	18th December 2003	Ver:	1
Sheet Size: B		Rev:	A

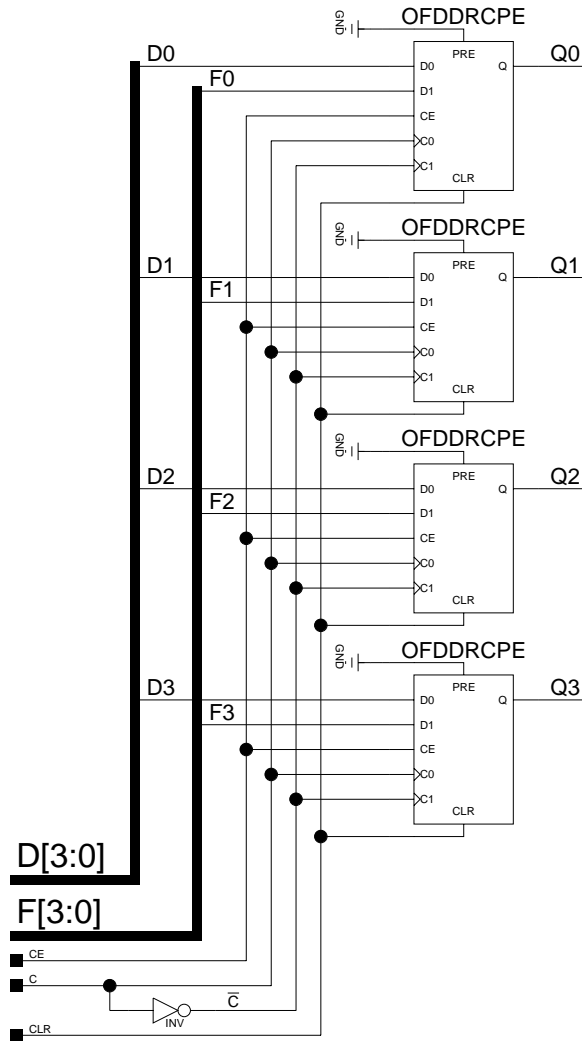
the D byte goes out last (on C rising edge)
 the F byte goes out first (on C falling edge)

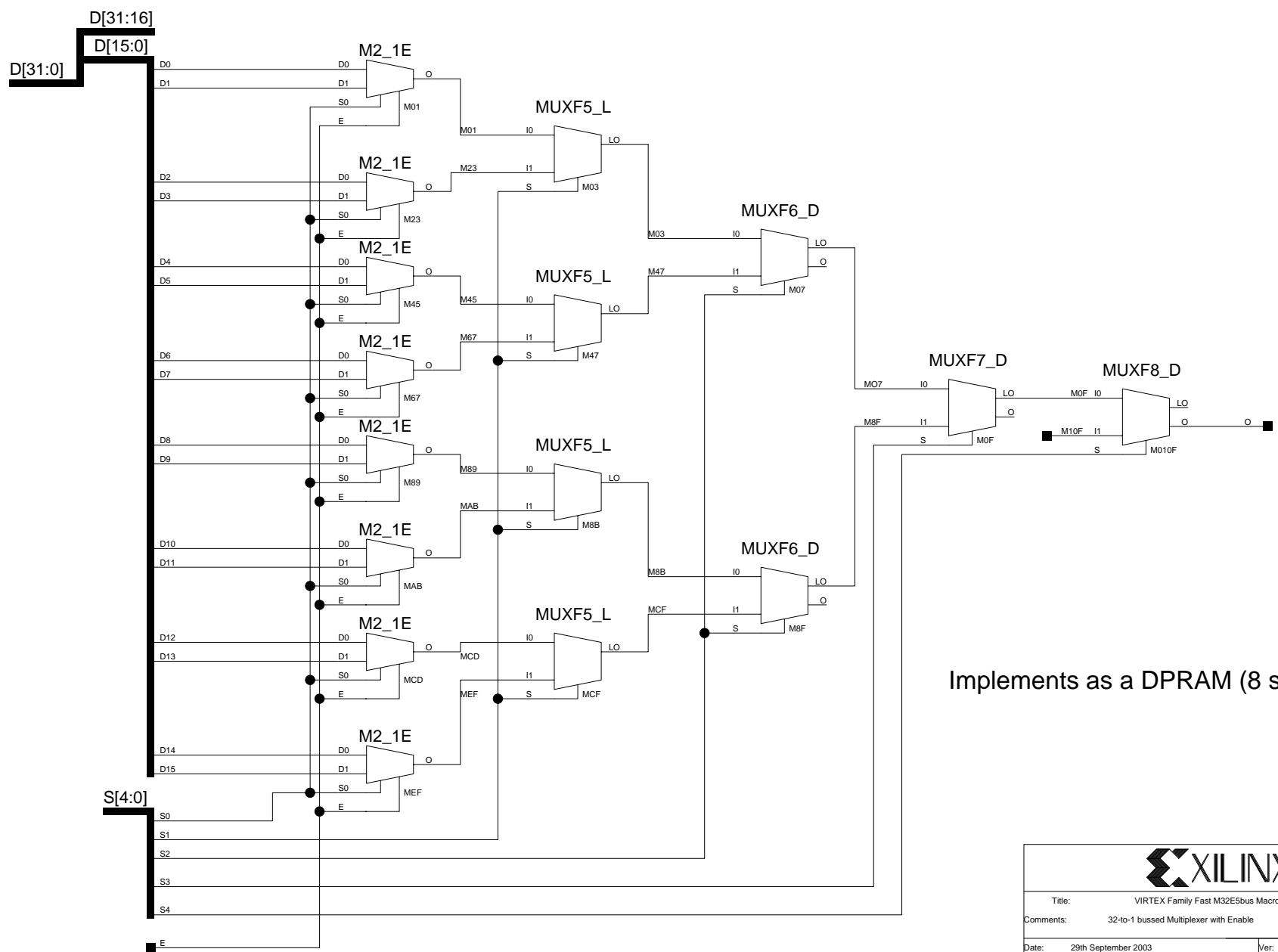


Title: VIREX Family OFDDR40 Macro	
Comment: 40-Bit Output DDR Flip-Flop w/Asynchronous Clear	
Date: 16th December 2003	Ver: 1
Sheet Size: B	Rev: A

the D byte goes out last (on C rising edge)
 the F byte goes out first (on C falling edge)

Q[3:0]



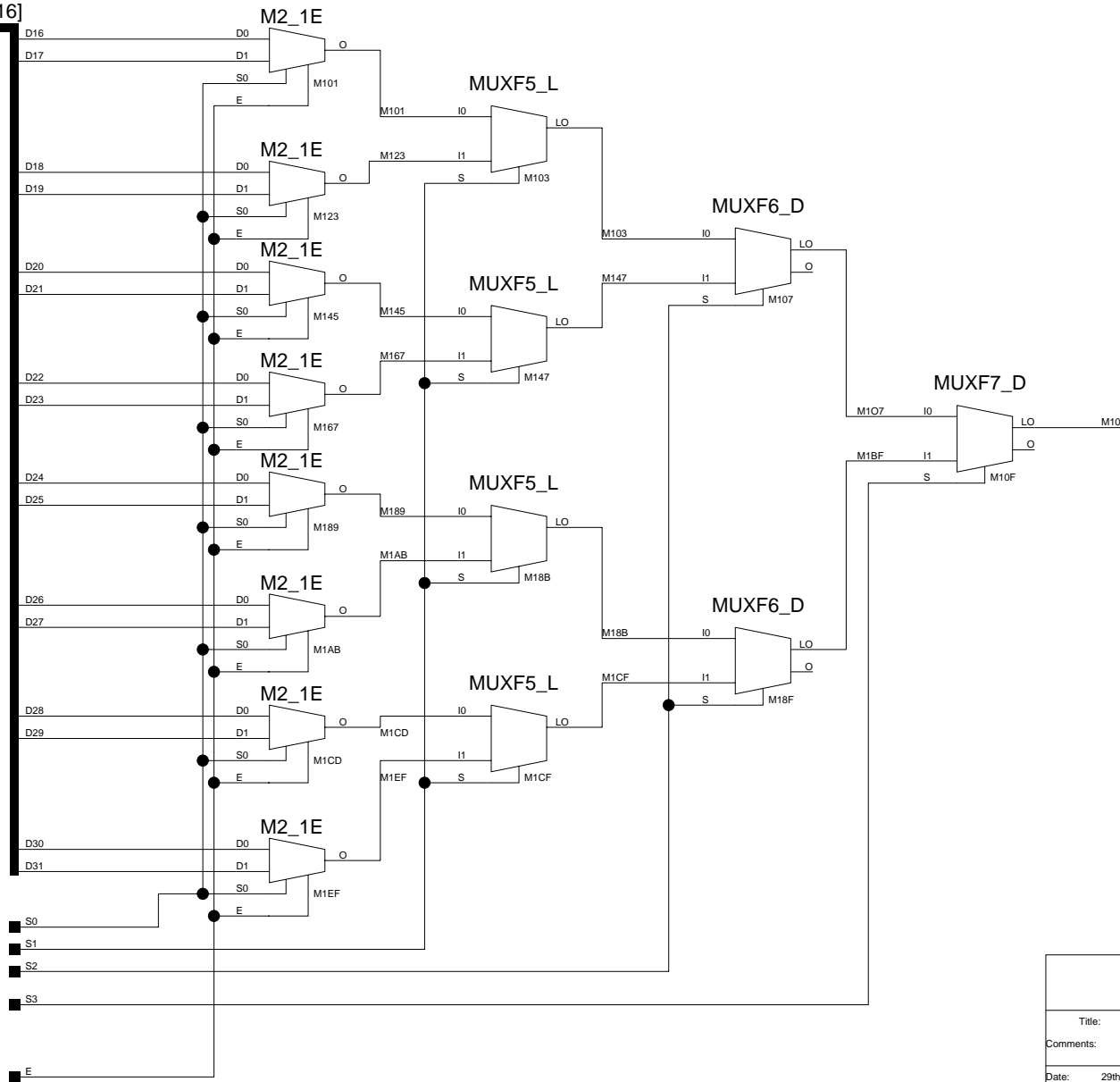


Implements as a DPRAM (8 slices)

drawn by KS
Copyright (c) 1993, Xilinx Inc.

		JRG
Title: VIRTEx Family Fast M32E5bus Macro		
Comments: 32-to-1 bus Multiplexer with Enable		
Date: 29th September 2003	Ver: 1	
Sheet Size: B	Rev: A	

D[31:16]



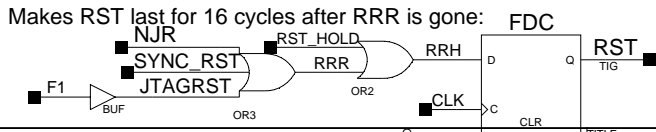
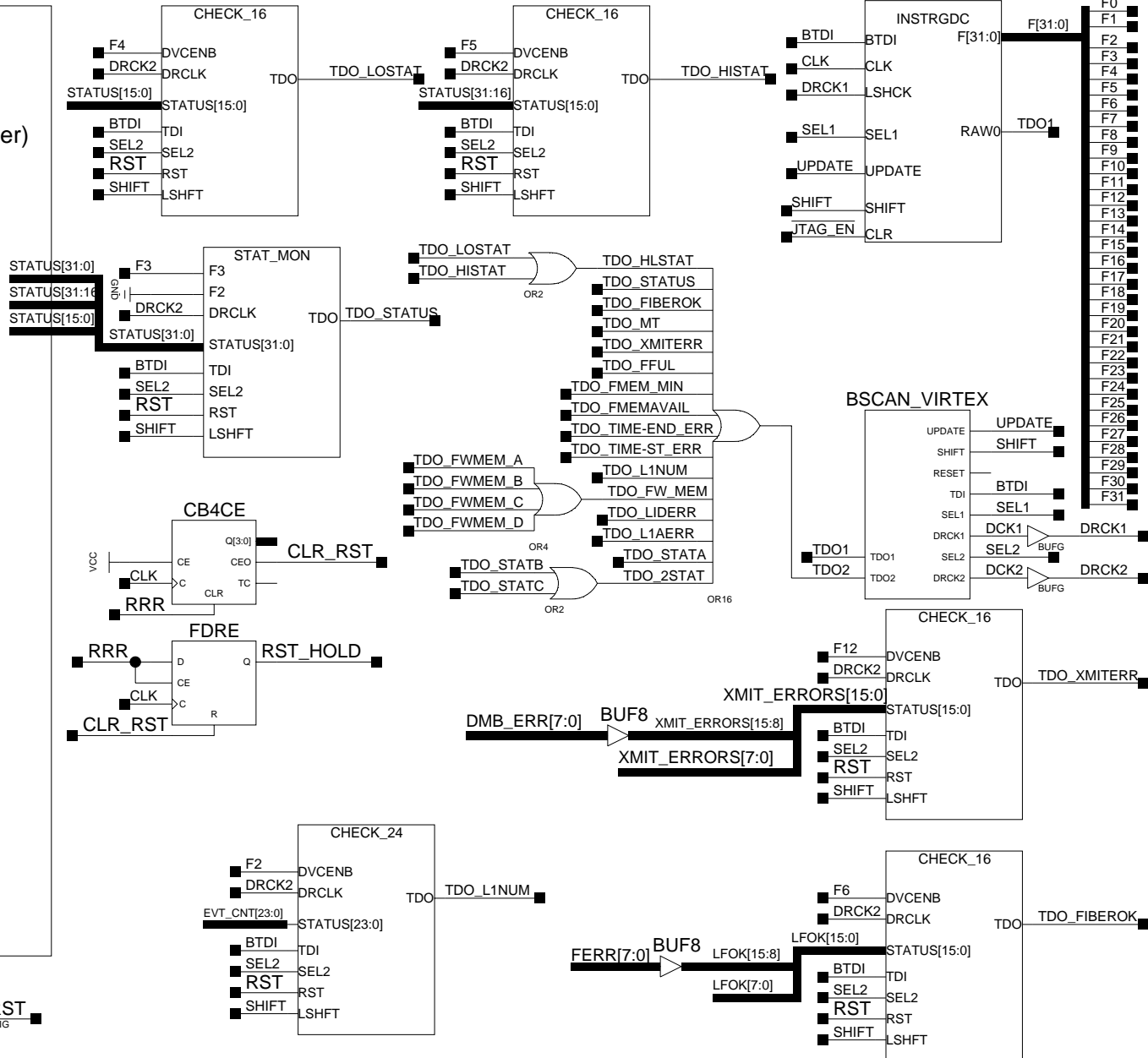
drawn by KS
Copyright (c) 1993, Xilinx Inc.

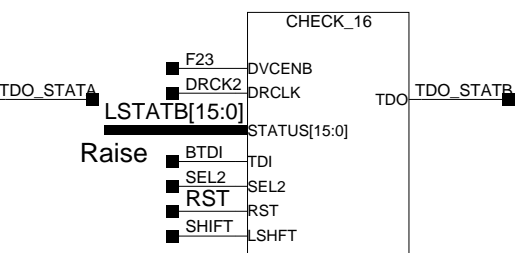
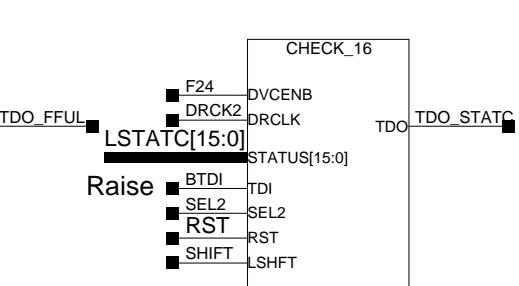
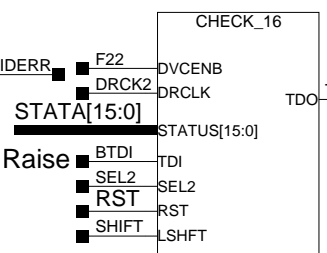
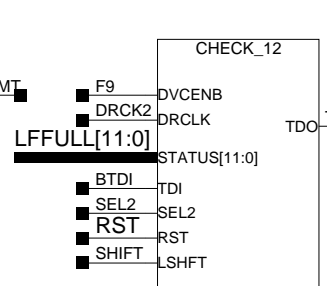
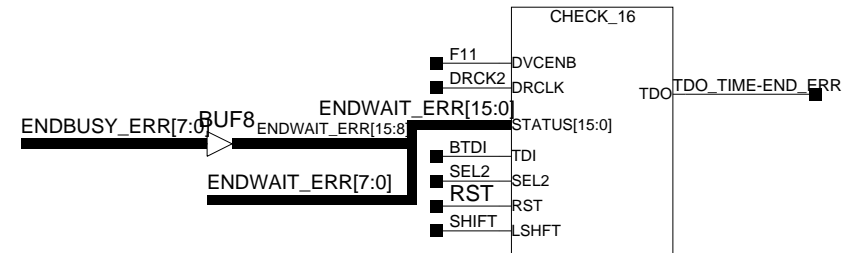
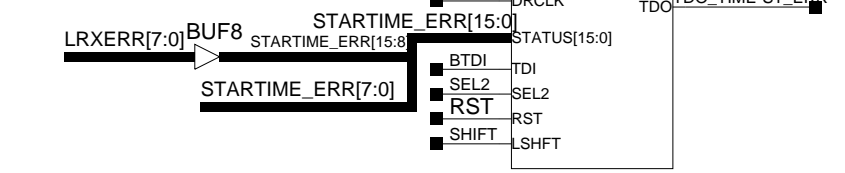
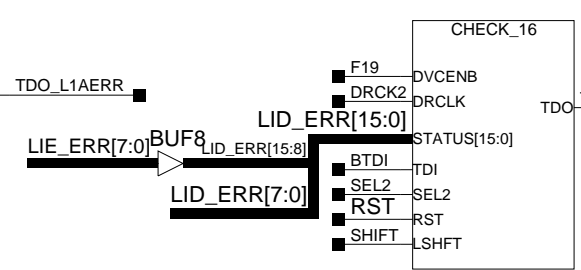
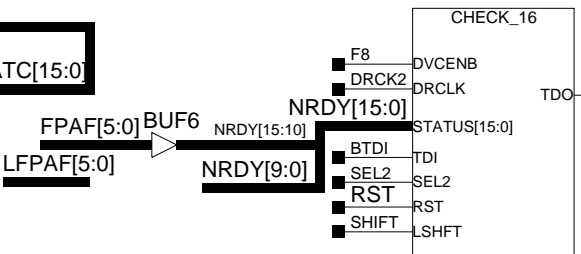
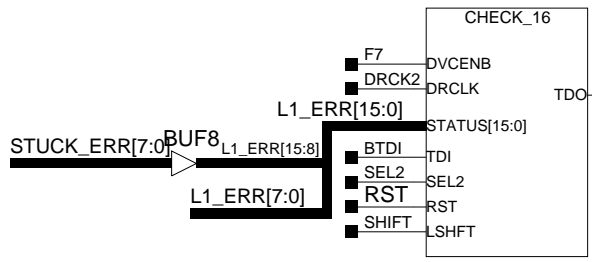
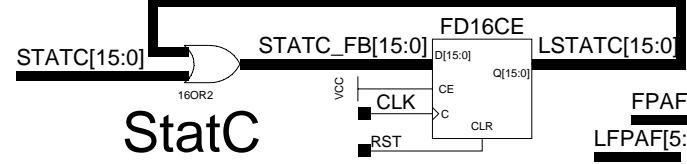
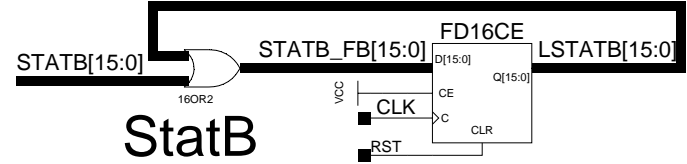
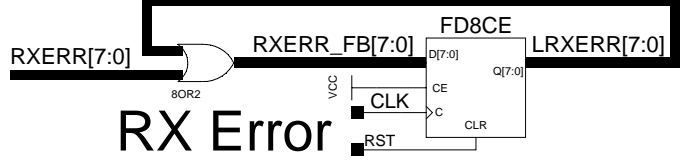
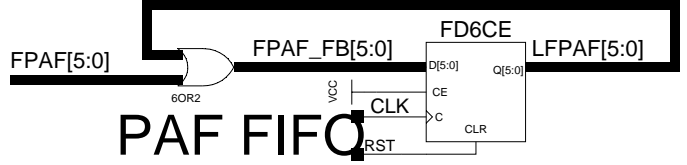
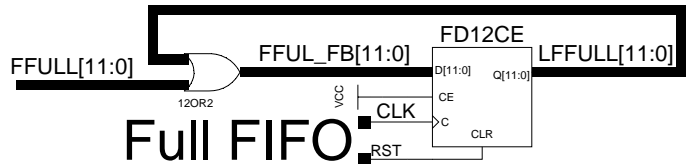


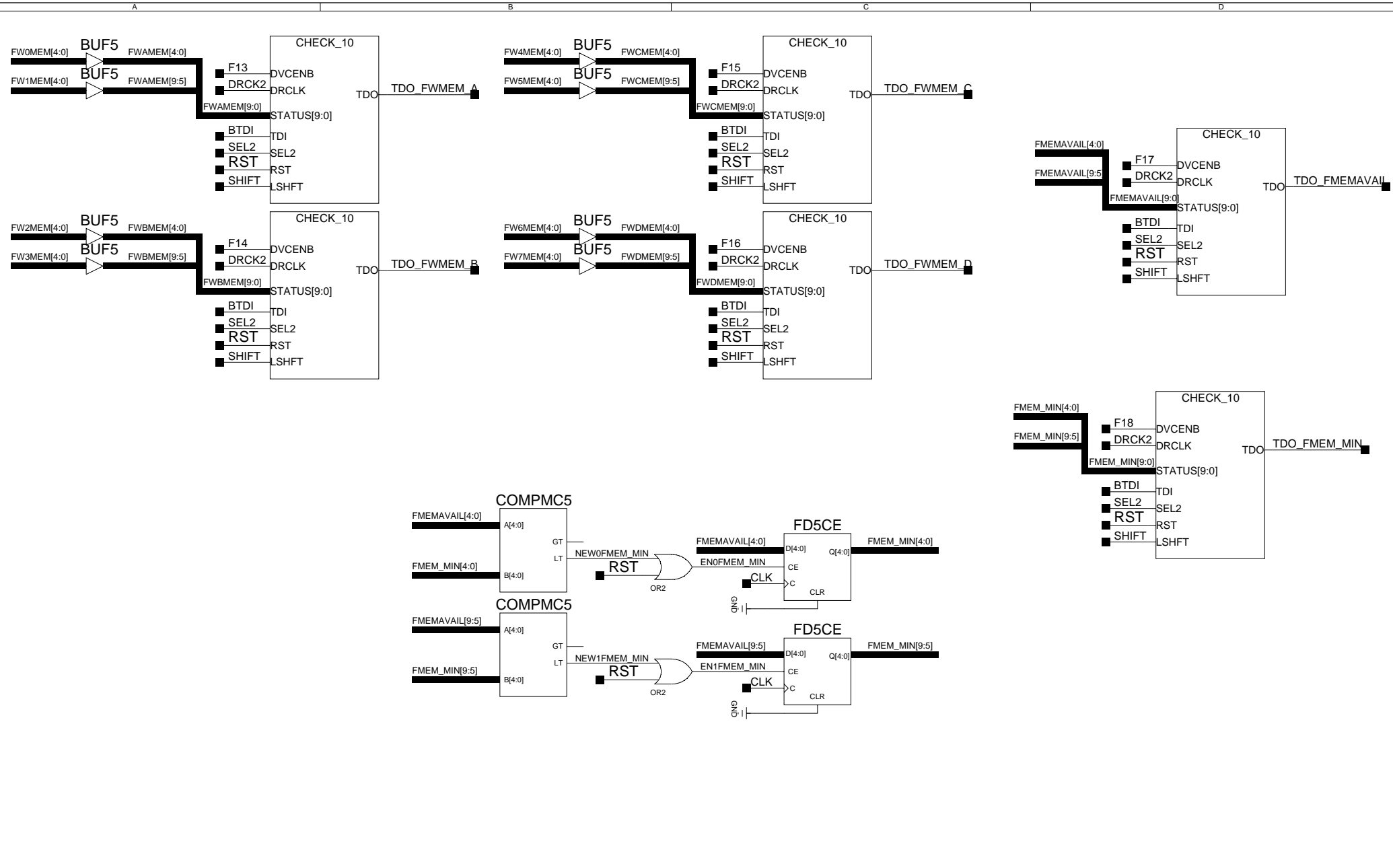
Title:	VIRTEX Family Fast M32E5bus Macro	
Comments:	32-to-1 bussed Multiplexer with Enable	
Date:	29th September 2003	Ver: 1
Sheet Size:	B	Rev: A

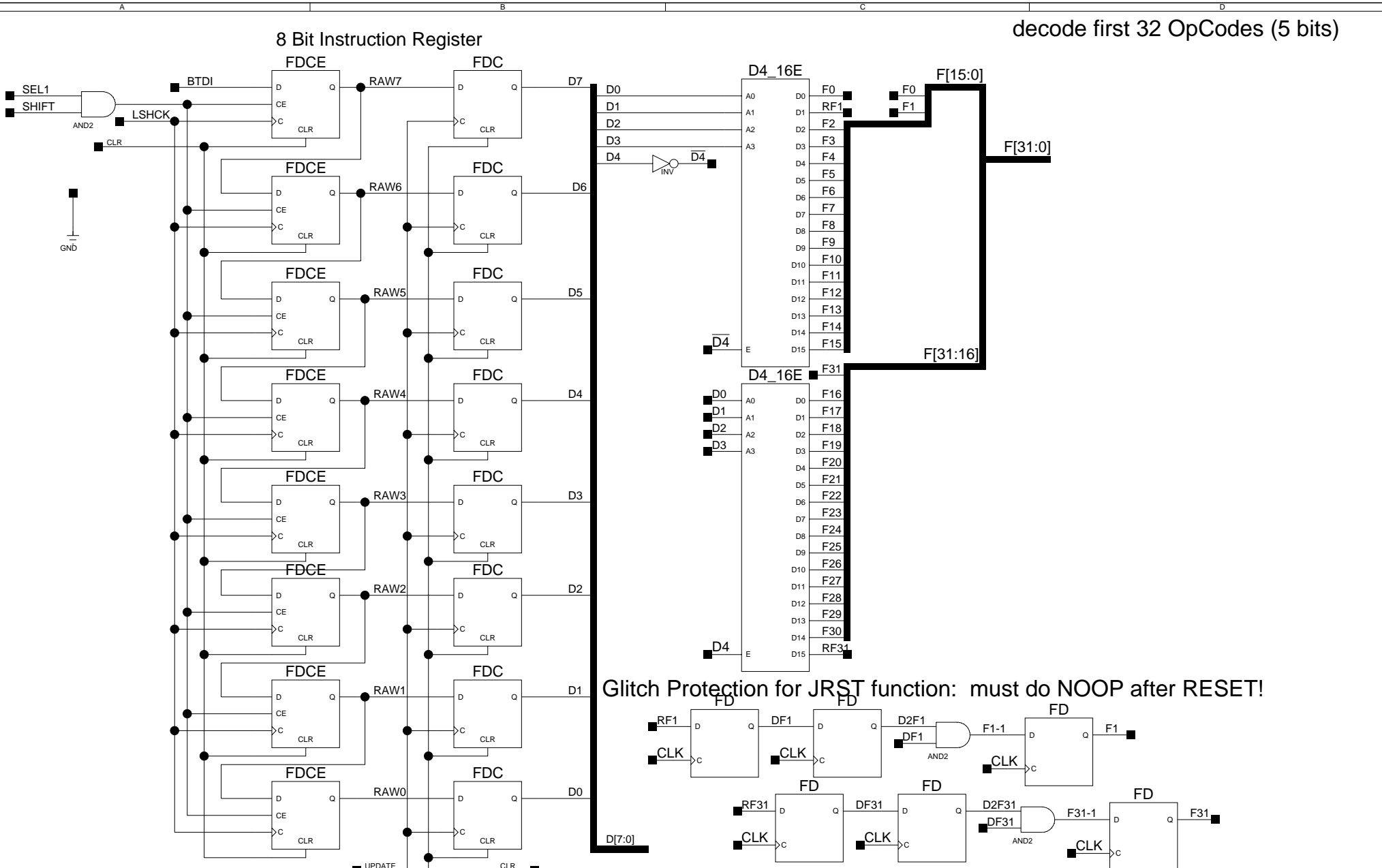
JTAG Instruction Decode

OpCode	Function [OpName]
0	No Operation [NOOP]
1	FPGA Reset [toggle]
2	Read Current DDU L1A Number (24-bit scaler)
3	Check status (capture and shift) [32 bits]
4	Check status, low-word [16 bits]
5	Check status, high-word [16 bits]
7	6a Check FiberOK status (Check_FOK) [8 bits]
6	6b Check FiberErr status (Check_Ferr) [8 bits]
11	7a L1A Mismatch [8-bits]
26	7b Stuck Buffer Errors [8-bits]
25	8a FIFO Empty Status [10-bits]
20	8b Almost Full FIFOs [6-bits]
21	9 Full FIFOs [12-bits]
13	10a Timeout: start [8-bits]
17	10b RX Errors [8-bits]
14	11a Timeout: end-wait [8-bits]
15	11b Timeout: end-active [8-bits]
12	12a Data Xmit Errors [8-bits]
16	12b DMB Errors [8-bits]
13	Fiber 1 & 0 Write Memory [10-bits]
14	Fiber 3 & 2 Write Memory [10-bits]
15	Fiber 5 & 4 Write Memory [10-bits]
16	Fiber 7 & 6 Write Memory [10-bits]
17	FIFO Mem Available 1 & 0 [10-bits]
18	FIFO Mem Min Available 1 & 0 [10-bits]
19	*19a Lost In Data [8-bits]
18	*19b Lost In Event [8-bits]
22	22 Status Register A [16-bits]
23	23 Status Register B [16-bits]
24	24 Status Register C [16-bits]





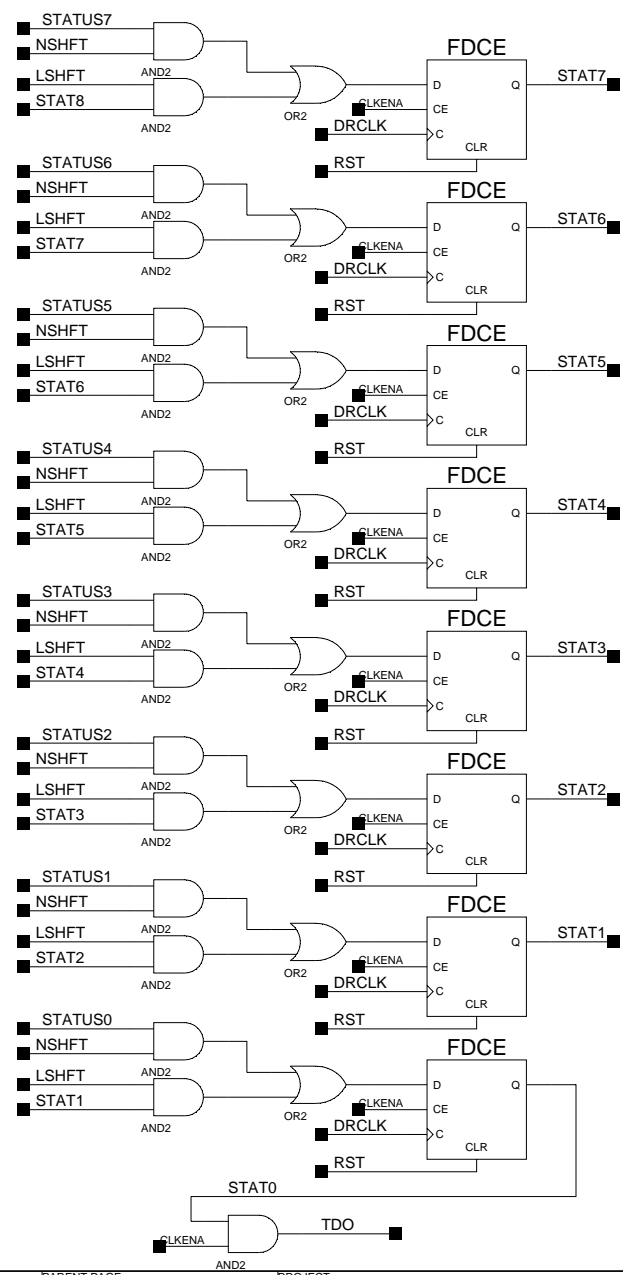
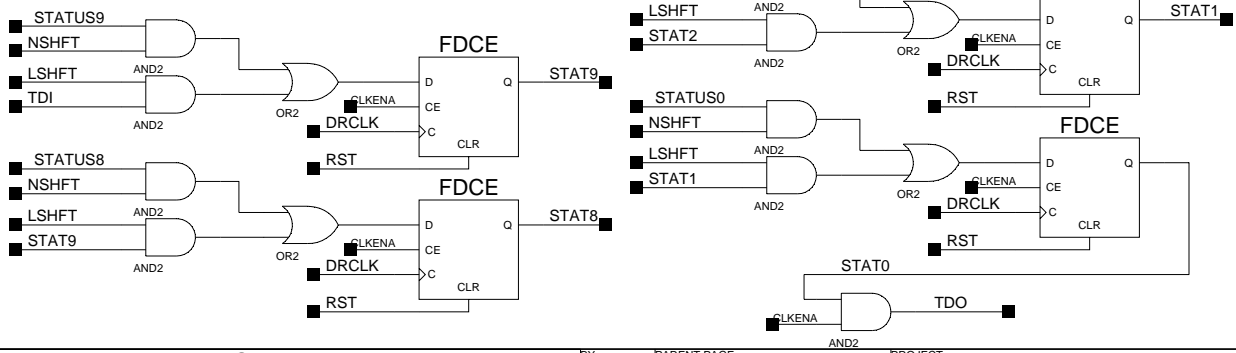
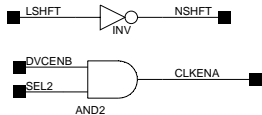




10-bit JTAG Register Read out (on DVCENB)

STATUS[9:0]

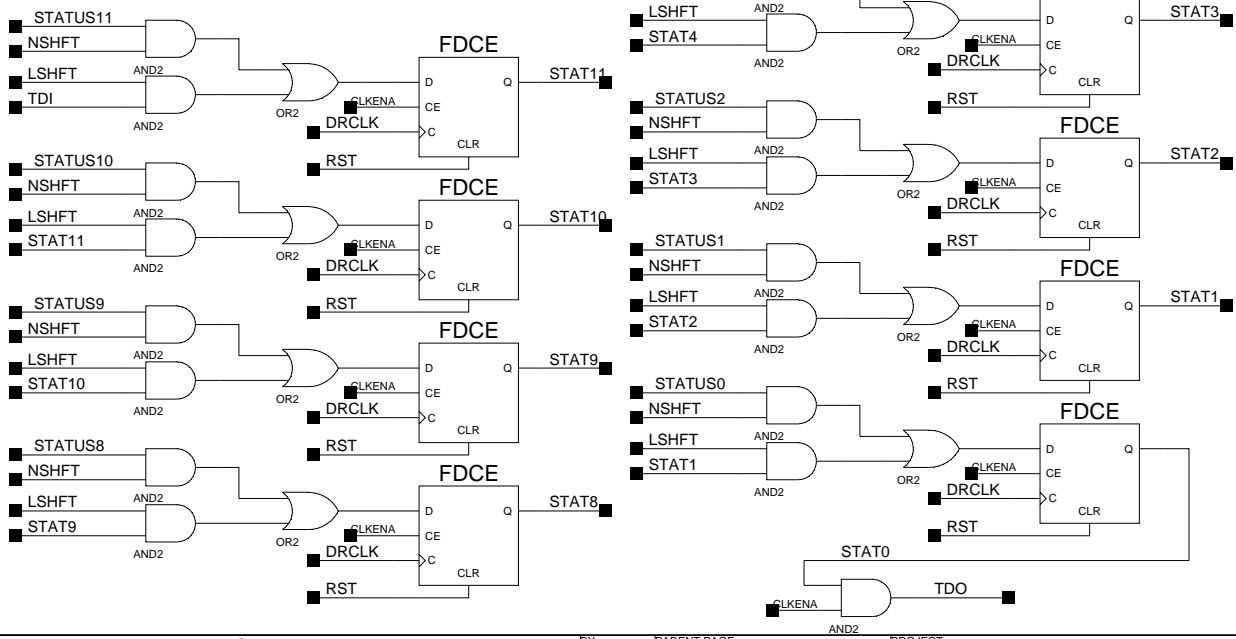
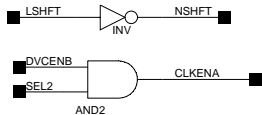
- STATUS9
- STATUS8
- STATUS7
- STATUS6
- STATUS5
- STATUS4
- STATUS3
- STATUS2
- STATUS1
- STATUS0



12-bit JTAG Register Read out (on DVCENB)

STATUS[11:0]

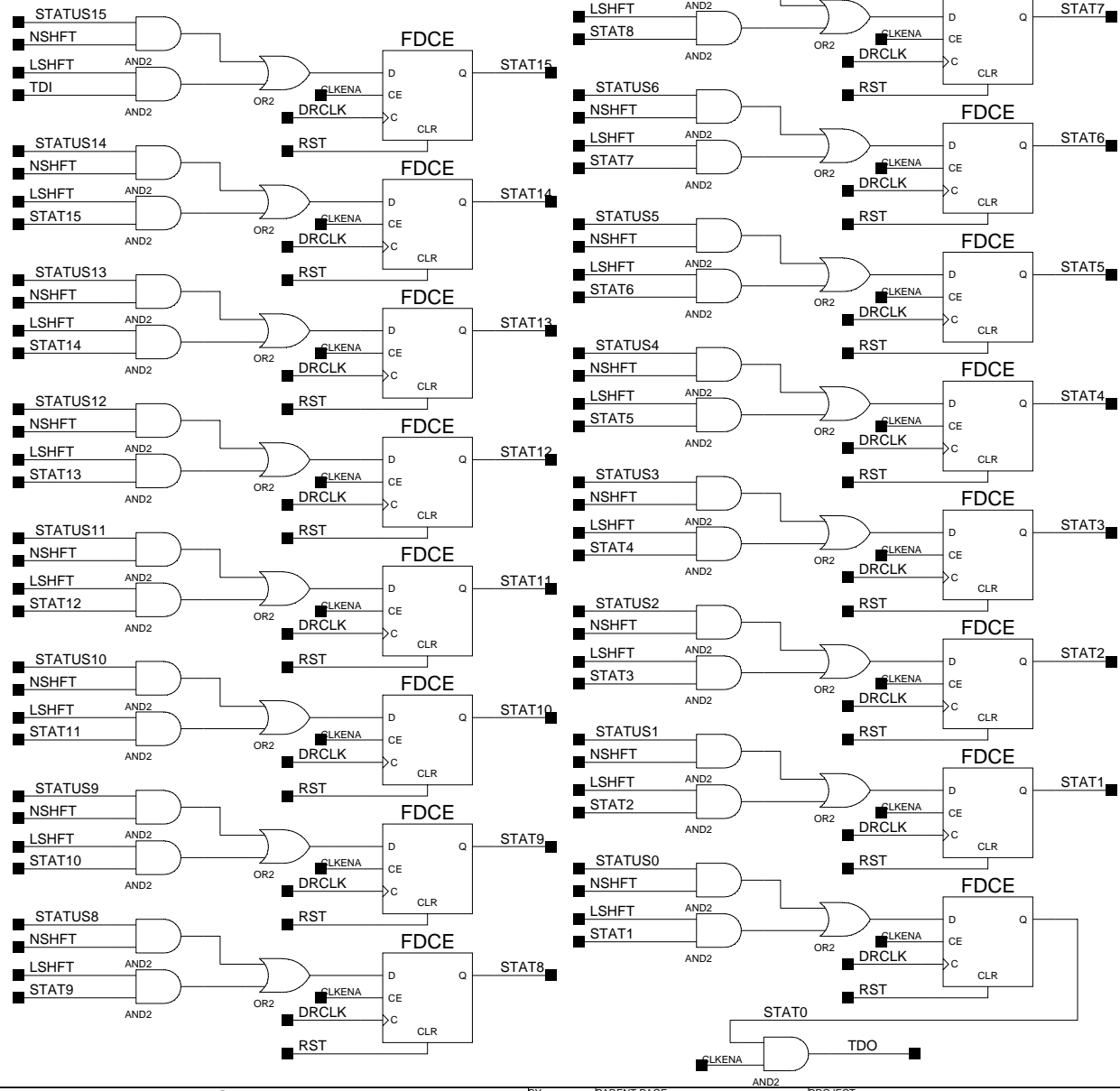
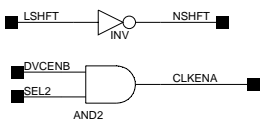
- STATUS11
- STATUS10
- STATUS9
- STATUS8
- STATUS7
- STATUS6
- STATUS5
- STATUS4
- STATUS3
- STATUS2
- STATUS1
- STATUS0



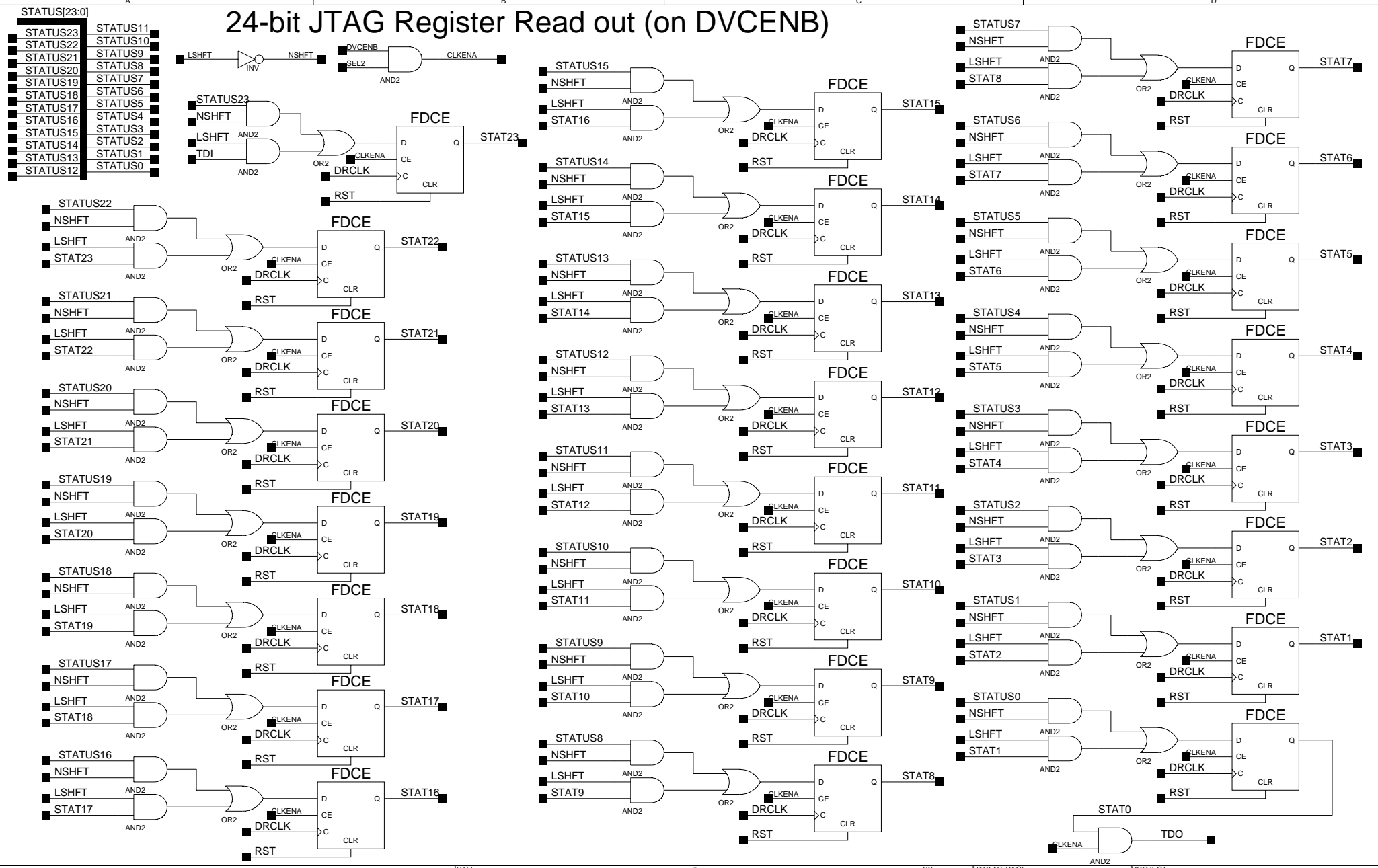
16-bit JTAG Register Read out (on DVCENB)

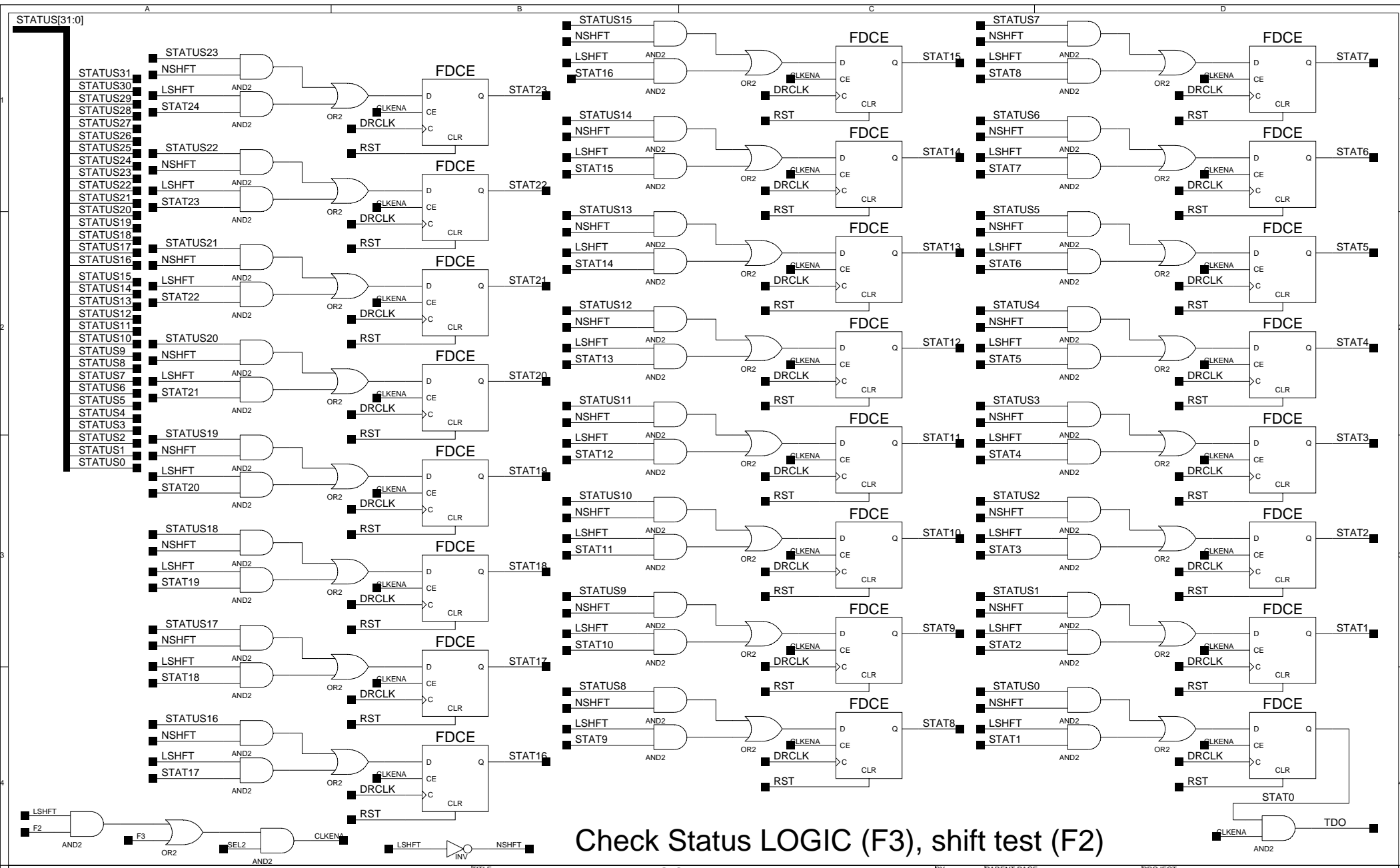
STATUS[15:0]

- STATUS15
- STATUS14
- STATUS13
- STATUS12
- STATUS11
- STATUS10
- STATUS9
- STATUS8
- STATUS7
- STATUS6
- STATUS5
- STATUS4
- STATUS3
- STATUS2
- STATUS1
- STATUS0

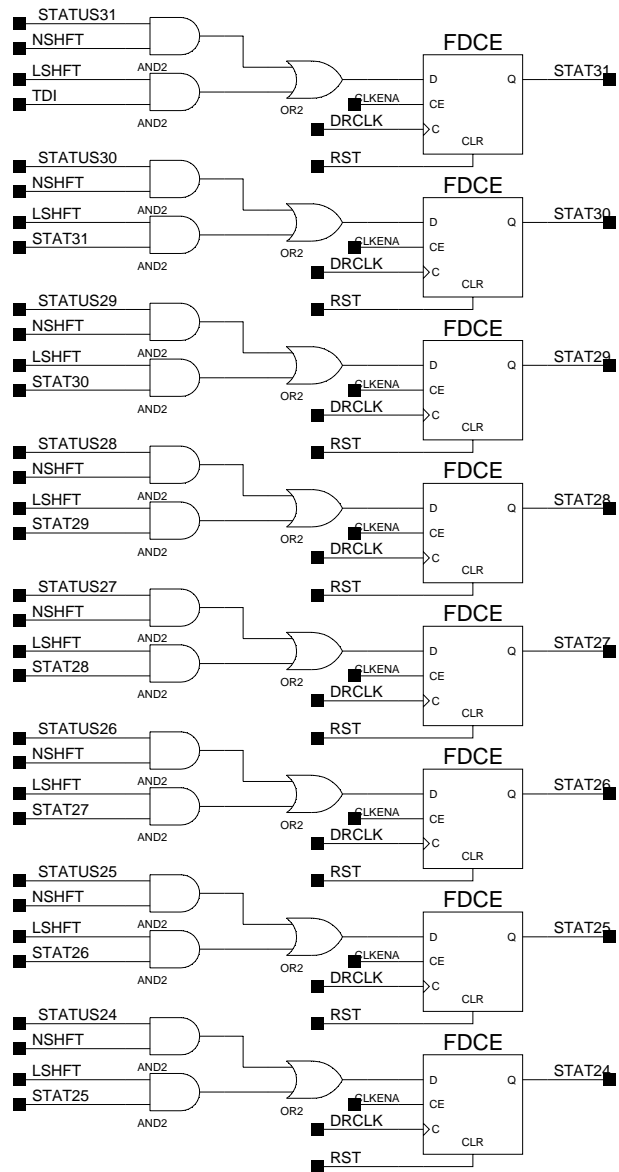


24-bit JTAG Register Read out (on DVCENB)





Check Status LOGIC (F3), shift test (F2)

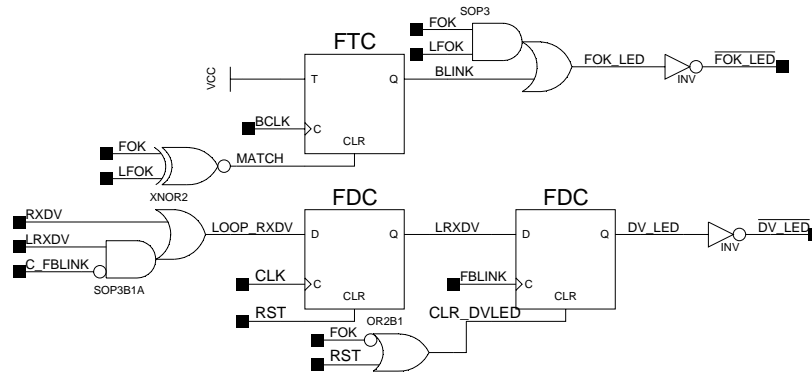


FOK LED

- LIT == Link is alive and well
- BLINK == Link not ready
- OFF == Link not present

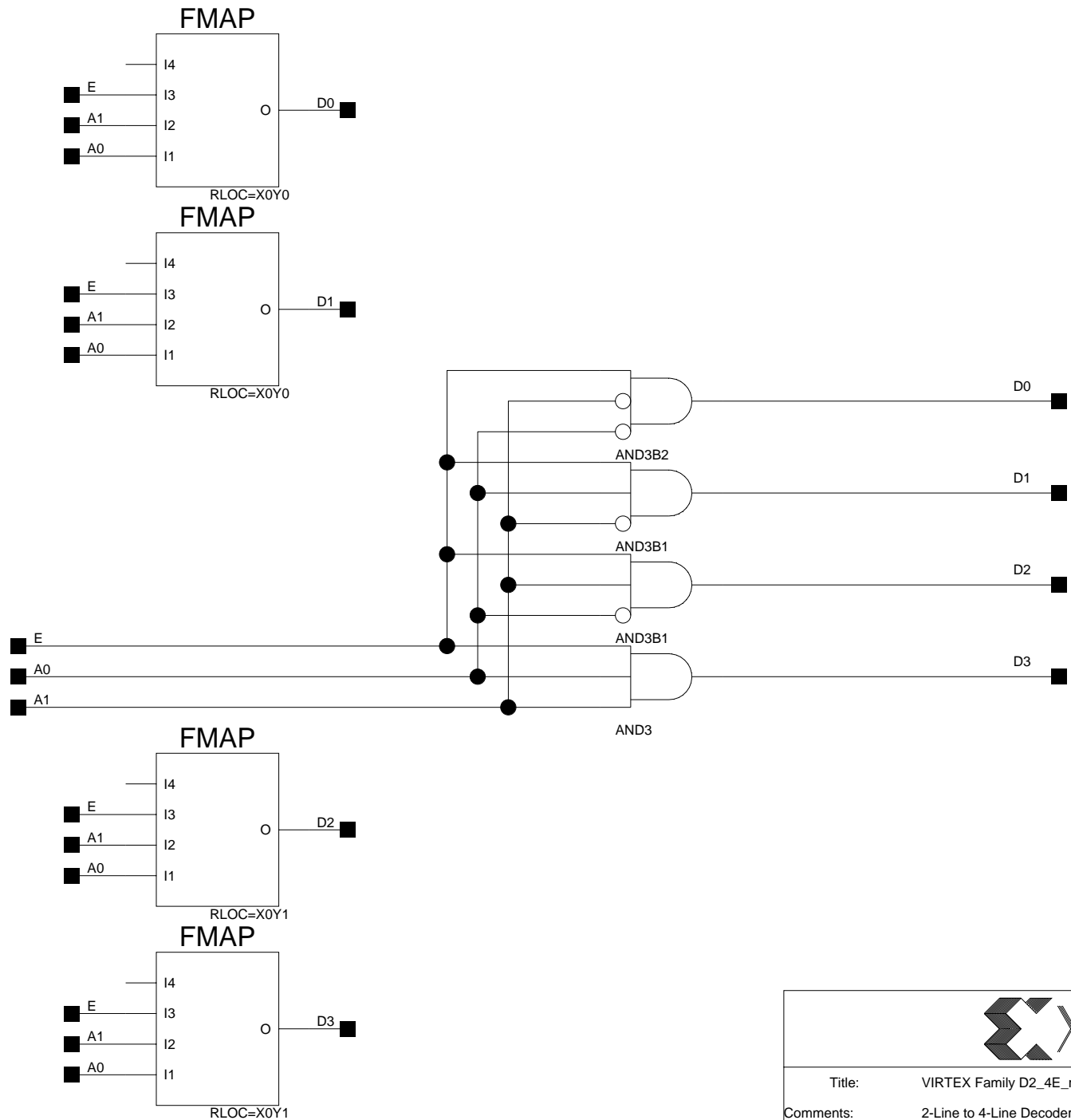
DAV LED

- LIT == Active Data Xmit
- OFF == No data to Xmit



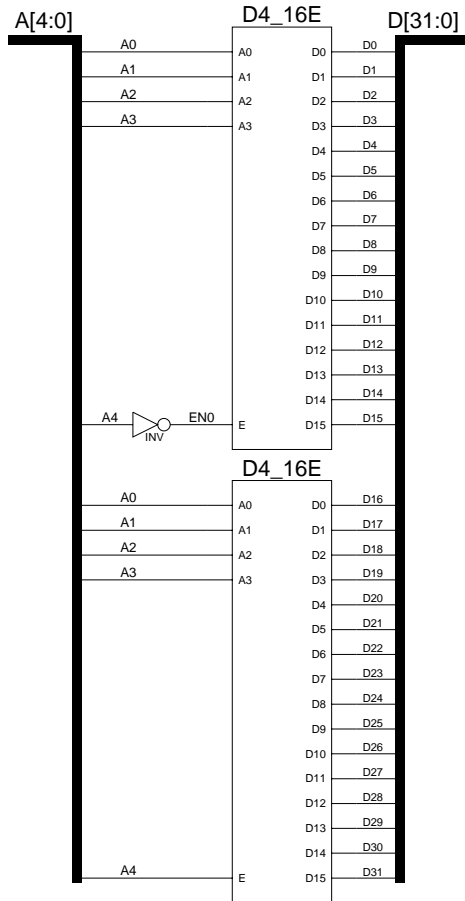
JRG

Title:	FIBERLED	
Comments:	Custom LED Slow-Blink Control for Fiber Inputs	
Date:	27th January 2004	Ver: 1
Sheet Size: B		Rev: A



JRG

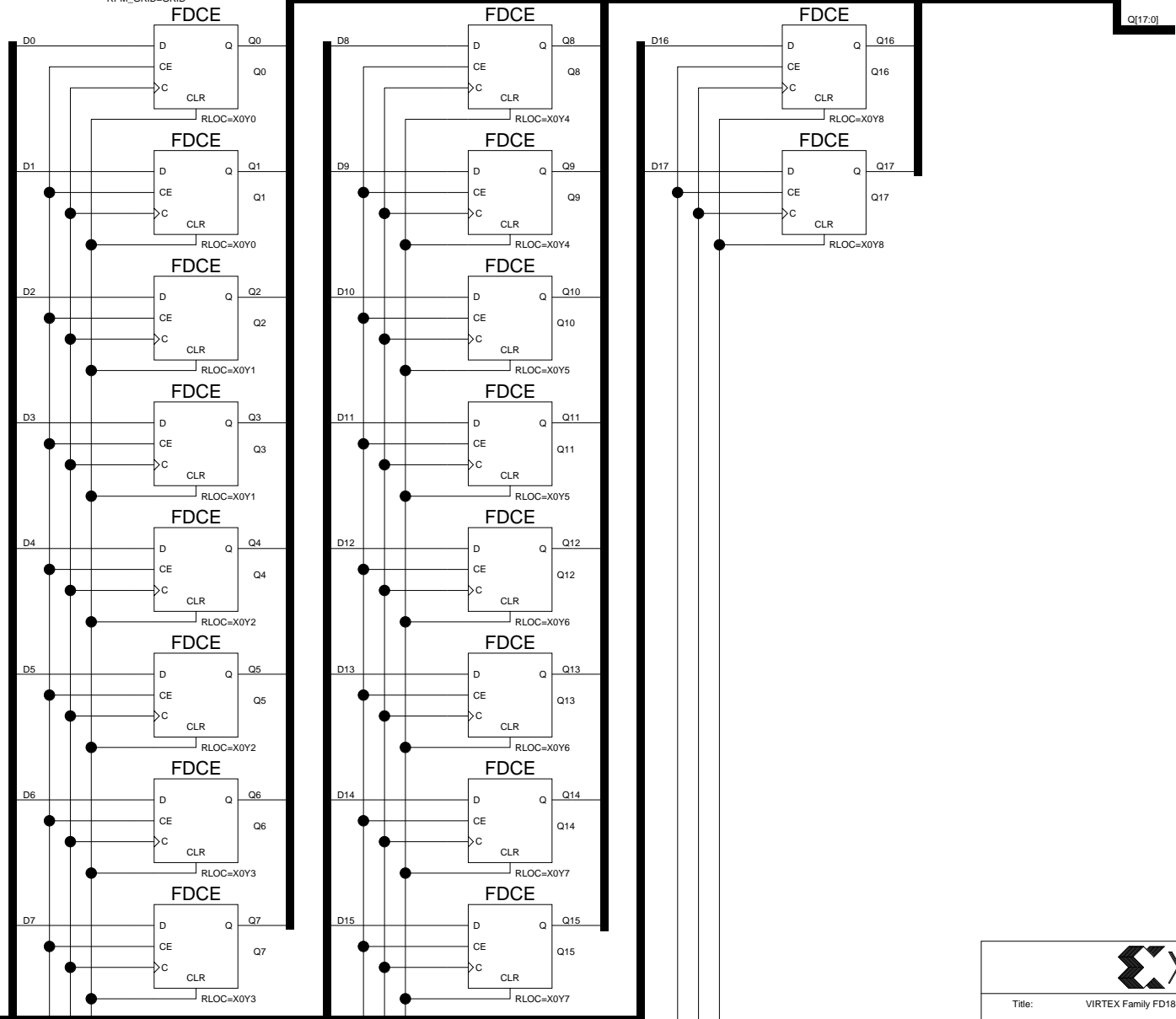
Title:	VIRTEX Family D2_4E_rpm Macro		
Comments:	2-Line to 4-Line Decoder/ Demultiplexer w/ Enable, RPM		
Date:	11th November 2003	Ver:	1
Sheet Size:	A	Rev:	A



Title: VIRTEX Family Decode 32E Macro		JRG
Comments: 5-bit to 32-bit Decoder		
Date: 30th September 2003	Ver: 1	
Sheet Size: B	Rev: A	

drawn by KS
Copyright (c) 1993, Xilinx Inc.

RPM_GRID=GRID



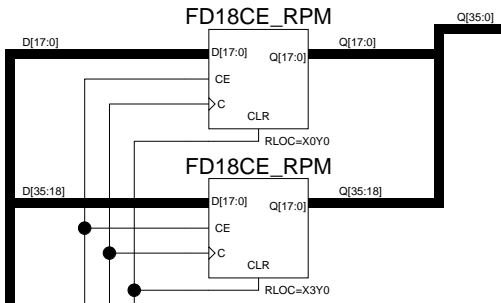
D[17:0]

- CE
- C
- CLR



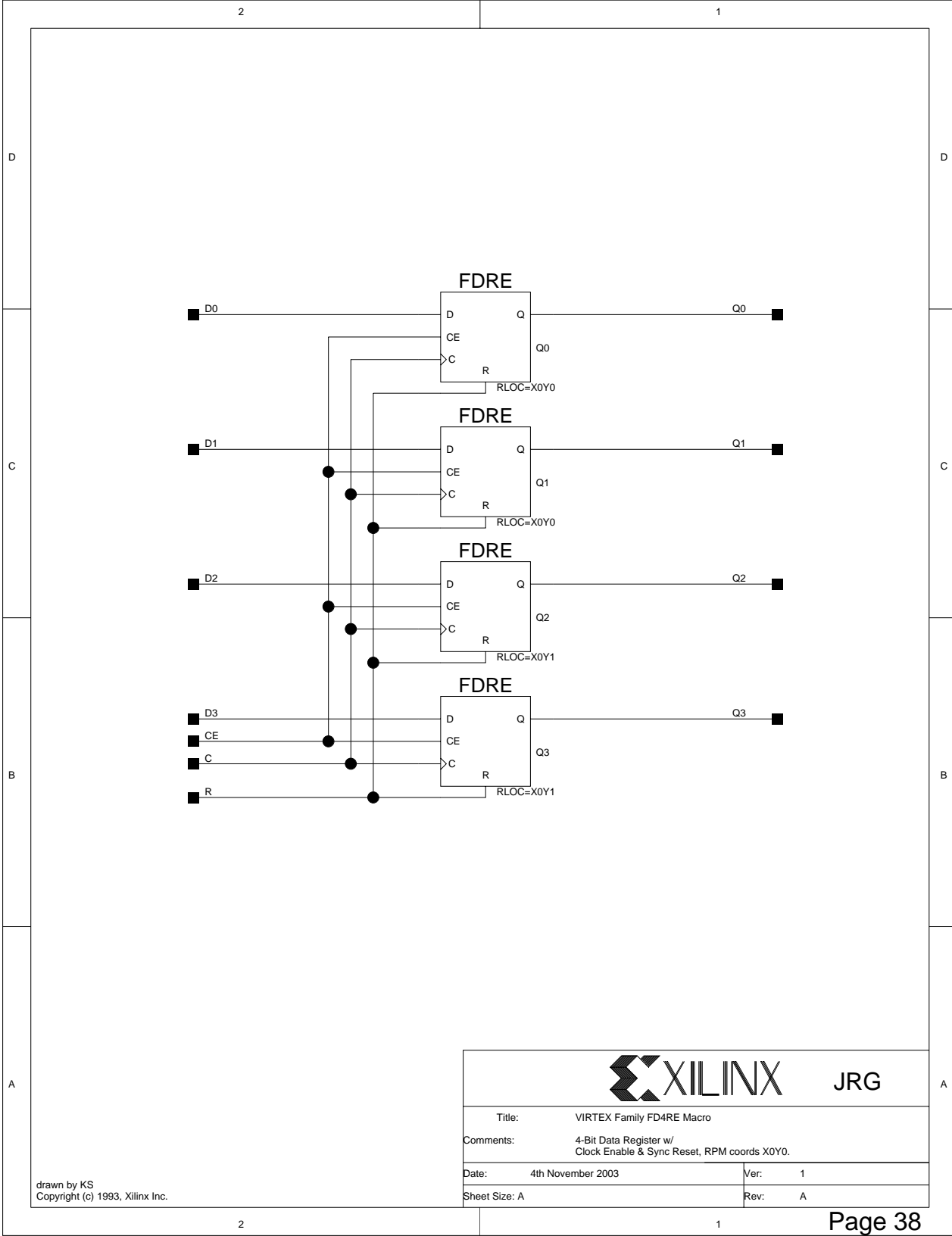
Title:	VIRTEX Family FD18CE_RPM Macro		JRG
Comments:	18-Bit Data Register w/ Clock Enable & Asynchronous Ctr, GRID Coords X0Y0-X0Y8		
Date:	29th October 2003	Ver:	1
Sheet Size:	B	Rev:	A

drawn by KS
Copyright (c) 1993, Xilinx Inc.




Title: VIRTEX Family FD36CE_RPM Macro	
Comments: 36-Bit Data Register w/ Clock Enable & Asynchronous Clr, GRID Coords X0Y0-X3Y8	
Date: 30th October 2003	Ver: 1
Sheet Size: B	Rev: A

JRG

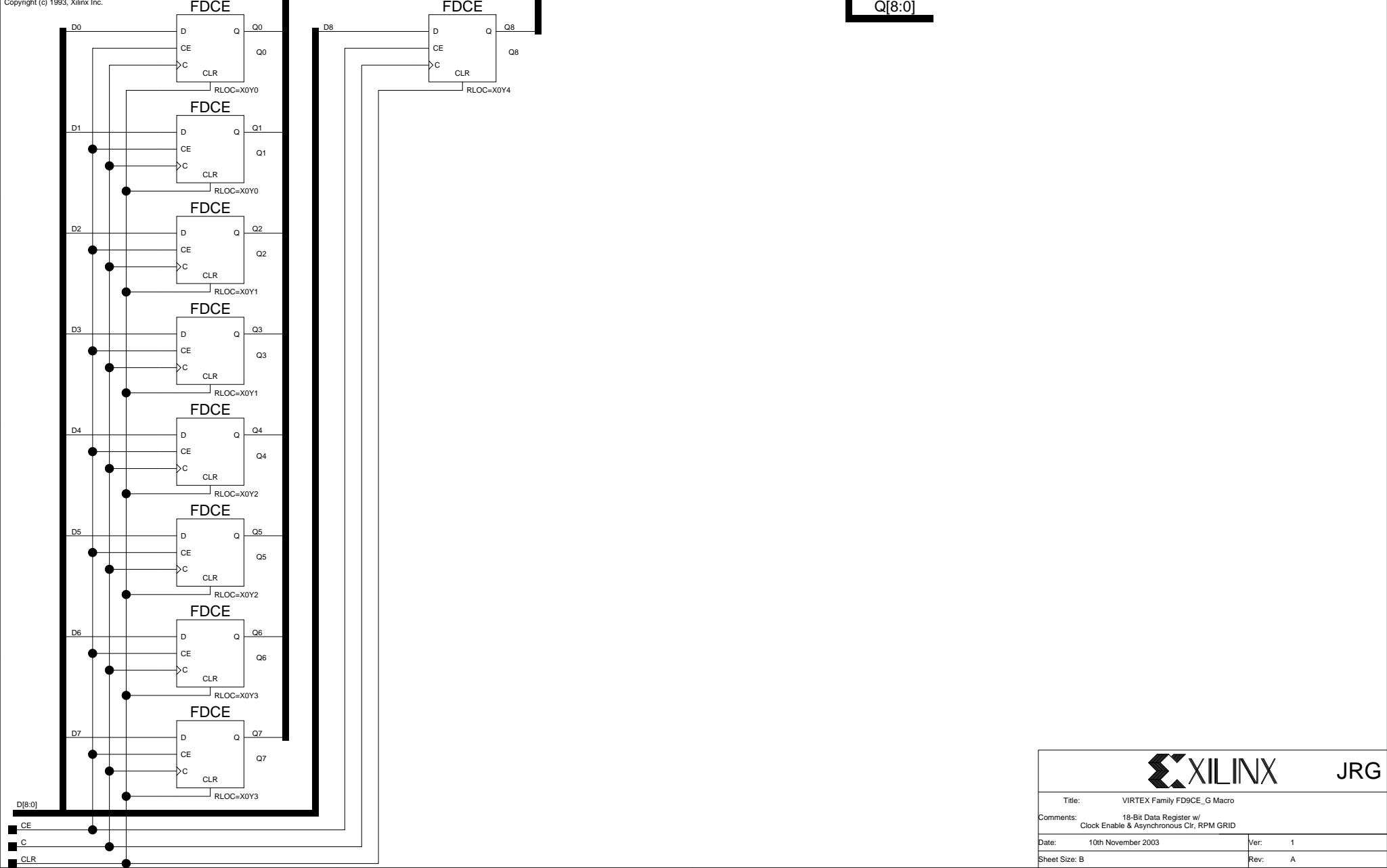


drawn by KS
Copyright (c) 1993, Xilinx Inc.

		JRG
Title: VIRTEX Family FD4RE Macro		
Comments: 4-Bit Data Register w/ Clock Enable & Sync Reset, RPM coords X0Y0.		
Date: 4th November 2003	Ver: 1	
Sheet Size: A	Rev: A	

drawn by KS
Copyright (c) 1993, Xilinx Inc.

RPM_GRID=GRID

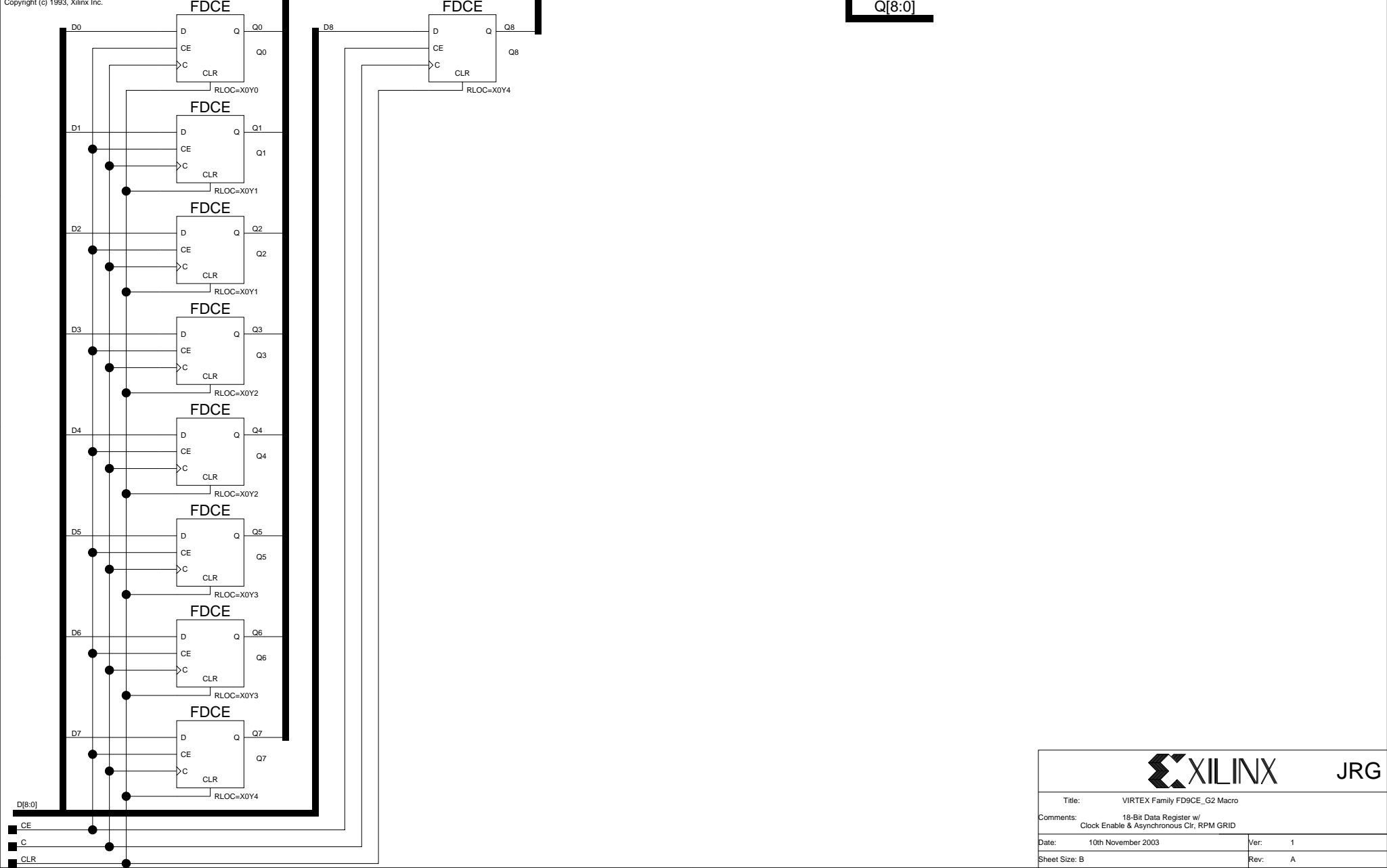


JRG

Title: VIRTEX Family FD9CE_G Macro	
Comments: 18-Bit Data Register w/ Clock Enable & Asynchronous Clr, RPM GRID	
Date: 10th November 2003	Ver: 1
Sheet Size: B	Rev: A

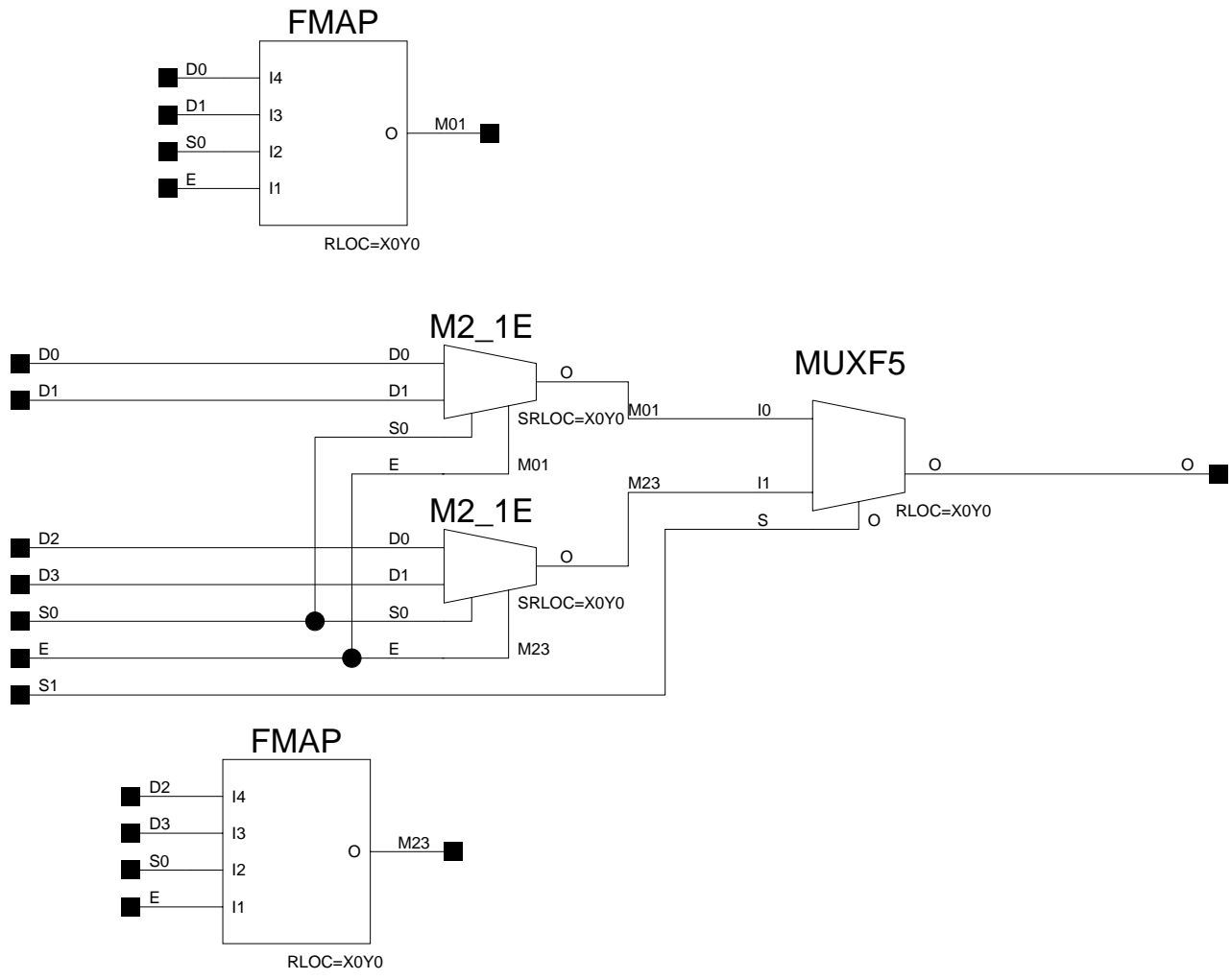
drawn by KS
Copyright (c) 1993, Xilinx Inc.

RPM_GRID=GRID

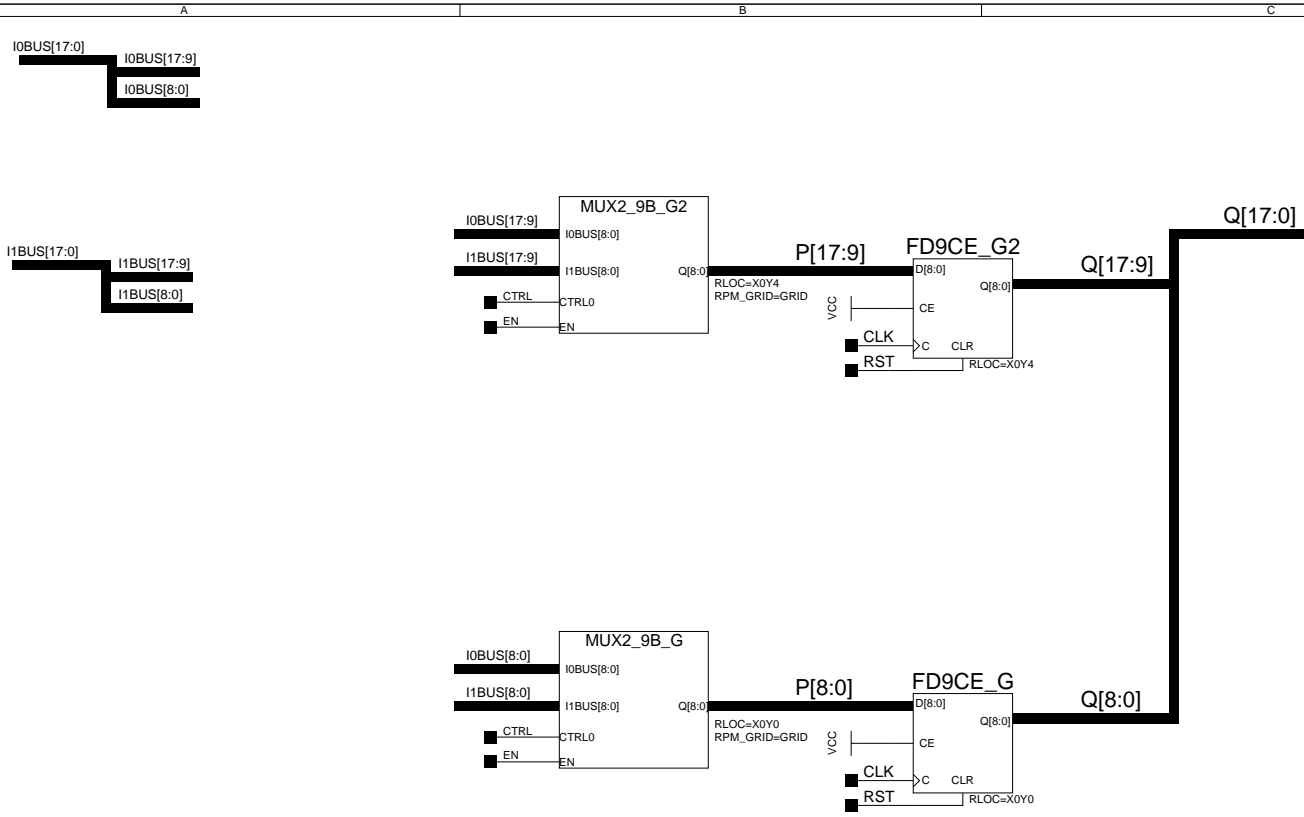


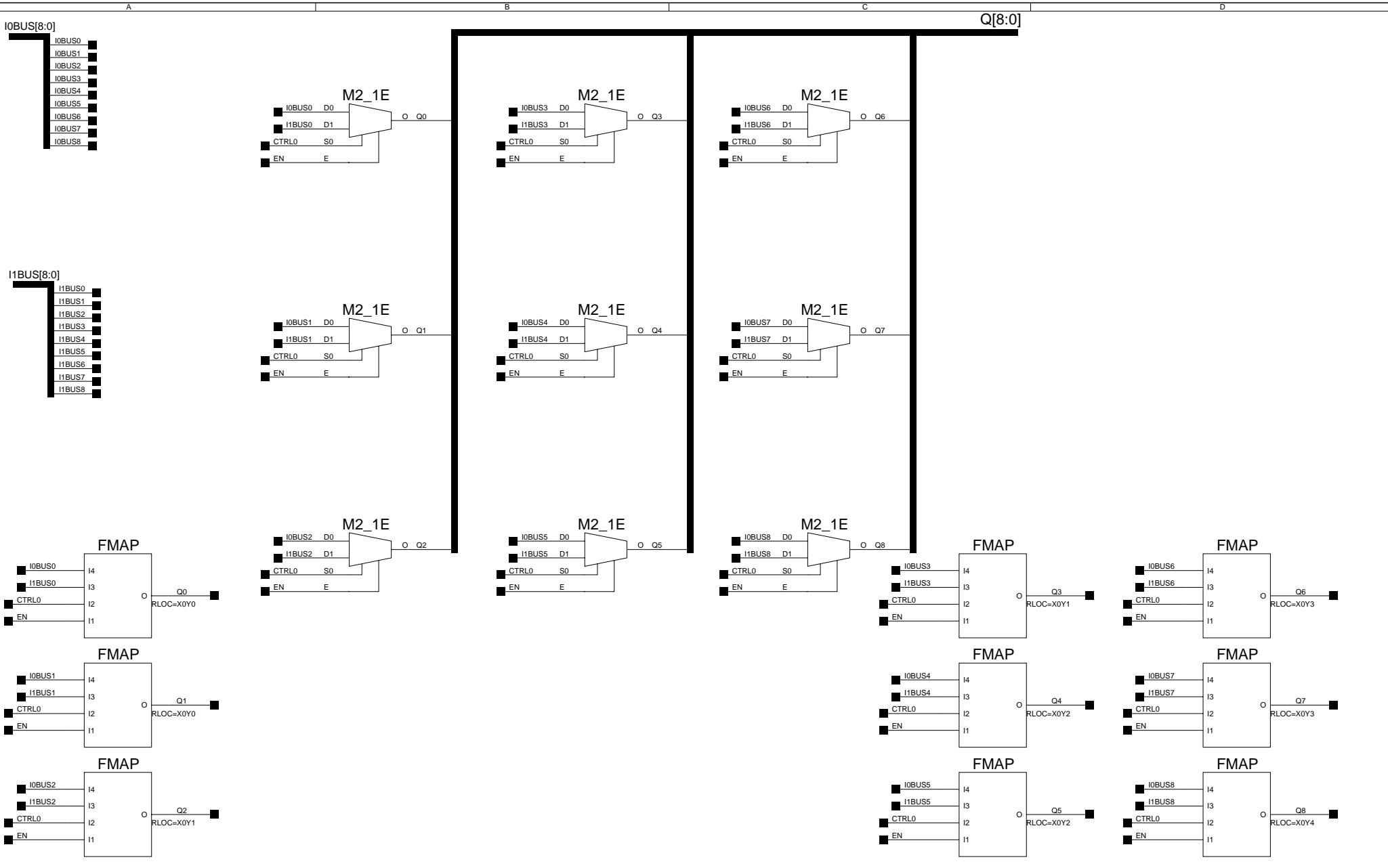
JRG

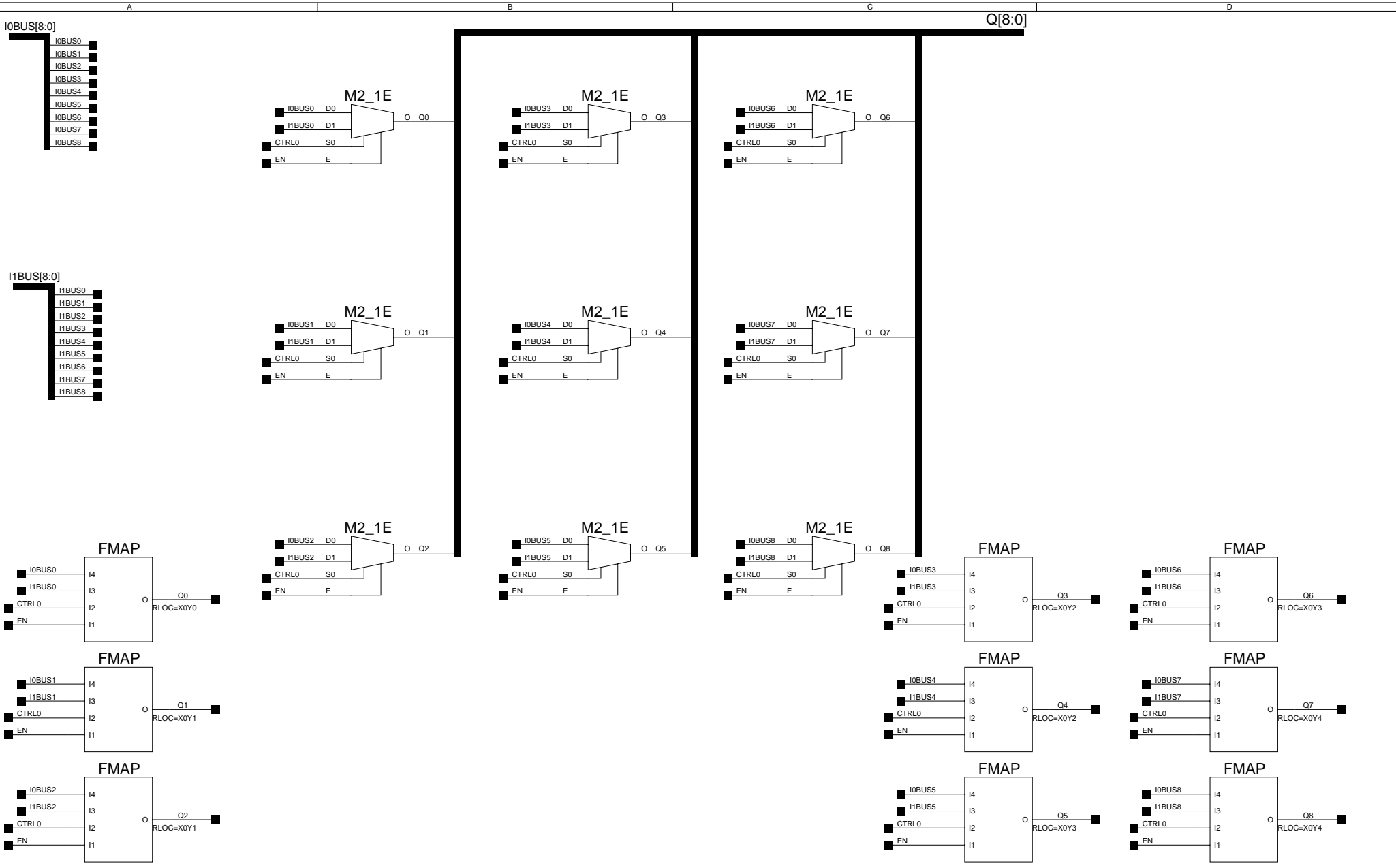
Title: VIRTEX Family FD9CE_G2 Macro	
Comments: 18-Bit Data Register w/ Clock Enable & Asynchronous Clr, RPM GRID	
Date: 10th November 2003	Ver: 1
Sheet Size: B	Rev: A

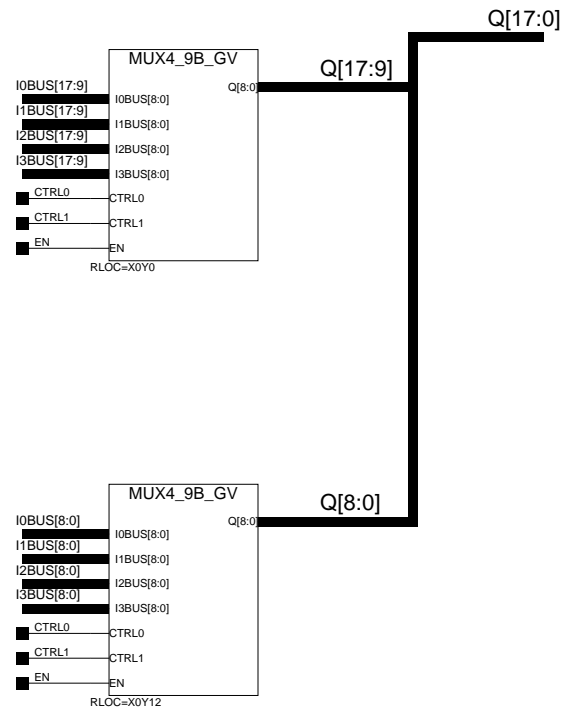
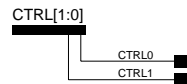
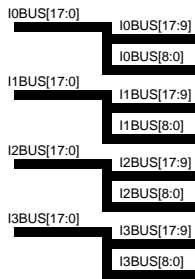


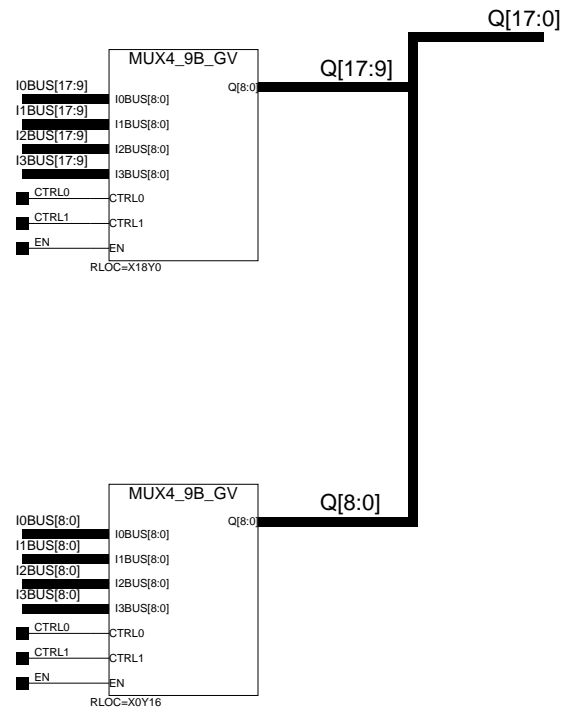
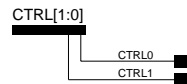
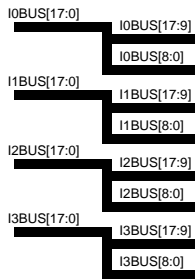
Title:	VIRTEX Family M4_1E_RPM Macro		
Comments:	4-to-1 Multiplexer with Enable		
Date:	31st October 2003	Ver:	1
Sheet Size:	A	Rev:	B

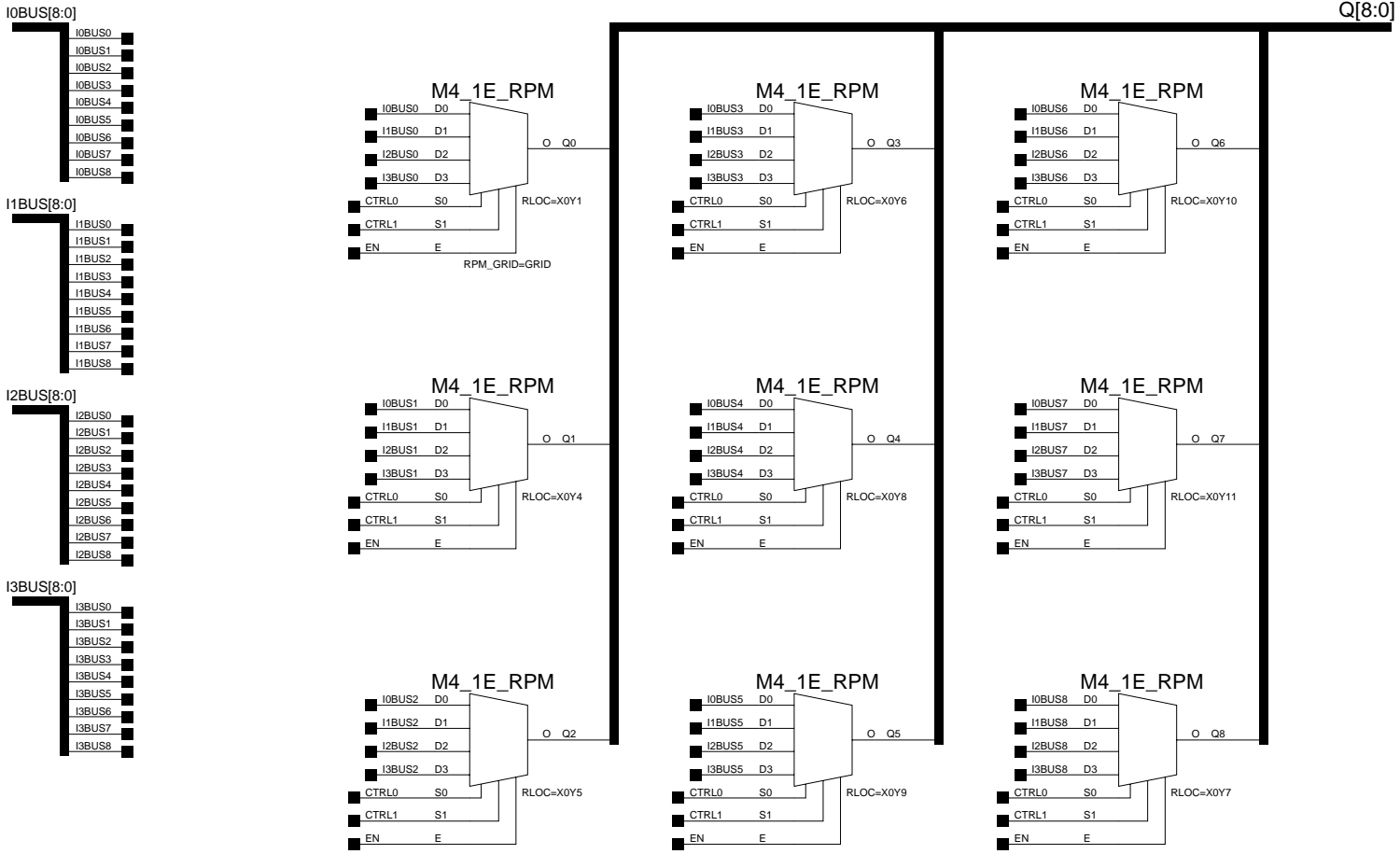




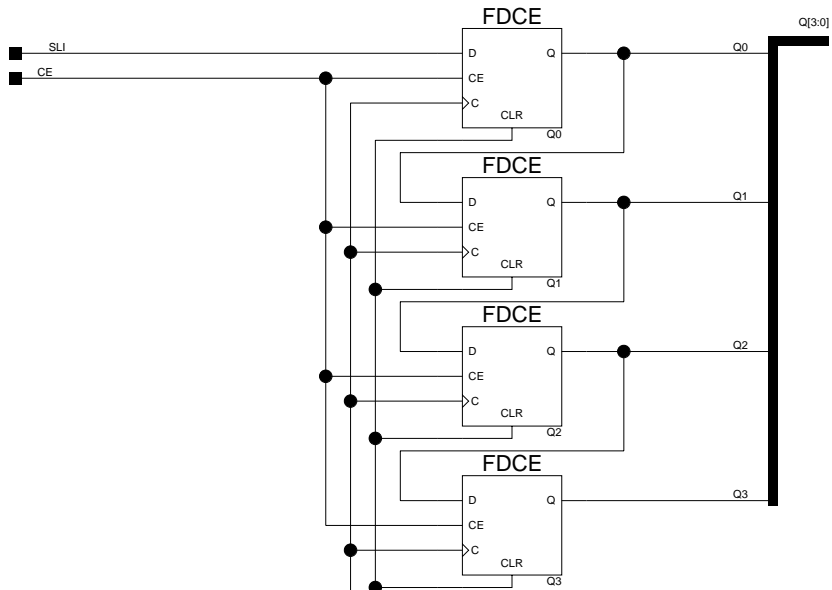








drawn by KS
Copyright (c) 1993, Xilinx Inc.



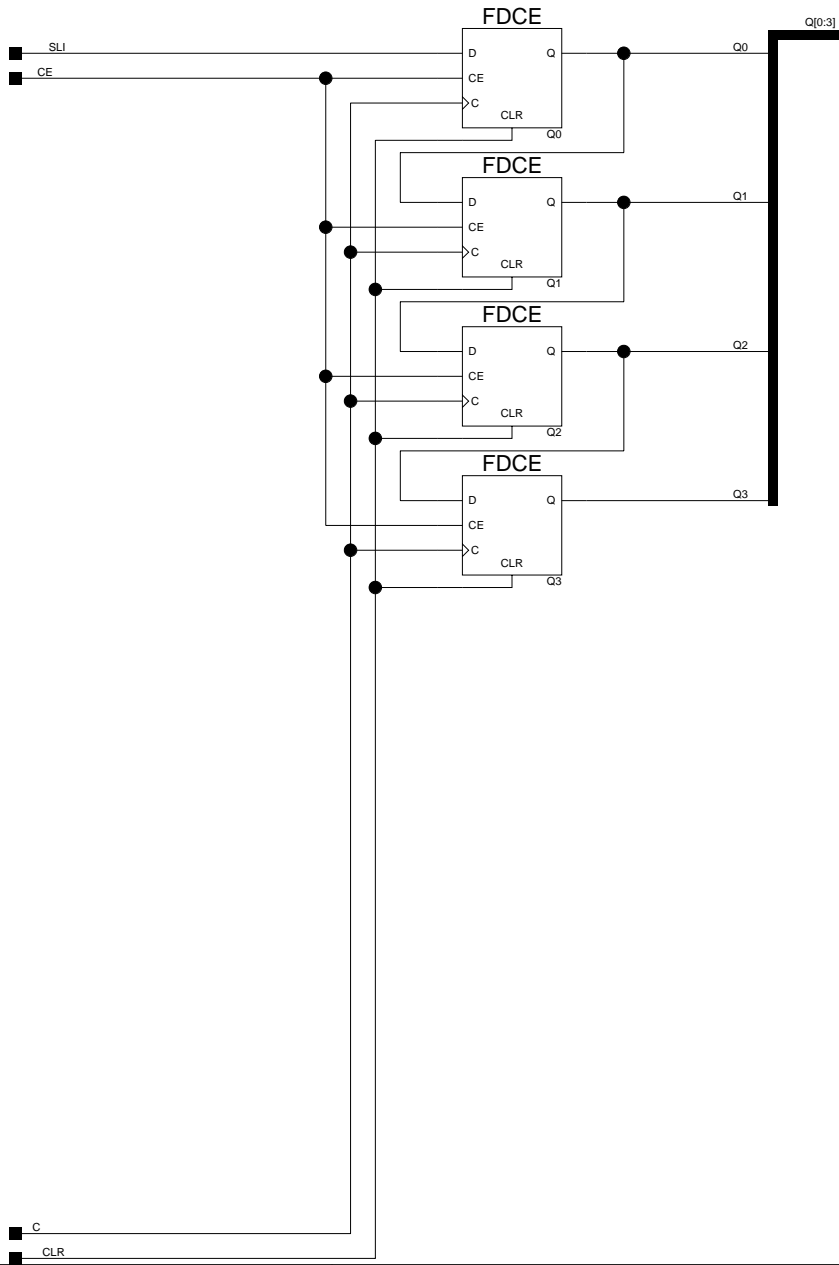
C
CLR



JRG

Title: VIRTEX Family SR4CE Macro	
Comments: 4-bit Serial-In Parallel-Out Shift Register w/ Enable and Async Clr	
Date: 3rd October 2003	Ver: 1
Sheet Size: B	Rev: A

drawn by KS
Copyright (c) 1993, Xilinx Inc.



Title: VIRTEX Family SR4CE_R Macro		JRG
Comments: 4-bit Serial-In Parallel-Out, REVERSED ORDER on OUTPUT BUS! Shift Register w/ Enable and Async Clr		
Date: 3rd October 2003	Ver: 1	
Sheet Size: B	Rev: A	