Set all Banks to 3.3V I/O

Mode 1 Switch Block, reversed labels on board
1: Mode Bit 0
2: Mode Bit 1
3: Mode Bit 2
4: Mode Bit 3
5: Mode Bit 4
6: Mode Bit 5
7: Disable Auto Serial Load
8: Set all LA bits HIGH, ~FPGA version on LEDs

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

Reset-to-DDU Ready time
Soft: < 20ms
Hard: < 52ms

DDU WordCount (64-bit words) for "No Data" event: 0x006.
DDU WordCount for one DMB (only one CFEB): 0x19A = 410 dec.
DDU WordCount for 2 DMB with 1 CFEB (nCFEB=2): 0x32E = 814 dec.
DDU WordCount for 2 DMB with 2 CFEB (nCFEB=4): 0x64E = 1614 dec.

DDU Format Since DDUctrl v15:
H1: 0xSTTNNNNNNN/DDS/5
H2: 0x800000/0001/0000
H3: 0xAAAADD/0001/0000/0000
H4: 0x00000000
H5: 0x00000000

Replace EmptyIN/FIFO.EMPTY PUs?

To Do:
- Test VME IRQ pin functions
- Check CAN-Throughput Latency
- Pull CAN Prompts into Flash RAM
- Connect and drive FPGA signals
   → Set correct default state on board!
- No logic for VMEW, VMEB, ...,

DDU WordCount = (6 + 25*Nts*nCFEB + 3*nDMB) <= 30050 (ignoring Trigger boards)

Ethernet ByteCount = 8*DDU WordCount

Default Startup Order:
4) DONE
5) En. Outputs
6) Release WE
CLK: 40MHz
SCLK: 10MHz = MIDCLK (max serial speed for FIFOs)
SLOWCLK: 2.5MHz (used for Serial ADC)
SLOWCLK2: 1.25MHz

CLK is in phase with MidClk, but they are not in phase with SlowClk
PROMs (max 2 MHz) use SlowClk2, Virtex2 (max 33MHz) use MidClk
Serial Flash PROM (max 20 MHz) use MidClk or SlowClk2
VMECLK: Not Used

FDC

STROBE

LSTROBE

JTAG "Device" List
1: Output FIFO
dvc7
dvc1

2: VME Ctrl Prom
dvc6
dvc4

3: DDU Ctrl Prom 1 & 0
dvc6
dvc4

4: InCtrl Prom 1 & 0
dvc4

5: DDU_Ctrl FPGA
dvc8
dvc2

6: InCtrl FPGA 0
dvc3
dvc5

7: InCtrl FPGA 1
dvc3
dvc5

8: SLINK JTAG
dvc5

9: VME Parallel (not JTAG) N/A
dvc7
dvc1

10: VME Serial (not JTAG) N/A
dvc7
dvc1

11: Serial ADC (not JTAG) N/A
dvc7
dvc1

12: Emergency Load for VME_Ctrl Prom N/A
dvc7
dvc1

"Device" List
1: Output FIFO
dvc7
dvc1

2: VME Ctrl Prom
dvc6
dvc4

3: DDU Ctrl Prom 1 & 0
dvc6
dvc4

4: InCtrl Prom 1 & 0
dvc4

5: DDU_Ctrl FPGA
dvc8
dvc2

6: InCtrl FPGA 0
dvc3
dvc5

7: InCtrl FPGA 1
dvc3
dvc5

8: SLINK JTAG
dvc5

9: VME Parallel (not JTAG) N/A
dvc7
dvc1

10: VME Serial (not JTAG) N/A
dvc7
dvc1

11: Serial ADC (not JTAG) N/A
dvc7
dvc1

12: Emergency Load for VME_Ctrl Prom N/A
dvc7
dvc1

13: Emergency Load for DDU_Ctrl Prom N/A
dvc7
dvc1

14: Emergency Load for VME_Ctrl Prom N/A
dvc7
dvc1

15: Emergency Load for DDU_Ctrl Prom N/A
dvc7
dvc1
The normal JTAG command can work at 10MHz, but for In_System_Programming, it must be slow, such as 1.25MHz.
The ISP does not work at 2.5MHz or faster.

Free LED Modes

LA0 free: 13    LA1 free: 12,13
LEDs Free: 13    TP 2-4 Used: 0,1

VME Communication Interface
DDU VME Controller Logic
CMS CSC Electronics
use "DMB15" for DDU_Ctrl status
use "DMB15" for DDU_Ctl status

Busy use "DMB15" for DDU_Ctl status
Ready

Ready + Error ./

Busy + Error ./

Logic for parallel FMM Register readout

The logic for parallel FMM Register readout includes

- Bit 0: Ready
- Bit 1: Busy
- Bit 2: Error
- Bit 3: Sync Lost
- Bit 4: Warning

DMB + DDU "STATx" definitions

0: BUSY (i.e. NotREADY)
1: Warning/NearFULL
2: Lost Sync, need SyncReset
3: Error, need HardReset
4: BUSY (i.e. NotREADY)
5: Error, need HardReset
6: Ready + Error

current status

"mcpdpara_status"

"mcpdpara_switch"

"mcpdpara_Ctrl"
Serial Device List (12 device functions, SEN = Serial Load)

0x 00: Read InFIFO 0
01: Read InFIFO 1
02: Read InFIFO 2
03: Read InFIFO 3
04: R/W Flash SRAM

0x 08: W Load DDR InFIFO 0
09: W Load DDR InFIFO 1
0A: W Load DDR InFIFO 2
0B: W Load DDR InFIFO 3

VME_SDEV>=8 or ==4 are Writeable; others are Read Only.

0C: W Load GBE Output FIFO (SEN=LD, set HI during MRST) --test?
0D: W Load DDR InFIFO 1

0E: Read Auto FIFO
0F: W Program page 7 (Board ID) [16 bit data]

VME SDV03: Read InFIFO 3

00: Read InFIFO 0
01: Read Status Register
02: Read InFIFO 1
04: Read page 4 (DDR offsets) to In DDR FIFO
05: Read page 5 (GBE offsets) to GBE Out FIFO

9 VME Reads, 4 VME Writes, 7 Auto commands

Flash Memory Access (9 VME commands on Dev4)

CMD=9 is Read Only.  CMD>=9 is Write Only

0x 00: Read Status Register
01: Read page 1 (Kill Ch.) to DDU_Ctrl
04: Read page 4 (DDR offsets) to In DDR FIFO
05: Read page 5 (GBE offsets) to GBE Out FIFO
07: Read page 7 (Board ID) to DDU_Ctrl

0x 09: W Program page 1 (Kill Ch.) [16 bit data]
0C: W Program page 4 (DDR offsets) [32 bit data]
0D: W Program page 5 (GBE offsets) [34 bit data]
0F: W Program page 7 (Board ID) [16 bit data]

9 VME-Serial commands on Dev4

0x 00: Read InFIFO 0
01: Read InFIFO 1
02: Read InFIFO 2
03: Read InFIFO 3
04: R/W Flash SRAM

0x 08: W Load DDR InFIFO 0
09: W Load DDR InFIFO 1
0A: W Load DDR InFIFO 2
0B: W Load DDR InFIFO 3

0C: Write DDR InFIFO 1
0D: Write GBE Out FIFO

0x 00: Read Status Register
01: Read page 1 (Kill Ch.) to DDU_Ctrl
04: Read page 4 (DDR offsets) to In DDR FIFO
05: Read page 5 (GBE offsets) to GBE Out FIFO
07: Read page 7 (Board ID) to DDU_Ctrl

Page 8


Serial Data Bit Counter for AutoLoad, RdFIFO, Read & Program SLD

CCS8LED

M XFER_CNT[7:0]

D_CNTIN[7:0]

Serial Data Bit Counter

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

VME Communication Interface
DDU VME Controller Logic
CMS CSC Electronics

JRG
5-24-2006,14:45

2G

D785
8-bit Opcode: Read Flash SRAM Status

- ROM16X1
- ROM32X1
- ROM64X1

32-bit OpCodes: Program Flash SRAM Pages 1,4,5,7

- ROM32X1
- ROM64X1

64-bit OpCodes: Readout Flash SRAM Pages 1,4,5,7

Note 4-bit VME command (VMEadr[5:2])
Serial ADC (12-bit, MAX1270/1271) Interface clock: 1.25MHz (Divided SLOWCLOCK) is used
The ADC1270/1271 can work at a frequency from 0.1MHz to 2.0MHz
Parallel Register Readout

Count how many CSC Reset bits are set:


Dev<8: Read Only, no CMD req'd.  Dev>=8 needs CMD, CMD>=128 is Write

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

VME Communication Interface
DDU VME Controller Logic
CMS CSC Electronics
Parallel Register Read/Write

**Required for Test firmware**

\[\text{CMD15} = \text{"FMM Test Reg"} \text{ FD16CE} \]

```
DEV9, CMD 0F/8Fh (R/W)
```

- bits15-4: \"F0Eh\"
- bits3-0: FMM Bit state to Force

must set \"FOE\" FMM Override Enable

**Required for Test firmware**

Read Only, VME Par Dev 8:

\[\text{CMD 3 = "Test Reg 0"} \text{ SCCLK} \]

```
Could be temporary
```

- \(\text{INPUT2} \text{ EVCNTRST}\)

**Required for Test firmware**

Read with InReg0

\[\text{CMD 4 = "Test Reg 1"} \text{ SCCLK} \]

```
Could be temporary
```

- \(\text{INPUT6} \text{ BC0}\)

**Required for Test firmware**

Read with InReg0

\[\text{CMD 5 = "Test Reg 2"} \text{ SCCLK} \]

```
Could be temporary
```

**Required for Test firmware**

Read with InReg0

\[\text{CMD 6 = "Test Reg 3"} \text{ SCCLK} \]

```
Could be temporary
```

**Required for Test firmware**

Read with InReg0

\[\text{CMD 7 = "Test Reg 4"} \text{ SCCLK} \]

```
```

Dev9, CMD 05/85h (R/W)

- bits2-0: Enable Fake L1A/Data Passthrough for each DDU FPGA
- bit3: Not Used

VME Communication Interface
DDU VME Controller Logic
CMS CSC Electronics

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

JRG JGOLEY PAGE 1
D785
1 5-24-2006 14:51
VMECNTRL : 13 2M
**VME-JTAG Device code:** Path IR bit length

<table>
<thead>
<tr>
<th>Device</th>
<th>ADR[15:12]=000b for VME-JTAG</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>VME_Ctrl PROMs (all PROMs=XC18V04)</td>
</tr>
<tr>
<td>01</td>
<td>Output FIFO <strong>FF</strong></td>
</tr>
<tr>
<td>02</td>
<td>JTAG command</td>
</tr>
<tr>
<td>03</td>
<td>Input FIFOs 0-3</td>
</tr>
<tr>
<td>04</td>
<td>Serial ADC (not JTAG)</td>
</tr>
<tr>
<td>06</td>
<td>Emergency PROM Programming via VME</td>
</tr>
</tbody>
</table>

**Type:**

ADR[18:16]=000b for VME-JTAG

**VME-Parallel Device code (all 16-bit data):**

1: R; CSCs w/Warning-NearFULL History
2: R; CSCs w/Warning-NearFULL History
3: R; CSCs w/Error, need HardReset
4: R; Quick summary of all boards that require a Reset
5: R; CSCs w/Warning-NearFULL History

**VME-Serial Device code:** "iadr" in scan.c

04 | Flash SRAM (64-bit or Program Page), NEEDS COMMAND
05 | Load GBE Output FIFO (SEN=LD, set HI during MRST)--N/A
06 | Load DDU_Ctrl FPGA (V2P7) 10
07 | Load DDU_Ctrl FPGA (Board ID)--N/A
08 | Input FIFOs 0-3

**Boundary Scan IR Codes**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**Device Bypass**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>04-10: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**User Code**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>04-10: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**ID Code**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>04-10: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**User2**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>04-10: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**ValidAMBOARDENB**

<table>
<thead>
<tr>
<th>Device</th>
<th>PROM XC18V04</th>
<th>FPGA Virtex2Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00-02: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>Read Output Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>03-07: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>04-10: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>05-11: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>06-12: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>07-13: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>08-14: Read Input Register 0-2 [16-bit data]</td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>09-15: Read Only RstTestReg 0-4 [16-bit data]</td>
<td></td>
</tr>
</tbody>
</table>

**2nd VME Data:**

- Set User DR (8-bit): "Select Function"
- Send IR (10-bit) "User2"
- Send IR (10-bit) "User1"
- Set User DR (8-bit): "No-Op"
- Send IR (10-bit): "Bypass"
- Send IR (10-bit) "User1"

**3rd VME Data:**

- Send IR (10-bit) "User2"
- Set User DR (8-bit): "Select Function"
- Send IR (10-bit) "User1"
- Write Input Register 0 [16-bit data]
- Read Input Register 0-2 [16-bit data]
- Read Only RstTestReg 0-4 [16-bit data]
DTACK for Load Instruction/Data Register command
CFEB JTAG commands:
- 00 || Shift data, no header, no tailer
- 01 || Shift data with header only
- 02 || Shift data with tailer only
- 03 || Shift data with header and tailer
- 04  
- 05 || Read TDO register
- 06 || Reset JTAG State machine
- 07 || Shift Instruction register with header and tailer

- 0C || Shift IR, no header, no tailer
- 0D || Shift IR with header only
- 0E || Shift IR with tailer only
- 0F || Shift Instruction register with header and tailer
4-bit Serial-In Parallel-Out Shift Register w/ Enable, loads a single "one" on Sync Reset

VIRTEX Family SR4RE Macro (SR4E_ONE)
was ~ADCENA1
Serial ADC Command Decoder:
00 || Write Control Byte to MAX1271's
01 || Read Data Back from 1271 Register
02 ||
04 ||
05 ||
06 ||

Serial ADC Command Decoder:
00 || Write Control Byte to MAX1271's
01 || Read Data Back from 1271 Register
02 ||
04 ||
05 ||
06 ||

CFEB JTAG command decode

- COMMAND0
- COMMAND1
- COMMAND2
- COMMAND3
- COMMAND4
- COMMAND5
- COMMAND6
- COMMAND7
- COMMAND8
- COMMAND9

- CMDHIGH
- CMDMID
- CMDLOW

- WRITE
- READ

- SELADC
- READMAX
- WRITEMAX

- COMMAND[9:0]

- DEVICE
- COMMAND4
- COMMAND5

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

Title: Serial ADC Commands decoder
DDU VME Controller Logic
CMS CSC Electronics
Custom LED Slow-Blink Control for FMM Outputs