



3<sup>rd</sup> December 2013

# ODMB user's manual

Optical DAQ MotherBoard for the ME1/1 stations of the CMS muon endcap detector

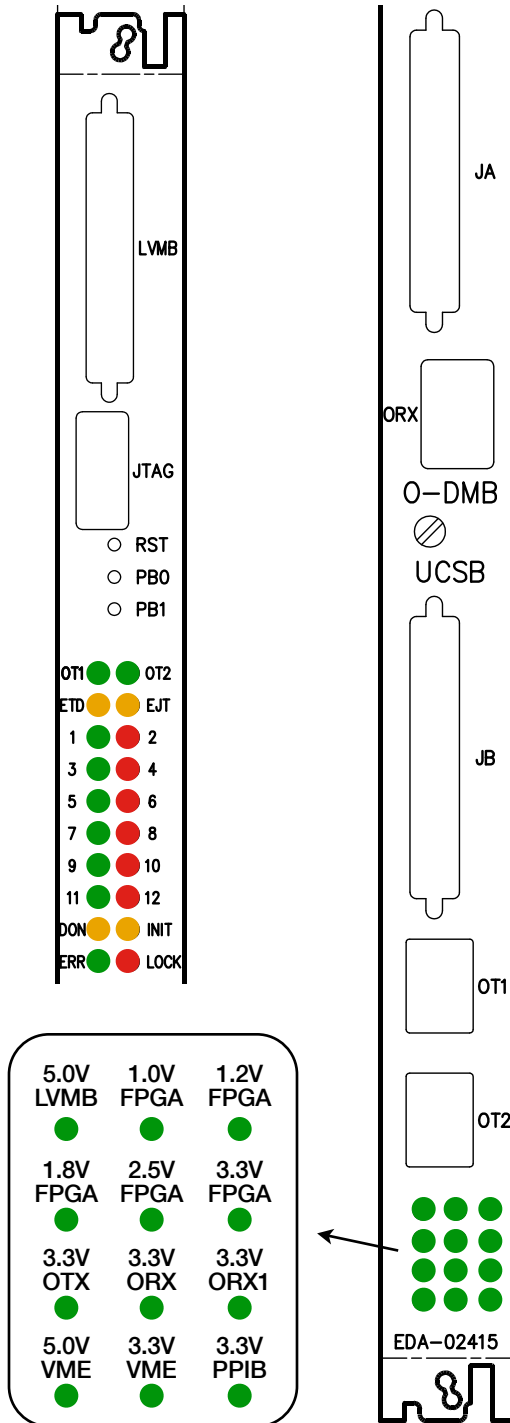
Firmware tag: V02-00

Manuel Franco Sevilla, Frank Golf, Guido Magazzù, Tom Danielson, Adam Dishaw, Jack Bradmiller-Feld  
UC Santa Barbara

# Table of Contents

<b>Front panel</b>	<b>2</b>
<b>General</b>	<b>3</b>
Firmware version	3
VME access through the board discrete “emergency” logic	3
Jumpers and test points	4
<b>Device 1: DCFEB JTAG</b>	<b>5</b>
Example: Read DCFEB UserCode	5
<b>Device 2: ODMB JTAG</b>	<b>6</b>
Example: Read ODMB UserCode	6
<b>Device 3: ODMB/DCFEB control</b>	<b>7</b>
Bit specification of ODMB_CTRL and DCFEB_CTRL	7
Information accessible via command “R 3YZC”	8
<b>Device 4: Configuration registers</b>	<b>9</b>
<b>Device 5: Test FIFOs</b>	<b>10</b>
Notes	10
<b>Device 6: BPI Interface</b>	<b>11</b>
<b>Device 7: ODMB monitoring</b>	<b>12</b>
Translation into temperatures and voltages	12
<b>Device 8: Low voltage monitoring</b>	<b>13</b>
<b>Device 9: System tests</b>	<b>14</b>
<b>Firmware block diagram</b>	<b>15</b>

# Front panel



## Push buttons

- **RST:** Reloads firmware in PROM onto FPGA
- **PB0:** Resets registers/FIFOs in FW. **LEDs 1-12** blink at different speeds for ~3s
- **PB1:** Sends L1A and L1A\_MATCH to all DCFEBs. Turns on **LED 12**

## LEDs set in firmware

- **1:** 4 Hz signal from clock for data → DDU
- **3:** 2 Hz signal from clock for data → PC
- **5:** 1 Hz signal from internal ODMB clock
- **7:** Data taking: ON normal, OFF pedestal
- **9:** Triggers: ON external, OFF internal
- **11:** Data: ON real, OFF simulated
- **2:** Bit 0 of L1A\_COUNTER
- **4:** Bit 1 of L1A\_COUNTER
- **6:** Bit 2 of L1A\_COUNTER
- **8:** Bit 3 of L1A\_COUNTER
- **10:** Bit 4 of L1A\_COUNTER
- **12:** Briefly ON when a VME command is received. Also ON when **PB1** is pressed

## LEDs set in hardware

- **OT1:** Signal Detected on OT1 (DDU)
- **OT2:** Signal Detected on OT2 (PC)
- **ETD:** DTACK enable for discrete logic (logic low)
- **EJD:** JTAG enable for discrete logic (logic low)
- **DON:** DONE signal from FPGA. ON when programmed
- **INIT:** INIT\_B signal from FPGA (logic low)
- **ERR:** Error on QPLL
- **LOCK:** QPLL is locked
- **Bottom 12:** Voltage monitoring

# General

## Firmware version

For a given firmware tag **VXY-ZK**:

- ❖ Usercode is **XYZKdbdb**
- ❖ Firmware version read via "R 4024" is **XYZK**

## VME access through the board discrete "emergency" logic

The FPGA may be accessed via JTAG through the discrete logic as follows

- ❖ The VME address is 0xFFFC
- ❖ The bit 0 of the data sent is TMS
- ❖ The bit 1 of the data sent is TDI

For example, to read the Usercode, starting from JTAG idle (five TMS = 1 & one TMS = 0), the commands are:

```

W FFFC 1 To Select-DR-Scan
W FFFC 1 To Select-IR-Scan
W FFFC 0 To Capture-IR
W FFFC 0 To Shift-IR

W FFFC 0 Shifting IR (Read UserCode IR = 3C8)
W FFFC 0 Shifting IR
W FFFC 0 Shifting IR
W FFFC 2 Shifting IR
W FFFC 0 Shifting IR
W FFFC 0 Shifting IR
W FFFC 2 Shifting IR
W FFFC 2 Shifting IR
W FFFC 2 Shifting IR
W FFFC 3 Shifting IR and to Exit1-IR

W FFFC 1 To Update-IR
W FFFC 0 To Run_Test/Idle
W FFFC 1 To Select-DR-Scan
W FFFC 0 To Capture-DR

W FFFC 0 Shifting DR
R FFFC 0 Shifting DR (Read bit 0 of UserCode)

```

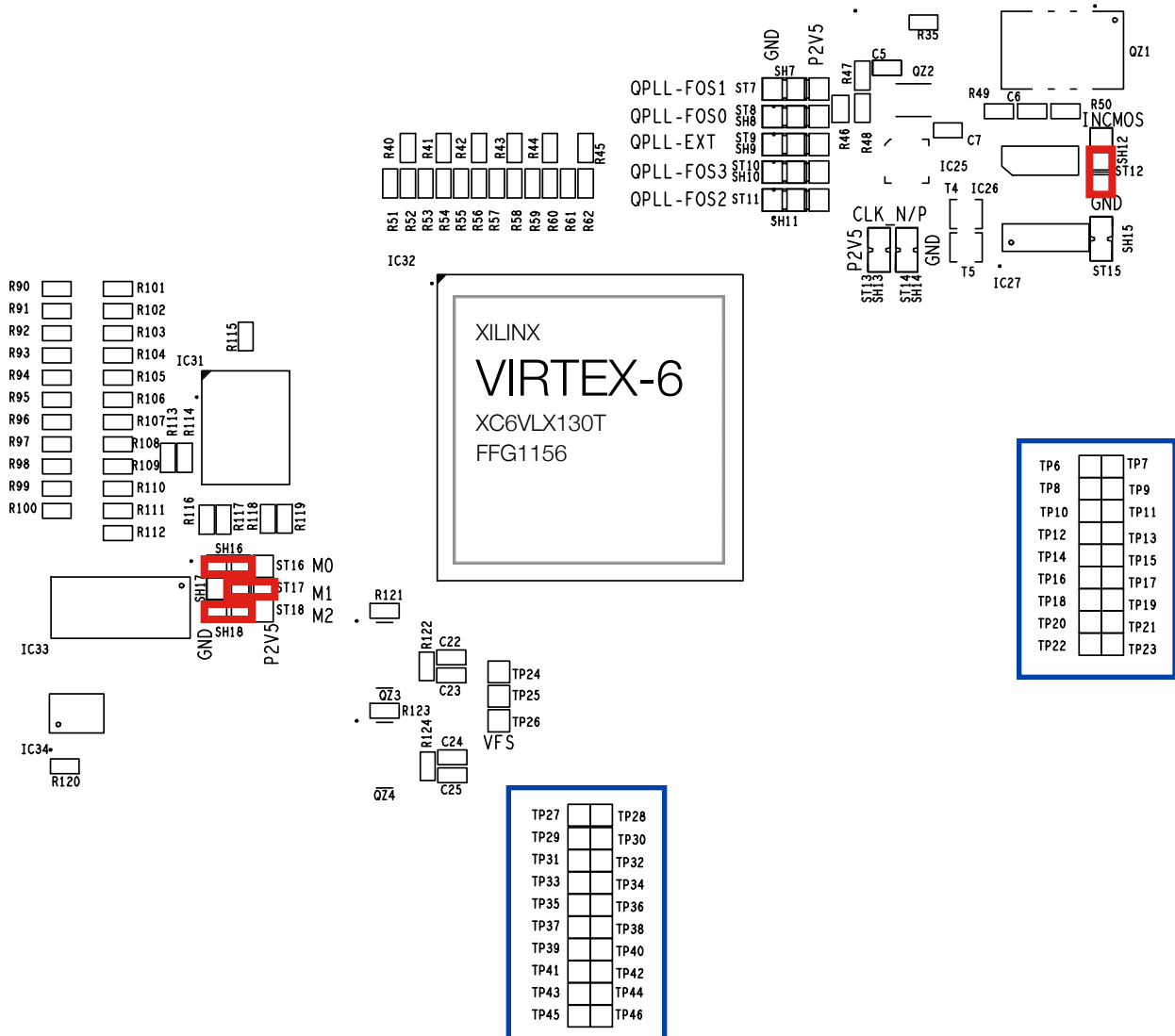
Since the Usercode register is 32 bits, the last two commands should be repeated 31 more times.

## Jumpers and test points

Place the **jumpers** marked in **red** in the diagram: M[2:0] = 010, and ST12 grounded to use clock from CCB.

The signals sent to the **test points** marked in **blue** are:

<b>TP6:</b>	RAW_LCT(1)	<b>TP7:</b>	L1A_MATCH(1)	<b>TP27:</b>	Defined by TP_SEL	<b>TP28:</b>	Defined by TP_SEL
<b>TP8:</b>	RAW_LCT(2)	<b>TP9:</b>	L1A_MATCH(2)	<b>TP29:</b>	DCFEB_DAV(1)	<b>TP30:</b>	DCFEB_DAV(2)
<b>TP10:</b>	RAW_LCT(3)	<b>TP11:</b>	L1A_MATCH(3)	<b>TP31:</b>	DDU_DATA_VALID	<b>TP32:</b>	PC_DATA_VALID
<b>TP12:</b>	RAW_LCT(4)	<b>TP13:</b>	L1A_MATCH(4)	<b>TP33:</b>	RAWLCT(1)	<b>TP34:</b>	RAWLCT(2)
<b>TP14:</b>	RAW_LCT(5)	<b>TP15:</b>	L1A_MATCH(5)	<b>TP35:</b>	RAWLCT(3)	<b>TP36:</b>	RAWLCT(4)
<b>TP16:</b>	RAW_LCT(6)	<b>TP17:</b>	L1A_MATCH(6)	<b>TP37:</b>	RAWLCT(5)	<b>TP38:</b>	RAWLCT(6)
<b>TP18:</b>	RAW_LCT(7)	<b>TP19:</b>	L1A_MATCH(7)	<b>TP39:</b>	RAWLCT(7)	<b>TP40:</b>	LCT_ERROR
<b>TP20:</b>	L1A	<b>TP21:</b>	DDU_DATA_VALID	<b>TP41:</b>	Defined by TP_SEL	<b>TP42:</b>	Defined by TP_SEL
<b>TP22:</b>	OTMBCDAV	<b>TP23:</b>	ALCTDAV				



# Device 1: DCFEB JTAG

“Y” refers to the number of bits to be shifted

Instruction	Description
<b>W 1Y00</b>	Shift Data; no TMS header; no TMS tailer
<b>W 1Y04</b>	Shift Data with TMS header only
<b>W 1Y08</b>	Shift Data with TMS tailer only
<b>W 1Y0C</b>	Shift Data with TMS header & TMS tailer
<b>R 1014</b>	Read TDO register
<b>W 1018</b>	Resets JTAG protocol to IDLE state (data sent with this command is disregarded)
<b>W 1Y1C</b>	Shift Instruction register
<b>W 1020</b>	Select DCFEB, one bit per DCFEB
<b>R 1024</b>	Read which DCFEB is selected

## Example: Read DCFEB UserCode

DCFEB registers are set and read via JTAG. The following procedure reads the 32-bit USERID of DCFEB 3:

```

W 1020 4 Select DCFEB 3 (one bit per DCFEB)

W 191c 3C8 Set instruction register to 3C8 (read UserCode)
W 1F04 0 Shift 16 lower bits
R 1014 0 Read last 16 shifted bits (DBDB)
W 1F08 0 Shift 16 upper bits
R 1014 0 Read last 16 shifted bits (XYZK)

```

# Device 2: ODMB JTAG

“Y” refers to the number of bits to be shifted

Instruction	Description
<b>W 2Y00</b>	Shift Data; no TMS header; no TMS tailer
<b>W 2Y04</b>	Shift Data with TMS header only
<b>W 2Y08</b>	Shift Data with TMS tailer only
<b>W 2Y0C</b>	Shift Data with TMS header & TMS tailer
<b>R 2014</b>	Read TDO register
<b>W 2018</b>	Resets JTAG protocol to IDLE state (data sent with this command is disregarded)
<b>W 2Y1C</b>	Shift Instruction register

## Example: Read ODMB UserCode

Read FPGA UserCode:

```

W 291c 3C8      Set instruction register to 3C8 (read UserCode)
W 2F04 0        Shift 16 lower bits
R 2014 0        Read last 16 shifted bits (DBDB)
W 2F08 0        Shift 16 upper bits
R 2014 0        Read last 16 shifted bits (XYZK)

```

# Device 3: ODMB/DCFEB control

Instruction	Description
<b>W/R 3000</b>	ODMB_CTRL register
<b>W/R 3010</b>	DCFEB_CTRL register
<b>W/R 3020</b>	TP_SEL register (selects which signals are sent to TP27, TP28, TP41, TP42)
<b>W/R 3100</b>	LOOPBACK: 0 → no loopback, 1 or 2 → internal loopback
<b>W/R 3110</b>	DIFFCTRL (TX voltage swing): 0 → minimum ~100 mV, F → maximum ~1100mV
<b>R 3120</b>	Read DONE bits from DCFEBs (7 bits)
<b>R 3YZC</b>	Read ODMB_DATA corresponding to selection <b>YZ</b> (see below)

## Bit specification of ODMB\_CTRL and DCFEB\_CTRL

- ▶ ODMB\_CTRL[3:0] - Selects CAL\_TRGEN (calibration mode).
- ▶ ODMB\_CTRL[4] - Selects CAL\_MODE (calibration mode).
- ▶ ODMB\_CTRL[5] - Selects CAL\_TRGSEL (calibration mode).
- ▶ ODMB\_CTRL[7] - Selects DCFEB data path: 0 → real data, 1 → dummy data.
- ▶ **ODMB\_CTRL[8] - Resets FPGA registers/FIFOs and LEDs 1-12 blink for ~3s. Bit is auto-reset.**
- ▶ ODMB\_CTRL[9] - Selects L1A and LCTs: 0 → from CCB, 1 → internally generated.
- ▶ ODMB\_CTRL[10] - Selects LVMB: 0 → real LVMB, 1 → dummy LVMB.
- ▶ ODMB\_CTRL[11] - Kills L1A.
- ▶ ODMB\_CTRL[12] - Kills L1A\_MATCH.
- ▶ ODMB\_CTRL[13] - 0 → normal, 1 → pedestal (L1A\_MATCHes sent to DCFEBs for each L1A).
- ▶ ODMB\_CTRL[14] - 0 → normal, 1 → pedestal (OTMB data requested for each L1A, needs spec. OTMB FW).
- ▶ **DCFEB\_CTRL[0] - Reprograms the DCFEBs. Bit is auto-reset.**
- ▶ DCFEB\_CTRL[1] - Resynchronizes the L1A\_COUNTER of ODMB and DCFEBs. Bit is auto-reset.
- ▶ DCFEB\_CTRL[2] - Sends INJPLS signal to DCFEBs. Bit is auto-reset.
- ▶ DCFEB\_CTRL[3] - Sends EXTPLS signal to DCFEBs. Bit is auto-reset.
- ▶ DCFEB\_CTRL[4] - Sends test L1A and L1A\_MATCH to all DCFEBs. Bit is auto-reset.
- ▶ DCFEB\_CTRL[5] - Sends LCT request to OTMB. Bit is auto-reset.
- ▶ DCFEB\_CTRL[6] - Sends external trigger request to OTMB. Bit is auto-reset.
- ▶ DCFEB\_CTRL[7] - Resets the optical transceivers. Bit is auto-reset.



## Information accessible via command "R 3YZC"

- ▶ YZ = 3F: Least significant 16 bits of L1A\_COUNTER
- ▶ YZ = 21-29: Number of L1A\_MATCHes for given DCFEB, OTMB, ALCT
- ▶ YZ = 31-37: Gap (in number of bunch crossings) between the last LCT and L1A for given DCFEB
- ▶ YZ = 38: Gap (in number of bunch crossings) between the last L1A and OTMBDAV
- ▶ YZ = 39: Gap (in number of bunch crossings) between the last L1A and ALCTDAV
- ▶ YZ = 41-49: Number of packets stored for given DCFEB, TMB, or ALCT
- ▶ YZ = 4A: Number of packets sent to the DDU
- ▶ YZ = 4B: Number of packets sent to the PC
- ▶ YZ = 51-59: Number of packets shipped to DDU and PC for given DCFEB, TMB, or ALCT
- ▶ YZ = 61-67: Number of data packets received with good CRC for given DCFEB
- ▶ YZ = 71-77: Number of LCTs for given DCFEB
- ▶ YZ = 78: Number of available OTMB packets
- ▶ YZ = 79: Number of available ALCT packets
  
- ▶ YZ = 5A: Read last CCB\_CMD[5:0] + EVTRST + BXRST strobed
- ▶ YZ = 5B: Read last CCB\_DATA[7:0] strobed
- ▶ YZ = 5C: Read toggled CCB\_CAL[2:0] + CCB\_BX0 + CCB\_BXRST + CCB\_L1ARST + CCB\_L1A + CCB\_CLKEN + CCB\_EVTRST + CCB\_CMD\_STROBE + CCB\_DATA\_STROBE
- ▶ YZ = 5D: Read toggled CCB\_RSV signals

# Device 4: Configuration registers

Instruction	Description
<b>W/R 4000</b>	LCT_L1A_DLY[5:0] → Set to LCT/L1A gap - 100
<b>W/R 4004</b>	OTMB_DLY[4:0] → Set to L1A/OTMBDAV gap - 1
<b>W/R 4008</b>	PUSH_DLY[4:0]
<b>W/R 400C</b>	ALCT_DLY[4:0] → Set to L1A/ALCTDAV gap - 1
<b>W/R 4010</b>	INJ_DLY[4:0] - Delay: 12.5*INJ_DLY [ns]
<b>W/R 4014</b>	EXT_DLY[4:0] - Delay: 12.5*EXT_DLY [ns]
<b>W/R 4018</b>	CALLCT_DLY[3:0] - Delay: 25*CALLCT_DLY [ns]
<b>W/R 401C</b>	KILL[9:1] (ALCT + TMB + 7 DCFEBs)
<b>W/R 4020</b>	CRATEID[6:0]
<b>R 4024</b>	Read firmware version
<b>W/R 4028</b>	Number of words generated by dummy DCFEBs, OTMB, and ALCT

# Device 5: Test FIFOs

**Z** refers to FIFO: 1 → PC TX, 2 → PC RX, 3 → DDU TX, 4 → DDU RX, 5 → OTMB, 6 → ALCT

Instruction	Description
<b>R 5000</b>	Read one word of selected DCFEB FIFO
<b>R 500C</b>	Read numbers of words stored in selected DCFEB FIFO
<b>W/R 5010</b>	Select DCFEB FIFO
<b>W 5020</b>	Reset DCFEB FIFOs (7 bits, one per FIFO, which are auto-reset)
<b>R 5Z00</b>	Read one word of FIFO
<b>R 5Z0C</b>	Read numbers of words stored in FIFO
<b>W 5Z20</b>	Reset FIFO

## Notes

- All these FIFOs can hold a maximum of 2,000 18-bit words (36 kb)
- The **OTMB**, **ALCT**, and **7 DCFEB FIFOs** store the data as it arrives in parallel to the standard data path
  - They can hold a maximum of 3 OTMB, 4 ALCT, and 2 DCFEB data packets
- The **DDU TX FIFO** stores DDU packets just before being transmitted
  - They include the DDU header (4 words starting with 9, 4 starting with A), ALCT data, TMB data, DCFEB data, and trailer (4 words starting with F, 4 starting with E)
- The **PC TX FIFO** stores DDU packets wrapped in ethernet frames just before being transmitted
  - They include the ethernet header (4 words) and trailer (4 words)
  - They need to be at least 32 words long
- The **DDU** and **PC RX FIFOs** can be used for loopback tests

# Device 6: BPI Interface (PROM)

**Important: Instruction 6000 takes ~1 second, during which Device 4 and 6 write commands are ignored**

Instruction	Description
<b>W 6000</b>	Write configuration registers to PROM
<b>W 6004</b>	Set configuration registers to retrieved values from PROM
<b>W 6020</b>	Reset BPI interface state machines
<b>W 6024</b>	Disable parsing commands in command FIFO while filling FIFO with commands (no data)
<b>W 6028</b>	Enable parsing commands in the command FIFO (no data)
<b>W 602C</b>	Write one word to command FIFO
<b>R 6030</b>	Read one word from read-back FIFO
<b>R 6034</b>	Read number of words in read-back FIFO
<b>R 6038</b>	Read BPI Interface Status Register
<b>R 603C</b>	Read Timer (16 LSBs)
<b>R 6040</b>	Read Timer (16 MSBs)

# Device 7: ODMB monitoring

## Reads output of the ADC inside the FPGA

Instruction	Description
R 7000	FPGA temperature
R 7100	LV_P3V3: input to FPGA regulators
R 7110	P5V: input to PPIB regulator and level for 5V chips
R 7120	THERM2: board temperature at the center-top
R 7130	P3V3_PP: voltage level for PPIB
R 7140	P2V5: voltage level for FPGA and 2.5V chips
R 7150	THERM1: board temperature close to the LVMB connector
R 7160	P1V0: voltage level for FPGA
R 7170	P5V_LVMB: voltage level for LVMB

## Translation into temperatures and voltages

The output of the 7YZ0 commands is a 12-bit number that we call  $R_{YZ}$ . The measurement is:

- The FPGA temperature is  $T_{\text{FPGA}} = \frac{R_{00} \times 503.975}{4096} - 273.15$  [ $^{\circ}\text{C}$ ]
- The temperature of the thermistors THERM1, THERM2 is given by

$R_{XY}$	377	455	55A	687	7DD	959	AF8	CB5	E87	FFF
$T$ [ $^{\circ}\text{C}$ ]	15	20	25	30	35	40	45	50	55	60

- The voltage levels are  $V_{YZ} = \frac{R_{YZ}}{2048} \times V_{YZ, \text{Nom}}$  [V], where  $V_{YZ, \text{Nom}}$  is the nominal voltage level for that register. That is,  $V_{10, \text{Nom}} = V_{13, \text{Nom}} = 3.3\text{V}$ ,  $V_{11, \text{Nom}} = V_{17, \text{Nom}} = 5\text{V}$ ,  $V_{14, \text{Nom}} = 2.5\text{V}$ , and  $V_{16, \text{Nom}} = 1\text{V}$ .

# Device 8: Low voltage monitoring

Instruction	Description
<b>W 8000</b>	Send control byte to ADC
<b>R 8004</b>	Read ADC
<b>W 8010</b>	Select DCFEBs/ALCT to be powered on (8 bits, ALCT + 7 DCFEBs)
<b>R 8018</b>	Read which DCFEBs/ALCT are powered on
<b>W 8020</b>	Select ADC to be read, 0 to 6
<b>R 8024</b>	Read which ADC is to be read

# Device 9: System tests

Instruction	Description
<b>W 9000</b>	Test the DDU TX/RX with a given number of PRBS $2^7-1$ sequences
<b>R 900C</b>	Read number of errors during last DDU PRBS test
<b>W 9100</b>	Test the PC TX/RX with a given number of PRBS $2^7-1$ sequences
<b>R 910C</b>	Read number of errors during last PC PRBS test
<b>W 9200</b>	Check N*10000 bits from the PRBS pattern sent by the DCFEB
<b>W/R 9204</b>	Select DCFEB fiber to perform PRBS test
<b>R 9208</b>	Read number of error edges during last DCFEB PRBS test
<b>R 920C</b>	Read number of bit errors during last DCFEB PRBS test
<b>W/R 9300</b>	Set PRBS type for DCFEB: 1 → PRBS-7, 2 → PRBS-15, 3 → PRBS-23, 4 → PRBS-31
<b>W 9400</b>	Check N*10000 bits from the PRBS pattern sent by the OTMB
<b>R 9404</b>	Read number of enables sent by the OTMB
<b>R 9408</b>	Read number of good 10000 bits sent by the OTMB
<b>R 940C</b>	Read number of bit errors during last OTMB PRBS test
<b>W 9410</b>	Reset number of errors in OTMB counter

# Firmware block diagram

The firmware can be downloaded from [http://github.com/odmb/odmb\\_ucsb\\_v2](http://github.com/odmb/odmb_ucsb_v2)

