MobiDick4

Introduction

This page contains information relative to the new front-end electronics stand-alone testbench MobiDick4. Its purpose is to provide tools for full certification of the TileCal super-drawers during maintenance periods. Previous version of this testbench can be found here, and a bunch of the schemas used for it can be found here. In that version, a custom aluminium box contains a modified VME crate populated with different VME modules as TTCvi, TTCex, Caen V972, RIO2 processor and custom boards. A schema showing the TTCvi connections is available here.

- Public note on MobiDICK

MobiDick4 is inspired by the second version of the digitizer testbench from Stockholm group. Where the core of the system is a Virtex 5 evaluation board that emulates the TTC and ROD electronics and a Java front-end GUI that analizes the data and presents the results to the user.

- CERN_digitest_preliminary_version_v2.pdf: Digitizer testbench documentation
- TWEPP-2008_-_Paper.pdf: Paper on Stockholm testbench
- Stockholm_testbench.pdf: Stockholm testbench
- Stockholm testbench repository

Stockholm test protocol

The tests that can be made on the Stockholm testbench are:

- Bias test
  This measures the ADC threshold of the digitizer. If the levels are too far apart or there is too much noise, this will be flagged as an error. To fix bias errors, calibration resistors are used to adjust the voltage levels.
- Memory test
  The memory test sends 16 events (with 16 samples) of zeroes followed by 16 events of ones and finishes with 16 events of running ones (111, 222, 44, 88, 111, etc...). This will run through the memory and catch any stuck bit errors. This is not a stress test and will not catch SI errors. An error here means that the DMU is bad (assuming transmission and power are ok).
- DAC test
  The DAC test (or linearity test) is a test of the linearity of the digitizer ADCs. An onboard DAC is ramped up event by event. A stuck bit here will show as a step in the graph and also in text in the test results. A stuck bit means that an error has occurred either in the ADC, the ADC output, the DMU input or in the lines between ADC and DMU.

Tests not currently working:

- Configuration test
  This is a pulse test. The 3-in-1 cards are configured to send a pulse. Four events are read out (4 pulses) with both hi and low gains. This procedure tests the range of the digitizer, since the high gain pulse should saturate the ADC.
- Pulse test
  This takes the average of several pulses and calculates a chi-square. This is a test of the pulse shape and ADCs stability between events.

Based on the errors returned by the front-end software, faulty DMU:s, TTC:S, DAC:s, clock-drivers, LVDS-drivers and in some cases ADC:s can be found relatively easy in addition to connection and power supply errors.

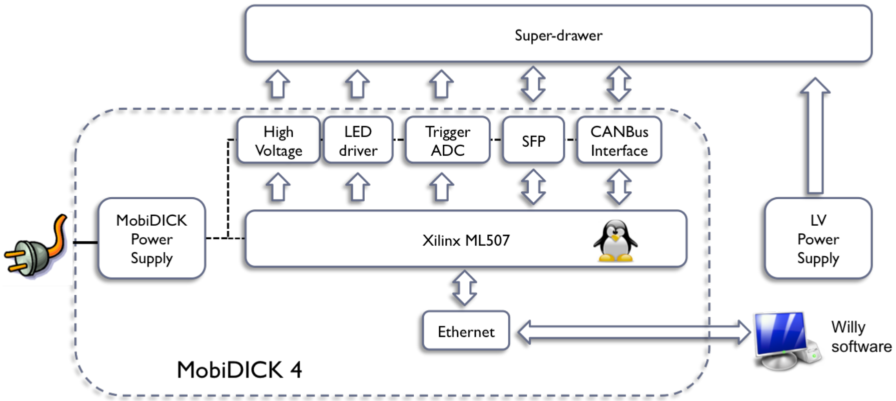
Further documentation can be found here

Description of the project

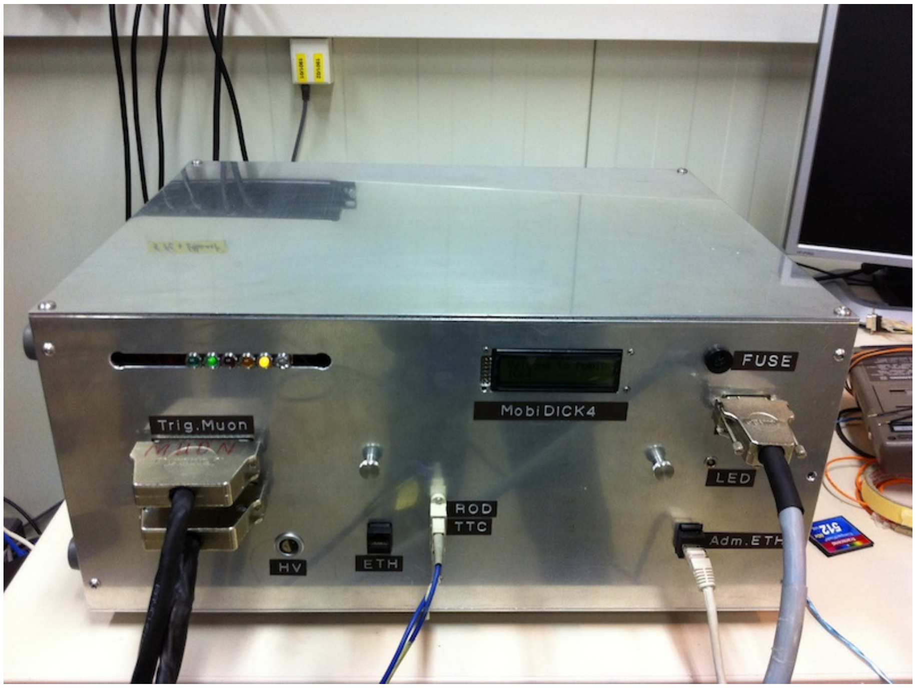
MobiDick4 is a stand-alone test-bench for the full certification of Tile front-end super-drawer electronics. This fourth version comprises:

A front-end GUI (Willy) already existing for previous versions of MobiDick, based on ROOT and C++ communicates with a back-end server running on a motherboard that is connected to the super-drawer to be tested.

Several state-of-the-art daugther-boards provide different functionalities to the motherboard: ADC trigger read-out, CAN bus interface, HV power supply and LED pulse generator. An additional daughter-board is required to distribute power to all the boards, and an external LV power supply is needed to power the super-drawer electronics.



Finished MobiDICK4:



Presentations:

- Project description at Upgrade session in Tile week by César Marin (17th June 2011): Slides
- Project status at Upgrade session in Tile week by Fernando Carrió (7th October 2011): Slides
- Project status at Tile operations weekly meeting by Fernando Carrió (1st December 2011): Slides
- Project status at Upgrade session in Tile week by Vinícius Schettino (8th February 2012): Slides
- Project status at Tile operations weekly by Fernando Carrió (29th March 2012): Slides
- Project status at Upgrade session in Tile week by Pablo Moreno (1st June 2012): Slides
- Project status at Tile operations weekly by Fernando Carrió (19th July 2012): Slides
- Project status at Tile operations weekly by José Alves (30th August 2012): Slides
- Project status at Upgrade session in Tile week by José Alves (28th September 2012): Slides
- Project status at Tile operations weekly by Julio Souza (25th October 2012): Slides
- Project status at Tile operations weekly by Fernando Carrió (6th December 2012): Slides
- Project status at Tile upgrade vidyo meeting by Carlos Solans (13th December 2012): Slides
- Project status at Upgrade session in Tile week by Hee Yeun Kim (1st February 2013): Slides
- Project status at Tile operations weekly by Fernando Carrió (14th March 2013): Slides
- Project status at Tile session in Upgrade week by Carlos Solans (17th May 2013): Slides
- Project status at Upgrade session in Tile week by Julio De Souza (14th June 2013): Slides
- Project status at Operations/Maintenance session in Tile week by Xifeng Ruan (2nd Oct 2013): Slides
- Project status at Operations/Maintenance session in Tile week by Daniel Bullock (5th Feb 2014): Slides
- Project status at Operations/Maintenance session in Tile week by Jeff Dandoy (1st Oct 2014): Slides

Publications:

- A new portable test bench for the ATLAS Tile Calorimeter front-end electronics presented at TWEPP12, Pablo Moreno et al., September 2012: Abstract Poster Proceedings Publication

- A new portable test bench for the ATLAS Tile Calorimeter front-end certification at ANIMMA 2013, J. Alves et al., June 2013: Abstract Poster Proceedings

- Design of an FPGA-based embedded system for the ATLAS Tile Calorimeter front-end electronics test-bench at TWEPP 2013, Fernando Carrio et al., September 2013: Abstract Poster Proceedings Publication

- Computing challenges in the certification of ATLAS Tile Calorimeter front-end electronics during maintenance periods at CHEP 2013, Carlos Solans et al., October 2013: Abstract Poster Proceedings Publication

- Upgrade to the portable test facility for ATLAS Tile Calorimeter front-end electronics presented at IEEE NSS/MIC 2013, H.Y. Kim et al., November 2013: Abstract Poster Proceedings

Documentation

- MobiDICK4 users's guide in SVN
- Technical note in CDS, EDMS, and in SVN.

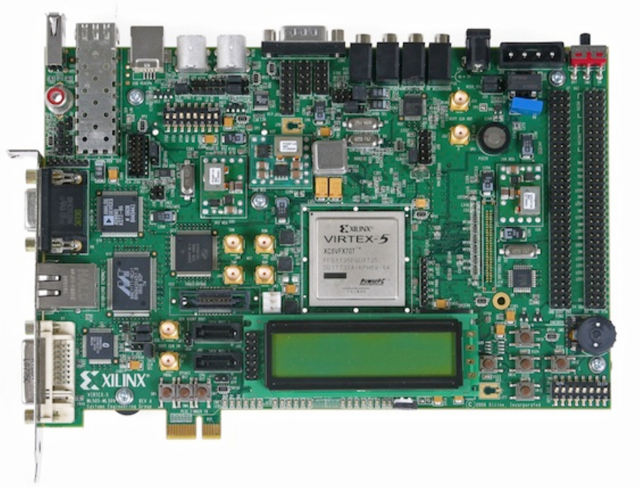
Motherboard

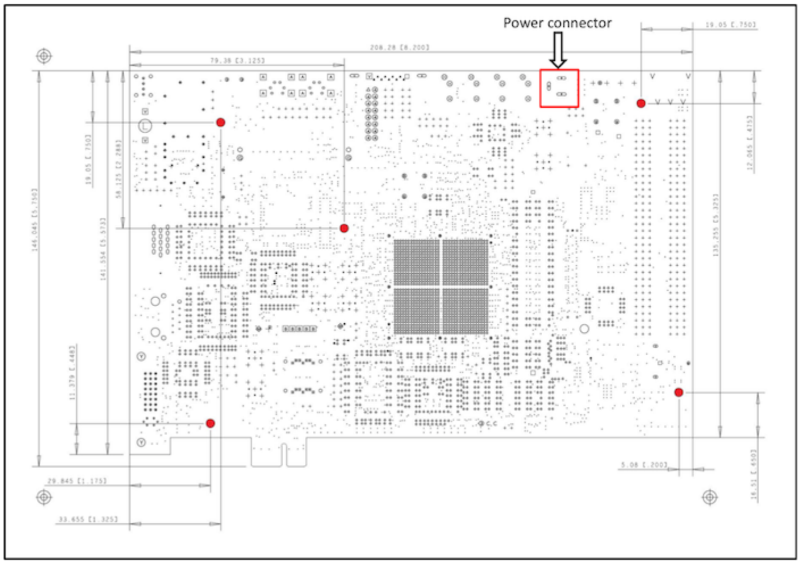
The motherboard of MobiDick4 is a Xilinx Virtex 5 evaluation board: ML507

It is equipped with an embedded IBM PowerPC440 processor, 256 MB of RAM memory and many communication interfaces (ethernet, USB, COM, ...) including an optical SFP module. Power is provided through the P20 connector using a standard connector with 5V.

List of motherboards

Serial number	Real MAC address	Comments
0431495-01-0936 / 0936-1488	00-0A-35-01-F4-F7	Replaced tantalum capacitors
0431495-04-1212 / 1212-3915	00-0A-35-02-79-E9	Ok
0431495-04-1212 / 1212-3927	00-0A-35-02-7A-E9	Ok
0431495-04-1025 / 1109-3402	00-0A-35-02-3D-93	Spare
0431495-04-1025 / 1109-3352	00-0A-35-02-3D-7F	Ethernet not working
0431495-04-1024 / 1109-3351	00-0A-35-02-3D-82	Ethernet not working





Configuration of switches on the board to boot from the Compact Flash card.

SW6 (1 to 8)	10101010
SW3 (1 to 8)	00010101

Configuration of jumpters for ethernet interface mode SGMII to copper no clock

J22	Jumper over pins 2-3
J23	Jumper over pins 2-3
J24	No jumper

Embedded system

The core of the embedded system is an IBM PowerPC 440 running at 400MHz which communicates with different IP cores using the Processor Local Bus (PLB). The embedded system controls all the external devices using these devices:

- 2 Serial port

- DDR2 memory controller
- Flash memory controller
- Hard Ethernet
- System Monitor
- HV and LED driver controller
- ADC trigger controller
- TTC_vi emulator
- G-Link decoder

This embedded system not only replaces the old server but also the TTCvi, TTCex and ODIN cards.

VHDL code is available to control the different devices of the board, including TTCvi and G-Link emulators.

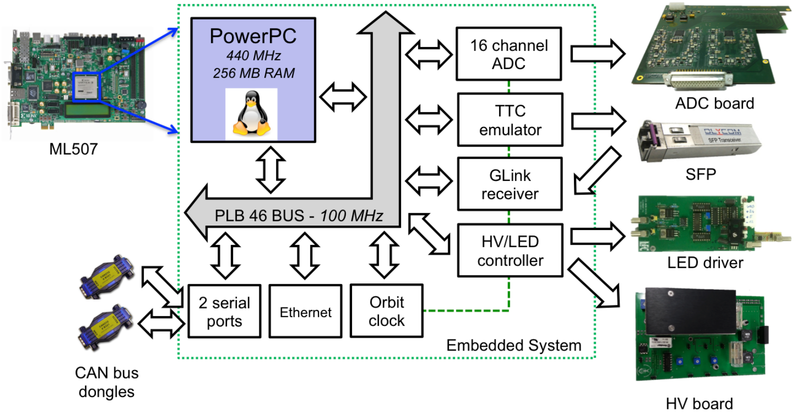


Table of registers

Device	Base	Register	Offset	Access	Description
HV+LED	0xC8600000	Dummy	0x0	R/W	Dummy register to test read and write functionality
		Status	0x4	R	Internal status of the IP Core FSM
		HV control	0x8	W/R	0=Off, 1=On. Turn on/off the HV and LED control through constant voltage on pin J6-12
		LED pulse	0xC	W	0=Off, 1=On, 2=Wait for a LED FIFO command, 3=Pulse. Generate a square pulse of 100ns on J6-10
ADC1	0xD8600000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Bits 23 - 0 = Release date
		Control	0x8	W/R	0x000=Normal mode, 0x100=Reset ADC, 0x200=General Reset, 0x300=Reset pointer, 0x400=Re-align clock
		Data1	0xC	R	Read sample from channel 1
		Data2	0x10	R	Read sample from channel 2
		...	...	...	...
		Data8	0x28	R	Read sample from channel 8
ADC2	0xD9600000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Bits 23 - 0 = Release date
		Control	0x8	W/R	0x000=Normal mode, 0x100=Reset ADC, 0x200=General Reset, 0x300=Reset pointer, 0x400=Re-align clock
		Data1	0xC	R	Read sample from channel 8
		Data2	0x10	R	Read sample from channel 9
		...	...	...	...
		Data8	0x28	R	Read sample from channel 15
TTC	0xE8600000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Bits 23 - 0 = Release date
		A channel	0x8	W/R	Register to manipulate A commands: Single and multiple L1As, fixed rate triggers
		B channel Asyn	0xC	W/R	Register to write asynchronous B channel commands
		B channel Syn Control	0x10	W/R	Control of the FIFO of synchronous commands. Bit 31 = 1 execute stored commands on each orbit. Bit 30 = 1 reset the FIFO.
		B channel Syn FIFO	0x14	W/R	Command (bits 31-23) and offset wrt orbit in units of BC (bits 15-0) to store in the FIFO. Special commands (longer than a short command) start at bit 16: L1A = 0x7FFF, LED trigger = 0x0FFF.
SysMon	0xE8610000	ALMout	0x8	R	Temperature alarm output status
		Temp	0xC8	R	Present temperature value
Orbit	0xE8620000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Internal status of the IP core
		Control	0x8	W/R	0=Off; 1=Generate the Orbit clock; 2=Single pulse
Glink	0xF8600000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Bit 31 = GTX is aligned, bit 30 = GTX is realigned, bit 29 = GTX detects comma words, Bits 23 - 0 = Release date
		FIFO_Status	0x8	R	Number of 32-bit words available to read. Diff pos write and pos read
		Control	0xC	W/R	Bit 0 = reset FIFO, bit 1 = reset CRC, bit 2 = CRC mode for long barrel, bit 3 = CRC mode for extended barrel
		Data	0x10	R	Read one 32-bit word from pos read and increment the pointer if pos read < pos write
		Error_Global	0x14	R	Number of Global CRC errors
		Error_DMU_Even	0x18	R	Number of DMU CRC errors in even bits
		Error_DMU_Odd	0x1C	R	Number of DMU CRC errors in odd bits
		Checked_Events	0x20	R	Number of checked events
		DMU error mask	0x24	R	16 low bits to indicate which DMU is giving errors
		ERROR_BIT31_17_DMU1_2	0x28	R	
		ERROR_BIT31_17_DMU3_4	0x2C	R	
		ERROR_BIT31_17_DMU5_6	0x30	R	
		ERROR_BIT31_17_DMU7_8	0x34	R	
		ERROR_BIT31_17_DMU9_10	0x38	R	
		ERROR_BIT31_17_DMU11_12	0x3C	R	
		ERROR_BIT31_17_DMU13_14	0x40	R	
		ERROR_BIT31_17_DMU15_16	0x44	R	
		ERROR_PARITY_DMU1_2	0x48	R	
		ERROR_PARITY_DMU3_4	0x4C	R	
		ERROR_PARITY_DMU5_6	0x50	R	
		ERROR_PARITY_DMU7_8	0x54	R	
		ERROR_PARITY_DMU9_10	0x58	R	

		ERROR_PARITY_DMU11_12	0x5C	R	
		ERROR_PARITY_DMU13_14	0x60	R	
		ERROR_PARITY_DMU15_16	0x64	R	
		ERROR_MEMORY_DMU1_2	0x68	R	
		ERROR_MEMORY_DMU3_4	0x6C	R	
		ERROR_MEMORY_DMU5_6	0x70	R	
		ERROR_MEMORY_DMU7_8	0x74	R	
		ERROR_MEMORY_DMU9_10	0x78	R	
		ERROR_MEMORY_DMU11_12	0x7C	R	
		ERROR_MEMORY_DMU13_14	0x80	R	
		ERROR_MEMORY_DMU15_16	0x84	R	
		ERROR_SSTROBE_DMU1_2	0x88	R	
		ERROR_SSTROBE_DMU3_4	0x8C	R	
		ERROR_SSTROBE_DMU5_6	0x90	R	
		ERROR_SSTROBE_DMU7_8	0x94	R	
		ERROR_SSTROBE_DMU9_10	0x98	R	
		ERROR_SSTROBE_DMU11_12	0x9C	R	
		ERROR_SSTROBE_DMU13_14	0xA0	R	
		ERROR_SSTROBE_DMU15_16	0xA4	R	
		ERROR_DSTROBE_DMU1_2	0xA8	R	
		ERROR_DSTROBE_DMU3_4	0xAC	R	
		ERROR_DSTROBE_DMU5_6	0xB0	R	
		ERROR_DSTROBE_DMU7_8	0xB4	R	
		ERROR_DSTROBE_DMU9_10	0xB8	R	
		ERROR_DSTROBE_DMU11_12	0xBC	R	
		ERROR_DSTROBE_DMU13_14	0xC0	R	
		ERROR_DSTROBE_DMU15_16	0xC4	R	
LCD	0xF9000000	Dummy	0x0	W/R	Dummy register to test read and write functionality
		Status	0x4	R	Bits 23 - 0 = Release date
		Reg0	0x8	W/R	first row, characters 1 to 4
		Reg1	0xC	W/R	first row, characters 5 to 8
		Reg2	0x10	W/R	first row, characters 9 to 12
		Reg3	0x14	W/R	first row, characters 13 to 16
		Reg4	0x18	W/R	second row, characters 1 to 4
		Reg5	0x1C	W/R	second row, characters 5 to 8
		Reg6	0x20	W/R	second row, characters 9 to 12
		Reg7	0x24	W/R	second row, characters 13 to 16

Release date format is: Bits [23-16] = Day, Bits [15-12] = Month, Bits [11-8] = Years since 2012, Bits [7-4] = Version, Bits [3-0] = Revision.

Firmware versions

Version (date)	Change
20140814	Updated Server and stand-alone tests
20130719	Use clock generated in ML507 for ADC board
20130620	
20130531	
20130520	
20130419	Developed with Xilinx ISE/EDK version 13.1
20130121	First release

Operating System

Linux is the operating system used in the motherboard, as it has to provide flexible, easy to use software, if possible in C/C++ to be able to develop software external communication.

- Linux kernel 2.6.29.
- Minimal BusyBox 1.3.0 based root file system.
- Initial RAM disk (intird) file system.
- Installed with Embedded Linux Development Kit v4.2 which is patent free.

Available boot options:

- 32MB Flash memory or FTP server using Universal bootloader U-boot
- 512MB Compact Flash memory. Instructions how to prepare the CF card can be found here

Code repository and instructions on how to compile the linux image are available in the valtical cluster under /work/Willy/FPGAprog

Operating System Alternatives

XilKernel operating system with MicroBlaze soft-processor / PowerPC

- http://www.xilinx.com/tools/platform.htm
- http://www.xilinx.com/support/documentation/sw_manuals/xilinx13_4/ug758.pdf
- http://www.xilinx.com/support/documentation/dt_edk.htm

PetaLinux

- PetaLinux does not provide support for PowerPC, thus a MicroBlaze has to be programmed into the FPGA.
- PetaLinux is not license free.

Previous exercise was done for the digitizer testbench using PetaLinux and MicroBlaze with Xilinx IDE 9.2.

- Software in stockholm repository

Compact Flash format

Initializing a New Compact Flash Card

The Compact Flash card needs a special format in order to boot properly the system. The recommended procedure is:

1. Format the card first from Windows XP, using FAT16 (FAT32 will not work) and the default settings for the rest.
2. After that, running in Linux, one has to make use of the mkdosfs tools:

```
$ mkdosfs -a -v -R 1 -F 16 /dev/sdb1
```

Being sdb1 the name assigned to the card by the system (may have to change to a different sdxx). After that, one has to verify that there is only one single reserved sector by doing:

```
$ hexdump -n 32 -C /dev/sdb1
```

Answer in the terminal prompt should be:

```
00000000 eb 3c 90 6d 6b 64 6f 73 66 73 00 00 02 20 01 00 |.<.mkdosfs... ..|
00000010 02 00 02 00 00 f8 f5 00 3f 00 40 00 00 00 00 00 |........?.@.....|
```

1. Following, the system.ace file should be copied to the compact flash for the system to boot.

Updating MobiDICK Server on a Compact Flash Card

Updated MobiDICK software may be installed to flash memory using the following steps. Steps for updating the MobiDICK server can be seen in Server (MobiDICK) section. The following requires ISE Design Suite version 13.1. Further information for compiling the MobiDICK image, making changes to the IP / MAC address, or changing the kernel configuration can be found in /work/ELDK/images_fernando/README on valtical.

1. Scripts for updating the MobiDICK .ace image are found on valtical at /work/ELDK/images_fernando/ .
2. Place the download.bit file in input_files/ (this should not need to be updated) .
3. Place the new MobiDICK server in input_files/soft/ .
4. Execute genaceimage.sh .
5. Copy output_files/myace.ace to the Compact Flash Card .

LCD display

A 2x16 character LCD backlight display is placed on the front-panel of the box.

Components:

- 32 characters backlight LCD display Farnell 2218964
- 14 way socket straight FARNELL 1098499
- 14 way plug right angle ~~FARNELL 9789421~~ FARNELL 8396000

Pinout to connect to motherboard

Pin on LCD	Pin on Motherboard	Power
1	2	
2	1	
3 to 14	3 to 14	
15		+5V
16		Ground

SFP module

A bi-directional optical transceiver module in Small Form-factor Pluggable (SFP) format is used for digital communication with the super-drawer interface card. Two cable latches are present, one for the receiver section (RX) and the other for the transmitter section (TX). Two coaxial optical cables are connected with a 6.25mm separation gap. This module has to be purchased separately. The current version is type SX working at a wave-length of 850nm feeding from 3.3V.

Working model is Infineon SFP SX 850nm 1.06/1.25/2.12 Gb/s Tri-Rate V23818-M305-B57.

ADC board

A daughter board with two ADCs with 8 12-bit channels digitizes the signal from the trigger towers. The differential input is amplified and provided to the ADCs where analog signals are digitized. These digitized samples are serialized and sent to the motherboard via an expansion connector at data rate of 640Mbps. A 40MHz local oscillator is used to provide the clock to both ADCs. The L-clock for the serialization is derived from this clock. A dual power supply is chosen to remove signal coupling problems and to improve the signal integrity.

The ADC board replaces the two diff2ADC boards and the CAEN 792 blade of the previous version.

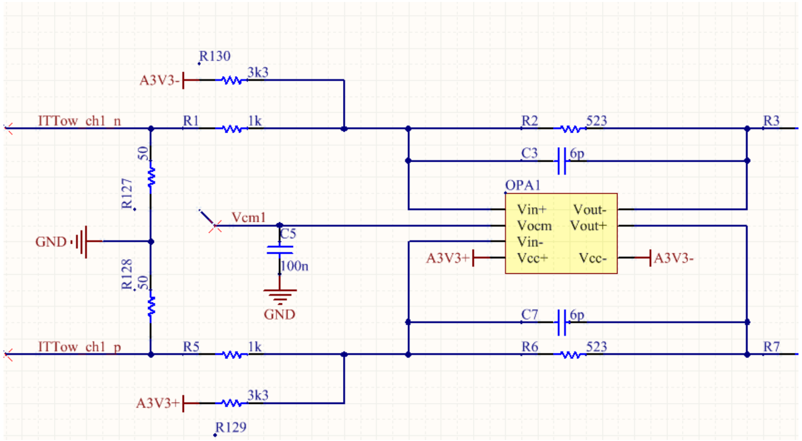
Requirements

- Input power voltages: +5V, -5V
- Input signal voltage range: [-2, +2] V
- Sampling capability of 16 signals at ~40 MHz synchronized with TTC clock
- Pass-through connector for HV and LED control

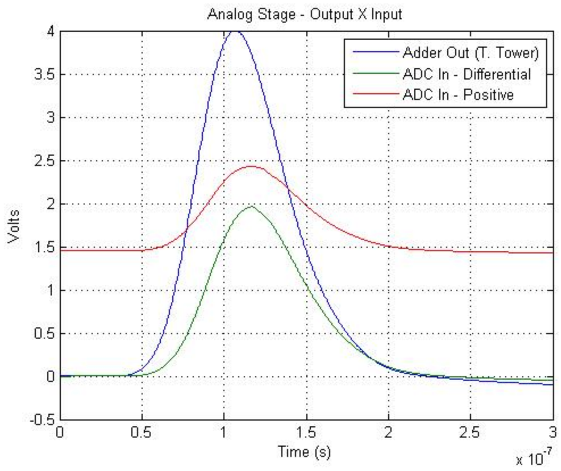
ADC board details

- Schematics of the ADC board.
- Trigger ADCs. Model ADS5271.
- Through hole connector used for the communication with the motherboard.
- Through hole HDD molex connector (4 way) for input power.
- Trigger cable input connector. RS reference: 740-7256
- Power regulators T46, LT1529, LT3015
- Operational amplifiers THS4151 compatible packages are: IDGK, IDGNR, IDGNRG4 and IDGKG4

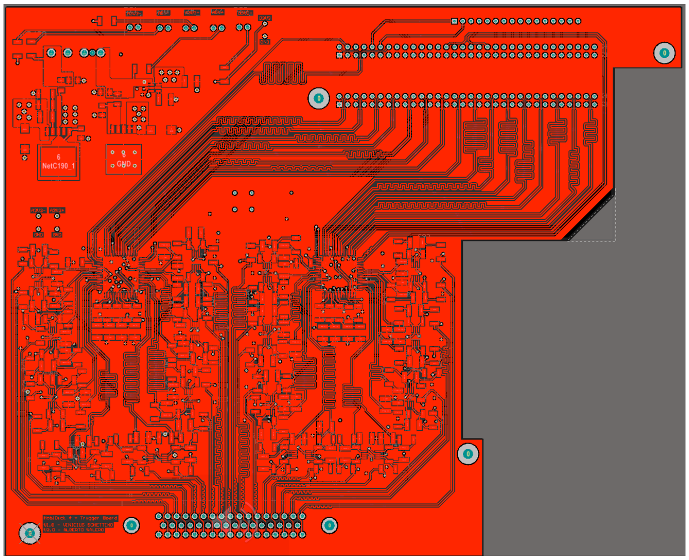
Schematics of the input state are shown below

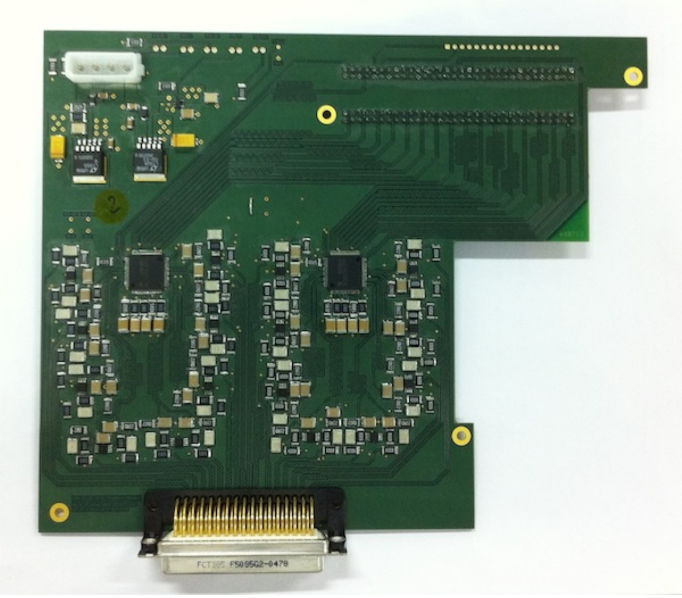


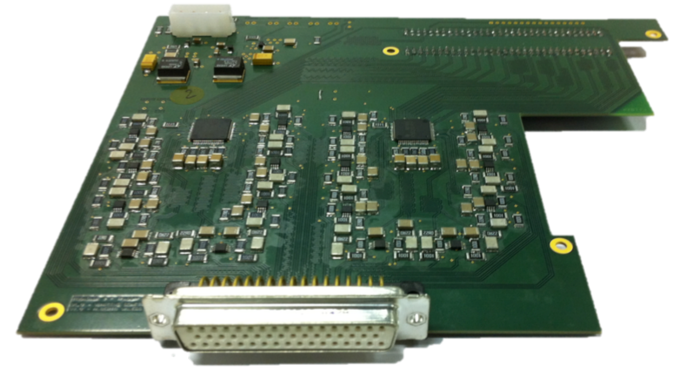
The following figure shows how the analog stage responds to a trigger signal with maximum energy. The input signal is labeled "Adder Out" and it ranges from 0 to 4V. The analog stage conforms this signal to the ADC's maximum input value, 2 Vpp, and add a commom mode of 1.5V, a requirement of the converter (ADS5271).



Follows the pcb design and pictures of the board.

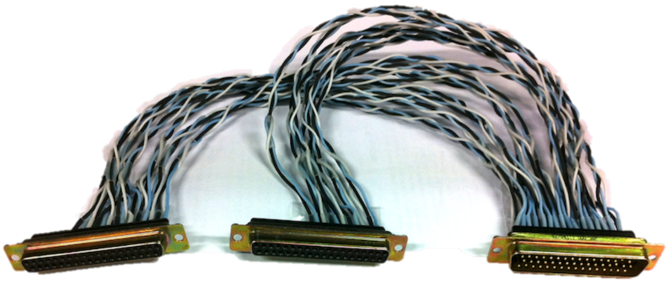




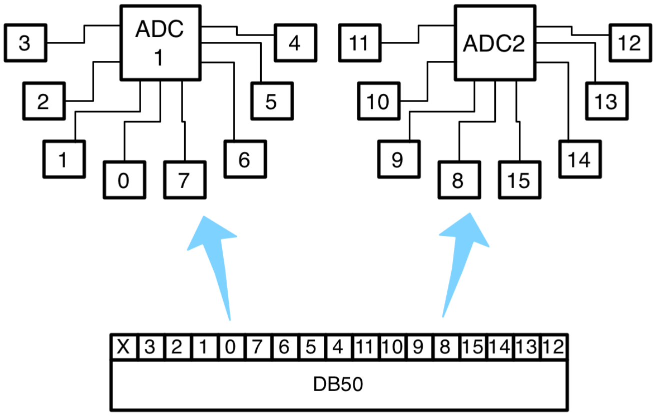


To connect the ADC board to the mobidick box we need a short conversion cable from two female db37 connectors to one male db50 connector.

- TE DB37 female housing ref 205209-2 SCEM 09.21.21.130.5
- TE DB50 male housing ref 1658641-3 SCEM 09.21.21.180.5
- ITT CANNON female pins ref 030-1953-000 SCEM 09.21.21.310.3
- ITT CANNON male pins ref 030-1952-000 SCEM 09.21.21.315.8



The mapping of the DB50 connector pins and the ADC chips inputs is showed in the following sketch:



Mapping of the ADC channel to PMT in Long and Extended Barrels. More info on the trigger tower to channel here

ADC[0-15]	Board index[TowMu 1-9]	EBX pmt[1-48]	EBX adder[0-7]	LBX pmt[1-48]	LBX adder[0-8]
0	muon 5	37	4 (muon)	26	3 (muon)
1	muon 6	38	3 (muon)	15	2 (muon)
2	muon 7	18	2 (muon)	14	1 (muon)
3	muon 1	17	1 (muon)	1	0 (muon)
4	tower 1	5,6,3,4,17	1 (tower)	1,2,3,4,5	0 (tower)
5	muon 2	-	-	43	6 (muon)
6	muon 3	-	-	40	5 (muon)
7	muon 4	1	0 (muon)	27	4 (muon)
8	tower 6	21,22,33,34	5 (tower)	20,21,22,23,27	4 (tower)
9	tower 7	7,8,15,16,37	4 (tower)	16,17,18,19,26	3 (tower)
10	tower 8	11,12,23,24,38	3 (tower)	10,11,12,13,15	2 (tower)
11	tower 9	9,10,18	2 (tower)	6,7,8,9,14	1 (tower)
12	tower 2	-	-	38,39,45,46,47,48	8 (tower)
13	tower 3	-	-	34,37,41,42	7 (tower)
14	tower 4	13,14,1,2	0 (tower)	28,31,35,36,43	6 (tower)
15	tower 5	29,30,43,44,41,42	7 (tower)	24,25,29,30,40	5 (tower)

Documents from the current L1Calo system

- Document repository at University of Heildelberg
- L1Calo pre-processor module
- PreProcessor analog input schematics
- More on the AnIn daughter board
- The Readout Manager FPGA, also details of the PPM
- Trigger cable specifications

High Voltage Board

The HV board provides the voltages for the operation of the PMTs.

First version designed by LPC. Contact person is Romeo Bonnefoy. HV and LED driver (DRAFT) and (Schematics)

Ultravolt HV cell 1AA24-N30-M.

Second version designed by Robert Reed (Wits)

- Schematics
- Layout

Requirements

- Input voltage: +24V@1A
- Input control: +3.3V (TTL) level provided by motherboard turns on/off the HV.
- Decoupled control input with HV off state in unknown state.

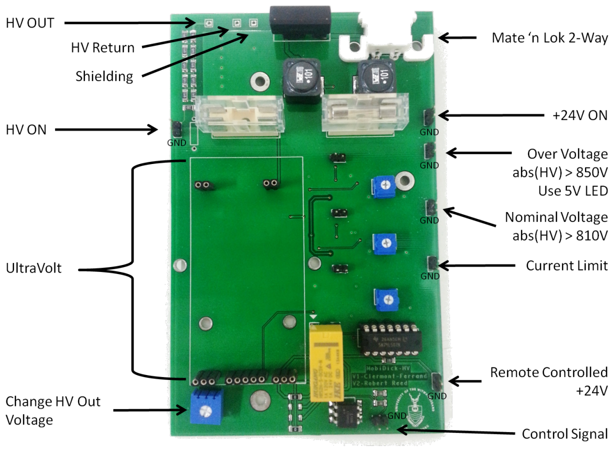
Rough estimate

- PCB : 15 EUR
- HV cell: 500 EUR
- Other components: 13 EUR
- Mounting: 19 EUR
- Total (Excluding HV cell): 47 EUR

Components:

- Input power receptacle on PCB is Mate-n-lok 2way 90degrees: RS 681-2345 (Cannot be found anymore)
- Input power cable connector is Mate-n-lok 1row 2way RS 280-1941
- Alternative input power cable connector is Mate-n-lok 1row 2way RS 280-5683
- Male pins for Mate-n-lok connector RS 172-9336
- HV cable for output with signal, return and shield (Standard used in TileCal) Ref: TWINAX 2x20 AWG 1kV DC IEC 60332-1 NOVACAVI MILANO

Produced board:



Testing Procedure:

1. Connect power 24V at 1A.
2. Connect a 3.3V to the Input pin.
3. Listen for relay to click over.
4. Check if HV ON LED turns on.
5. Check voltage between HV Out pin and GND
6. Adjust resistor under UltraVolt to alter the output voltage.

LED board

The LED board provides a sharp voltage peak to a passive LED device that illuminates the PMTs. It is controlled by the rising edge of a pulse from the ML507 in J6-12.

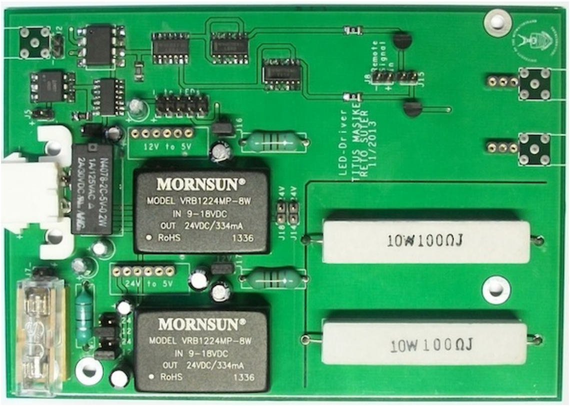
First version was designed by LPC. Contact person is Romeo Bonnefoy. HV and LED driver (DRAFT) and (Schematics)

Second version designed by Reto Suter and Titus Masike based on the concept by Andrey Shalyugin (Schematics) and User's guide. The measured propagation time through the board is 70 ns ± 10 ns.

Requirements:

- Input voltages: +24V@500mA
- Input control: +3.3V (TTL) 100ns wide pulse provided by the motherboard is used to fire the output voltage for a LED pulser. Nothing will happen if a constant level is asserted.
- Decoupled control input with power consumption off in unknown state.

Produced board:



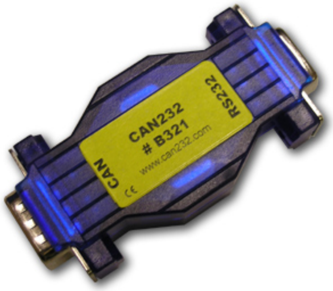
CANBus interface

Each super-drawer has 2 different CAN bus interfaces merged into a single connector. One bus establishes connection with the mother board side of the drawer, through the ADC-I card. From this bus we have access to all 3-in-1 cards of the drawer. The second bus communicates with the high voltage side of the drawer, through the HV Micro card. This one allows us to control the high voltage applied to each individual PMT, and thus their gain. Description of the HV_micro commands can be found here

We use two commercial CAN232 dongles to communicate with the HV and integrator ADC through the RS232 ports of the motherboard. These two CANBus dongles replace the TVME200 card which holds the TIP816 in the previous version.

Estimated cost:

- Dongle: 100 EUR
- Db9 and db15 connectors and other cabling: 100 EUR
- Db9 plastic cover: SCEM 09.21.23.140.5

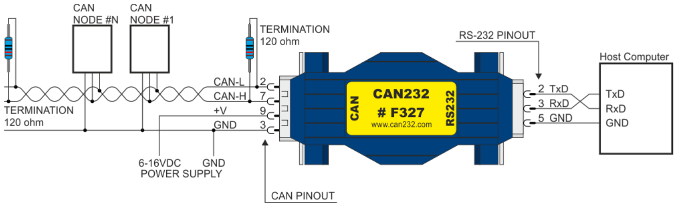


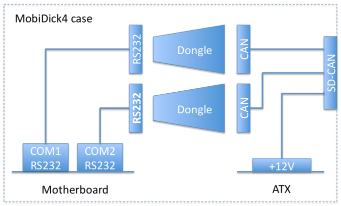
Custom cabling is needed for the connection between the super-drawers and the motherboard. These include the cables from the super-drawer to the patch-panel, from the patch-panel and the dongles (db15 to two db9) and the dongle and the motherboard (db9 to IDC10). More documentation on these cables can be found here. CANbus standard on DB9 can be found here. Note that the dongles are powered from the CAN side from the ATX.

Cabling patch-panel to dongles (db15 to two db9).

- Pins CAN_L (2) and CAN_H (7) must be connected via a 120 Ohm resistor.
- Pins CAN_V (9) and CAN_G (3) must be connected to +12V and G respectively.

Signal	Pin in db15	Pin in db9 (ADC)	Pin in db9 (HV)
Reserved	1		
CAN HV L	2		2
CAN HV G	3		3
Reserved	4		
CAN shield	5	5	5
ADC CAN G	6	3	
ADC CAN L	7	2	
Reserved	8		
Reserved	9		
HV CAN H	10		7
Reserved	11		
HV CAN V+	12		9
Reserved	13		
ADC CAN V+	14	9	
ADC CAN H	15	7	





Once the cabling has been setup, the dongles need to be programed to their new default baud rate of 115 kbps for serial communication with the motherboard. The super-drawer interfaces are 250 kpbs, and the default serial rate is 56 kbps. This can be achieved with the `test-dongle` binary in the server.

Power distribution

For the power distribution we rely on a commercial ATX power supply and a commercial 24V power supply.

Voltage requirements by the different boards are listed in the following table.

Voltage	Source	Color	ML507	Router	ADC board	LS150	HV driver	LED driver
+5V	ATX	Red	yes	yes	yes			
-12V	ATX	Blue			yes			
+12V	ATX	yellow						
GND	ATX	black	yes	yes	yes	yes (-V)		
220VAC	ATX	-				yes		
+V	LS150	purple					yes (+24V)	yes (+24V)
-V	LS150	black					yes (0V)	yes (0V)

ATX has to be modified to provide through a hole in the top plate voltages 220AC (two wires) and 0V. These are terminated with a LUGS connector routed to the 24V power supply indide the box and the rest of the power cables are grouped as follows:

- Cables for LS150: Connectors LUGS EDH 04.76.22.224.1

Voltage	Color	LS150 Pin
220 VAC+	Grey	L
220 VAC-	Blue	N
0V	Black	V-

- Connector for ML507 and Router : Pole Lumberg Farnell 1568397

Voltage	Color	Pinout
0V	black	GND
+5V	red	Pole

- Connector for ADC board: AMP Mate-n-lock 1-480424-0 with female pins

Voltage	Color	Pin#
-12V	blue	1
0V	black	2
0V	black	3
5V	red	4

- Connector for LED board: Mate-n-lock 2 way with male pins

Voltage	Color	Pin#
5V	red	1
0V	black	2

- Connector for HV board: Mate-n-lock 2 way with male pins

Voltage	Color	Pin#
+24V	pink	1
0V	black	2

Networking

The motherboard needs TCP/IP connectivity with the client, this is achieved with commercial networking hardware and some configuration. In order to have one single firmware version, the firmware forces ethernet physical address to CA:CA:CA:CA:CA:CA, and a router placed inside the box provides static IP to the motherboard (192.168.0.102) via a DHCP service. Two network ports are made available through the front-panel, one connected to a LAN port of the router and another one connected to the internet port. Furthermore, the router forwards the ports needed by the system to the motherboard IP address.

Forwarded ports are:

- 21: FTP
- 23: telnet
- 1570: MobiDICK server

In order to make MobiDick4 wireless, a conventional router Linksys N600 is used as wireless network adapter. Instructions on how to flash this device with proper firmware are listed here. Specific information on the firmware is found here.

Components

- DLink router model Dir-100/E placed inside the box link.
- Cisco Linksys model N600 for WIFI connectivity link.

Casing and assembly

The system is closed inside a box with minimum a projective size of 208.28 mm x 146.045 mm with 5 mounting holes at least (given by motherboard)

- LED holder: 8mm diameter
- LEDs: EDH document
- HV: Female LEMO conector for cable FFA.2S.302.CLAC62 (Hole for cable diameter is 6.7mm in the current one)
- HV: Male socket for box ERN.2S.302.CLL
- Ethernet ports: RJ45 cable to cable category 5 socket
- LED test: Female LEMO connector for LED signal output EDH 09.46.11.130.6
- ML507 reset button: 15mm diameter
- Fuse holder: EDH 067.73.01.105.3

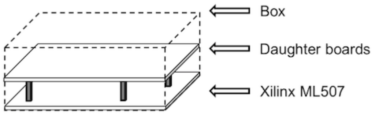
Inside the box

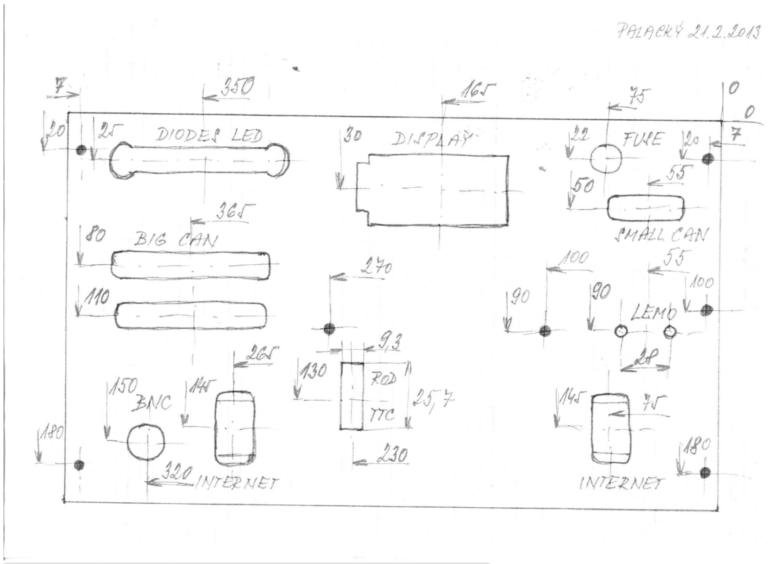
- 2 way housing for 2.54 mm pins: HARWIN M20-1060200
- Female pins for 2.54 mm pins: HARWIN M20-1180042 or HARWIN M20-1180046
- LS150-24 AC/DC: link
- Trigger cable connector: EDH 09.21.21.130.5
- Trigger cable pins ITT CANNON 030-1952-000 link
- ATX female socket cable to cable: link
- ATX pins: link

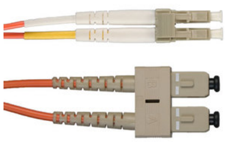
- Router for Wifi: Distrelec 842482
- Router for DHCP: Distrelec 848412
- Short network cables (3 for each box): EDH 80.02.08.013.0
- Short ROD and TTC fibers inside the box: Short patch-cord fiber pair LC-SC
- Power connector for router and ML507: link

Other non-referenced references from farnell are: 4138442, 1012168, 2101972, 2008466, 299479, 148085

Casing ilustration





 LED LEMO	 HV LEMO
 LC-SC patch-cord	 Reset button

Stickers

The front-panel needs the following stickers

- Trig.Muon
- Trig.Calo
- HV
- ETH
- MobiDICKX
- TTC
- ROD
- Fuse
- CAN
- LED

- Adm.ETH
- Reset

Assembling a new box

1. Open the ATX power supply and solder 220VAC pair of cables to input and route them to the outside.
2. Connect the pair of cables to the LS120 power supply
3. Mount the following components on the box front-panel
    1. Admin (cat 4) and data (cat 5) ethernet passthrough connectors
    2. Optical feed-through connector
    3. 2 LEMO connectors
    4. Reset button and Reset CAN button
    5. HV female connector

LEDs on the front-panel

Position (left to right)	Label	Color	Cabling
1	CAN fuse	Green	CANbus power after fuse + ground
2	HV fuse	Green	HV board J1 (+24V on)
3	HV on	Blue HLMP-CB50	HV board J6 (HV output)
4	OVC	Red	HV board J4 (over voltage)
5	LED on	Blue HLMP-CB50	LED board J2 (trigger input)

Laptop

A laptop is needed to run the front-end GUI software (Willy). These are the following:

Name	Model	Comments
mobidick04pc	HP PROBOOK 4545S	Not recommended
mobidick05pc	HP PROBOOK 4520S	Not recommended
mobidick06pc	DELL LATITUDE E6540	
mobidick07pc	DELL LATITUDE E6540	

Recommended laptop configuration

This works with all laptops

1. Install SLC6 1. Install necessary packages:

```
yum install -y xterm telnet spawn-fcgi root root-physics
```

1. Install argparse module for Python 2.6

```
easy_install argparse
```

Specific information for MOBIDICK04PC

Purchased model is HP4545s. We had problems with the graphics card + synaptics touchpad, and Ralink wireless card (model 3290 that can be identified by lspci). After many trials, the selected OS was SLC5, supported at CERN.

Mouse problem was fixed by removing the synaptics configuration from /etc/X11/xorg.conf.

Wireless kernel module can be found here (details). Instructions to install it are the following:

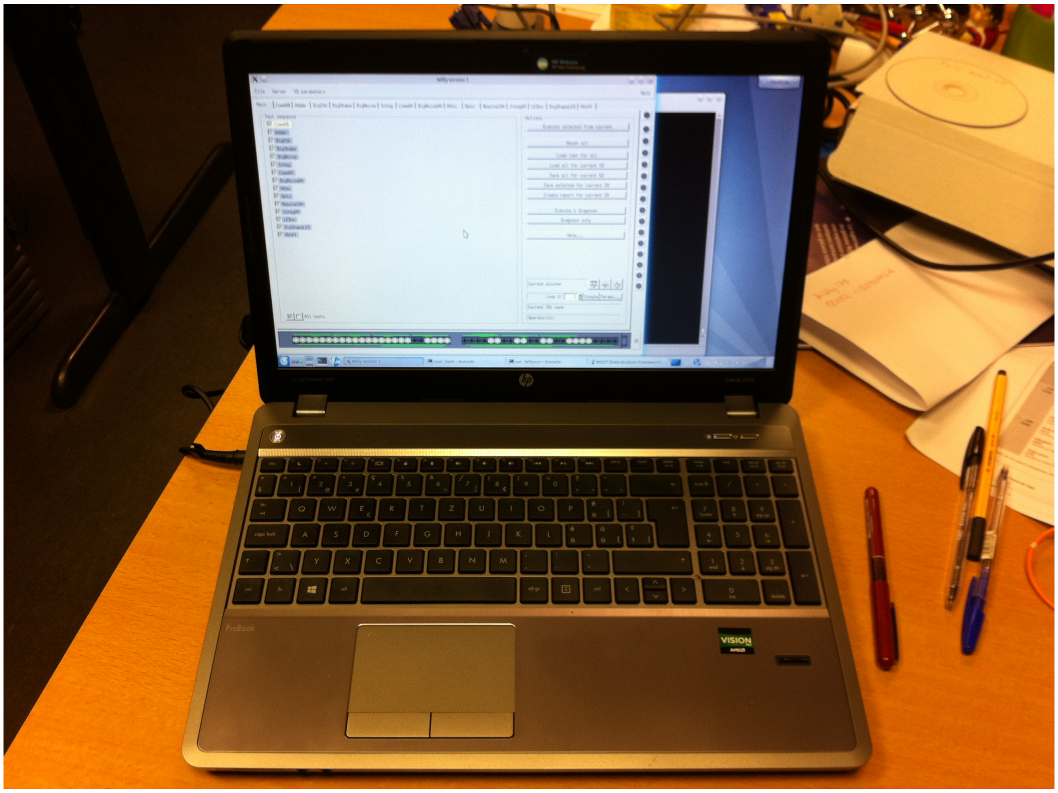
1. Extract the package contents.
2. Change os/linux/config.mk and set HAS_WPA_SUPPLICANT=y and HAS_NATIVE_WPA_SUPPLICANT_SUPPORT=y (for NetworkManager)
3. Edit RT2860STA.dat to remove any hardcoded settings: SSID... (should all be auto).
4. Compile with: make
5. Install module with: make install (as root); this will:
    1. Place inside: /lib/modules/`uname -r`/kernel/drivers
    2. Run: depmod

Additionally:

1. Disable network service, we will use NetworkManager: service network stop; chkconfig --level 2345 network off; chkconfig --del network
2. Install NetworkManager if not there: yum install NetworkManager
3. Create the file `/etc/sysconfig/modules/wifi.modules` with the following contents

    ```
    #!/bin/sh
    exec modprobe rt3290sta
    ```

4. Add to /etc/modprobe.conf : alias ra0 rt3290sta
5. Reboot



Specific information for MOBIDICK05PC

Model is HP4520s model Delphi D40 Rev AM5. Wireless adapter is Ralink radio model RT3090BC4.

To compile the wireless drivers follow these instructions.

To start the networkmanager follow these instructions.

Additionally

1. Disable network service, we will use NetworkManager: service network stop; chkconfig --level 2345 network off; chkconfig --del network
2. Install NetworkManager if not there: yum install NetworkManager
3. Create the file `/etc/sysconfig/modules/wifi.modules` with the following contents

```
#!/bin/sh
exec modprobe wl
```

1. Reboot

Details for Ubuntu

We had troubles with SLC and WIFI adapter. Administrator user is admin, pass is Amparo's password.

1- Remove the following packages:

```
sudo apt-get purge bcmwl-kernel-source broadcom-sta-common broadcom-sta-source
```

2 - Install the following packages:

```
sudo apt-get install b43-fwcutter firmware-b43-installer sudo echo b43 >> /etc/modules
```

3 - Use the wl driver, execute:

```
sudo modprobe -r wl
sudo modprobe b43
```

Reboot the machine and the wireless should be working.

To remove annoying messaging system:

```
sudo apt-get remove indicator-messages
```

Software

MobiDICK4 software is split into server (MobiDICK) and client (Willy) applications. Willy sends commands to the server. The server controls the front-end, reads data out and transfers back results and parts of the data to Willy.

Software repository is available in SVN.

```
https://svnweb.cern.ch/trac/atlasgroups/browser/Detectors/TileCal/MobiDICK
```

Installation instructions:

1. Create a MobiDICK directory and change into it
   ```
   mkdir MobiDICK; cd MobiDICK
   ```
2. Checkout the Release package from the SVN repository
   ```
   svn co https://user@svn.cern.ch/reps/atlasgroups/Detectors/TileCal/MobiDICK/Release/trunk Release
   ```
3. Run the initialization script specifying the requirements file and the installation directory (Run it with "-h" to see the options)
   ```
   python Release/scripts/init_release.py Release/cmt/requirements .
   ```
4. Run the compilation script specifying to install CMT and the installation directory
   ```
   python Release/scripts/compile_release.py -installcmt .
   ```
5. It Failed? Try previous step again

Running instructions:

1. Change into the Run directory that was checkout by the initalization script
   ```
   cd MobiDICK/Run
   ```
2. Run Willy.exe
   ```
   Willy.exe
   ```

Server (MobiDICK)

The glue software is the server daemon that runs on the Motherboard. This includes the libraries for hardware access to the different boards and the MobiDICK server daemon that is the back-end application.

Part of the data analysis is performed on the server, which is implemented in C with posix limitation.

- A hierarchy diagram showing a complete track of functions calls for the main tests executed on the server side of Willy can be found here.
- A simplified and printable version can be found here.
- A description of the functionalities of the server can be found here.

Compilation and test instructions are:

- ssh -Y tilerod@valtical01
- cd /work/Willy/Server/src/
- ./make_g++ ppc
- ./inst.py MobiDICK mobidick04.cern.ch (will copy to /tmp)

This is the command to start the server from Willy:

```
xterm -e "expect -c 'spawn telnet -l root mobidick05; expect \"Password:\"; send \"root\r\"; send \"./soft/MobiDICK\r\"; interact'"
```

Server software versions

Version (date)	Change
2013-12-02	Speed-up the integrator tests by factor 2

Client (Willy)

Willy is the user friendly GUI for MobiDICK. It is implemented in C++ using ROOT libraries as window manager and analysis framework.

Requirements for Willy

1. ROOT 5.22 or greater
2. xterm
3. telnet
4. spawn
5. CMT

Instructions to install Willy

1. Download and install CMT
2. Download and install ROOT >5.22.00
3. Checkout MobiDICK repository
4. Compile Willy from `MobiDICK/Willy/cmt` with `cmt bro make`
5. Add `$HOME/Willy/Willyrun` to the `Startup programs` under `System > Preferences > More Preferences > Sessions`.

Client software versions

Version (date)	Change
20140124	Stable tag of Willy
20140827	New trigger and stuckbits tests

Other software tools

It is possible to operate with the box through telnet. All you need is to connect your laptop to the ethernet input and access it at its local ethernet address.

```
telnet -l root 192.168.0.102
```

An expert software tool called `test-device` allows the direct communication with the registers in the FPGA.

```
root:~> /soft/test-device -h
Usage: /soft/test-device <baseaddress> <arguments>
 arguments:
    read offset
    write offset value
```

Front-end electronics tests

The different test implemented in Willy are summarized in the following table, where ID represents the name displayed in Willy and the device that it uses is enumerated.

ID	Description	TTC	Glink	CAN	HV	LED	ADC	Status
CommMB	Communication with MB CAN			X				Working
CommHV	Communication with ADC CAN			X				Working
Adder	Trigger tower injection test	X					X	Working
DigChk	Data integrity test	X	X					Working
DigShape	Charge injection test	X	X					Working
DigShapeLED	LED test	X	X			X		Working
Noise	Noise test	X	X					Working
NoiseHV	Noise test with HV	X	X		X			Working
Integrator	Integrator read-out test	X		X				Working
IntegratorHV	Integrator read-out test with HV	X		X	X			Working
HVon	Turn on HV				X			Working
HVoff	Turn off HV				X			Working
Opto	Communication with optocoupler			X				Working
NominalHV	HV regulation test			X	X			Working
LEDon	Turn on LED					X		Working

Digitizer check (DigChk)

This is a test of the digital read-out. It comprises 5 sub-tests, four of which use the same digitizer mode (calibration) and different read-out mode from 0xA8 to 0xAB. The last sub-test is performed in the test mode of the digitizers (rolling bits) and read-out mode 0xA8. For each sub-test, one single event is read-out and data quality is assessed by checking the CRC of each of the 16 DMUs along with the global one.

For each of the four first sub-tests the functions DIGI and DIGK that control the trigger and read-out of the data respectively, are executed sequentially. For the last sub-test, DIGI is followed by DIGR for the read-out of the special data.

Digitizer shape (DigShape)

This test checks the two amplification circuits of the digital read-out via the Charge Injection System. At the beginning of the test, the digitizers are configured in auto-gain and read-out mode 0xAB. Two events are triggered, one for a small charge that will read-out the high gain samples, and another one with a big charge that will read-out the low gain samples. Each event is analyzed separately, and compared to the expected pulse height and width for the given charge. This test also detects a miss-connection between the 3-in-1 and the digitizer channel.

For this test the execution of functions does not follow a sub-test structure as DIGI is executed in the first place, followed by two calls to DIGA.

Digitizer Noise (DigNoise)

This test checks the average RMS of the samples of each channel over many events. All the samples are returned to Willy for offline analysis in a software controlled trigger (trigger, read-out, data transfer, trigger,...)

1. DIGI is executed
2. DIGN is executed for N events, all the samples are returned for each channel every time.

Stress test (FastTest inside DigNoise)

This implements the CRC checking code

Integrator motherboard side communication (CommMB)

This is a test of the communication with the Mother Board side of the drawer. It performs several actions:

1. The communication with the ADC-I using CANbus is established
2. The serial number of the ADC-I is read out, the super-drawer associated with this serial number is searched for in the local or central databases
3. The super-drawer parameters are compared between the central and local databases
4. The firmware version of ADC-I is checked
5. The 3in1 cards are configured with CAN bus and their configuration are read back with CAN bus
6. The 3in1 cards are configured with TTC and their configuration are read back with CAN bus
7. The TTCrx address of the mezzanine is checked.

There is also the possibility of execute only the CAN bus part of the test.

Integrator high voltage side communication (CommHV)

This is a test of the communication with the High Voltage side of the drawer, and also a test of its status. It performs several actions:

1. The communication with HVmicro using CAN bus is established
2. The software version of HVmicro is checked
3. The global status register of HVmicro is checked
4. The input low voltages are checked
5. The temperature probes are checked
6. The serial numbers of the HVmicro and the two HVopto cards are checked

Integrator linearity (Integ)

The purpose of this test is to check the linearity and noise level of the ADC-I and of the charge integrator circuits of the 3in1 cards. This is done sequentially on each 3in1 card:

1. Pedestal RMS is computed over one hundred measurements, results are sent back as single string value. Remote test checks RMS value.
2. DAC value is raised from 0 to maximum value for each of the five gains with a different number of steps for each gain. Measurements are returned independently for each DAC value and gain. Remote test checks linearity, slope and pedestal value for each gain.

The panel is common with the integrator HV test (TestInteg).

Integrator HV (IntegHV)

The purpose of this test is to check the noise linearity and noise (pedestal RMS level) of the ADC-I with the high voltage distribution electronics and the high voltage input enabled. This is done sequentially on each 3in1 card.

1. Pedestal RMS is computed over one hundred measurements, results are sent back as single string value. Remote test checks RMS value.

Note it is also possible to change the parameters of the test to perform the linearity test too. The panel is common with the integrator linearity test (TestInteg).

Opto and NominalHV

The purpose of these tests is to check the functionalities of the high voltage distribution electronics.

For the Opto test:

1. The four HV switches (odd/even internal/external) are all switched off. Then, they are switched on one by one and verified;
2. The HV is set to 700 V on all the channels. Then, for each channel, the measured voltage is read out and compared to the expected one;
3. The HV is set to 600 V on all the channels. Again, for each channel, the measured voltage is compared to the expected one.

For the NominalHV test the same action is performed, but the voltage value set is the one stored in the HVMicro's EEPROM. The value read back is compared against the super-drawer data base.

List of MobiDICK units

List of MobiDICK units

Name in LanDB	MAC address	Description
MobiDICK04	00-0A-35-01-F4-F7	1st production unit. HV and LED from LPC
MobiDICK05	00-0A-35-02-3D-7F	2nd production unit. Missing HV and LED
MobiDICK06	1C-7E-E5-88-DD-55	3rd production unit. Missing HV and LED
MobiDICK07	C8-D3-A3-28-A7-C7	4th production unit. Missing HV, LED, Router

Turn on sequence

With wired connection

1. Power up MobiDICK box
2. Power up laptop
3. Connect ethernet cable between MobiDICK box ethernet port (not the admin port) and to the laptop port
4. Disable wireless connection on laptop
5. Press reset button on MobiDICK box (and wait 30 seconds)
6. Open Willy in the laptop and start server

With wireless connection

1. Turn on power to MobiDICK box
2. Wait 30 seconds
3. Press reset button
4. Wait 30 seconds
5. Try to connect remotely. It could happen that network connection is broken, in this case repeat procedure from first step.

Notes

So that we are all on the same page. These are the voltages of the different logical levels.

Technology	Low voltage	High voltage	Notes
TTL	0 V to 0.8 V	2 V to VCC	VCC is 4.75 V to 5.25 V
ECL	-1.175 V to -VEE	0.75 V to 0 V	VEE is about -5.2 V. VCC=Ground

Further Needs

1. Adder test : Fix PMT to ADDER mapping
2. Adder test : Add more infomration when we get "more than one adder reply"
3. Adder test : Test of noise should be made optimal and have a switch to activate
4. CommMB : Include a paramtter for which type of drawer, and in case of Demonstrator drawer do not run can_mse_high, can_mse_low
5. Server's INTtest::INTS should take hard gains as a parameter, and Willy's DoTestInteg::singleTest should accept drawer type as a parameter, and send out the relevant hard gains
6. DigNoise test : fails because it checks old format errors which are not reproduced in the new sROD packer. Emulate CRC and DMU headers, or remove checks.
7. DigChk test : add pattern functionality to DB or remove checks.

Major updates:
-- CarlosSolans - 24-Jun-2011

Responsible: CarlosSolans
Last reviewed by: **Never reviewed**

I	Attachment	History	Action	Size	Date	Who	Comment
	ADC-board-perspective.png	r1	manage	472.1 K	2014-02-28 - 15:43	CarlosSolans	
	ADC-board-small.jpg	r1	manage	98.3 K	2014-02-28 - 15:48	CarlosSolans	
	ADC-schematic-sample.png	r1	manage	39.1 K	2013-05-24 - 15:51	CarlosSolans	ADC schematic sample
	ADCboard-to-box-cover.png	r1	manage	385.7 K	2014-02-18 - 20:32	CarlosSolans	ADC board to box cable
	ADCboard_layout.png	r1	manage	383.2 K	2014-02-28 - 12:33	AlbertoValero	ADC board layout version2
	BlockDiagram.pdf	r1	manage	39.6 K	2011-07-20 - 19:41	UnknownUser	
	CAN232-connection-diagram.png	r1	manage	52.7 K	2013-09-11 - 10:18	CarlosSolans	CAN dongle connection diagram
	CERN_digitest_preliminary_version_v2.pdf	r1	manage	9148.3 K	2011-06-24 - 16:08	CarlosSolans	Digitizer testbench documentation
	CablesPinout.pdf	r1	manage	31.5 K	2012-04-02 - 13:54	ViniciusSchettino	Schemes used for custom CAN bus cables
	DB50_ADC_mapping.png	r1	manage	61.6 K	2014-09-04 - 16:15	AlbertoValero	DB50 to ADC chip channel map
	HV-Board-V2-Layout.pdf	r1	manage	47.3 K	2014-08-04 - 14:56	RobertReed	Final HV Board Layout
	HV-Board-V2-Schematics.pdf	r1	manage	36.3 K	2014-08-04 - 14:53	RobertReed	Final HV Board Schematics
	HV-Board-V2.png	r1	manage	1905.5 K	2014-08-07 - 17:37	RobertReed	Final HV Board Design
	HV-LEMO-connector-FFA2S302CLAC.gif	r1	manage	10.0 K	2013-05-24 - 17:45	CarlosSolans	HV LEMO connector FFA.2S.302.CLAC
	HV.jpg	r2 r1	manage	63.2 K	2011-12-01 - 11:18	CarlosSolans	HV layout
	HV_LED_driver.pdf	r1	manage	62.5 K	2011-06-27 - 17:41	CarlosSolans	LPC HV and LED driver for new system (DRAFT)
	HV_part.JPG	r2 r1	manage	67.5 K	2012-11-21 - 13:37	CarlosSolans	HV board
	LED-LEMO-connector.gif	r1	manage	15.2 K	2013-05-25 - 00:14	CarlosSolans	LED LEMO connector
	LED_board_schematics.pdf	r1	manage	92.7 K	2014-08-22 - 14:46	CarlosSolans	LED board schematics
	LED_driver.jpg	r4 r3 r2 r1	manage	105.0 K	2014-02-14 - 19:51	CarlosSolans	LED driver layout
	LED_driver_User_Guide.pdf	r1	manage	752.4 K	2014-02-14 - 19:47	CarlosSolans	
	Mezzanines_for_Mobidick.pdf	r1	manage	1713.6 K	2012-09-26 - 17:51	CarlosSolans	LPC HV and LED driver schematics for new system
	MobiDICK4_logo_small.png	r1	manage	71.9 K	2012-03-30 - 14:31	CarlosSolans	MobiDICK4 logo small
	MobidickInstruction.pdf	r1	manage	14066.8 K	2014-07-04 - 11:49	CarlosSolans	
	Mobidick_schemas.zip	r1	manage	3647.7 K	2011-09-30 - 16:19	CarlosSolans	MobiDick schemas
	MobyDickPowerSupplySchema.pdf	r1	manage	257.2 K	2012-10-10 - 14:23	AlbertoValero	New Power Supply schema
	MobyDickPowerSupplySchema.png	r2 r1	manage	281.2 K	2012-10-10 - 14:48	AlbertoValero	New Power Supply schema
	Reset-button.jpg	r1	manage	6.7 K	2013-05-30 - 17:02	CarlosSolans	Reset button for ML507
	Stockholm_testbench.pdf	r1	manage	58.3 K	2011-06-24 - 16:09	CarlosSolans	Stockholm testbench
	TTCvi_config.jpg	r2 r1	manage	32.7 K	2012-07-18 - 18:31	FernandoCarrio	
	TWEPP-2008_-_Paper.pdf	r1	manage	1313.3 K	2011-07-06 - 16:27	CarlosSolans	Paper on Stockholm testbench
	TriggerBoardAnalogStage_MaxInputXOutput.jpg	r1	manage	38.1 K	2012-12-14 - 17:13	ViniciusSchettino	
	Trigger_Board_schematics.pdf	r4 r3 r2 r1	manage	1976.9 K	2013-07-17 - 13:42	CarlosSolans	ADC board schematics
	Willy_FlowChart_-_Simplified_Version.pdf	r1	manage	150.4 K	2012-02-28 - 13:39	ViniciusSchettino	
	Willy_Hierarchy_Diagram.pdf	r1	manage	170.9 K	2012-02-28 - 13:42	ViniciusSchettino	
	Willy_Hierarchy_Diagram_-_Simplified_Version.pdf	r1	manage	150.4 K	2012-02-28 - 14:32	ViniciusSchettino	
	box-front-panel-20130221.png	r1	manage	568.8 K	2013-04-24 - 16:52	CarlosSolans	Box front-panel 21 Feb 2013
	box_2_small.jpg	r1	manage	194.6 K	2012-07-18 - 15:24	CarlosSolans	Box picture
	cable_canbus_dongle_patchpanel.jpg	r1	manage	26.3 K	2012-07-18 - 10:55	CarlosSolans	Can bus cable - drawer to dongled
	cable_canbus_motherboard_dongle.jpg	r1	manage	20.2 K	2012-07-18 - 10:53	CarlosSolans	Can bus cable - dongle to mother board
	can-bus-dongle.png	r2 r1	manage	142.0 K	2012-11-02 - 09:58	CarlosSolans	
	canbus-pin-layout.png	r1	manage	26.6 K	2012-11-28 - 15:52	CarlosSolans	CANbus cabling
	carte_Led_driver.JPG	r1	manage	215.7 K	2012-03-30 - 14:23	CarlosSolans	LED driver produced

diff2adc_bloc_ampli.jpg	r2 r1	manage	69.7 K	2011-09-28 - 18:15	CarlosSolans	
dir100e.jpg	r1	manage	20.4 K	2013-08-28 - 10:50	CarlosSolans	DLink router inside box
hvbrdupgrade-schematics-20130524.pdf	r1	manage	69.2 K	2013-05-24 - 15:48	CarlosSolans	HV board schematics as of 24th May 2013
lc-sc-patchcord.jpg	r1	manage	12.2 K	2013-05-28 - 14:37	CarlosSolans	LC-SC patchcord
led-driver-proposal-1.png	r1	manage	39.0 K	2013-05-12 - 16:48	CarlosSolans	LED driver optimization proposal diagram
led-driver-proposal-2.png	r1	manage	69.7 K	2013-05-12 - 16:48	CarlosSolans	LED driver optimization proposal simulation
ml507_front.jpeg	r1	manage	726.0 K	2011-10-18 - 12:16	CarlosSolans	Virtex 5 ML507
mobidick4pc.JPG	r1	manage	1529.6 K	2013-01-17 - 19:18	CarlosSolans	Laptop
monstro3.jpeg	r1	manage	54.2 K	2012-03-30 - 12:43	CarlosSolans	MobiDICK4 logo raw image. Courtesy of http://www.amazing3dmodels.com/store/images/uploads/170/monstro3.jpg
power_cables.png	r5 r4 r3 r2 r1	manage	1149.1 K	2012-09-21 - 18:45	CarlosSolans	Power cables

Topic revision: r249 - 2016-11-10 - CarlosSolans

Ideas, requests, problems regarding TWiki? Send feedback