The Development of the Embedded Local Monitor Board (ELMB)

H.Boterenbrood [1], B.Hallgren [2]

[1] NIKHEF, P.O.Box 41882, 1009 DB Amsterdam,
[2] CERN, 1211 Geneva 23, Switzerland
boterenbrood@nikhef.nl, Bjorn.Hallgren@cern.ch

Abstract

The Embedded Local Monitor Board (ELMB) is a credit-card sized plug-on board, designed as a general-purpose CAN-bus based building block for various control and monitoring tasks in the LHC experiments. The latest and final version of the hardware in combination with the firmware shows to have even better radiation tolerance than before. This and the ELMB's low cost, low power consumption, software support – both low-level and high-level – have made the ELMB the solution of choice in a wide variety of control and monitoring applications. For the LHC experiments a total of ca. 10000 pieces will be produced.

I. INTRODUCTION

The Detector Control System (DCS) of the ATLAS experiment [1] is using the CAN industrial fieldbus and the *CANopen* protocol [2] for its front-end I/O, where possible and appropriate. Examples are monitoring of environmental parameters such as temperature and magnetic field, configuration and monitoring of detector front-end electronics and power supply control and monitoring. Many of the electronics systems in and around the detector are custom designs, done by various groups all over the world. Important requirements for the DCS frontend I/O are:

- low cost (i.e. use of COTS components only)
- low power consumption
- high I/O channel density (i.e. many channels per control unit)

Where the front-end I/O is located on or inside the detector there are additional requirements to be met:

- remote firmware upgrades must be possible (*In-System-Programmable*)
- insensitivity to magnetic fields
- tolererance to the radiation levels found in that location over the lifetime of the experiment

To reduce the design and test effort by individual control system designers and to promote a common solution for interfacing custom designs in a standard way to the DCS, the ELMB was conceived and developed, since there was no commercial solution to satisfy all requirements.

The ELMB is meant to be embedded in custom designs, where it interfaces to other electronics and/or sensors, depending on the application, while providing a CAN interface for communication with higher levels of the DCS hierarchy.

The ELMB features a powerful 8-bit microcontroller with single clock-cycle instructions running at 4 MHz, analog and digital I/O, and a CAN-controller and CAN-bus interface.

Care was taken to select components that comply with the magnetic and radiation requirements.

The ELMB design has gone through two prototypes, named *LMB* and *ELMB103*. Of the ELMB103 300 pieces were produced, which were used in sub-detector test and measurement set-ups. A number of the ELMB103s were extensively tested in radiation [3].

The ELMB103 has two on-board microcontrollers, one of which is dedicated to provide *In-Application-Programming* (IAP) capability, i.e. new or upgraded program code can be downloaded into the ELMB, in this case via the CAN-bus and using the *CANopen* protocol. This can even be done without disturbing other CAN-bus traffic.

In the final design the main microcontroller of the ELMB103 was replaced by a compatible but more advanced type, with self-programming capability, whereby a section in the microcontroller's flash memory can be programmed with a bootloader program. The bootloader provides the IAP capability. This meant that the second microcontroller was obsolete and could be removed. The final design of the ELMB is named *ELMB128*.

A preproduction series of 650 pieces of the ELMB128 has been made to satify the short-term needs of ELMB users, while the final production is being prepared.

II. THE ELMB128 HARDWARE

Figure 1: shows a block diagram of the ELMB128. The ELMB128 is divided into three separately powered sections, separated by opto-couplers. Each section has a voltage regulator with current and thermal limitations, providing protections against Single-Event-Latch-up (SEL). Typical current consumption is also indicated in Figure 1:.

The ELMB provides the user with 64 differential analog inputs, as well as 32 digital I/O lines. The ADC can be dynamically configured for 6 different voltage ranges, unipolar or bipolar, between -2V and +5V. In addition there is a 3-wire Serial Peripheral Interface (SPI), which may be used but with care, because the same interface is used on-board for access to the CAN-controller.

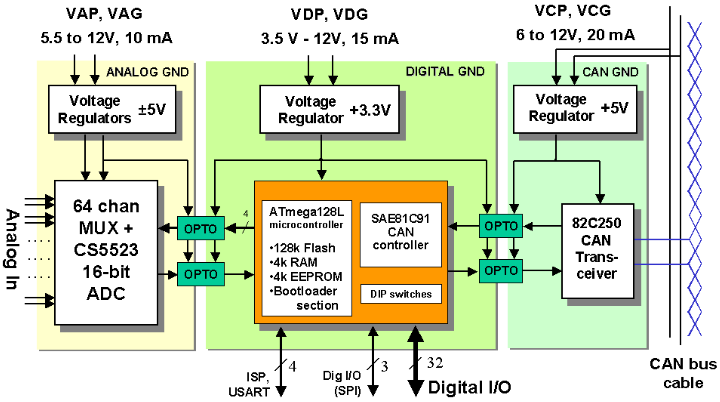

Figure 1: Block diagram of the ELMB128, showing the separately powered board sections. Currents are typical when there is no activity.

And finally there are an In-System-Programming (ISP) and USART (serial) interface, which are available to the ELMB user on a separate connector.

The analog section with ADC and multiplexor circuitry for 64 analog-in channels is optional and may be left off. Part of the ELMB128 production will consist of boards without the analog section. This type is called ELMB128D (with 'D' for Digital) and the fully-equipped type is called ELMB128A (with 'A' for Analog). Figure 2: shows pictures of front and back side of the ELMB128A board. The back side carries the ADC and the 64-channel analog input multiplexor circuitry. The board measures 50x67mm^2.

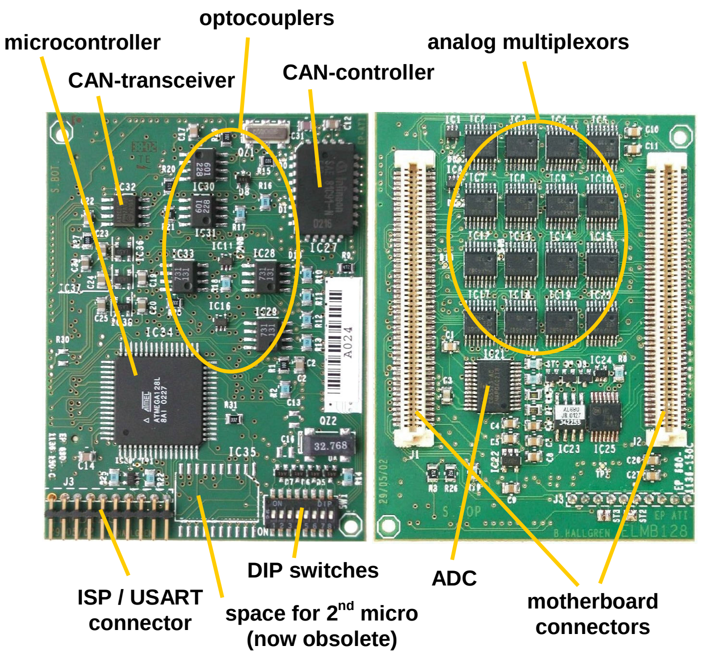

Figure 2: ELMB128A module front (left) and back side (right) with main components indicated. Note the empty space (left) for a second microcontroller, necessary in the ELMB prototypes, but now no longer needed.

To help users of the ELMB get started quickly a general-purpose motherboard was developed, shown in Figure 3:. The front side is occupied by connectors for digital I/O, analog inputs, CAN and power. The back side carries the ELMB and has sockets for signal conditioning adapters for all 64 analog input channels. Different adapters for different sensors, for example NTC, 2-wire PT1000 or 4-wire PT100, are available. With these adapters sensors can be directly connected to the motherboard.

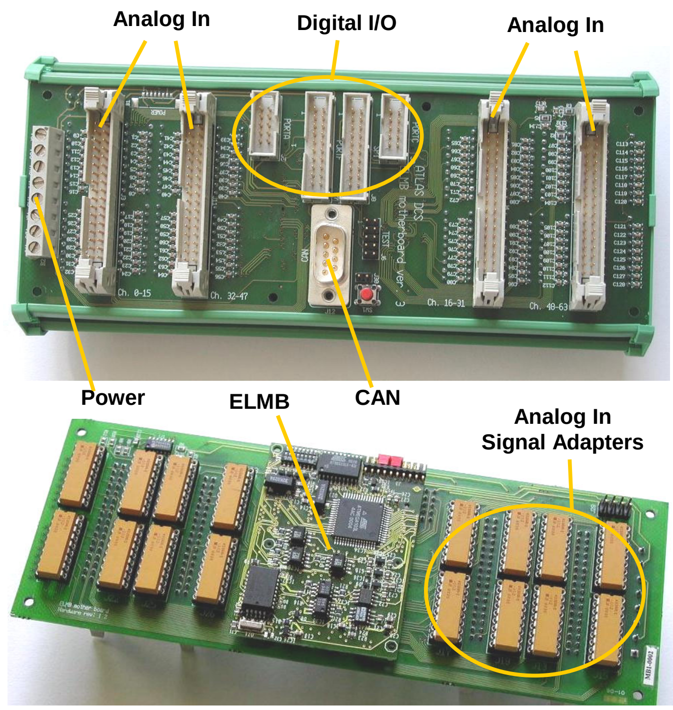

Figure 3: General-purpose motherboard for ELMB, front (top) and back side (bottom).

III. THE ELMB128 FIRMWARE

During production the ELMB128 is programmed with 2 applications in firmware:

1. a bootloader
2. a general-purpose *CANopen* I/O application

The bootloader is important, because it enables upgrade of the main application firmware, remotely via the CAN-bus. At power-up of the ELMB128 it is always the bootloader that takes control, for a few seconds at least. In this way a corrupt program can be replaced or new code downloaded, at all times.

The general-purpose *CANopen* application supports the analog in- and digital in- and outputs of the ELMB128, plugged on the general-purpose motherboard. In addition an external analog output module (the result of a separate development [8]) is supported. So an *out-of-the-box* ELMB128 plugged on the motherboard is ready-to-use. At production the application is used to test the ELMB128 hardware, as well as calibrating the 6 voltage ranges of the ADC. The calibration constants are stored on-board.

The use of the *CANopen* standard protocol on the CAN-bus guarantees that the ELMB128 will work with other (commercial) modules on the same CAN-bus. Users doing their own ELMB software development must also adhere to this standard, thus facilitating integration into the rest of the DCS. For ELMB application developers a software framework (including all *CANopen* functionality but without any I/O, except CAN) is available in the form of source code [4]. Alternatively developers can take the general-purpose application source code and modify it to suit their needs.

In the software special attention is given to prevent faulty program behaviour due to SEE effects (bit flips in memories and registers), which mainly affects RAM, as shown in the next section. The software measures taken include:

- use of the on-chip Watchdog Timer
- periodic microcontroller, CAN-controller and ADC register refresh where possible (values copied from flash memory or EEPROM)
- rarely changing configuration parameters are (re)read from EEPROM just before each use
- majority voting scheme for CAN buffer management variables
- full CAN-controller reset when no message has been received for a while
- CRC on parameter blocks in EEPROM and program code in flash memory
- mask off unnecessary bits in variables (e.g. *boolean*)

IV. RADIATION TESTS

The ELMB128 will be used on the ATLAS MDT Muon chambers, where the accumulated levels of radiation over a period of 10 years of operation in LHC are:

- Total Ionising Dose (TID): 4.7 Gy
- Non-Ionising Energy Loss (NIEL): $3.0*10^{10}$ neutrons/cm^2 (1 MeV eq.)
- Single Event Effect (SEE): $5.4*10^9$ hadrons/cm^2 (> 20 MeV)

These values have been derived from calculations and simulations for the barrel region of the MDT muon chambers.

In doing radiation tests certain safety factors have to be applied, according to the "ATLAS Policy on Radiation Tolerant Electronics" [5]: a factor of 3.5 or 5 for simulated levels, a factor for Low Dose Rate, which can be 1 for COTS components of homogeneous batches and a factor for the components lot, which is 2 in preselection tests with components from homogeneous batches. This leads to the following minimum levels for the radiation tests:

- TID: 4.7 * 3.5 * 1 * 2 = 33 Gy
- NIEL: $3.0*10^{10} * 5 * 1 * 2 = 3.0*10^{11}$ n/cm^2
- SEE: $5.4*10^9 * 5 * 1 * 2 = 5.4*10^{11}$ n/cm^2

A. NIEL Test

Six ELMB128 boards were irradiated at the reactor of ITN Portugal in Feb 2003 [6]. Three boards received $2*10^{12}$ neutrons/cm^2 and the other three received $8*10^{12}$ neutrons/cm^2. In addition to the neutron dose each board received a TID dose of about 20 Gy and 80 Gy, respectively.

The ELMB128s that received the lower dose were all still working fine, and all parameters were still inside specifications. The lower dose is about 7x higher than the dose it should at least be able to tolerate.

The ELMB128s that received the higher dose did not work anymore, but after replacing the 5V regulators on these boards, they were working again, and within specifications.

B. TID Test

In April 2003 at the cyclotron of the Universite Catholique de Louvain in Louvain-la-Neuve, Belgium, 12 ELMB128s were irradiated [7]. Each ELMB128 was irradiated with a minimum of $1*10^{11}$ protons/cm^2, which corresponds to a TID of 140 Gy, about 4x the dose it should be able to tolerate. The graph in Figure 4: shows the increase of the current of the analog section of the ELMB128 as a function of received TID. The results of all ELMB128s tested are shown (indicated by their serial number). One ELMB128 received a much higher dose. It was found that the currents of the CAN and digital sections were not affected.

However it was found that none of the ELMB128s, although still completely functional, could be reprogrammed. This will be further investigated in coming radiation tests.

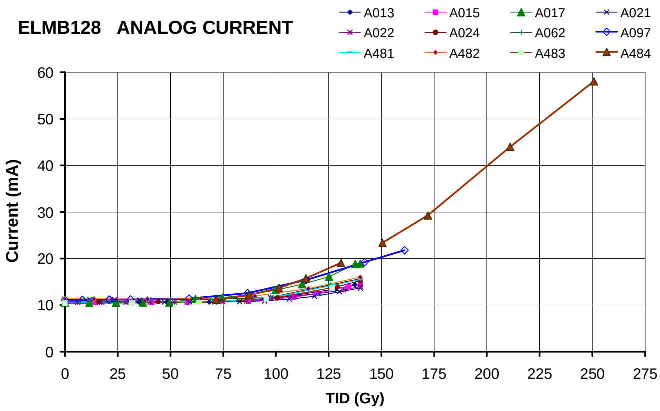

Figure 4: ELMB128 analog current as function of the received TID for twelve ELMB128s (one was irradiated up to 250 Gy).

C. SEE Test

The TID test described in the previous section was a combined SEE/TID test. The ELMB128s in this test were powered and running the general-purpose *CANopen* application, described in the previous section, with additions to run periodic checks on (unused) parts of memories and registers in devices which were filled with pattern data.

The so-called *systematic* SEE test consists of counting the errors found by the memory and register checks. The so-called *functional* SEE test consists of counting the number of times 'abnormal' behaviour by the ELMB128 was seen by observing and analysing the continuous communication exchange via CAN-bus between the ELMB128, which was periodically monitoring a number of analog inputs, and a host PC. If necessary a soft (via a CAN-message) or hard reset (power off) is given.

Figure 5: shows the total number of SEE detected in the microcontroller SRAM, as a function of received fluence. It clearly shows the improvement (factor 6) gained by using the new type of microcontroller on the ELMB128, which is made in 0.35 •m technology, while the one it replaced on the ELMB103 was made in 0.50 •m technology. Similar improvement was seen for the ADC registers, but a technology change by the chip manufacturer could not be confirmed.

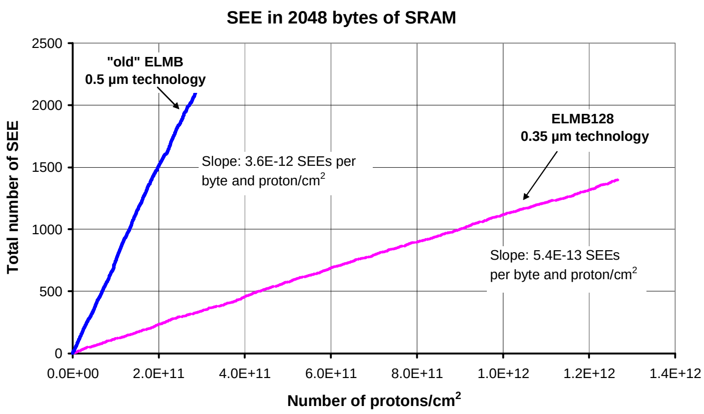

Figure 5: Total number of SEE detected in SRAM of the ATmega103L (left graph, ELMB103) and ATmega128L (ELMB128) microcontrollers as a function of fluence.

Table 1: summarizes the results of the SEE *functional* test and compares this to a previous test with the ELMB103 prototype. The 'misbehaviour' of the ELMBs is categorized according to the necessary action to be taken to correct it. Note the difference in fluence. This considerable improvement of SEE behaviour is thanks to both the microcontroller with its smaller feature size and the firmware which was improved as a result of the outcome of the earlier SEE test.

Recovery Method	**ELMB128** 13.0*10^{11} p/cm^2	**ELMB103 proto** 3.3*10^{11} p/cm^2
Power Cycling	0	4
Software reset	1	5
Automatic	13	20

Table 1: Observed number of functional SEE in ELMB128 and ELMB103 prototype during proton irradiation.

V. Production of the ELMB128

Currently a final production batch of 9000 ELMB128s is being prepared. Of these 5800 are for the ATLAS experiment, 1800 for 'services' (rack and gas systems) and 1400 for the other LHC experiments. This includes spares.

Components of the same production batch were ordered, in order to lower the variation in radiation tolerance of individual components. A manufacturer will produce the first 30 pieces in October 2003. These will be programmed with the latest firmware and subjected to another series of radiation tests. Provided these tests have a positive outcome, the rest will be produced immediately following. All components have been received. The production is scheduled to be completed by the end of 2003.

VI. Conclusions

The ELMB128 has been accepted as an excellent choice for various control tasks in and around the LHC detectors, due to its low-cost and proven radiation tolerance. It will be widely used, especially in the ATLAS detector.

Thanks to the ELMB128's versatility, ease of use and low cost, it has found several other applications outside the pure LHC detector environment, such as in the electronics rack control system and the experiment gas systems.

VII. References

[1] *CAN in Automation* (CiA) international users' and manufacturers' group, http://www.can-cia.org/

[2] H.Boterenbrood et al., "Design and Implementation of the ATLAS Detector Control System", IEEE Real Time Conference, Montreal, May 2003.

[3] B.Hallgren et al., "The Embedded Local Monitor Board (ELMB) in the LHC Front-end I/O Control System", Proceedings of the 7th Workshop on Electronics in LHC Experiments, Sep 2001.

[4] http://www.nikhef.nl/pub/departments/ct/po/html/ELMB/ELMBresources.html or http://elmb.web.cern.ch/

[5] M.Dentan, "ATLAS Policy on Radiation Tolerant Electronics", ATLAS Document No. ATC-TE-QA-0001, Nov 1997.

[6] B.Hallgren, "Results from Neutron Irradiations of the ELMB128", CERN ATLAS Internal Working Note, DCS-IWN19, Sep 2003.

[7] H.Boterenbrood, B.Hallgren, "SEE and TID Tests of the Embedded Local Monitor Board with the ATmega128 processor, CERN ATLAS Internal Working Note, DCS-IWN20, Sep 2003.

[8] J.Kuijt et al., "ELMB-DAC 16-channel 12-bit DAC-module for the ELMB", user manual, NIKHEF, May 2002.