ABBA2

A short description

G. Siringo, MPIfR
15.04.2008

1 Introduction

A new data acquisition system, ABBA2, has been provided by the MPIfR bolometer group for operation as backend of the MAMBO (37 elements) and MAMBO-2 (117 elements) receivers. ABBA2 is going to replace the previous system, ABBA (ADC Backend for Bolometer Arrays) and is based on the same kind of analog-to-digital conversion hardware: two National Instrument PCI6033E boards, providing 64 analog input each, digitized over 16 bit. The two boards, however, are now installed in a new industrial computer. The SBC (Single Board Computer) is an IEI PCIE-9450 (PICMG1.3 specification, PCIe bus) equipped with an Intel Core2 Duo E6600 (two cores at 2.4 GHz, for high performance in parallel processing) and 4 Gb of DDR2 RAM (800 MHz FSB). The backplane is an IEI PXE-13S-R10 (PICMG1.3) providing 8 PCI, 3 PCIe x1 and one PCIe x16 slots and equipped with a PLX PEX8111 PCIe-to-PCI bridge, a chipset which can boost the performance of a standard 32 bit/33 MHz PCI card.

Apart from the new hardware, the ABBA2 backend differs from the old ABBA for the use of Linux as operative system. The porting of the software has been possible since the release in 2005 by National Instrument of official drivers (DAQmx) to operate the PCI6033E boards under Linux.

The new computer is mounted in a 19” rackmount chassis and located in the same electronics rack where the old ABBA was.

2 Principles of operation

The new ABBA2 data acquisition system differs in several aspects from its predecessor, the good old ABBA, and not only because of the new hardware or the different operative system. The principle of operation, in fact, is completely different. The old ABBA was hardware triggered to the blank pulses generated, at a specified delay, by the so-called “delay box” which was triggered by the wobbler sync signal. The new ABBA uses directly the wobbler sync and operates the blanking via software. There will be no more need for a delay box and for an external blanking triggering. The wobbler sync reference is the only signal required for operation.

Time synchronization is provided by the *ntpd* server, always running in the background. Although this new concept is less accurate than the use of a GPS clock interface, as it was in the old ABBA, the errors are small enough to be compatible with our application (2 ms at most) without the need for an extra piece of hardware.

I will describe now briefly the main points of the software.

2.1 Wobbler sync detection

When the program starts, ABBA2 runs a short data acquisition (a few seconds, user defined) at high sample rate on one single channel, the one connected to the wobbler sync reference signal. The properties of the reference (frequency/period, duty cycle, pulse duration, hi/low states levels) are then analyzed and stored, to be used during operation.

2.2 Main loop

The first level of the ABBA2 program runs a TCP listener, waiting for an incoming connection, mainly from the main telescope’s control software. When a connection is established, the main loop runs a command parser, in order to evaluate the commands sent by the control software and will take the corresponding action, in case the command is recognized as valid, otherwise it sends back the error message “unknown command”.

2.3 DAQ loop

Once a valid start command is received at the TCP socket, the DAQ (data acquisition) loop is started. This loop acquires a continuous stream of samples from the ADC boards and then produces the corresponding stream of timestamps, at the specified sample rate (in the range 1 – 1000 Hz). These raw values are temporarily stored in a FIFO (first in/first out) buffer which acts as interface between the data acquisition hardware and the program memory.

2.4 Data preparation loop

In parallel, another loop runs continuously extracting the raw data from the DAQ buffer and performing some operations in real-time, to prepare a new data stream (merging channels acquired by different ADC boards, re-ordering of the samples by channel, labeling with phase and timestamp information per sample) for the following data manipulation. An intermediate FIFO buffer is filled by this loop.

2.5 Data manipulation and storage loop

In parallel, another loop extracts the data from the intermediate buffer and operates the separation of the samples per wobbler phase, applying the requested blanking. The identification of the first phase (on or off) can be selected by the user, as it was in the old ABBA, to compensate for the opposite sign of the bias current in MAMBO-37 and MAMBO-117. The data stream is divided in phases, the samples of each phase are averaged and multiplied by 100 to increase the resolution of the digitization to ~21 bit. The resulting data stream (one dump per phase) is then stored in the corresponding output data file, on the local hard disk, following the same data format used by the old ABBA.

2.6 Broadcast channel for quick look

Once the *stop!* command is received, in addition to the output data file, ABBA2 will send the data of the selected broadcast channel through the TCP connection back to the telescope's control software (or any other TCP client connected) in order to provide the observer with a small set of raw data for quick data look (e.g. pointing and focus corrections, skydips). ABBA2 uses the same data format used by the old ABBA.

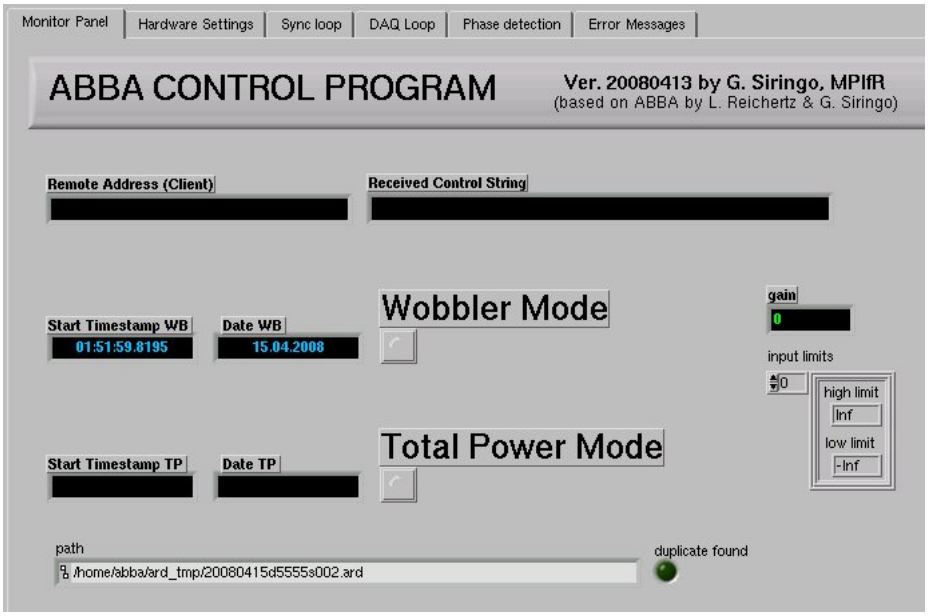
3 – GUI description

The software of ABBA2 is written using LabVIEW 8.2, a platform independent software development environment. In this installation, it will be operated under Linux, however the same program could also be used under Windows without any changes.

The main GUI (Graphical User Interface) is divided in 6 menus, as listed below.

3.1 Monitor

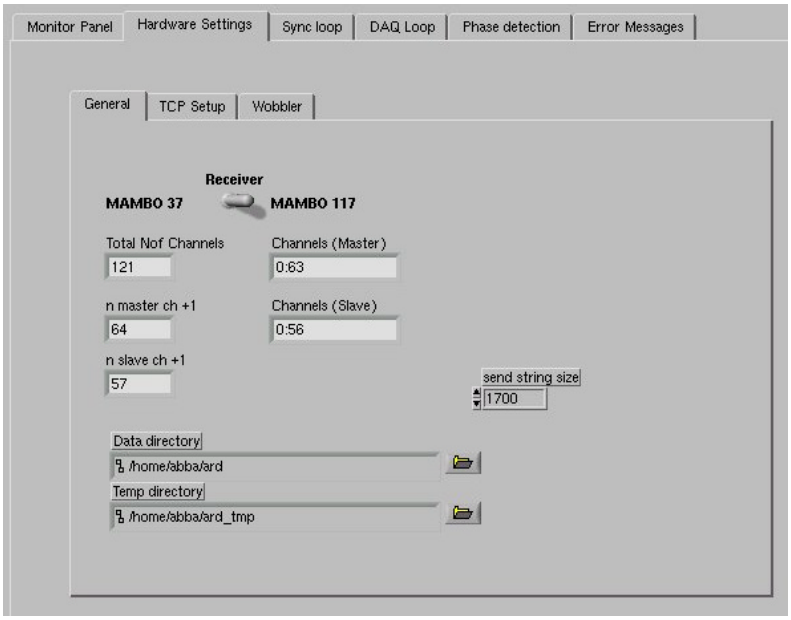
This panel shows some general information to follow the operation of the backend. In particular, the time/date and filename of the last scan are displayed.

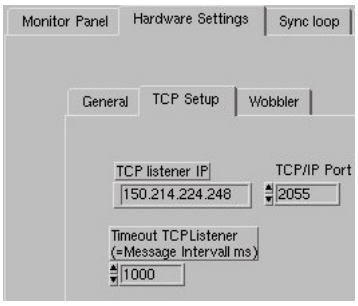


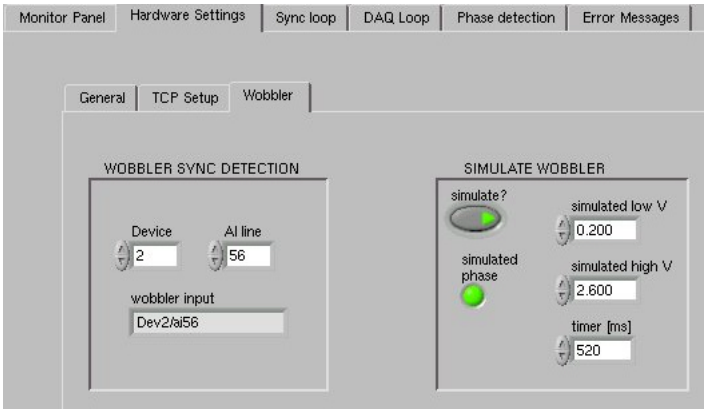
3.2 Hardware settings

This panel is divided in three sub-menus.

- **General**: some general settings, like the choice of the receiver (37 or 117 channel).
- **TCP**: setting specific to the network communication (default IP is 150.214.224.248, default TCP port 2055)
- **Wobbler**: settings for the sync detection loop (and a wobbler simulator, only for testing purposes).

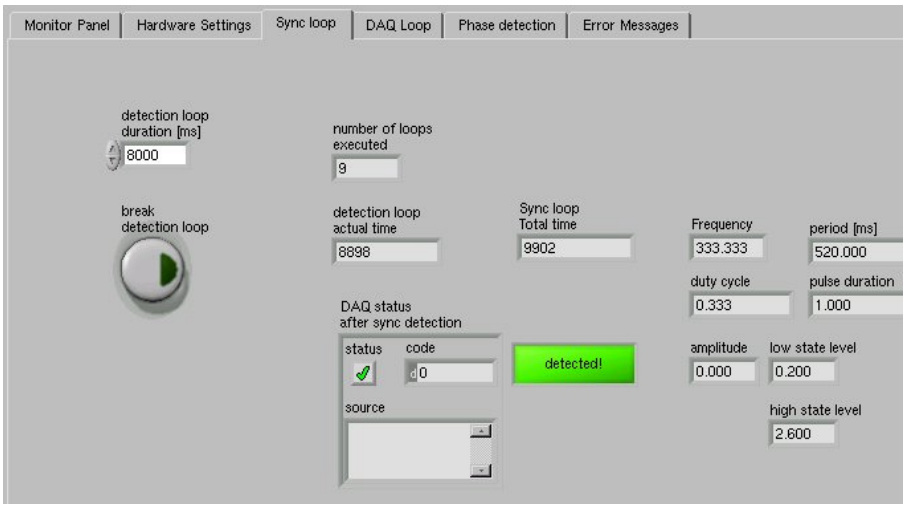






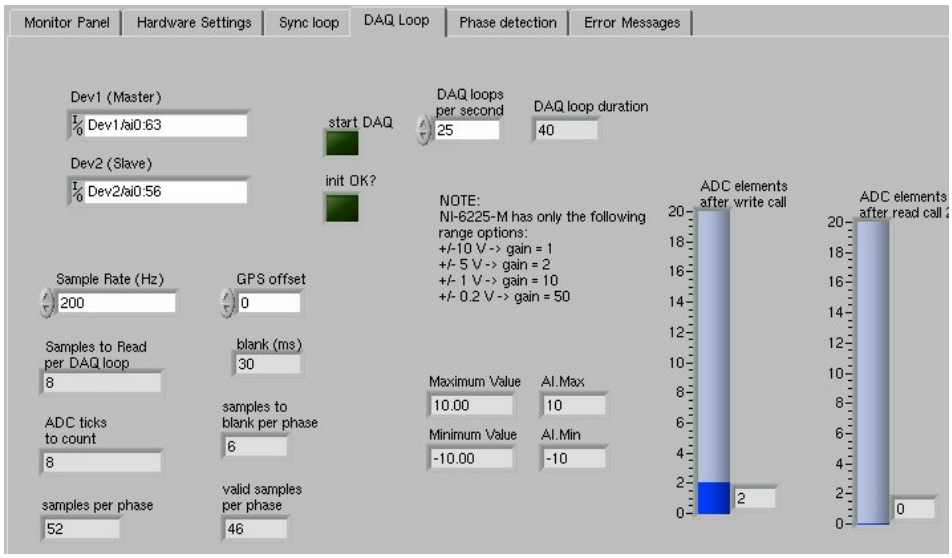
3.3 Sync loop

Settings specific to the wobbler sync detection. The values detected from the analysis of the reference are also displayed.



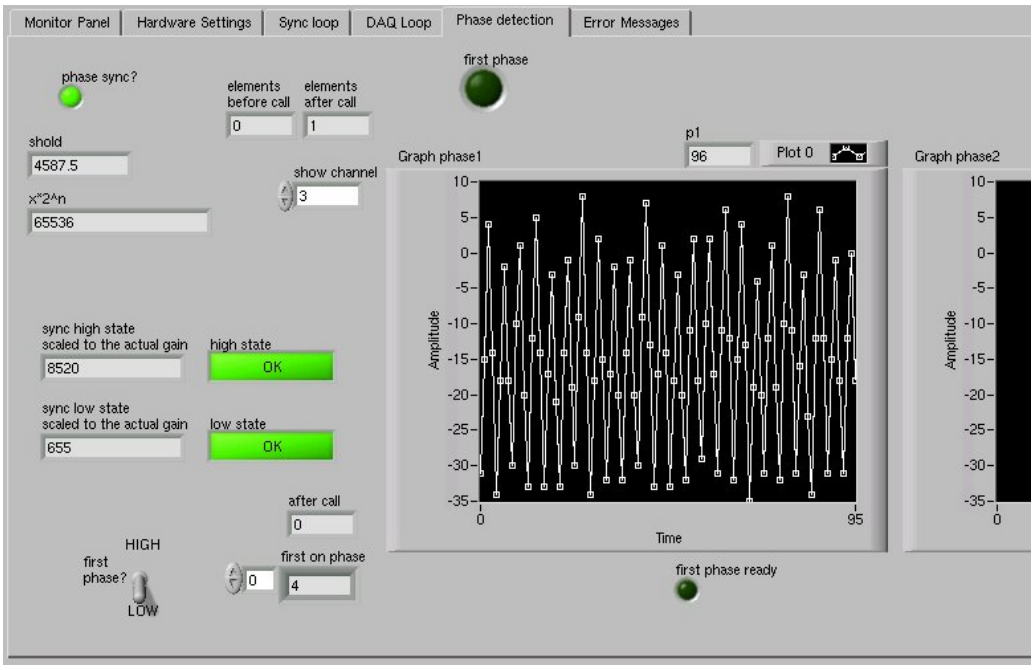
3.4 DAQ loop

This panel shows settings specific to the data acquisition loop. In particular, the definition of the two data acquisition channels (ADC devices), the sampling rate, the blocking factor (DAQ loops per second), gain value and dynamic range is use, status of the FIFO buffers.



3.5 Phase detection

Apart from the settings of the first phase (on/off), this panel mainly shows the status of the phase detection in real time. In particular, two plots show the samples of each phase, before they are averaged, for a selected channel.



3.6 Error messages

This panel shows the status of the two DAQ devices (Dev 1,2), of the TCP socket, of the file open/write operations. In case of failure, the indicator corresponding to the faulty element will show an error messages including a specific error code and a brief description.