We present a full-time beam-former for antenna phase arrays, designed on a single FPGA (development kit Xilinx ML523 with one Virtex-5 FXT-100 on it), whose first tests occurred at the Nançay radio telescope[1] on the FAN[2] antenna array. This beam-former have been designed for the BAO-radio instrument[3] which is a demonstrator dedicated to the study of dark energy by HI probing. This instrument involves two groups of antenna placed along a North-South line in two half-pipe grating-made reflectors.

The number of channels of the beam-former is 24, gathered in two groups of 12, each channel corresponding to one antenna. Prior to be sent to the beam-former, the RF antenna signals are amplified and their frequency lowered down in the 0-250MHz band by signal mixing for a digitization at 500MHz/8bits. The digitizing boards involves FPGAs with high-speed transceivers (5GbPS), and their firmware includes real-time FFT and frame-building capabilities. An external device dispatches a clock and a trigger (for frame-sending) to the digitizing boards. This electronics have already been used with a software correlator and off-line processing[4].

When the hardware beam-former is implemented in the acquisition chain previously mentioned, its incoming data comes from the frames (two channels each) of 4096 complex coefficients per channel, 2 x 8 bits each, sent on optical fibers at a 4GbPS effective rate by the digitizing boards. Twelve optical SFP to differential SMA converting boards are used to send the data to the ML523 Xilinx kit. The incoming frames are received by the MGT transceivers (of the Virtex FPGA) and their data parallelized in 16 bits-words at the rate -250MHz- of the recovered clocks, and 12 clock domains (and regional buffers) are involved.

A frame-checking is done by 12 independent state-machines whose other task is to read in the frames the data of the two channels and store it. The whole incoming data flow of the beam-former is stored in 24 dual-port RAMs. All the frames are not exactly stored at the same time in the RAMs, as the electronic delays, boards manufacturing, fiber and/or cable lengths cannot be perfectly matched all together. Then, a clock-independent logic waits until the data of the last frame has been stored, and then triggers the second part of the FPGA logic.

This said part has a unique 250MHz clock domain, where, after the emptying of the dual-port RAMs, a complex gain is applied on each frequency of each channel, and the 12 channels of each group can be summed. Afterward, the square of modules of the two summed groups are calculated and a time-averaging is done. The number of such averaged square modules can be tuned between 2 and 65536 and the results stored in two dedicated RAMs, one per channel group. The complex-gains are applied for balancing, after calibration, the discrepancies of delay an/or electronic transfer function between channels or to introduce an artificial weight and phase shift on each channel. For the BAO-radio application, as an example, a time delay (i.e. a phase shift varying linearly with frequency) proportional to the antenna index can also be added in the complex gains, prior the summation of the channels belonging to a same group, for an synthesized pointing on the sky.

There are 24 sets of complex gains in the FPGA, whose real and imaginary parts are coded on 12 bits each for each frequency. In order to save the RAM in the FPGA, there is a differential coding of the complex coefficients on 4 bits, except for the frequency 0. So, all the complex gains can be stored on 24 x 4095 bytes and 48 x 12-bits registers.

Each of the two adders of the beam-former that sums 12 channels together can be configured to behave as a multiplexer which selects randomly one channel. On the other hand, the logic that builds the square of modules can also compute a visibility between the two selected channels and make a time-averaging similar to the one done with squares of modules. In this case, the RAM that stores the averaged square of modules of the first beam will store the real-part of of the averaged visibility and the other RAM will store the imaginary part. Of course, a visibility can also been built on the summed channels if necessary.

This gives obviously some versatility to the beam-former. Nevertheless, because a real-time visibility can be built on a pair of channels only, the effective trigger-rate when the beam-former behaves as a correlator is low (one can notice that, in this case, the two groups of channels must be twins and the effective number of channels be 12 and not 24. This issue will be evoked hereafter). For example, the duration of a frame provided by the digitizing board at the present time is ~33µs and the beam-forming can be done with almost (< 40ns) no dead-

time, at a trigger rate of ~30KHz. If, instead, the beam-former is configured to built a 12 x 12 visibility matrix, the number of computations is 78 and the effective trigger-rate becomes ~390Hz.

A special firmware has been realized for having two identical groups of twelve channels, and its architecture presented on figure 1. The difference versus the BAO-dedicated firmware is that only 6 incoming fibers (and MGTs) are used, instead of 12, and, in the firmware, the 6 recovered clocks feed two regional buffers each, and the parallelized 16-bits data buses are split in two. The remaining part of the two firmwares are perfectly identical.

The firmware that splits the clocks and the data is suited for the FAN array at NRT where 96 antenna are summed by 8 to form 12 pixels that can collect the focal spot of the radio telescope. The FAN array is placed on a carriage and source tracking or transits can be observed. Because of the splitting of the incoming data in two identical flows, the beam-former can then build two beams (Two identical 12 channels group and 24 gains) and a visibility matrix if the source is tracked, notwithstanding the reduced effective trigger rate.

 The whole behavior of the beam-former is controlled by one of the PPC-440 processors where a C code handles the communication with an external (called CC) program written in Visual Basic®.
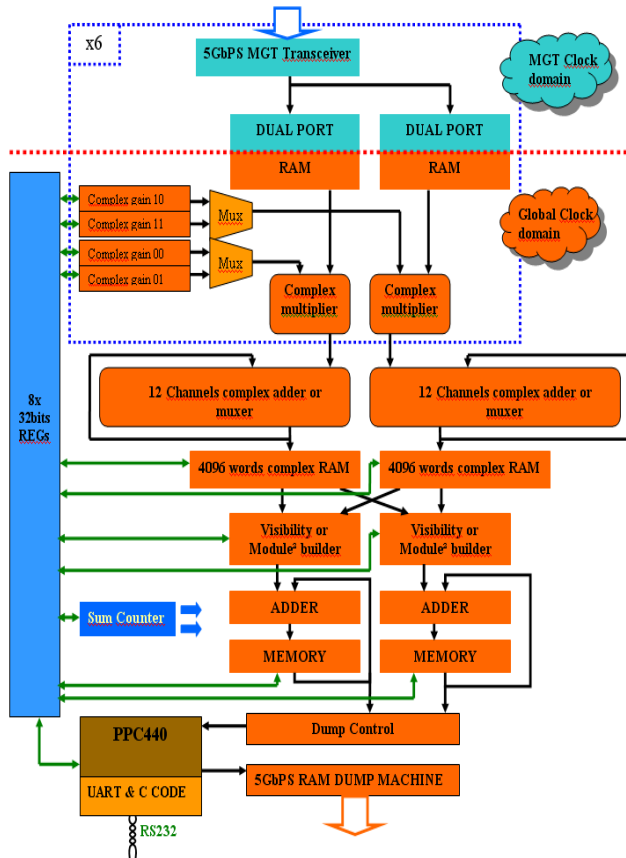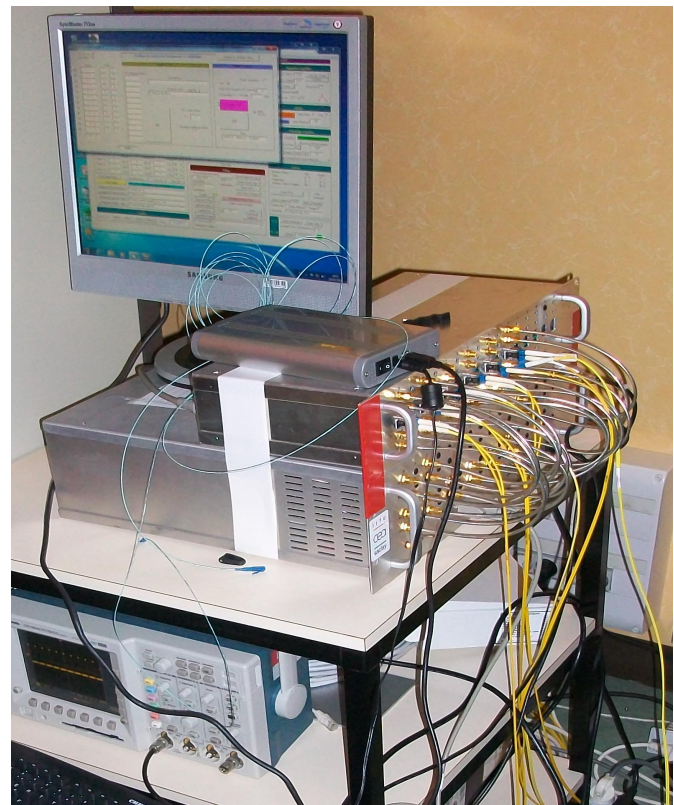

Figure 1 : Architecture of the beam-former

Figure 2 : The beam-former and its control-screen installed at NRT

The CC program, via the RS-232 link available on the kit ML523, can modify some internal registers implemented in the beam-former design. This control allows to reset the FPGA (GTX, regional and global logic), load the gain memories, choose between beam-forming or visibility, start and stop the sequencing for the time-averaging of square of modules or visibilities, start the PPC440 for the reading of the last RAMs of the path and send the data to the 5GbPS RAM dump machine.

This dump-machine is the output port of the computations made by the beam-former and uses the Tx channel of

one MGT of the FPGA. A single character sent by the CC program starts a routine of the C code running in the PPC440, which sets the 4096 addresses of the two last RAMs and transmits their data to the input FIFO of the dump-machine. This FIFO is permanently read by the machine. When it is empty, the machine sends to the GTX a flow of idle words made by control characters of the 8b/10b alphabet, and, otherwise, inserts 32-bits relevant data words in this flow.

The CC program controls the beam-former behavior and can group in predefined sequences the basic operations described above. For example, a window of CC automatically selects the channels and starts a loop of time-averaging and RAM-dump sequences (figure 3)
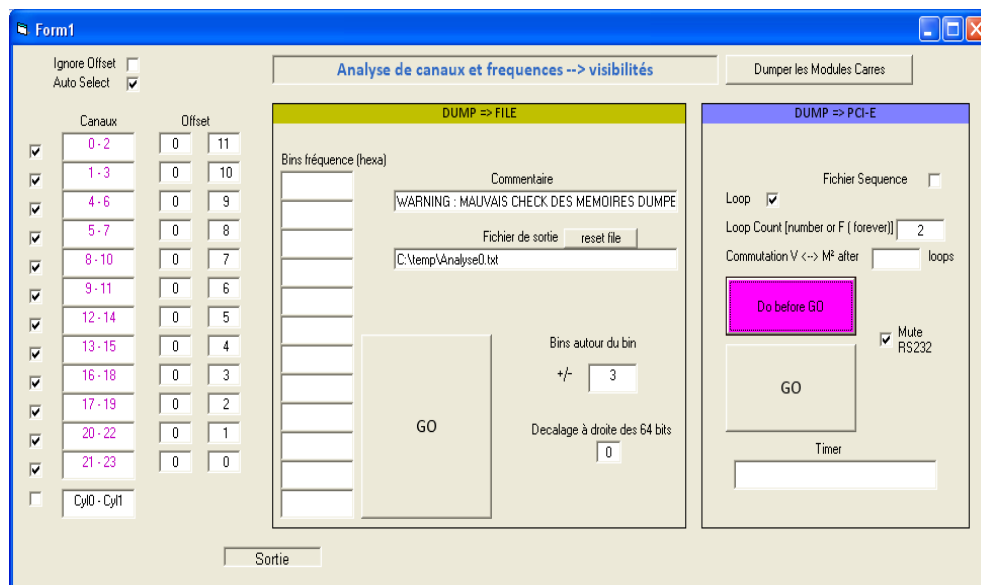

Figure 3 : Window of CC program for the launching of the beams or visibilities sequences.

After a setup of the beam-former, CC can compute 24 complex gains, 4096 frequency each, to compensate fixed-delays or amplitude discrepancies between channels. The real and complex part of these gains are coded on 12 bits. For simple delay compensation, the gains are sine and cosine of angles varying linearly with the frequency, and, to increase the precision, they can be multiplied by an integer constant (gain_max) smaller than 2048. The 24 x 4096 complex gains and their differential encoding, mentioned above, are computed by CC. It was mentioned that, for a differential encoding of the 12 bits-words of the gain, 4 bits suits in most of the cases. If it is not, CC advises the user and gain_max can be reduced.

It can occur that the gains computed by CC from a fixed-delay model does not fit the reality. So, CC is able to download any gain files, especially those built from calibrations performed on sky radio-sources. After a smoothing of the gain curves, if needed, and differential encoding, CC uploads the files in the RAMs of the beam-former.

References :

[1] http://www.obs-nancay.fr/index.php?option=com_content&view=category&layout=blog&id=5&Itemid=5

[2] http://www.obs-nancay.fr/index.php?option=com_content&view=article&id=77&Itemid=125

[3] Peterson et al. "The Hubble sphere hydrogen survey", astro-ph/0606104

[4] Charlet et al. "The BAO Radio Acquisition System," *Nuclear Science, IEEE Transactions on* , vol.58, no.4, pp.1833-1837. 2011