# Uniboard Digital Receiver Programming Manual Revision 1.0

G. Comoretto[1], A. Russo[1], A. Baudry[2], P. Camino[2], B. Quertier[2]

[1]INAF - Osservatorio Astrofisico di Arcetri
[2]Observatoire de Bordeaux

**Arcetri Technical Report 6-2011**

## Abstract

*The Uniboard is a project for a general purpose board, containing several large FPGAs, to be used as a general component for the next generation VLBI correlator. The Digital Receiver Working package deals with the implementation of a wideband frequency demultiplexer, that divides the received radio band (up to a bandwidth of 8 GHz) into enough narrow band channels (64 MHz bandwidth or lower) to completely cover it. Each channel can be independently positioned across the input band, and may have a different bandwidth and signal representation (number of bits per sample).*

*The document describes the programing model for the hardware components in the board.*

# 1 Glossary

The following terms and acronyms are used in this document:

**ADC:** Analog to Digital Converter. An ADC converts a continuous signal to a discretely sampled digital signal, represented by a finite number of bits.

**DBBC:** Digital Baseband Converter. A component that selects a portion of its input signal, converts it to near zero frequecy, and filters it to a selectable bandwidth.

**DSP:** Digital Signal Processing. Any technique used to digitally process a signal. All components in a digital receiver inplement DSP operations.

**FFT:** Fast Fourier Transform

**channel:** A portion of the input signal in the frequency domain, as processed by the digital receiver. Here we will use the

**FFT channel:** The portion of the input signal at each FFT output port.

**DBBC channel:** The portion of a FFT channel filtered and frequency translated by a DBBC.

**Digital receiver:** A system that processes a radio signal using only digital components. The signal to be processed is sampled by an ADC and downconverted, filtered, resampled by various digital components. [1]

Bandwidth: The frequency span of a signal. For a complex signal it is equal to its samplig rate, for a real signal it is half the sampling rate. The **effective bandwidth** is the portion of the available bandwidth that is not distorted/attenuated by the signal processing operations.

**Channel group:** A set of four FFT channels that share the same outputs of the FFT processor first section.

**Polyphase filter:** A filter in which different samples are convolved by a different function. Polyphase filters can be used to control the shape of the FFT channels, or to implement

**Recirculation:** A technique in which the same component is used at a higher clock rate to perform several operations at a lower clock rate. In the DBBC the filter multipliers are recirculated to compute the signal convolution on a much longer filter length.

**FPGA:** Field Programmable Gate Array: a digital component whose functionality is defined by a externally loadable configuration. The Uniboard is composed by 8 FPGAs.

**BN:** Back Node. The four FPGA closer to the Uniboard backplane, and connected to the ADC input busses.

**FN:** Front Node. The four FPGA closer to the front side of the Uniboard, and connected to hte high-speed 10G links.

**SFDR:** Spurious Free Dynamic Range: is the ratio (usually in dB) between the maximum input signal (usually a RFI) and the strongest spurious interferring signal.

**Uniboard:** The general purpose board used in this project.

**SOPC:** System on a Programmable Chip: a general purpose processor and its peripherals, instantiated using general logic in a FPGA.

**avalon:** The proprietary processor used in the Altera SOPC development system

[1] In a common use of this term, it is implied that only signal amplification, not frequency translation, is performed between the antenna and the ADC. Here we do not make this assumption.

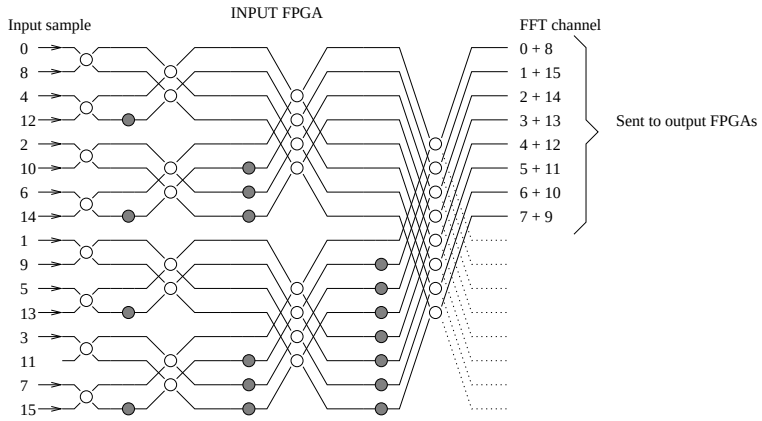


Figure 1: Structure of the FFT algorithm. Butterfly stages are divided between 4 input FPGAS (one shown) and 4 output FPGAs(next figure)

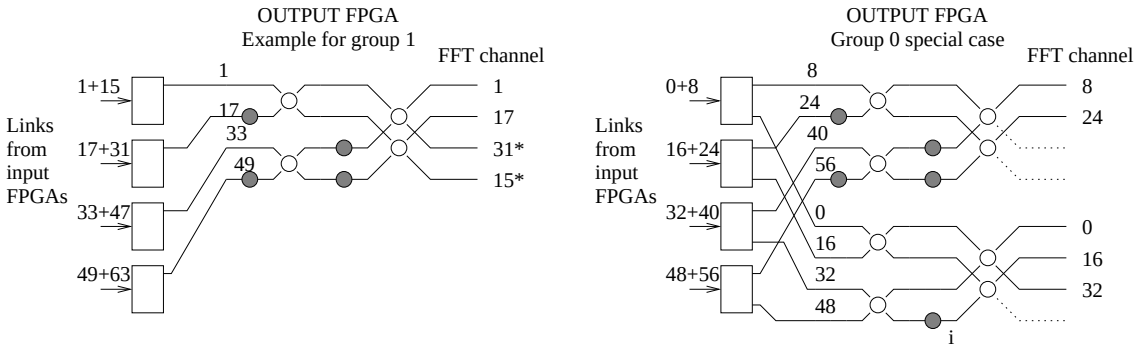


Figure 2: Each output FPGA processes two channel groups, examples shown for group 1 and for the special case of group 0

**Memory Mapped (MM) interface:** an interface between components of a SOPC (usually between the processor and a peripheral) where the peripheral is seen as a range of memory locations.

**Streaming (ST) interface:** A point-to-point interface between two components of a SOPC in which the data flow is unidirectional, very fast, and without direct intervention of the processor.

**VDIF:** a format for the transmission and storage of digitally sampled radio data, used in VLBI applications. Each VDIF packet contains a contiguous time frame of radio samples from one or more radio channels, together with all the informations needed for its identification and time-tagging.

# 2 Introduction

The *digital receiver* application for the Uniboard consists in an array of digital receivers that transform the radio signal sampled by a fast ADC to a series of digital streams, packetized according to the VDIF format. The Uniboard can processes 1,2 or 4 input signals with a total bandwidth of 4 GHz (8 GS/s), and produces up to 64 output streams, representing (up to) 64 portions of the input signal(s).

Each output channel is 1 to 64 MHz wide, and can be codes with 1,2,4 or 8 bit real samples. It can be positioned arbitrarily in the input band, with a high degree of flexibility. Each group of 8 channels sends its output packets to a 10G optical link, as UDP/IP packets. The destination IP address and port can be specified for each channel, simplifying the implementaiton for a distributed correlator.

The architecture of the digital receiver application consists in a 64 point polyphase FFT filterbank, that divides the input signal(s) into 32 overlapping sub-bands [2] with a fixed spacing of 128 MHz, followed by an array of 64 digital BBC. Each BBC is composed of a tunable mixer/LO, a programmable low-pass filter, and a complex-to-real output stage. The architecture is distributed over the Uniboard, using two personalities resp. for the front node and back node chips.

The back node (fig. 1) receives the ADC samples using a time multiplexed scheme. Each chip uses 4 LVDS busses clocked at 500 (512) MHz, with one, two or four chips used for 2, 4 or 8 GS/s ADCs, respectively. Samples are presented in bit-reversed order, as shown in fig. 1.

The personality in the back node provides the following fonctionalities:

- Demultiplexing of each input stream to $2 \times 256$MS/s samples
- Time alignment, total power measurement, statistics and data integrity check of the input samples
- Polyphase filtering for FFT band shaping
- First 4 stages of a 64 point FFT
- Total power measurements of the FFT output signals
- Requantization to 8 bit and transmission over the Uniboard interconnection mesh to the front node chips.

Signals from the back node are further processed in the 4 front node FPGAs. The personality of each of the front node chips implements the following functionalities:

- Receiver side of the mesh interconnection, with data integrity check and sample realignment
- Last stages of the 64 point FFT, partially or totally bypassed for 4 and 2 MS/s modes (fig 2)
- Total power measurement of final FFT outputs
- Array of 16 digital BBCs
- VDIF, UDP and IP packet formatting, with packet buffering
- Internal routing of each block of 8 BBCs to one 10G IP link
- 10G link formatting (using Altera IP macrofunction)

All the above functionalities are embedded into the framework of a NIOS SOPC. The basic architecture of the NIOS processor is taken from the generic processor developed as part of the Uniboard main project, with the specific front node and back node functionalities implemented inside two custom NIOS components.

Each component is seen by the SOPC as a memory mapped peripheral, with streaming interfaces for the radio input and output signals. Separate, standaridized streaming components are used for the LVDS input, for the interconnecting mesh and for the 10G links. Also the IP packet formatting is performed in a separate streaming component, as it is a standard function in the system.

# 3 Input (back side) FPGA

The input FPGA structure is shown in fig. 3. Apart from the components included in the Uniboard reference design, the SOPC includes:

- A LVDS receiver. It receives data from the four LVDS ports, and converts them to a simple streaming connection. It has no programmable parameters, and therefore it does not provide a MM port.

[2] Since the input signal is real, only the 32 sub-bands with positive frequencies are retained. If the input bandwidth is 1 or 2 GHz, the 32 sub-bands represent portions of the input band of different ADCs.

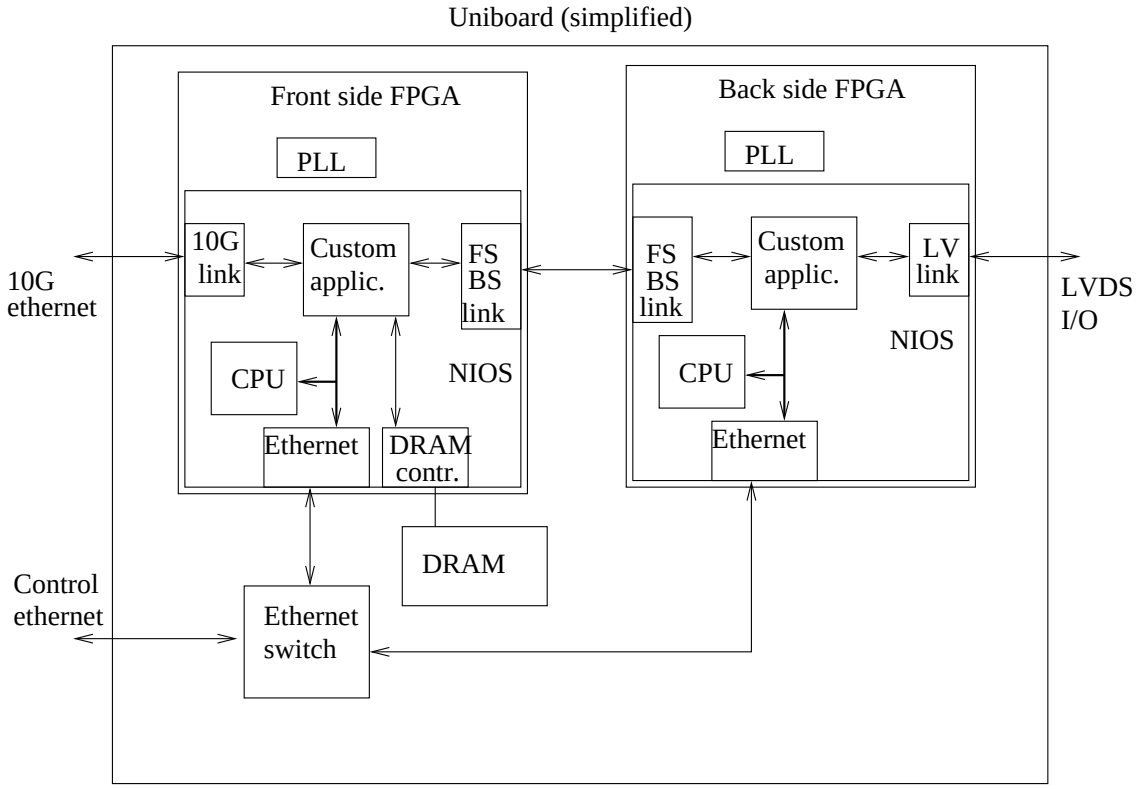


Figure 3: Structure of the input and output FPGAs. Each FPGA design is implemented as a system-on-a-chip, using the Avalon processor structure

- The back end Avalon peripheral (`dr_1st_stage`), that performs the bulk of the signal processing
- A streaming interface to the fast interconnecting mesh of the Uniboard. 2 fast lanes at up to 6.25 Gbps are available between each front-end and back-end FPGAs. The interface is based to the library `gxtr` macrofunction.

In this chapter we describe the programming model for the `dr_1st_stage` peripheral, while the `gxtr` is described in chapter 4.

The peripheral uses an address space of 128 bytes, organized as 32 32-bit registers. Not all bits of each register are used, or read back. A summary of the registers usage is shown in table 1.

Bit assignment for most registers is still preliminar, and may be subjected to change during personality testing.

## 3.1 General registers

The first 8 positions are reserved for global functionalities of the component. Register 0 is used to specify the signals associated with the 8 `testio` lines Register 1 is used to specify general parameters for the component. Register 2 is unused, but reserved for interrupt control, Interrupt s not currently implemented, but can be useful for total power reading, for the serial lines from/to the ADC, and for error conditions. Registers 3–7 are unused.

### 3.1.1 Test point selection

Register 0 is used to select the functionality of the test point signals. Each chip has 8 `testio` lines. The 32 bits are grouped in 8 4-bit groups, and each group select one of 16 lines.

| Address | Write | Read |
|---|---|---|
| 0x00 | Test point select | Test point readback |
| 0x01 | Global control | Global status |
| 0x02 | Interrupt control | Interrupt status |
| 0x03-07 | unused | unused |
| 0x08 | Input total power control | Input total power |
| 0x09 | FFT total power control | unused |
| 0x0a | ADC Serial control/transmit | ADC status/receive |
| 0x0b | unused | unused |
| 0x0c-0f | Input link control | Input link status |
| 0x10-18 | FFT gain | FFT total power |
| 0x19-1b | unused | unused |
| 0x1c-1f | Output link control | Output link status |

Table 1: Register mapping for Uniboard Digital Receiver input chip

Lines 0 and 1 are inputs (jumpers), their use as output test points is TBD. Lines 2 and 3 drive two LEDs, so they are used for pulsed signals (with pulse stretcher), or slow status signals. Lines 4-7 are connected to posts in the board, that can be probed using an oscilloscope.

Each group of 4 bits in the register select the signal associated to one of the lines, with LS bits specifiying `testio(0)`. To reduce possible RFI generation, a setting of 0 (default at startup) disables the associated line.

This register can be read back, as part of a simple integrity test.

### 3.1.2 General control

Register 1 controls the general behavior of the board. Bits are assigned as in table 2.

| bit | control | status |
|---|---|---|
| 0x00 | Low power mode | = |
| 0x02-03 | Chip position | = |
| 0x04-05 | ADC bandwidth | = |
| 0x10 | Input TP ready reset | Input TP ready |
| 0x11 | FFT TP ready reset | FFT TP ready |
| 0x12 | FFT overflow reset | FFT overflow |

Table 2: Register mapping for general control register. Status bits specified as "=" are a copy of the corresponding control bits

If bit 0 is set, the component is disabled. Input signals are set to zero, thus producing almost no transitions in the logic.

Bits 2–3 specify sample sequence position, i.e. which samples of the ADC output are processed. This binary number usually corresponds to the chip position in the board, that is hardcoded in the FPGA ID input port.

Bits 4–5 specify the input bandwidth of the ADC connected to the board. as shown in table 3. As the input ADC can be implemented as multiple interleaved ADCs operating at lower data rate, this can be changed dynamically if these ADCs are fed with interleaved or independent signals. Values of 0, 1 and 2 specify a bandwidth of 4,2 and 1 GHz respectively.

Most bits in the status register are simply a copy of the corresponding control bit. Unused bits always return a "0". Bits 10, 11 and 12 are set by the specified events. Events are latched, e.g. an overflow on a single FFT frame will set bit 12, and must be explicitly cleared after the condition has been acknowledged by pulsing the corresponding control register bit.

| Bits 3:2 | Sample pos. | 8GS 4GHz | 4GS 2GHz | 2GS 1GHz |
|---|---|---|---|---|
| 00 | 0 | 0,4,8,12 | 0,2,4,6 | 0,1,2,3 |
| 01 | 1 | 2,6,10,14 | 1,3,5,7 | 0,1,2,3 |
| 10 | 2 | 1,5,9,13 | 0,2,4,6 | 0,1,2,3 |
| 11 | 3 | 3,7,11,15 | 1,3,5,7 | 0,1,2,3 |

Table 3: Input samples processed by the component, for input bandwidths of 4, 2 and 1 GHz. The sample position (0=older, 3 = newer) must be specified using bits 3:2 of the general control register

## 3.2 Total power control

Total power measurements are performed in a number of places across the design, and therefore a consistent programming interface has been adopted. Slight differences are still present, mainly as functionalities that are not implemented in all total power instantations.

The total power meter used to measure the power level and various statistics of the input link is the most complete, so it is fully described. Other instantations will be described referring to this one.

Register 8 (table 4 controls the integration time and general functionality of the input total power meter. Total power is read from the same register.

| bit | control |
|---|---|
| 0x00-01 | TP function:<br>0 = Total power, 1 = DC offset<br>2 = State counter, 3 = RD check |
| 0x02 | General enable |
| 0x08-0f | Reference status or RD check line |
| 0x10-1c | Integration time minus 1 |
| 0x1d-1f | Integration prescaler |

Table 4: Total power control register

Integration time is expressed in 1ms intervals. Minimum integration time is 2 ms, specified as 1. As it is expressed as a 13 bit quantity, maximum integration time is about 8 seconds.

Bits 0–1 are used to specify the quantity to be monitored, that can be selected between the following:

- Total power: the square of the input signal
- DC offset: the average of the input signal
- Status: counts the number of samples identical to a predefined value, specified as a 8 bit value in the control register. This is useful to build an histogram of the ADC sampled data.
- Random data check: counts the number of errors in one of the 8 input lines, specified as a 3 bit value. Is used to check electric integrity over the input lines

The integration is performed on a sum of these values over the 8 parallel input samples. Each sample can be individually enabled/disabled using the input link control register. For most cases, all inputs are used, but for debug purposes (e.g. random data check) a single line or a single data stream can be examined.

Integration result must be rescaled to compensate for different integration time. The number of bits discarded range from zero to 14, in steps of 2. The total power reading is a 31 bit quantity, with the most significant bit used to signal an overflow.

If bit 2 is set to 0, the total power is reset. If it is set to 1, it continues to integrate providing a new value at each integration interval. Each time a new value is available, bit `0x10` of the status register is set.

Register 9 controls the integration time and number of bits discarded in the FFT total power detector. The register is similar, but only total power and DC offset functions are available. The corresponding read register is unused as total power is read for each individual FFT channel.

Register `0xa` is used to send/receive data over the ADC serial data line. The line is basically a bidirectional UART, and its exact functionality is still TBD.

## 3.3 Input control

Registers `0x0c` to `0x0f` control each of the 4 input blocks, with each block corresponding to one LVDS port. This register is used to specify setting of the alignment FIFO, to read the syncronization detector, and to enable/disable each individual input channel for total power metering.

The register is actually composed of two 16 bit portions, with the MS bits controlling sample 0 (the older) of the two and the MS bits controlling sample 1.

| bit | control | status |
|---|---|---|
| 0x00-07 | Reference state | Sample alignment |
| 0x08 | Enable | - |
| 0x09-11 | Alignment delay | - |

Table 5: Input link control and status register

The alignment delay is used to align all samples from the ADC, compensating any chip-to-chip and sample-to-sample line mismatch. The alignment delay line is short (8 samples), i.e. only bits `0b:09` are actually used.

The actual delay is measured inserting a specific sample code at the ADC, equal to the reference state specified in bits 7-0. A different sync code can be specified for each time multiplexed sample. The sync code is inserted in the 32 samples just after the 1 millisecond strobe, and is catched in the input circuitry in a small window ($\pm$128 samples) around the 1 millisecond strobe. The arrival time is then available in the input status register as a 8 bit signed value. If the specified sample is not detected in the specified window, result is equal to the maximum value (127 samples). Alignment offset is returned in bits 7-0 of the input status word.

## 3.4 FFT output control

Registers `0x10` to `0x18` are used to read the signal level at the 9 FFT outputs, and to adjust the gain of each output channel before the final 8 bit requantization. Channels 0 and 8 are combined together in the output link, but they have independent total power meters and usually have different rescaling factors.

Gain is expressed as a unsigned in the range 0 to 255. The FFT output is multipled by the specified gain, divided by $2^{16}$ and truncated to 8 bit, with clipping. The output value is not rounded, i.e. a value of 0 corresponds to samples in the range 0 to $2^{16} - 1$ (after gain multiplication).

The same address range is used to read the measured total power. Each total power reading is a 31 bit quantity, with the most significant bit used to signal an overflow.

| bit | control |
|---|---|
| 0x00-07 | Gain |
| 0x08-1f | Unused |

Table 6: Output link control and status register

## 3.5 Output link control

The input bandwidth is divided into 32 overlapping sub-channels, ranging from frequency $(k-1)*128$ MHz to $(k+1)*128$ MHz, $k = 0, 31$. Actual subchannel width is smaller (approximately $\pm$90 MHz around

central frequency), for example channel 1 ranges from 38 to 218 MHz, but there is a large overlap between adjacent subchannels.

As the FFT is performed in two steps, partly in the first and partly in the second FPGA, sub-channels are divided into 8 groups, corresponding to the eight FFT outputs of the first chip. Subchannels within the same group are computed using only a single output form the four input chips. Sub-channel group $k$ includes the four sub-channels $k$, $k+16$, $32-k$ and $16-k$. Group 0 is an exception, with sub-channels 0, 8, 16 and 24.

For 1 GHz mode the FFT is completed in the first chip, and each group conatins only the sub-channel $k$. For 2 GHz mode, the group cuntains sub-channels $k$ and $16-k$ (or 0,8 for group 0).

Each output chip can analyze the portion of the input band relative to two consecutive sub-channel groups.

Each output link register specifies which FFT outputs are routed to each output FPGA, and optionally if the actual signal is substituted by a pseudorandom sequence. The groups are specified using bits 0:2 of the link control register, with bit 0 unused (always seen as 0). For correct operations these registers mus be set with the same configuration in all input FPGAs.

Bits 3 and 4 control the Pseudorandom Data Generator. If bits 6-5 are set to "01" the output signals are substituted with a pseudorandom sequence, identical for the two channels in the link. The PRDG is initialized at each millisecond if bit 4 is set. Bit 3 enables the generator, the sequence is identically zero if it is not set.

Bit 6-5 select various test signals to be sent to the output. Normal operation requires these bits to be set to 0. If they are set to 1, the output consists of pseudo random samples, generated by the PRDG. Different bytes are sent to the real and imaginary parts of each sample, but the samples for the two channels are identical.

If these bits are set to 2, the byte specified in bits 17-10 is sent on the real and impaginary parts of each channel every ms pulse. All other samples are identically zero. Samples are always zero (channel off) if these bits are set to 3.

| bit | control |
|---|---|
| 0x00-02 | Channel group |
| 0x03 | Enable PRDG |
| 0x04 | Initalize seed for PRDG |
| 0x05-06 | Output mode: |
| | 0 = normal samples |
| | 1 = PRDG samples |
| | 2 = sync word |
| | 3 = off (all 0) |
| 0x10-17 | sync word |

Table 7: Output link control register

# 4 Interconnect link module

Back side and front side FPGAs are interconnected using a $4 \times 4$ fast interconnect mesh. Each front side FPGA is connected to each back-side FPGA by an independent, bidirectional link.

Each link in turn is composed of 4 fast serial links, with a maximum speed between 6 and 8 Gbit/s. 3 of these links have a hard IP block that provides advanced communication functions (8B/10B encoding/decoding, word and byte alignment, etc.), while the fourth (not used in this design) provides only physical interface and serialization/deserialization.

# 5 Output (front side) FPGA

Each FPGA receives samples from the 4 input FPGAs, corresponding to two outputs different phases, from the 4 input FPGAs. Each signal is composed of complex samples at 256 MS/s, representing the output of the previous stages of the 64 channel FFT. For each group the four signals represent FFT outputs at index $j + 16k$, with $j$ and $k$ the index of the channel group and of the input FPGA, respectively.

The complex conjugate of each input sample represents the output at index $16 - j + 16k$, apart for channel group 0, where the real and imaginary parts correspond to channels $16k$ and $16k + 8$ respectively.

From these samples the FPGA compute the final part of the FFT, obtaining up to 8 frequency channels, each one representing a 256 MHz wide portion of the input signal as a 256 MS/s complex signal.

From these signals, up to 16 real data streams are computed, with independently selectable bandwidth and position. Bandwidth can be chosen from 1 to 64 MHz (2 to 128 MS/s), with a position resolution of 0.01 MHz.

## 5.1 Address map

The addressing space of the $2^{nd}$ stage component is 256 bytes long, organized as 64 32 bit registers. The register address mapping is shown in table 8. The first 8 positions (only 3 used) are used for general control. The next 8 are used for the formatter and output sections. Next 16 registers control the input and FFT blocks, and the last 32 control each of the 16 DBBCs.

Registers 0, 1 and 2 are similar to corresponding registers in the input FPGA. FFT and BBC total power registers (`0x10, 0x11`) specify the integration time, the number of bits discarded in the result, and the function integrated with the same format used in first stage component, and described in chapter 4.

The FFT total power can be used to check the statistics and the signal integrity for the input lines, by placing the FFT stage in the *16 channel FFT mode*.

| address | write | read |
|---|---|---|
| 00 | Test point select | Test point readback |
| 01 | General control | General status |
| 02 | Interrupt register | Interrupt status |
| 03-07 | unused | unused |
| 08 | Output formatter control | Formatter status |
| 09 | Output formatter time | unused |
| 0a-0b | Output formatter ext reg. | unused |
| 10 | FFT Total power ctl | FFT Total power status |
| 11 | BBC Total power ctl | BBC Total power status |
| 12-13 | unused | unused |
| 14-17 | Input 0-3 FIFO ctl | Input 0-3 FIFO status |
| 18-1f | FFT 0-7 control | FFT 0-7 total power |
| 20 | BBC 0 ctl | BBC 0 status |
| 21 | BBC 0 frequency | BBC 0 total power |
| ... | ... | ... |
| 3e | BBC 15 ctl | BBC 15 status |
| 3f | BBC 15 frequency | BBC 15 total power |

Table 8: Register mapping for Uniboard Digital Receiver output chip

## 5.2 General control register

The general control register is used to specify global settings of the chip.

The low power mode is used to place everything in the lowest comsumption mode. Most functions are disabled, inputs and outputs are zeroed.

| bit | control | status |
|---|---|---|
| 0x00 | Low power mode | = |
| 0x01 | FFT reset | = |
| 0x02-07 | FFT mode | = |
| 0x08-0d | Reserved | = |
| 0x0e | RDG enable | = |
| 0x0f | RDG seed load | = |
| 0x11 | FFT TP ready reset | FFT TP ready |
| 0x12 | FFT overflow reset | FFT overflow |

Table 9: Register mapping for general control register. Status bits specified as "=" are a copy of the corresponding control bits

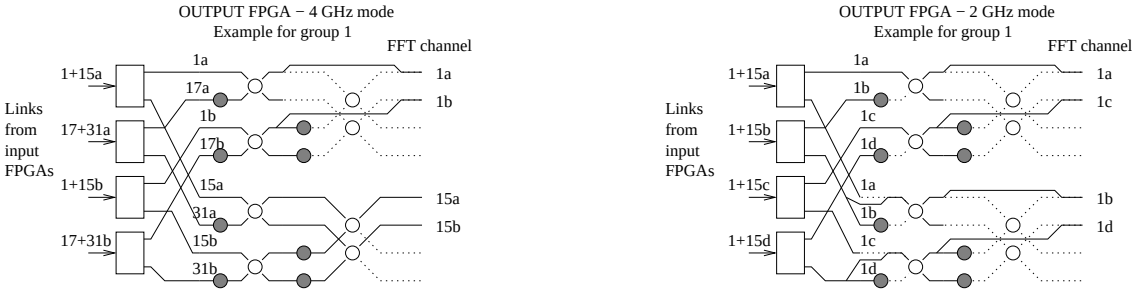


Figure 4: 32 channels (2 GHz) and 16 channels (1 GHz) modes

FFT modes are listed in table 10.

First 16 modes are the normal 64 channel FFT modes. If the input signals are for group $k$, output channels can be either $k, k+16$ (first half), or $32-k, 16-k$ (second half).

Modes `0x10-1f` are used for a 4 GHz input bandwidth (32 output channels). In this case both outputs correspond to channel $k$ (first half) and $32-k$ (second half) for the two ADCs connected to the top and bottom input FPGAs. In this mode, the first butterfly is not modified, and the second one is bypassed, with the two outputs corresponding to the top outputs of the first butterflies (fig. 4a).

Modes `0x20, 21` are used for 2 GHz ADCs. In this mode, each input FPGA completely process the signal sampled by a single ADC, and the butterfly simply copies its inputs (either the upper or the lower 2) to its outputs.

| Mode | Function |
|---|---|
| 00-07 | 64 channel FFT, channel group 0-7, first half |
| 08-0f | 64 channel FFT, channel group 0-7, second half |
| 10-17 | 32 channel FFT, channel group 0-7, first half |
| 18-1f | 32 channel FFT, channel group 0-7, second half |
| 20 | 16 channel FFT, inputs 0 & 2 |
| 28 | 16 channel FFT, inputs 1 & 3 |
| 30-3f | disabled |

Table 10: FFT modes

Modes for the 4 FFT blocks are linked. Channel groups from the first chip are always consecutive, with the first one always even (e.g. 0&1), and couples of FFT blocks related to the same group must use modes $k$ and $k+8$. Therefore the only possible modes available are those in the list below. For each mode the output channels available at the FFT block outpu are listed.

| | 4 GHz | | | | 4 GHz | | | | 1GHz |
|---|---|---|---|---|---|---|---|---|---|
| Mode | 0 | 2 | 4 | 6 | 10 | 12 | 14 | 16 | 20 |
| 0 | 0 | 2 | 4 | 6 | 0a | 2a | 4a | 6a | 0a |
| 1 | 16 | 18 | 20 | 22 | 0b | 2b | 4b | 6b | 0b |
| 2 | 8 | 14 | 12 | 10 | 8a | 14a | 12a | 10a | 0c |
| 3 | 24 | 30 | 28 | 26 | 8b | 14b | 12b | 10b | 0c |
| 4 | 1 | 3 | 5 | 7 | 1a | 3a | 5a | 7a | 1a |
| 5 | 17 | 19 | 21 | 23 | 1b | 3b | 5b | 7b | 1b |
| 6 | 15 | 13 | 11 | 9 | 15a | 13a | 11a | 9a | 1c |
| 7 | 31 | 29 | 27 | 25 | 15b | 13b | 11b | 9b | 1d |

## 5.3 FFT control registers

The FFT stage is composed of 4 identical blocks, each one processing 4 signals from 4 input chips, and producing two outputs (the other two are the associated complex conjugates), for a total of 8 independent outputs.

Each output has an independent gain setting (unsigned in the range 0 to 255) and total power meter. All total power measurements are done in parallel, using the setting specified in register `0x11`.

Output signal is requantized to 8 bit, with an independent rescaling factor output gain for each output, and two total power monitor points.

## 5.4 DBBC block

DBBC control register specifies the band, gain, and other functions for each one of the 16 DBBCs. Bit mapping for this register is shown in fig. 11.

| Bit | Control | Status |
|---|---|---|
| 2-0 | Band select<br>0 = 1/256 (1 MHz)<br>7 = 1/2 (128 MHz) | = |
| 3 | Bypass mode | = |
| 6-4 | Input selection | = |
| 7 | reverse frequency scale | = |
| 8 | Test pattern generation | = |
| 9 | Reset | = |
| 10 | Load LO frequency, syncronous | = |
| 11 | Load LO frequency, immediate | = |
| 13-12 | Real mode | = |
| 14 | TP OVF reset | TP OVF |
| 15 | TP ready reset | TP Ready |
| 23-16 | Phase offset | = |
| 31-24 | Output gain | = |

Table 11: Register mapping for the BBC Control/Status registers

### 5.4.1 Bandwidth selection

Bits $0-2$ select the decimation, and thus the band.

The DBBC selects a portion of the complex input bandwidth, converting it to a real valued signal with a reduced band. The maximum possible band is 1/2 the clock frequency (Nyquist), corresponding to the band select code of 7, and the band can be reduced in binary steps down to 1 MHz (band select = 0). Output signal is resampled at the appropriate Nyquist frequency.

The relative bandshape is similar for all the bands, as the number of filter taps is increased with the decimation factor (tap recirculation).

A *bypass mode* (bit 3) is possible. In this case output samples are just the input real samples. This mode produces meaningful data only for FFT channel 0, and is used mainly to reduce power dissipation when a DBBC is not used.

Bit 7 is used to reverse the frequency scale. This is useful both for the last FFT channel, and to correct different sideband schemes in the receiver for different antennas.

### 5.4.2 Input selection

Bits $4-6$ set the BBC input selection. Each BBC has 8 complex inputs corresponding to 8 complex FFT channels or 7 complex channels plus 2 real channels. The first and last FFT outputs are in fact real, and are pcked together as the real and imaginary parts of FFT channel 0.

To select which real signal to use, bits 11-12 are used. If these bits are both set to 0 (or both to 1), the input signal is considered a complex value. If bit 11 or 12 are set to 1, the real or the imaginary portions of the input are used. The chosen portion is always considered a real valued signal, with the imaginary part set to zero.

### 5.4.3 Local oscillator

The local oscillator is set using the second register of each BBC. The frequency is interpreted as a signed value, in the range $[-f_c/2 - -f_c/2]$i, and $f_c$ the main clock frequency (nominal 256 MHz).

If $F_n$ is the 32 bit signed number written in the register, $f_{lo}$ is given by

$$f_{lo} = \frac{F_n f_c}{2^{32}} = 0.059604645 F_n (\text{MHz})$$

The hardware ignores the 4 least significant bits, and therefore the actual frequency resolution is 0.95367432 Hz.

It is possibile to apply a constant phase offset to the local oscillator output, thus introducing a controlled phas eoffset to the output signal. Phase offset is specified in bits 18-23 of the control word, in steps of 1/64 turn (5.625 degrees).

Frequency is loaded either asyncronously, when bit 11 of the control word is asserted, or syncronously to the 1 ms strobe signal, when bit 10 is asserted. Syncronous assertion is essential to guarantee that multiple BBCs have their local oscillators in a known phase relation.

Frequency change are phase continuous if the reset bits are not used. In this way is for example possible to track an accelerating monochromatic source (typically a spacecraft beacon)

Phase is reset to the phase offset value either asyncronously, using the reset bit, or syncronously to the 1 ms strobe signal, asserting bit 11.

Syncronous reset is useful both during frequency load, if phase continuity is not desired across a frequency change, and to produce LO values multiple of 1 KHz. Setting the local oscillator as close as possible to a multiple of 1 KHz produce a maximum phase error of 1/2000 turn (0.17 degrees or 3.0 milliradians) in one millisecond. If the phase accumulator is reset every millisecond, the phase error does not accumulate over time, and the resulting phase jitter corresponds to a phase noise of less than -50 dB.

### 5.4.4 Output gain and pseudorandom test signal

Bits 31-24 of the control word are used to specify the level of the output real signal. The gain is determined using the result of a total power measurement, and depends on the quantization scheme adopted, using the formula

$$G = K_q \sqrt{\frac{t_i}{2^b P}}$$

where $G$ is the optimum gain value, $P$ is the reading of the total power detector, $t_i$ is the integration time in milliseconds, $b$ is the number of discarded bits in the total power readout, and $K_q$ is a constant

| Quantization scheme | Scaling factor $K_q$ |
|---|---|
| 1 bit | |
| 2 bit | |
| 4 bit | |
| 8 bit | |

Table 12: Scaling factors for output gain for different quantization schemes

that depends on the quantizaton scheme used. The values for $K_q$ for 1, 2, 4 and 8 bit quantizations are given in table 12.

If bit 16 is asserted, the output is replaced by a 8 bit pseudorandom code. The pseudorandom generator can be either free running, with a cycle time of 4194 seconds, or reset to the same starting point every ms if bit 17 is asserted. All other settings are meaningless when the pseudorandom signal is selected.

## 5.5 Output formatter

The signal is then formatted in a VLBI packet using the VDIF format, encapsulated in a UTP *jumbo packet*, and sent to the correlator using one of the output links.

The output link bandwidth is limited by the actual physical connection between the antenna and the correlator, and is usually much less than the card output bandwidth. For example using 16 output channels with 3 bit representation and 60 MHz per channel (128 MSample/s), a total of 6 Gb/s is required. Considering packetization and coding efficiency, this data rate is within the bandwidth of a single 10G link.

Each BBC output is formatted by an individual VDIF formatter, which parameters can be independently specified. Some parameters are common to all formatters (e.g. physical time tag), while others can be specified either individually or together for an arbitrary set of formatters.

Two registers are used to specify all the parameters common to all formatters (control register) and the time (time register). Other two registers are used to access 8 32-bit registers in each formatter.

### 5.5.1 Formatter control and status register

| Bit | Control | Status |
|---|---|---|
| 4-9 | Reference epoch | = |
| 31 | General reset | = |

Table 13: Register mapping for the formatter Control/Status registers

### 5.5.2 Formatter time register

According to the VDIF standard, the time in each frame is measured in units of seconds from a reference epoch.

The epoch is specified as a 6-bit field (in word 1, see below) with a value of 0 corresponding to the first 6 months of 2000, starting at 00H UT 1 Jan 2000, 1 corresponding to the 6 months starting 00H 1 July 2000, etc. After setting, this 6-bit field is static and is not incremented. This field rolls over to 0 again when the reference epoch corresponds to the 6-month period starting 00H 1 Jan 2032.

Seconds from reference epoch are specified as a 30-bit field, and is updated by the formatter every 1pps pulse. The count includes any leap second occurred from the reference time, and increases beyond the 6 month boundary if an observation spans across such a boundary.

To load this value in the formatter clock, the time code corresponding to the *next second* is written in register `0x09`, bits 29:0, and bit 31 set to zero. At the next pps pulse, the value is loaded in the second counter of each formatter, and at this point bit 31 (or the whole register) must be cleared.

Reading the time register returns the time currently counted by formatter 0.

### 5.5.3 Formatter auxiliary registers

Each formatter has 8 extra registers, that are accessed using registersr `0x0a` (auxiliary register address) and `0x0b` (auxiliary register value). The value to be loaded in aux. register $n$ is loaded in register `0x0b`, and then register `0x0a` is loaded with the value $n$ in bits 2:0. The upper 16 bits of register `0x0a` correspond each to a different formatter, with bit $i + 16$ enabling formatter $i$.

For example, to load the value `0xabcd` to register 3 in the formatters from 4 to 7, the value `0xabcd` is loaded in register `0x0b`, and the value `0x00f00003` is loaded in register `0x0a`.

Registers are loaded on a low-to-high transition of the corresponding bit in register `0x0a`, and therefore this register must be cleared before it is loaded.

Auxiliary registers usage is specified in table 14.

| Bit | Control |
|---|---|
| 0 | Destination IP |
| 1 | Souce and dest. UDP ports |
| 2 | Frame length and bits/sample |
| 3 | Thread and station ID |
| 4-7 | VDIF header aux. words 4-7 |

Table 14: Auxiliary register usage

Words 0 and 1 are used to specify the parameters for the UDP link. The source IP is specified in the IP interface module, while other link parameters are speficied on a per-BBC basis. This allows dynamic routing of each output channel e.g. to a different correlator or recorder. Word 0 is used to specify the destination IP addres (IPv4, 4 bytes in network order, i.e. with the first octet at the MS byte). Word 1 is used to specify the source (most significant 16 bits) and destination (least significant 16 bits) ports.

Word 2 is used to specify the frame length and the sample (symbol) length. The frame length is specified, in bytes, in bits 15:0. Since an integer number of 32 bit words is always used (incomplete words in a frame are not supported), bits 0 and 1 are actually ignored and the length must be always a multiple of 4. Bits per sample are specified as a 2-bit code in bits 25:24. Sample length can be only a power of 2, with 1, 2, 4 or 8 bits/sample supported. A code of 0 specifies 1 bit per sample, with 32 samples per word, and code of 3 specifies 8 bits per sample, with 4 samples per word.

Word 3 is used to specify a 16-bit station ID (bits 15:0), and a 10-bit thread ID (bits 25-16). These values will be placed unchanged in word 3 of the VDIF header.

The remaining 4 words are simply copied to the corresponding words of the VDIF header.

# References


[1] G. Comoretto, A. Russo, G. Tuccari, A. Baudry, P. Camino, B. Quertier: "Uniboard Digital Receiver - Initial design document"

[2] "VLBI Data Interchange Format (VDIF) Specification"

# Contents