

FEBEX 4A ADCFIFO

Summary:

This document describes the functionality and interface of the ADCFIFO that has been implemented on the FEBEX4A. The FIFO provides clock domain translation with latency 60nS + phase offset between the 100 MHz sclk0 domain of the ADC and the 100 MHz txclk which is used for GOSSIP communications and crucially the time stamp. The phase relationship between sclk0 is not know but is presumed to be deterministic (otherwise any time calibrations will not persist with less than 10nS uncertainty through power up). This unit is not used when obtaining ADC channel traces, in that circumstance the clock domain transition occurs in the dpm buffers.

Addressing:

Only one register is used to communicate with the ADCFIFO peripheral with the GOSIP address 0x200034. The reads/writes to this register are 32 bits. Write to read is used to read diagnostic bits for the state of the FIFOs, there is no writeable register for this peripheral. For example to read the status word the data payload is set to the value indicated in the table and written to 0x200034, then a read of 0x200034 is initiated which will return the status bits. The channel is selected using the 3rd most significant nibble. Channels are numbered 0 through to 15 (16 total).

Parameter	Data word (32bit, x = don't care)
Status	1000 0001 [chan] xxxx xxxx xxxx xxxx [DATA]
Test Word	1000 0010 xxxx xxxx xxxx xxxx xxxx xxxx

Read example:

In this example the status bits are read out of channel 0 and then the test word is read out:

```
LIPC-2 mbsdaq > goc -w -x 1 0 0x200034 0x81000000
```

```
LIPC-2 mbsdaq > goc -r -x 1 0 0x200034
```

```
0x0
```

```
LIPC-2 mbsdaq > goc -w -x 1 0 0x200034 0x82000000
```

```
LIPC-2 mbsdaq > goc -r -x 1 0 0x200034
```

```
0xdeadbeef
```

```
LIPC-2 mbsdaq >
```

Detailed description of input parameters:

Status:

The status bits occupy the least significant nibble of the read out 32 bit word:

Status(3) = underflow

Status(2) = empty

Status(1) = overflow

Status(0) = full

Test word:

Sets the return value to "DEADBEEF" (32 bits)