Threshold Scan

This module performs a threshold scan on a single VFAT2 by varying its threshold from a minimum value *min* to a maximum value *max* by steps of *steps* and by counting the number of events where the SBits are fired in a set of N events.

Addressing

OptoHybrid Wishbone address: 0b 0001 0000 0000 000X XXXX 0000 0YYYY

Y register	Mode	Function
Control		
0	Write only	Start a threshold scan on the VFAT2 addressed using the X parameter (ID of the VFAT2 encoded on 5 bits). This will also empty the FIFO holding the data of the previous scan. The written value is ignored.
Parameters		
1	Read / write	`min   – 8 bits  – [0x0, max[`
2	Read / write	`max   – 8 bits  – ]min, 0xFF]`
3	Read / write	`steps – 8 bits  – [0x0, 0xFF]`
4	Read / write	`N     – 24 bits – ]0x0, 0xFFFFFF]`
Results		
5	Read only	FIFO holding the results of the scan. This register will return the data points collected by the scan using the following data format: 8 MSBits hold the threshold value of the point 24 LSBits hold the number of events that have fired If no data is present, an error is returned.
Reset		
6	Write only	Local reset of the module

Description

One Threshold Scan module is present per sector on the GEB (6 sectors of 4 VFAT2s). This means that 6 scans can be operated in parallel. The configuration registers ARE NOT shared between sectors but ARE shared between VFAT2s of the same sector. For example, setting the parameters for a scan of VFAT2 #0 means that a scan of VFAT2 #1 can be launched immediately afterwards without having to set the parameters in the registers. However, the parameters set for VFAT2 #0 do not propagate to VFAT2 #7 which is in another sector.

Note that the module will store the value of the threshold before the scan and reapply the latter after the end of the operation.

Errors

Two types of errors can be returned by the module when running the scan: global errors and local errors.

A global error occurs if the VFAT2 is not present or running at the start of the scan. In that case, a single 32 bits word of value 0xFF000000 is stored in the FIFO. No other read operations of the FIFO should occur afterwards.

A local error occurs if one of the I2C operations used to change the threshold did not succeed. In that case, the value of the 24 LSBits of that particular point is 0xFFFFFF. Other data points will still be saved and be present in the FIFO.

Latency Scan

This module performs a latency scan on a single VFAT2 by varying its latency from a minimum value *min* to a maximum value *max* by steps of *steps* and by counting the number of events where the strips are fired in a set of *N* events.

Addressing

OptoHybrid Wishbone address: 0b 0010 0000 0000 000X XXXX 0000 0YYYY

Y register	Mode	Function
Control		
0	Write only	Start a latency scan on the VFAT2 addressed using the X parameter (ID of the VFAT2 encoded on 5 bits). This will also empty the FIFO holding the data of the previous scan.
Parameters		
1	Read / write	`min   – 8 bits  – [0x0, `*max*`[`
2	Read / write	`max   – 8 bits  – ]`*min*`, 0xFF]`
3	Read / write	`steps – 8 bits  – [0x0, 0xFF]`
4	Read / write	`N     – 24 bits – ]0x0, 0xFFFFFF]`
Results		
5	Read only	FIFO holding the results of the scan. This register will return the data points collected by the scan using the following data format: 8 MSBits hold the latency value of the point 24 LSBits hold the number of events that have fired If no data is present, an error is returned.
Reset		
6	Write only	Local reset of the module

Description

One Latency Scan module is present per sector on the GEB (6 sectors of 4 VFAT2s). This means that 6 scans can be operated in parallel. The configuration registers ARE NOT shared between sectors but ARE shared between VFAT2s of the same sector. For example, setting the parameters for a scan of VFAT2 #0 means that a scan of VFAT2 #1 can be launched immediately afterwards without having to set the parameters in the registers. However, the parameters set for VFAT2 #0 do not propagate to VFAT2 #7 which is in another sector.

Note that the module will store the value of the latency before the scan and reapply the latter after the end of the operation.

Errors

Two types of errors can be returned by the module when running the scan: global errors and local errors.

A global error occurs if the VFAT2 is not present or running at the start of the scan. In that case, a single 32 bits word of value 0xFF000000 is stored in the FIFO. No other read operations of the FIFO should occur afterwards.

A local error occurs if one of the I2C operations used to change the latency did not succeed. In that case, the value of the 24 LSBits of that particular point is 0xFFFFFF. Other data points will still be saved and be present in the FIFO.

T1 Controller

This module sends T1 commands to the VFAT2s according to different operation modes defined by *op_mode*.

Addressing

OptoHybrid Wishbone address: 0b 0011 0000 0000 0000 0000 0000 0YYYY

| Y register	Mode	Function
Control		
0	Read / write	Enable (1) / disable (0) the module.
1	Read / write	`op_mode  - 2 bits  - {0, 1, 2}`
Mode 0 & 1 parameters		
2	Read / write	`type     - 2 bits  - {0, 1, 2, 3}` `0 = LV1A - 1 = Calpulse - 2 = Resync - 3 = BC0`
3	Read / write	`N        - 32 bits - [0, 0xFFFFFFFF]` `0 = infinite`
4	Read / write	`interval - 32 bits - [4, 0xFFFFFFFF]`
5	Read / write	`Delay    - 32 bits - [interval + 4, 0xFFFFFFFF]`
Mode 2 parameters		
7 & 6	Read / write	`lv1a_sequence     - 64 bits`
9 & 8	Read / write	`calpulse_sequence - 64 bits`
11 & 10	Read / write	`resync_sequence   - 64 bits`
13 & 12	Read / write	`bc0_sequence      - 64 bits`
Reset		
14	Write only	Local reset of the module

Operation modes

Mode 0

Send *N* T1 commands of type *type* with an interval of *interval* BXs. Note that *interval* cannot be smaller than 4 BXs which is the time needed to encode a T1 command on the wire. Example with an *interval* of 4 BXs:

```
CLK   _|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_

T1     _|‾|_____|‾|_____|‾|_____|‾|_____
```
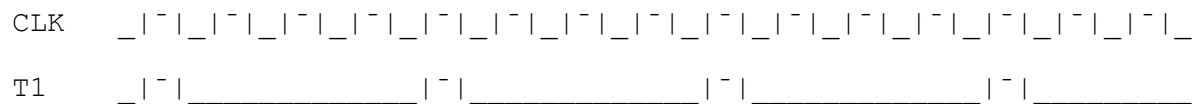
Mode 1

Send *N* packets composed of an LV1A followed by a Calpulse separated by *interval* BXs. The packets are spaced by *delay* BXs. Note that *interval* cannot be smaller than 4 BXs which is the time needed to encode a T1 command on the wire and that *delay* must be greater or equal to (*interval* + 4) BXs. Example with an *interval* of 4 BXs and a *delay* of 10 BXs:

```
CLK    _|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_

LV1A  _|‾|_____|‾|_____

CAL    _____|‾|_____|‾|_
```
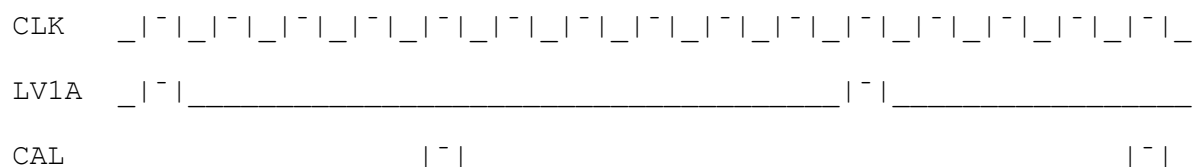
Mode 2

Send T1 commands according to a pattern defined by the sequence registers: `lv1a_sequence`, `calpulse_sequence`, `resync_sequence`, and `bc0_sequence`. Every 4 BXs, the module reads a bit in each of the registers and sets/resets the T1 line according to the asserted bits. This operation mode allows the user to create custom patterns of T1 commands. The module will loop over the registers indefinitely. Example of a generated pattern using the `lv1a_sequence` and `calpulse_sequence` registers.

```
CLK       _|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_|‾|_

SEQ_LV1A  __1_____1_____

SEQ_BC0   _____1_____

LV1A      _|‾|_____|‾|_____

BC0       _____|‾|_____
```