Configuration Block

William Wulff

Upstream DAQ Firmware

26-11-2020

Content

- Goals
- Planned design
- Design structure
- Update of parameters
- Comments
- Status

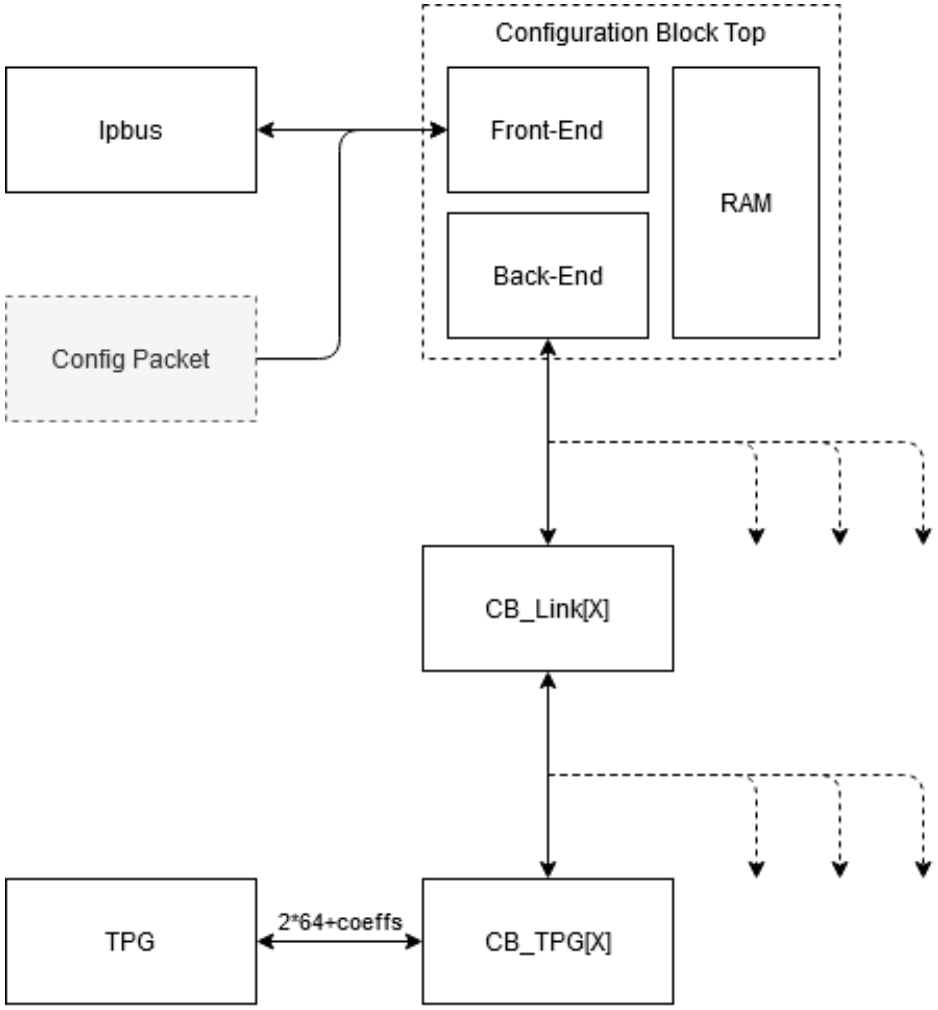
Goals

- Minimise ressource usage – low speed is ok
- Provide easy-to-use interface for configuration
- Scalability and expandability – should be easy to introduce new functionality in the form of parameters and blocks
- Compatibility with two different front-ends – i.e. configuration packet and Ipbus
- Offer Read/Write functionality and global/local parameter addressing

Planned design

- Interface based on Ipbus dRAM + registers
  - RAM sets parameters – 2 * #links * #TPGs * 64 + 32 integers
  - Control interface via registers – CB status, commands, I/O(?)
- Simple bus for getting and setting parameter values
  - Flexible addressing
  - Endpoint with connection to SSR storage – buffer as little data as possible

Structure


Configuration Block Top
Ipbus
Front-End
RAM
Back-End
Config Packet
CB_Link[X]
TPG
2*64+coeffs
CB_TPG[X]

Updating

- Bus system
  - Outgoing: Address | Command | <Data>
  - Ingoing: Data
  - Control: bsy, req, etc
  - Can be parallel or bit-serial
- Commands
  - Write (pedestal, threshold, coefficients)
  - Read (pedestal, threshold, coefficients)
  - Status
  - Target: Wire, TPG, link, global

Some comments

Benefits

- Fewer registers needed
- Simplified outward interface
- Compatibility with future Configuration Packet
- All parameters are handled in a similar manner
- Flexibility…?

Issues

- Addition of new bus system
- Complex structure and solution

Current status

Done

- Basic test-bench
- Initial design

In progress

- Expanded test-bench
- Design finalisation
- Tests of bus concept

To-do

- Configuration block implementation
- Documentation
- Configuration packet specification