

OTU1 Framer Core

16-bit Data Width

Revision History

| Version | Date | Details |
|----------------|-------------|--------------------|
| Version 0.1 | 26.4.2004 | First Draft |
| Version 0.2 | 8.11.2004 | Updated pin-out |
| Version 0.3 | 17.11.2004 | Added Virtex4 info |

Contents

| | |
|---|----|
| 1. Features | 4 |
| 2. Functional Description | 4 |
| 2.1. Frame Generation | 4 |
| 2.2. FEC Interface | 4 |
| 2.3. OTUk Overhead | 4 |
| 2.4. ODUk Overhead | 5 |
| 2.5. OPuk Overhead | 5 |
| 2.6. Payload Interface | 5 |
| 3. Signal Description | 6 |
| 3.1. Data Output Interface | 6 |
| 3.2. FEC Interface | 7 |
| 3.3. Upstream Timing Output | 8 |
| 3.4. Upstream Timing/Data Input | 9 |
| 3.5. Status and Configuration Interface | 10 |
| 3.5.1. Frame Generation | 10 |
| 3.5.2. Overhead | 11 |
| 4. Implementation Details | 13 |
| 4.1. Resource Utilisation | 13 |
| 4.1.1. Altera | 13 |
| 4.1.2. Xilinx | 13 |
| 4.2. Ordering Information | 14 |

1. Features

The Aliathon OTU1 Framers core provides a flexible, resource-efficient, programmable-logic based solution for OTU interfacing, with a 16-bit (2 byte) wide data path. The core...

- Generates the OTU1 frame structure and inserts framing overhead.
- Scrambles the OTU frame, inserts OTUk Overhead, and calculates Section Monitoring BIP.
- Inserts ODUk Overhead, and calculates Path Monitoring BIP.
- Inserts synchronous payload mappings (eg: ATM, GFP), and asynchronous SONET/SDH mappings (accommodating positive/negative stuffing).
- Provides a contiguous 16 bit-wide payload output.
- Interfaces to Aliathon's OTU1 encoder core for efficient, low-latency G.709 FEC.

2. Functional Description

Figure 2 illustrates the major functional blocks within the OTU Framers Core.

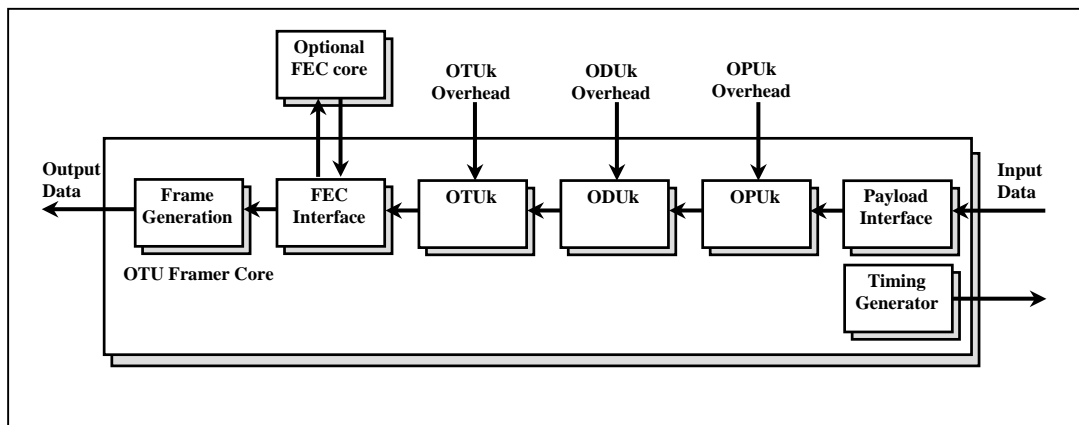


Figure 2 – OTU Framers Core Functional Blocks

2.1. Frame Generation

The Frame generation block inserts the OTU framing pattern and performs frame scrambling.

2.2. FEC Interface

This interface connects to Aliathon's G.709 OTU1 Encoder to provide efficient, low-latency FEC functionality.

2.3. OTUk Overhead

The OTUk Overhead block inserts OTUk overhead and calculates the Section Monitoring BIP.

2.4. ODUk Overhead

The ODUk Overhead block inserts ODUk overhead and calculates the Path Monitoring and TCM BIP.

2.5. OPUk Overhead

The OPUk Overhead block inserts OPUk overhead and optionally inserts the SONET/SDH specific positive/negative stuffing control bits.

2.6. Payload Interface

This block accepts the OTU payload, accommodating any SONET/SDH payload stuffing. It aligns the data to provide a contiguous 16 bit-wide payload output, accommodating any asynchronous payload stuffing.

3. Signal Description

The input and output signals are grouped by function into the following interfaces.

- Data output
- FEC Interface
- Upstream Timing/Data input
- Status and configuration

3.1. Data Output Interface

This interface provides output data from the framer core. Typically an external LIU or transponder device would connect to these signals.

| Name | Type | Description |
|-------------------------|------|--|
| reset | I | Asynchronous reset input. |
| otn_clk | I | otn_clk is typically supplied by the external LIU. All inputs and outputs from the core are synchronous to this clock unless otherwise noted. |
| otn_req | I | General chip-enable input to core. In most applications this will be tied high. |
| otn_sync | I | Presets the framer timing generator to a known value. In most applications this will be tied low. |
| otn_data (15..0) | O | otn_data is the 16 bit wide OTN data output. Note that bit 15 is the most significant (first received). |
| otn_vld | O | Indicates that the otn_data output carries valid data. If the otn_req input is tied high then this output will always be high. |
| otn_sof | O | Indicates that otn_data carries the first word of an OTN frame. |

3.2. FEC Interface

This interface connects to Aliathon's FEC encoder core. If FEC is not being used then the outputs of this interface should be hard-wired to the inputs.

| Name | Type | Description |
|-------------------------|------|--|
| dfo_vld | O | When asserted this indicates that the following signals are valid. |
| dfo_data (15..0) | O | OTN frame data. The FEC overhead locations are all 0. |
| dfo_sof | O | Indicates that dfo_data is the first data word of an OTU frame. |
| ufi_vld | I | When asserted this indicates that the following inputs are valid. |
| ufi_data (15..0) | I | OTN frame data with added FEC overhead. |
| ufi_sof | I | Indicates that ufi_data is the first data word of an OTU frame. |

3.3. Upstream Timing Output

This interface will typically drive an upstream device with OTN timing signals, such as Aliathon's STM16/OC48 framer core.

| Name | Type | Description |
|---------------------------|------|--|
| uto_vld | O | Indicates that the following signals are valid |
| uto_sof | O | Indicates that this cycle represents the start of an OTU frame. |
| uto_somf | O | Indicates that this cycle represents the start of an OTU multi-frame. |
| uto_cs_vld | O | Indicates that this cycle represents valid client-signal data (ie: not OUT overhead). |
| uto_cs_jst (1..0) | O | This signal indicates what payload justification is being applied to the current frame. Valid values are... 00 : None 01 : Negative 10 : Positive |
| uto_cs_bv (1..0) | O | This output indicates how many payload bytes are valid in this cycle. A value of 1 (01) may occur due to payload justification. |
| uto_pyld_vld | O | Indicates a valid payload cycle for data sources providing contiguous 16-bit payload data. |
| uto_pyld_os (1..0) | O | This indicates that state of the shift register used to absorb payload justification. It is only used to specialised applications. |

3.4. Upstream Timing/Data Input

This interface will typically be driven by an upstream device with OTN timing and data signals, such as Aliathon's STM16/OC48 framer core.

| Name | Type | Description |
|-------------------------------|------|---|
| uti_vld | | Indicates that the following signals are valid. |
| uti_sof | | Indicates that this clock-cycle is the start of an OTU frame. |
| uti_somf | | Indicates that this clock-cycle is the start of an OTU multi-frame. |
| uti_data (15..0) | | Raw OTU payload data. |
| uti_cs_vld | | Indicates valid client-signal data (ie : not OTU overhead). |
| uti_cs_jst (1..0) | | Indicates what payload justification is occurring this frame. Values are 00 : None 01 : Negative 10 : Positive |
| uti_cs_bv (1..0) | | Indicates the number of valid payload bytes (which may be 1 due to payload justification). |
| uti_pyld_vld | | Indicates that the following payload inputs are valid. |
| uti_pyld (15..0) | | Contiguous 16 bit payload data input. |
| uti_pyld_os (1..0) | | Infers the state of the shift register used to align contiguous payload data. Only used in specialized applications. |

3.5. Status and Configuration Interface

This interface allows the core to be dynamically configured, and provides status outputs. To lower resource utilisation, any unused outputs should be left open, and configuration inputs should be hard-wired to the required value if they do not need to change. The following configuration and status signals are grouped by function.

3.5.1. Frame Generation

| Name | Type | Description |
|---------------------------------|------|---|
| cfg_rate (1..0) | I | This configuration inputs sets the OTU rate of the core. Defined values are... 00 : OTU1 01 : OTU2 (not valid for this core) 10 : OTU3 (not valid for this core) 11 : not defined |
| cfg_scram_ off | I | When asserted the OTU frame is not scrambled. |
| cfg_pyld_ type(3..0) | I | This defines the type of OTU payload that the core generates. Valid values are... 0000 : Disabled – no payload will be generated. 0001 : Transparent – raw OTN data is passed through unprocessed. 0010 : Experimental payload (synchronous payload, pt=0x01) 0011 : Asynchronous CBR – The core inserts positive/negative stuffing to accommodate the asynchronous CBR (usually SONET/SDH) mapping. (pt=0x02) 0100 : Synchronous CBR – The core inserts synchronous CBR (usually SONET/SDH) mapping. (pt=0x03) 0101 : ATM mapping (pt=0x04) 0110 : GFP mapping (pt=0x05) 0111 : Octect-aligned bit stream mapping (pt=0x10) 1000 : Non-octet-aligned bit stream mapping (pt=0x11) 1001 : Null mapping (all 0s transmitted) (pt=0xFD) 1010 : Test PRBS mapping (PRBS-31 transmitted) (pt=0xFE) |
| cfg_otuk_ais | I | When asserted the core transmits OTU AIS. |
| cfg_oduk_ais | I | When asserted the core transmits ODU AIS. |
| cfg_oduk_ oci | I | When asserted the core transmits ODU OCI (open connection indication) |
| cfg_oduk_lck | I | When asserted the core transmits ODU LCK (locked signal). |

| Name | Type | Description |
|--------------------|------|--|
| cfg_jst_vld | O | When asserted this indicates a request from the core to determine what payload stuffing to insert. The following two signals should be valid in the following clock cycle to influence payload stuffing. |
| cfg_pos_jst | I | When asserted in response to cfg_jst_vld the core generates positive payload justification. This may be under the control of a register interface or DDS for test purposes, or based on the status of a fifo for asynchronous signal mapping. |
| cfg_pos_jst | I | When asserted in response to cfg_jst_vld the core generates negative payload justification. This may be under the control of a register interface or DDS for test purposes, or based on the status of a fifo for asynchronous signal mapping. |

3.5.2. Overhead

Overhead is inserted on this interface.

| Name | Type | Description |
|----------------------------|------|---|
| otu_oh_vld | O | When asserted this indicates that there is the opportunity to insert overhead data on otu_oh_data . Note that all overhead types (OTUk, OTDk, OPUk) are indicated here. The overhead inputs should be valid in the clock-cycle following this signal being asserted. |
| otu_oh_row (1..0) | O | This indicates which OTN row the overhead data is for. |
| otu_oh_col (11..0) | O | This indicates which OTN column the overhead data is for (0-15). |
| otu_oh_mf (7..0) | O | The 8 bit OTN multi-frame count, which is used in conjunction with some overhead locations. |
| otu_oh_bip (7..0) | O | The calculated BIP value for this frame, which may be inserted in overhead locations. |
| otu_oh_data (15..0) | I | The overhead data. |
| otu_oh_mask (15..0) | I | This mask is applied to the otu_oh_data input. Only bits qualified with a matching '1' in this input are inserted into the OTU frame. |

4. Implementation Details

4.1. Resource Utilisation

The following figures are calculated assuming that all core IOs are routed off-chip. This results in a worst-case resource utilisation figure, and for any given application the resource utilisation is likely to be lower. The core exceeds OTN performance requirements (168MHz for OTU1 at 16 bits wide).

The core can be targeted for devices from other families and manufactures. Contact Aliathon for further details.

4.1.1. Altera

| | Stratix Family Eg : EP1S10F484C6 | | | Stratix GX Family Eg : EP1SGX10CF672C6 | | |
|----------------------|-------------------------------------|-----------------|-----------------|---|-----------------|-----------------|
| | Used by Core | In example Part | Percentage used | Used by Core | In example Part | Percentage used |
| Logic Elements (LEs) | 498 | 10570 | 5% | 498 | 10570 | 5% |
| M512 RAM Blocks | 0 | 94 | 0% | 0 | 94 | 0% |
| M4k RAM Blocks | 0 | 60 | 0% | 0 | 60 | 0% |
| M-RAM Blocks | 0 | 1 | 0% | 0 | 1 | 0% |
| Fmax | > 170MHz | | | > 170MHz | | |

| | Stratix2 Family Eg : EP2S15F484C5 | | |
|-----------------|--------------------------------------|-----------------|-----------------|
| | Used by Core | In example Part | Percentage used |
| ALMs | 281 | 6240 | 5% |
| M512 RAM Blocks | 0 | 104 | 0% |
| M4k RAM Blocks | 0 | 78 | 0% |
| M-RAM Blocks | 0 | 1 | 0% |
| Fmax | > 170MHz | | |

4.1.2. Xilinx

| | Virtex-II Family Eg : XC2V1000-5FG456 | | | Virtex-II Pro Family Eg : XC2VP7-6FG456 | | | Spartan3 Family Eg : XC3S1000-5FG456 | | |
|-----------|--|-----------------|-----------------|--|-----------------|-----------------|---|-----------------|-----------------|
| | Used by Core | In example Part | Percentage used | Used by Core | In example Part | Percentage used | Used by Core | In example Part | Percentage used |
| Slices | 270 | 5120 | 6% | 270 | 4928 | 6% | 270 | 7680 | 4% |
| Blockrams | 0 | 40 | 0% | 0 | 44 | 0% | 0 | 24 | 0% |
| Fmax | > 170MHz | | | > 170MHz | | | > 170MHz | | |

| | Virtex-IV Family Eg : XC4VLX15FF668-10 | | |
|-----------|---|-----------------|-----------------|
| | Used by Core | In example Part | Percentage used |
| Slices | 265 | 6140 | 5% |
| Blockrams | 0 | 40 | 0% |
| Fmax | > 170MHz | | |

4.2. Ordering Information

For technical enquiries and ordering please contact Aliathon Ltd at:

*Aliathon Ltd
Evans Business Centre
Pitreavie Court, Dunfermline
United Kingdom
KY11 5UU*

Phone: +44 (0)1383 737 736
Fax: +44 (0)1383 749 501
Email: info@aliathon.com
Web: www.aliathon.com