

OTU1 Deframer 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 with new pinout
Version 0.3	17.11.2004	Added Virtex4 info

Contents

1. Features	4
2. Functional Description	4
2.1. Frame Alignment	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 Input Interface	6
3.2. FEC Interface	7
3.3. Downstream Timing/Data Output	8
3.4. Status and Configuration Interface	9
3.4.1. Frame Synchronisation	9
3.4.2. Overhead	11
4. Implementation Details	12
4.1. Resource Utilisation	12
4.1.1. Altera	12
4.1.2. Xilinx	12
4.2. Ordering Information	13

1. Features

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

- Detects and aligns to the OTU framing pattern.
- Performs frame synchronisation for OTU1.
- Descrambles the OTU frame, extracts OTUk Overhead, and detects Section Monitoring BIP errors.
- Extracts ODUk Overhead, and detects Path Monitoring BIP errors.
- Extracts 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 decoder core for efficient, low-latency G.709 FEC.

2. Functional Description

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

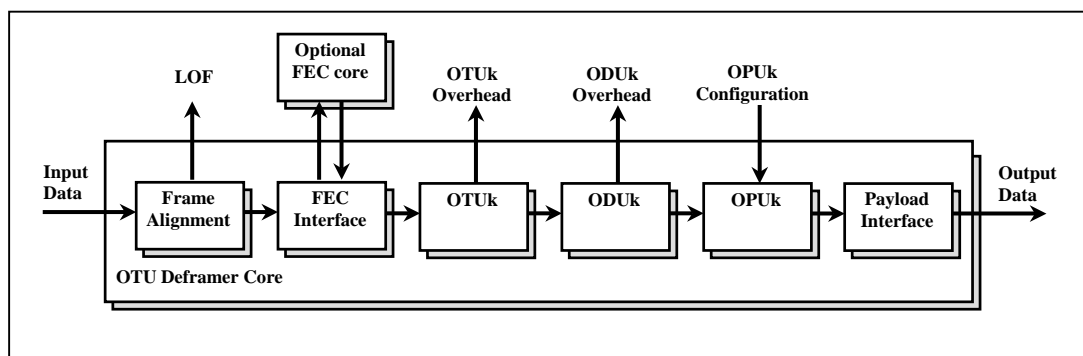


Figure 2 – OTU Deframer Core Functional Blocks

2.1. Frame Alignment

The Frame alignment block detects the OTU framing pattern at any bit offset within the input data stream, and aligns to it.

2.2. FEC Interface

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

2.3. OTUk Overhead

The OTUk Overhead block extracts OTUk overhead and verifies the Section Monitoring BIP calculation.

2.4. ODUk Overhead

The ODUk Overhead block extracts ODUk overhead and verifies the Path Monitoring and TCM BIP calculations.

2.5. OPUk Overhead

The OPUk Overhead block extracts OPUk overhead and optionally monitors the SONET/SDH specific positive/negative stuffing control bits. Detected stuff locations are passed on to the Payload Interface block.

2.6. Payload Interface

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

3. Signal Description

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

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

3.1. Data Input Interface

This interface provides the input clock and data to the deframer core. Typically an external LIU or transponder device would drive these signals, but the core provides internal frame alignment and a generic chip-enable input that allows other data sources to drive the core.

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_data (15..0)	I	otn_data is the 16 bit wide OTN data input. When otn_vld is asserted it must contain valid data. Note that bit 15 is the most significant (first received).
otn_vld	I	When asserted this indicates that there is valid data on otn_data . If an external CDR device is used then this input would typically be tied high, as CDR devices provide data on every clock cycle. This input is useful for driving the core with data from another clock domain, via an asynchronous fifo, for example.

3.2. FEC Interface

This interface connects to Aliathon's FEC decoder 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	Pre-corrected OTN frame data.
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	Corrected OTN frame data.
ufi_sof	I	Indicates that ufi_data is the first data word of an OTU frame.

3.3. Downstream Timing/Data Output

This interface will typically drive a downstream device with OTN timing and data signals, such as Aliathon's STM16/OC48 Deframer core. Note that in most applications the downstream deframer logic (ATM, GFP, SONET/SDH) only needs to connect to the `dto_pyld_vld` and `dto_pyld` signals. The other signals are used in specialized applications (eg: test/measurement, Mux/Demux, etc).

Name	Type	Description
<code>dto_vld</code>	○	This output indicates that the following signals are valid.
<code>dto_sof</code>	○	When asserted this output indicates that <code>dto_data</code> contains the first word of an OTU frame.
<code>dto_somf</code>	○	This output is only asserted when <code>dto_sof</code> is also asserted. It indicates that <code>dto_data</code> contains the first word of an OTU multi-frame.
<code>dto_data</code> (15..0)	○	This output carries raw OTU payload data.
<code>dto_cs_vld</code>	○	This output indicates that <code>dto_data</code> contains valid client-signal data (ie: not OTU overhead).
<code>dto_cs_bv</code> (1..0)	○	In some cases payload justification may result in only the least-significant byte of <code>dto_data</code> being valid. This output indicates this case when its value is "01". Otherwise it is "00" or "10" to indicate 0 to 2 valid bytes.
<code>dto_pyld_vld</code>	○	When asserted this indicates that the following payload data outputs are valid.
<code>dto_pyld</code> (15..0)	○	The OTN frame payload data. When valid this output always contains 16 valid data bits. For asynchronous payloads (eg: asynchronous SONET/SDH) the core absorbs payload justification to provide contiguous payload on this output.
<code>dto_pyld_os</code> (1..0)	○	This output indicates the state of the shift register used to absorb payload justification. It is only used in conjunction with Aliathon's OTU1 framer core in specialized applications.

3.4. 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.

3.4.1. Frame Synchronisation

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 OTU descrambling is disabled. This input should typically be deasserted.
cfg_pyld_ type(3..0)	I	This defines the type of OTU payload that the core processes. Valid values are... 0000 : Disabled – no payload will be output. 0001 : Transparent – raw OTN data is passed through unprocessed. 0011 : Asynchronous – The core processes positive/negative stuffing to accommodate the asynchronous SONET/SDH mapping. All other values result in a synchronous payload mapping.
st_oof	O	This output indicates that the OTN Deframer core is not synchronized to the OTN frame structure (out-of-frame).
st_lof	O	When asserted this output indicates that the fs_oof state has persisted for 3ms or more (loss-of-frame).
st_los	O	When asserted this output indicates that the input data stream has been all 0s for 20µs or more (loss-of-signal).
st_mf_lof	O	When asserted this indicates that the core is not synchronized to the OTU multiframe count.

Name	Type	Description
st_otuk_ais	O	When asserted this indicates that the core has detected OTU AIS.
st_oduk_ais	O	When asserted this indicates that the core has detected ODU AIS.
st_oduk_oci	O	When asserted this indicates that the core has detected OTU OCI (open channel indication).
st_oduk_lck	O	When asserted this indicates that the core has detected OTU LCK (locked channel).

3.4.2. Overhead

All OTU overhead is available on this interface

Name	Type	Description
otu_oh_vld	O	When asserted this indicates that there is valid overhead data on otu_oh_data . Note that all overhead types (OTUk, OTDk, OPUk) are available here.
otu_oh_row (1..0)	O	This indicates which OTN row the overhead data is from.
otu_oh_col (11..0)	O	This indicates which OTN column the overhead data is from.
otu_oh_data (15..0)	O	The overhead data.
otu_oh_mf(7..0)	O	The 8 bit OTN multi-frame count, which is used to conjunction with some overhead locations.
otu_oh_bip (7..0)	O	The calculated OTU BIP value. This value may be compared with the received OTU overhead to detect BIP errors.

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)	689	10570	7%	689	10570	7%
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	418	6240	7%
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	472	5120	10%	472	4928	10%	472	7680	7%
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	438	6144	8%
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