



Welcome to [E-XFL.COM](https://www.e-xfl.com)

Understanding [Embedded - FPGAs \(Field Programmable Gate Array\)](#)

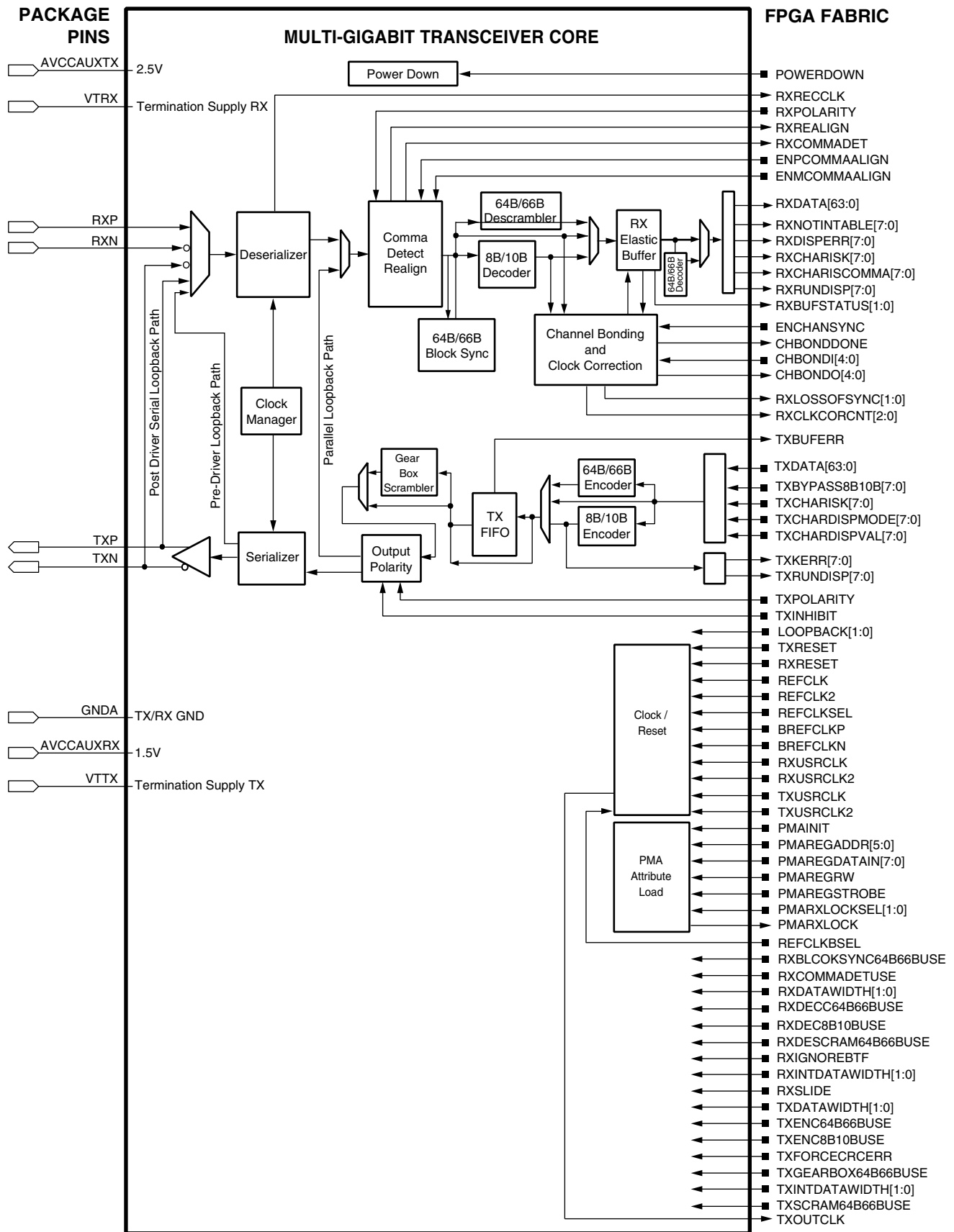
Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details

Product Status	Obsolete
Number of LABs/CLBs	4848
Number of Logic Elements/Cells	43632
Total RAM Bits	3538944
Number of I/O	416
Number of Gates	-
Voltage - Supply	1.425V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	676-BGA
Supplier Device Package	676-FBGA (27x27)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc2vp40-5fg676i



DS083-2_37_050704

Figure 4: RocketIO X Transceiver Block Diagram

cation is given at the receiver interface. The realignment indicator is a distinct output.

The transceiver continuously monitors the data for the presence of the 10-bit character(s). Upon each occurrence of a 10-bit character, the data is checked for word alignment. If comma detect is disabled, the data is not aligned to any particular pattern. The programmable option allows a user to align data on comma+, comma–, both, or a unique user-defined and programmed sequence.

Comma detection has been expanded beyond 10-bit symbol detection and alignment to include 8-bit symbol detection and alignment for 16-, 20-, 32-, and 40-bit paths. The ability to detect symbols, and then either align to 1-word, 2-word, or 4-word boundaries is included. The RXSLIDE input allows the user to “slide” or “slip” the alignment by one bit in each 16-, 20-, 32- and 40-bit mode at any time for SONET applications. Comma detection can be bypassed when needed.

Clock Correction

RXRECCLK (the recovered clock) reflects the data rate of the incoming data. RXUSRCLK defines the rate at which the FPGA fabric consumes the data. Ideally, these rates are identical. However, since the clocks typically have different sources, one of the clocks will be faster than the other. The receiver buffer accommodates this difference between the clock rates. See [Figure 6](#).

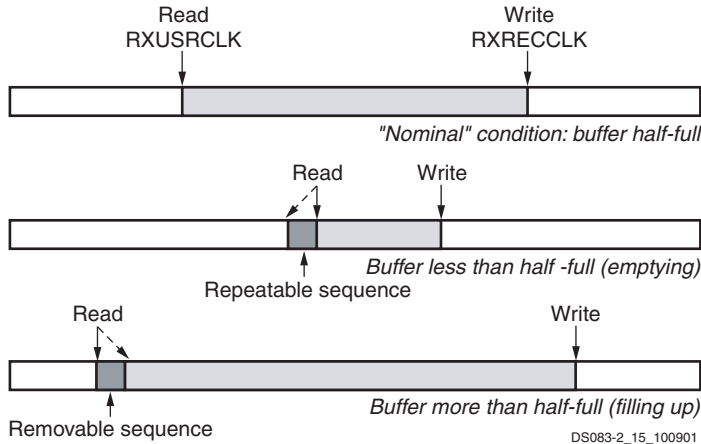


Figure 6: Clock Correction in Receiver

Nominally, the buffer is always half full. This is shown in the top buffer, [Figure 6](#), where the shaded area represents buffered data not yet read. Received data is inserted via the write pointer under control of RXRECCLK. The FPGA fabric reads data via the read pointer under control of RXUSRCLK. The half full/half empty condition of the buffer gives a cushion for the differing clock rates. This operation continues indefinitely, regardless of whether or not “meaningful” data is being received. When there is no meaningful data to be received, the incoming data will consist of IDLE characters or other padding.

If RXUSRCLK is faster than RXRECCLK, the buffer becomes more empty over time. The clock correction logic corrects for this by decrementing the read pointer to reread a repeatable byte sequence. This is shown in the middle buffer, [Figure 6](#), where the solid read pointer decrements to the value represented by the dashed pointer. By decrementing the read pointer instead of incrementing it in the usual fashion, the buffer is partially refilled. The transceiver design will repeat a single repeatable byte sequence when necessary to refill a buffer. If the byte sequence length is greater than one, and if attribute CLK_COR_REPEAT_WAIT is 0, then the transceiver may repeat the same sequence multiple times until the buffer is refilled to the desired extent.

Similarly, if RXUSRCLK is slower than RXRECCLK, the buffer will fill up over time. The clock correction logic corrects for this by incrementing the read pointer to skip over a removable byte sequence that need not appear in the final FPGA fabric byte stream. This is shown in the bottom buffer, [Figure 6](#), where the solid read pointer increments to the value represented by the dashed pointer. This accelerates the emptying of the buffer, preventing its overflow. The transceiver design will skip a single byte sequence when necessary to partially empty a buffer. If attribute CLK_COR_REPEAT_WAIT is 0, the transceiver may also skip two consecutive removable byte sequences in one step to further empty the buffer when necessary.

These operations require the clock correction logic to recognize a byte sequence that can be freely repeated or omitted in the incoming data stream. This sequence is generally an IDLE sequence, or other sequence comprised of special values that occur in the gaps separating packets of meaningful data. These gaps are required to occur sufficiently often to facilitate the timely execution of clock correction.

Channel Bonding

Some gigabit I/O standards such as Infiniband specify the use of multiple transceivers in parallel for even higher data rates. Words of data are split into bytes, with each byte sent over a separate channel (transceiver). See [Figure 7](#).

The top half of the figure shows the transmission of words split across four transceivers (channels or lanes). PPPP, QQQQ, RRRR, SSSS, and TTTT represent words sent over the four channels.

The bottom-left portion of [Figure 7](#) shows the initial situation in the FPGA’s receivers at the other end of the four channels. Due to variations in transmission delay—especially if the channels are routed through repeaters—the FPGA fabric might not correctly assemble the bytes into complete words. The bottom-left illustration shows the incorrect assembly of data words PQPP, QRQQ, RSRR, and so forth.

To support correction of this misalignment, the data stream includes special byte sequences that define corresponding points in the several channels. In the bottom half of [Figure 7](#), the shaded “P” bytes represent these special characters. Each receiver recognizes the “P” channel bond-

The top half of the figure shows the transmission of words split across four transceivers (channels or lanes). PPPP, QQQQ, RRRR, SSSS, and TTTT represent words sent over the four channels.

The bottom-left portion of Figure 13 shows the initial situation in the FPGA's receivers at the other end of the four channels. Due to variations in transmission delay—especially if the channels are routed through repeaters—the FPGA fabric might not correctly assemble the bytes into complete words. The bottom-left illustration shows the incorrect assembly of data words PQPP, QRQQ, RSRR, and so forth.

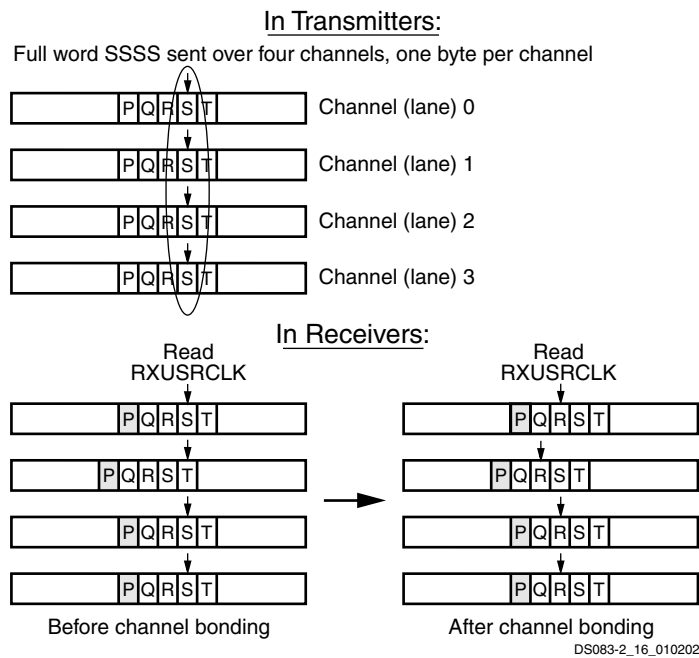


Figure 13: Channel Bonding (Alignment)

To support correction of this misalignment, the data stream includes special byte sequences that define corresponding points in the several channels. In the bottom half of Figure 13, the shaded "P" bytes represent these special characters. Each receiver recognizes the "P" channel bonding character, and remembers its location in the buffer. At some point, one transceiver designated as the master instructs all the transceivers to align to the channel bonding character "P" (or to some location relative to the channel bonding character).

After this operation, words transmitted to the FPGA fabric are properly aligned: RRRR, SSSS, TTTT, and so forth, as shown in the bottom-right portion of Figure 13. To ensure that the channels remain properly aligned following the channel bonding operation, the master transceiver must also control the clock correction operations described in the previous section for all channel-bonded transceivers.

Transmitter Buffer

The transmitter's buffer write pointer (TXUSRCLK) is frequency-locked to its read pointer (REFCLK). Therefore, clock correction and channel bonding are not required. The purpose of the transmitter's buffer is to accommodate a phase difference between TXUSRCLK and REFCLK. A simple FIFO suffices for this purpose. A FIFO depth of four will permit reliable operation with simple detection of overflow or underflow, which could occur if the clocks are not frequency-locked.

RocketIO Configuration

This section outlines functions that can be selected or controlled by configuration. Xilinx implementation software supports 16 transceiver primitives, as shown in Table 6.

Each of the primitives in Table 6 defines default values for the configuration attributes, allowing some number of them to be modified by the user. Refer to the [RocketIO Transceiver User Guide](#) for more details.

Table 6: Supported RocketIO MGT Protocol Primitives

GT_CUSTOM	Fully customizable by user
GT_FIBRE_CHAN_1	Fibre Channel, 1-byte data path
GT_FIBRE_CHAN_2	Fibre Channel, 2-byte data path
GT_FIBRE_CHAN_4	Fibre Channel, 4-byte data path
GT_ETHERNET_1	Gigabit Ethernet, 1-byte data path
GT_ETHERNET_2	Gigabit Ethernet, 2-byte data path
GT_ETHERNET_4	Gigabit Ethernet, 4-byte data path
GT_XAUI_1	10-gigabit Ethernet, 1-byte data path
GT_XAUI_2	10-gigabit Ethernet, 2-byte data path
GT_XAUI_4	10-gigabit Ethernet, 4-byte data path
GT_INFINIBAND_1	Infiniband, 1-byte data path
GT_INFINIBAND_2	Infiniband, 2-byte data path
GT_INFINIBAND_4	Infiniband, 4-byte data path
GT_AURORA_1 ⁽¹⁾	1-byte data path
GT_AURORA_2 ⁽¹⁾	2-byte data path
GT_AURORA_4 ⁽¹⁾	4-byte data path

Notes:

- For more information on the Aurora protocol, visit <http://www.xilinx.com>.

Other RocketIO Features and Notes

CRC

The RocketIO transceiver CRC logic supports the 32-bit invariant CRC calculation used by Infiniband, FibreChannel, and Gigabit Ethernet.

On the transmitter side, the CRC logic recognizes where the CRC bytes should be inserted and replaces four placeholder bytes at the tail of a data packet with the computed CRC. For Gigabit Ethernet and FibreChannel, transmitter

Functional Description: Processor Block

This section briefly describes the interfaces and components of the Processor Block. The subsequent section, **Functional Description: Embedded PowerPC 405 Core** beginning on [page 20](#), offers a summary of major PPC405 core features. For an in-depth discussion on both the Processor Block and PPC405, see the [PowerPC Processor Reference Guide](#) and the [PowerPC 405 Processor Block Reference Guide](#) available on the Xilinx website at <http://www.xilinx.com>.

Processor Block Overview

Figure 14 shows the internal architecture of the Processor Block.

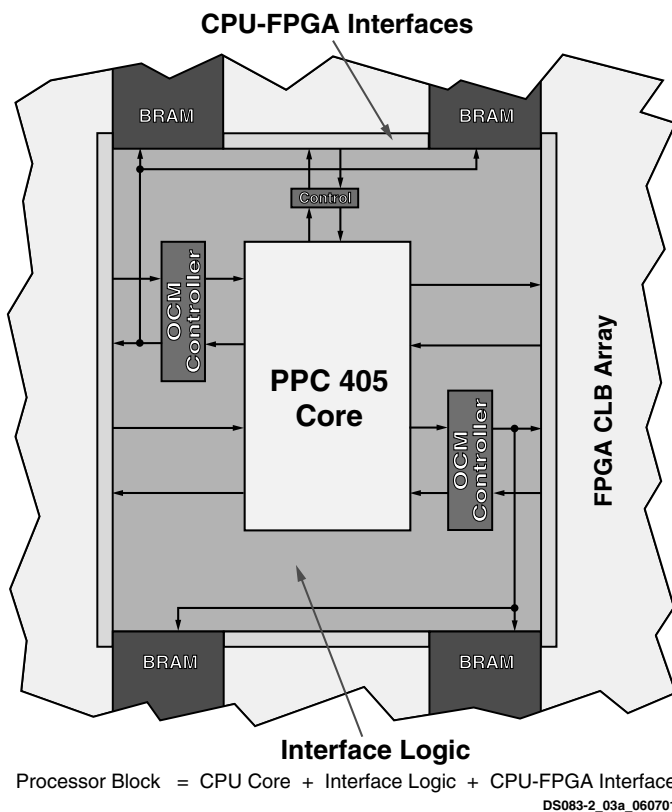


Figure 14: Processor Block Architecture

Within the Virtex-II Pro Processor Block, there are four components:

- Embedded IBM PowerPC 405-D5 RISC CPU core
- On-Chip Memory (OCM) controllers and interfaces
- Clock/control interface logic
- CPU-FPGA Interfaces

Embedded PowerPC 405 RISC Core

The PowerPC 405D5 core is a 0.13 μm implementation of the IBM PowerPC 405D4 core. The advanced process technology enables the embedded PowerPC 405 (PPC405)

core to operate at 300+ MHz while maintaining low power consumption. Specially designed interface logic integrates the core with the surrounding CLBs, block RAMs, and general routing resources. Up to four Processor Blocks can be available in a single Virtex-II Pro device.

The embedded PPC405 core implements the PowerPC User Instruction Set Architecture (UISA), user-level registers, programming model, data types, and addressing modes for 32-bit fixed-point operations. 64-bit operations, auxiliary processor operations, and floating-point operations are trapped and can be emulated in software.

Most of the PPC405 core features are compatible with the specifications for the PowerPC Virtual Environment Architecture (VEA) and Operating Environment Architecture (OEA). They also provide a number of optimizations and extensions to the lower layers of the PowerPC Architecture. The full architecture of the PPC405 is defined by the PowerPC Embedded Environment and PowerPC UISA documentation, available from IBM.

On-Chip Memory (OCM) Controllers

Introduction

The OCM controllers serve as dedicated interfaces between the block RAMs in the FPGA fabric (see [18 Kb Block SelectRAM+ Resources, page 44](#)) and OCM signals available on the embedded PPC405 core. The OCM signals on the PPC405 core are designed to provide very quick access to a fixed amount of instruction and data memory space. The OCM controller provides an interface to both the 64-bit Instruction-Side Block RAM (ISBRAM) and the 32-bit Data-Side Block RAM (DSBRAM). The designer can choose to implement:

- ISBRAM only
- DSBRAM only
- Both ISBRAM and DSBRAM
- No ISBRAM and no DSBRAM

One of OCM's primary advantages is that it guarantees a fixed latency of execution for a higher level of determinism. Additionally, it reduces cache pollution and thrashing, since the cache remains available for caching code from other memory resources.

Typical applications for DSOCM include scratch-pad memory, as well as use of the dual-port feature of block RAM to enable bidirectional data transfer between processor and FPGA. Typical applications for ISOCM include storage of interrupt service routines.

Functional Features

Common Features

- Separate Instruction and Data memory interface between processor core and BRAMs in FPGA
- Dedicated interface to Device Control Register (DCR) bus for ISOCM and DSOCM

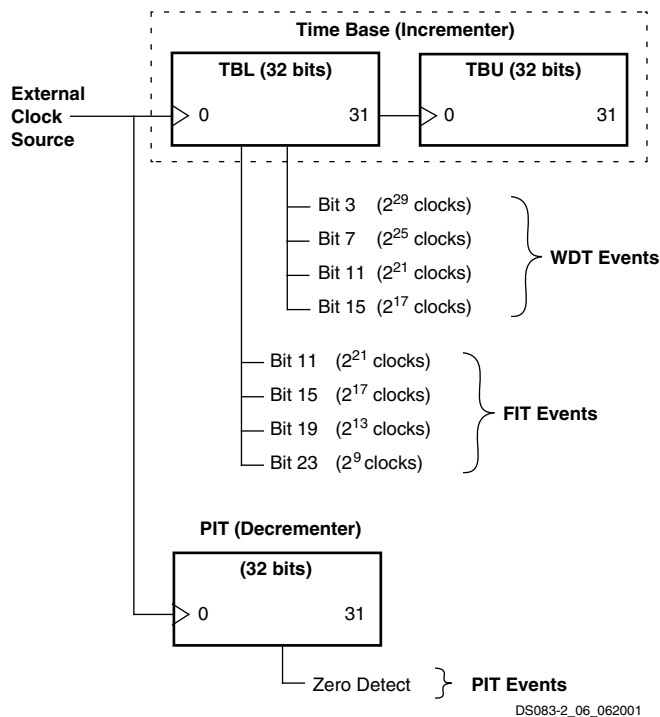


Figure 17: Relationship of Timer Facilities to Base Clock

Interrupts

The PPC405 provides an interface to an interrupt controller that is logically outside the PPC405 core. This controller combines the asynchronous interrupt inputs and presents them to the embedded core as a single interrupt signal. The sources of asynchronous interrupts are external signals, the JTAG/debug unit, and any implemented peripherals.

Debug Logic

All architected resources on the embedded PPC405 core can be accessed through the debug logic. Upon a debug event, the PPC405 core provides debug information to an external debug tool. Three different types of tools are supported depending on the debug mode: ROM monitors, JTAG debuggers, and instruction trace tools.

In internal debug mode, a debug event enables exception-handling software at a dedicated interrupt vector to take

over the CPU core and communicate with a debug tool. The debug tool has read-write access to all registers and can set hardware or software breakpoints. ROM monitors typically use the internal debug mode.

In external debug mode, the CPU core enters stop state (stops instruction execution) when a debug event occurs. This mode offers a debug tool read-write access to all registers in the PPC405 core. Once the CPU core is in stop state, the debug tool can start the CPU core, step an instruction, freeze the timers, or set hardware or software breakpoints. In addition to CPU core control, the debug logic is capable of writing instructions into the instruction cache, eliminating the need for external memory during initial board bring-up. Communication to a debug tool using external debug mode is through the JTAG port.

Debug wait mode offers the same functionality as external debug mode with one exception. In debug wait mode, the CPU core goes into wait state instead of stop state after a debug event. Wait state is identical to stop state until an interrupt occurs. In wait state, the PPC405 core can vector to an exception handler, service an interrupt and return to wait state. This mode is particularly useful when debugging real time control systems.

Real-time trace debug mode is always enabled. The debug logic continuously broadcasts instruction trace information to the trace port. When a debug event occurs, the debug logic signals an external debug tool to save instruction trace information before and after the event. The number of instructions traced depends on the trace tool.

Debug events signal the debug logic to stop the CPU core, put the CPU core in debug wait state, cause a debug exception or save instruction trace information.

Big Endian and Little Endian Support

The embedded PPC405 core supports big endian or little endian byte ordering for instructions stored in external memory. Since the PowerPC architecture is big endian internally, the ICU rearranges the instructions stored as little endian into the big endian format. Therefore, the instruction cache always contains instructions in big endian format so that the byte ordering is correct for the execution unit. This feature allows the 405 core to be used in systems designed to function in a little endian environment.

Table 3: DC Characteristics Over Recommended Operating Conditions

Symbol	Description	Virtex-II Pro X			Virtex-II Pro			Units
		Min	Typ	Max	Min	Typ	Max	
V_{DRINT}	Data retention V_{CCINT} voltage (below which configuration data might be lost)	1.25			1.25			V
V_{DRI}	Data retention V_{CCAUX} voltage (below which configuration data might be lost)	2.0			2.0			V
I_{REF}	V_{REF} current per pin			10			10	μA
I_L	Input or output leakage current per pin (sample-tested)			10			10	μA
C_{IN}	Input capacitance (sample-tested)			10			10	pF
I_{RPU}	Pad pull-up (when selected) @ $V_{in} = 0V$, $V_{CCO} = 2.5V$ (sample tested)			150			150	μA
I_{RPD}	Pad pull-down (when selected) @ $V_{in} = 2.5V$ (sample-tested)			150			150	μA
$I_{BATT}^{(1)}$	Battery supply current	Note (2)			Note (2)			nA
I_{CCAUTX}	Operating AVCCAUTX supply current		115			60	105	mA
I_{CCAUXX}	Operating AVCCAUXRX supply current		85			35	75	mA
I_{TTX}	Operating I_{TTX} supply current when transmitter is AC-coupled		55			30		mA
	Operating I_{TTX} supply current when transmitter is DC-coupled	N/A	N/A	N/A		15		mA
I_{TRX}	Operating I_{TRX} supply current when receiver is AC-coupled		15			0		mA
	Operating I_{TRX} supply current when receiver is DC-coupled	N/A	N/A	N/A		15		
P_{CPU}	Power dissipation of PowerPC™ 405 processor block		0.9			0.9		mW/MHz
$P_{RXTX}^{(3)}$	Power dissipation of MGT @ 1.25 Gb/s per channel	N/A	N/A	N/A		230		mW
	Power dissipation of MGT @ 2.5 Gb/s per channel		290			310		mW
	Power dissipation of MGT @ 3.125 Gb/s per channel		310			350		mW
	Power dissipation of MGT @ 4.25 Gb/s per channel		450		N/A	N/A	N/A	mW
	Power dissipation of MGT @ 6.25 Gb/s per channel		525		N/A	N/A	N/A	mW

Notes:

1. Characterized, not tested.
2. Battery supply current (I_{BATT}):

	Device Unpowered	Device Powered	Units
25°C:	< 50	< 10	nA
85°C:	N/A	< 10	nA

3. Total dissipation of fully operational PMA and PCS combined. This power is the average power supply dissipation per MGT. The averaging was done by simultaneously turning on all eight transceivers and dividing the total power supply dissipation by eight.

Table 7: FG676/FGG676 — XC2VP20, XC2VP30, and XC2VP40

Bank	Pin Description	Pin Number	No Connects		
			XC2VP20	XC2VP30	XC2VP40
3	IO_L49N_3	T24			
3	IO_L49P_3	U24			
3	IO_L48N_3	U23			
3	IO_L48P_3	U22			
3	IO_L47N_3	T19			
3	IO_L47P_3	U19			
3	IO_L45N_3/VREF_3	V26			
3	IO_L45P_3	V25			
3	IO_L43N_3	V24			
3	IO_L43P_3	V23			
3	IO_L42N_3	V22			
3	IO_L42P_3	V21			
3	IO_L41N_3	V20			
3	IO_L41P_3	V19			
3	IO_L39N_3/VREF_3	W26			
3	IO_L39P_3	W25			
3	IO_L37N_3	W21			
3	IO_L37P_3	W20			
3	IO_L36N_3	Y26			
3	IO_L36P_3	Y25			
3	IO_L35N_3	Y24			
3	IO_L35P_3	Y23			
3	IO_L33N_3/VREF_3	W22			
3	IO_L33P_3	Y22			
3	IO_L31N_3	AA26			
3	IO_L31P_3	AA25			
3	IO_L24N_3	AA24	NC		
3	IO_L24P_3	AA23	NC		
3	IO_L23N_3	Y21	NC		
3	IO_L23P_3	AA21	NC		
3	IO_L06N_3	AB26			
3	IO_L06P_3	AB25			
3	IO_L05N_3	AA22			
3	IO_L05P_3	AB23			
3	IO_L03N_3/VREF_3	AC26			

Table 10: FF1152 — XC2VP20, XC2VP30, XC2VP40, and XC2VP50

Bank	Pin Description	Pin Number	No Connects			
			XC2VP20	XC2VP30	XC2VP40	XC2VP50
4	IO_L73P_4	AG17				
4	IO_L74N_4/GCLK3S	AH17				
4	IO_L74P_4/GCLK2P	AJ17				
4	IO_L75N_4/GCLK1S	AK17				
4	IO_L75P_4/GCLK0P	AL17				
5	IO_L75N_5/GCLK7S	AL18				
5	IO_L75P_5/GCLK6P	AK18				
5	IO_L74N_5/GCLK5S	AJ18				
5	IO_L74P_5/GCLK4P	AH18				
5	IO_L73N_5	AG18				
5	IO_L73P_5	AF18				
5	IO_L69N_5/VREF_5	AL19				
5	IO_L69P_5	AK19				
5	IO_L68N_5	AJ19				
5	IO_L68P_5	AH19				
5	IO_L67N_5	AE18				
5	IO_L67P_5	AD18				
5	IO_L57N_5/VREF_5	AL20				
5	IO_L57P_5	AL21				
5	IO_L56N_5	AJ20				
5	IO_L56P_5	AH20				
5	IO_L55N_5	AG19				
5	IO_L55P_5	AF19				
5	IO_L54N_5	AM22				
5	IO_L54P_5	AM21				
5	IO_L53_5/No_Pair	AK21				
5	IO_L50_5/No_Pair	AJ21				
5	IO_L49N_5	AE19				
5	IO_L49P_5	AD19				
5	IO_L48N_5	AL23				
5	IO_L48P_5	AL22				
5	IO_L47N_5	AH21				
5	IO_L47P_5	AG21				
5	IO_L46N_5	AF20				
5	IO_L46P_5	AE20				
5	IO_L45N_5/VREF_5	AM24				
5	IO_L45P_5	AL24				

Table 10: FF1152 — XC2VP20, XC2VP30, XC2VP40, and XC2VP50

Bank	Pin Description	Pin Number	No Connects			
			XC2VP20	XC2VP30	XC2VP40	XC2VP50
7	IO_L43N_7	M31				
7	IO_L42P_7	L32				
7	IO_L42N_7	L31				
7	IO_L41P_7	N28				
7	IO_L41N_7	N27				
7	IO_L40P_7	M33				
7	IO_L40N_7/VREF_7	L33				
7	IO_L39P_7	M29				
7	IO_L39N_7	M28				
7	IO_L38P_7	N26				
7	IO_L38N_7	N25				
7	IO_L37P_7	L34				
7	IO_L37N_7	K34				
7	IO_L36P_7	L30				
7	IO_L36N_7	L29				
7	IO_L35P_7	L28				
7	IO_L35N_7	L27				
7	IO_L34P_7	K33				
7	IO_L34N_7/VREF_7	J33				
7	IO_L33P_7	K31				
7	IO_L33N_7	K30				
7	IO_L32P_7	M26				
7	IO_L32N_7	M25				
7	IO_L31P_7	H34				
7	IO_L31N_7	H33				
7	IO_L24P_7	H32	NC			
7	IO_L24N_7	H31	NC			
7	IO_L23P_7	K28	NC			
7	IO_L23N_7	K27	NC			
7	IO_L22P_7	J32	NC			
7	IO_L22N_7/VREF_7	J31	NC			
7	IO_L21P_7	J30	NC			
7	IO_L21N_7	J29	NC			
7	IO_L20P_7	G34	NC			
7	IO_L20N_7	G33	NC			
7	IO_L19P_7	H30	NC			
7	IO_L19N_7	H29	NC			
7	IO_L18P_7	L26	NC			

Table 10: FF1152 — XC2VP20, XC2VP30, XC2VP40, and XC2VP50

Bank	Pin Description	Pin Number	No Connects			
			XC2VP20	XC2VP30	XC2VP40	XC2VP50
N/A	GND	V19				
N/A	GND	V20				
N/A	GND	V21				
N/A	GND	W1				
N/A	GND	W14				
N/A	GND	W15				
N/A	GND	W16				
N/A	GND	W17				
N/A	GND	W18				
N/A	GND	W19				
N/A	GND	W20				
N/A	GND	W21				
N/A	GND	W34				
N/A	GND	Y8				
N/A	GND	Y14				
N/A	GND	Y15				
N/A	GND	Y16				
N/A	GND	Y17				
N/A	GND	Y18				
N/A	GND	Y19				
N/A	GND	Y20				
N/A	GND	Y21				
N/A	GND	Y27				
N/A	GND	AA14				
N/A	GND	AA15				
N/A	GND	AA16				
N/A	GND	AA17				
N/A	GND	AA18				
N/A	GND	AA19				
N/A	GND	AA20				
N/A	GND	AA21				
N/A	GND	AC5				
N/A	GND	AC8				
N/A	GND	AC27				
N/A	GND	AC30				
N/A	GND	AE3				
N/A	GND	AE32				
N/A	GND	H23				

Table 12: FF1517 — XC2VP50 and XC2VP70

Bank	Pin Description	Pin Number	No Connects	
			XC2VP50	XC2VP70
3	IO_L28P_3	AH3		
3	IO_L27N_3/VREF_3	AJ7		
3	IO_L27P_3	AJ8		
3	IO_L26N_3	AF8		
3	IO_L26P_3	AF9		
3	IO_L25N_3	AJ5		
3	IO_L25P_3	AJ6		
3	IO_L24N_3	AJ3		
3	IO_L24P_3	AJ4		
3	IO_L23N_3	AF10		
3	IO_L23P_3	AG10		
3	IO_L22N_3	AJ1		
3	IO_L22P_3	AJ2		
3	IO_L21N_3/VREF_3	AK6		
3	IO_L21P_3	AK7		
3	IO_L20N_3	AF11		
3	IO_L20P_3	AF12		
3	IO_L19N_3	AK4		
3	IO_L19P_3	AK5		
3	IO_L18N_3	AK1		
3	IO_L18P_3	AK2		
3	IO_L17N_3	AG9		
3	IO_L17P_3	AH8		
3	IO_L16N_3	AL6		
3	IO_L16P_3	AL7		
3	IO_L15N_3/VREF_3	AK3		
3	IO_L15P_3	AL3		
3	IO_L14N_3	AG11		
3	IO_L14P_3	AH11		
3	IO_L13N_3	AL1		
3	IO_L13P_3	AL2		
3	IO_L12N_3	AM6		
3	IO_L12P_3	AM7		
3	IO_L11N_3	AH10		
3	IO_L11P_3	AJ9		
3	IO_L10N_3	AL5		
3	IO_L10P_3	AM4		
3	IO_L09N_3/VREF_3	AM2		

Table 12: FF1517 — XC2VP50 and XC2VP70

Bank	Pin Description	Pin Number	No Connects	
			XC2VP50	XC2VP70
7	IO_L24N_7	L37		
7	IO_L23P_7	P31		
7	IO_L23N_7	P32		
7	IO_L22P_7	L34		
7	IO_L22N_7/VREF_7	L35		
7	IO_L21P_7	L32		
7	IO_L21N_7	L33		
7	IO_L20P_7	N29		
7	IO_L20N_7	M29		
7	IO_L19P_7	K38		
7	IO_L19N_7	K39		
7	IO_L18P_7	J37		
7	IO_L18N_7	K37		
7	IO_L17P_7	N30		
7	IO_L17N_7	P30		
7	IO_L16P_7	K35		
7	IO_L16N_7/VREF_7	K36		
7	IO_L15P_7	K34		
7	IO_L15N_7	K33		
7	IO_L14P_7	N31		
7	IO_L14N_7	M32		
7	IO_L13P_7	J38		
7	IO_L13N_7	J39		
7	IO_L12P_7	J35		
7	IO_L12N_7	H36		
7	IO_L11P_7	M30		
7	IO_L11N_7	L31		
7	IO_L10P_7	J33		
7	IO_L10N_7/VREF_7	J34		
7	IO_L09P_7	H37		
7	IO_L09N_7	H38		
7	IO_L08P_7	K31		
7	IO_L08N_7	K32		
7	IO_L07P_7	H33		
7	IO_L07N_7	H34		
7	IO_L84P_7	G38	NC	
7	IO_L84N_7	G39	NC	
7	IO_L82P_7	G36	NC	

Table 12: FF1517 — XC2VP50 and XC2VP70

Bank	Pin Description	Pin Number	No Connects	
			XC2VP50	XC2VP70
N/A	GND	W18		
N/A	GND	V18		
N/A	GND	U18		
N/A	GND	T18		
N/A	GND	AD17		
N/A	GND	AC17		
N/A	GND	AB17		
N/A	GND	AA17		
N/A	GND	Y17		
N/A	GND	W17		
N/A	GND	V17		
N/A	GND	U17		
N/A	GND	P20		
N/A	GND	L20		
N/A	GND	G20		
N/A	GND	C20		
N/A	GND	AD19		
N/A	GND	AC19		
N/A	GND	AB19		
N/A	GND	AA19		
N/A	GND	Y19		
N/A	GND	W19		
N/A	GND	V19		
N/A	GND	U19		
N/A	GND	T19		
N/A	GND	AD18		
N/A	GND	AC18		
N/A	GND	U21		
N/A	GND	T21		
N/A	GND	AU20		
N/A	GND	AN20		
N/A	GND	AJ20		
N/A	GND	AF20		
N/A	GND	AD20		
N/A	GND	AC20		
N/A	GND	AB20		
N/A	GND	AA20		
N/A	GND	Y20		

Table 13: FF1704 — XC2VP70, XC2VPX70, and XC2VP100

Bank	Pin Description		Pin Number	No Connects	
	Virtex-II Pro Devices	XC2VPX70 (if Different)		XC2VP70, XC2VPX70	XC2VP100
2	IO_L44P_2		U10		
2	IO_L45N_2		U3		
2	IO_L45P_2		U4		
2	IO_L46N_2/VREF_2		U1		
2	IO_L46P_2		U2		
2	IO_L47N_2		T12		
2	IO_L47P_2		U12		
2	IO_L48N_2		V10		
2	IO_L48P_2		V11		
2	IO_L49N_2		V7		
2	IO_L49P_2		V8		
2	IO_L50N_2		U11		
2	IO_L50P_2		V12		
2	IO_L51N_2		V4		
2	IO_L51P_2		V5		
2	IO_L52N_2/VREF_2		V1		
2	IO_L52P_2		V2		
2	IO_L53N_2		W9		
2	IO_L53P_2		W10		
2	IO_L54N_2		W7		
2	IO_L54P_2		W8		
2	IO_L55N_2		W5		
2	IO_L55P_2		W6		
2	IO_L56N_2		W11		
2	IO_L56P_2		W12		
2	IO_L57N_2		W3		
2	IO_L57P_2		W4		
2	IO_L58N_2/VREF_2		W1		
2	IO_L58P_2		W2		
2	IO_L59N_2		Y9		
2	IO_L59P_2		Y10		
2	IO_L60N_2		Y6		
2	IO_L60P_2		Y7		
2	IO_L85N_2		Y3		
2	IO_L85P_2		Y4		
2	IO_L86N_2		Y11		

Table 13: FF1704 — XC2VP70, XC2VPX70, and XC2VP100

Bank	Pin Description		Pin Number	No Connects	
	Virtex-II Pro Devices	XC2VPX70 (if Different)		XC2VP70, XC2VPX70	XC2VP100
N/A	RXPPAD17		BB15		
N/A	GNDA17		AY16		
N/A	TXPPAD17		BB16		
N/A	TXNPAD17		BB17		
N/A	VTTXPAD17		BA17		
N/A	AVCCAUXTX17		BA16		
N/A	AVCCAUXRX18		BA18		
N/A	VTRXPAD18		BA19		
N/A	RXNPAD18		BB18		
N/A	RXPPAD18		BB19		
N/A	GNDA18		AY21		
N/A	TXPPAD18		BB20		
N/A	TXNPAD18		BB21		
N/A	VTTXPAD18		BA21		
N/A	AVCCAUXTX18		BA20		
N/A	AVCCAUXRX19		BA22		
N/A	VTRXPAD19		BA23		
N/A	RXNPAD19		BB22		
N/A	RXPPAD19		BB23		
N/A	GNDA19		AY22		
N/A	TXPPAD19		BB24		
N/A	TXNPAD19		BB25		
N/A	VTTXPAD19		BA25		
N/A	AVCCAUXTX19		BA24		
N/A	AVCCAUXRX20		BA26		
N/A	VTRXPAD20		BA27		
N/A	RXNPAD20		BB26		
N/A	RXPPAD20		BB27		
N/A	GNDA20		AY27		
N/A	TXPPAD20		BB28		
N/A	TXNPAD20		BB29		
N/A	VTTXPAD20		BA29		
N/A	AVCCAUXTX20		BA28		
N/A	AVCCAUXRX21		BA30		
N/A	VTRXPAD21		BA31		
N/A	RXNPAD21		BB30		

Table 13: FF1704 — XC2VP70, XC2VPX70, and XC2VP100

Bank	Pin Description		Pin Number	No Connects	
	Virtex-II Pro Devices	XC2VPX70 (if Different)		XC2VP70, XC2VPX70	XC2VP100
N/A	GND		AB18		
N/A	GND		AB17		
N/A	GND		AB11		
N/A	GND		AB8		
N/A	GND		AB5		
N/A	GND		AC41		
N/A	GND		AC26		
N/A	GND		AC25		
N/A	GND		AC24		
N/A	GND		AC23		
N/A	GND		AC22		
N/A	GND		AC21		
N/A	GND		AC20		
N/A	GND		AC19		
N/A	GND		AC18		
N/A	GND		AC17		
N/A	GND		AC2		
N/A	GND		AD26		
N/A	GND		AD25		
N/A	GND		AD24		
N/A	GND		AD23		
N/A	GND		AD22		
N/A	GND		AD21		
N/A	GND		AD20		
N/A	GND		AD19		
N/A	GND		AD18		
N/A	GND		AD17		
N/A	GND		AE37		
N/A	GND		AE34		
N/A	GND		AE26		
N/A	GND		AE25		
N/A	GND		AE24		
N/A	GND		AE23		
N/A	GND		AE22		
N/A	GND		AE21		
N/A	GND		AE20		

Table 14: FF1696 — XC2VP100

Bank	Pin Description	Pin Number	No Connects
			XC2VP100
2	IO_L22N_2/VREF_2	L4	
2	IO_L22P_2	L5	
2	IO_L23N_2	T8	
2	IO_L23P_2	T9	
2	IO_L24N_2	L3	
2	IO_L24P_2	K3	
2	IO_L25N_2	L1	
2	IO_L25P_2	L2	
2	IO_L26N_2	U12	
2	IO_L26P_2	V12	
2	IO_L27N_2	M7	
2	IO_L27P_2	L6	
2	IO_L28N_2/VREF_2	M5	
2	IO_L28P_2	M6	
2	IO_L29N_2	U10	
2	IO_L29P_2	U11	
2	IO_L30N_2	M3	
2	IO_L30P_2	M4	
2	IO_L31N_2	N6	
2	IO_L31P_2	N7	
2	IO_L32N_2	U7	
2	IO_L32P_2	U8	
2	IO_L33N_2	N3	
2	IO_L33P_2	N4	
2	IO_L34N_2/VREF_2	N2	
2	IO_L34P_2	M2	
2	IO_L35N_2	V10	
2	IO_L35P_2	V11	
2	IO_L36N_2	P6	
2	IO_L36P_2	P7	
2	IO_L37N_2	P1	
2	IO_L37P_2	P2	
2	IO_L38N_2	V8	
2	IO_L38P_2	V9	
2	IO_L39N_2	R6	
2	IO_L39P_2	P5	
2	IO_L40N_2/VREF_2	R4	

Table 14: FF1696 — XC2VP100

Bank	Pin Description	Pin Number	No Connects
			XC2VP100
3	IO_L19N_3	AM3	
3	IO_L19P_3	AN3	
3	IO_L18N_3	AN1	
3	IO_L18P_3	AN2	
3	IO_L17N_3	AG12	
3	IO_L17P_3	AH12	
3	IO_L16N_3	AP6	
3	IO_L16P_3	AP7	
3	IO_L15N_3/VREF_3	AP3	
3	IO_L15P_3	AP4	
3	IO_L14N_3	AH10	
3	IO_L14P_3	AH11	
3	IO_L13N_3	AR6	
3	IO_L13P_3	AR7	
3	IO_L12N_3	AR4	
3	IO_L12P_3	AR5	
3	IO_L11N_3	AH8	
3	IO_L11P_3	AH9	
3	IO_L10N_3	AR2	
3	IO_L10P_3	AR3	
3	IO_L09N_3/VREF_3	AP2	
3	IO_L09P_3	AR1	
3	IO_L08N_3	AJ10	
3	IO_L08P_3	AJ11	
3	IO_L07N_3	AT7	
3	IO_L07P_3	AT8	
3	IO_L72N_3	AT3	
3	IO_L72P_3	AT4	
3	IO_L71N_3	AJ12	
3	IO_L71P_3	AK12	
3	IO_L70N_3	AT1	
3	IO_L70P_3	AT2	
3	IO_L69N_3/VREF_3	AT6	
3	IO_L69P_3	AU6	
3	IO_L68N_3	AK10	
3	IO_L68P_3	AK11	
3	IO_L67N_3	AT5	

Table 14: FF1696 — XC2VP100

Bank	Pin Description	Pin Number	No Connects
			XC2VP100
4	IO_L26P_4	AU12	
4	IO_L27N_4	AR12	
4	IO_L27P_4/VREF_4	AP12	
4	IO_L28N_4	AW13	
4	IO_L28P_4	AW12	
4	IO_L29N_4	BA12	
4	IO_L29P_4	AY12	
4	IO_L30N_4	AN13	
4	IO_L30P_4	AM13	
4	IO_L34N_4	AU13	
4	IO_L34P_4	AT13	
4	IO_L35N_4	BA13	
4	IO_L35P_4	AY13	
4	IO_L36N_4	AM14	
4	IO_L36P_4/VREF_4	AL14	
4	IO_L76N_4	AR15	
4	IO_L76P_4	AT14	
4	IO_L77N_4	AV14	
4	IO_L77P_4	AU14	
4	IO_L78N_4	AP14	
4	IO_L78P_4	AN14	
4	IO_L79N_4	AW15	
4	IO_L79P_4	AY14	
4	IO_L80_4/No_Pair	BB14	
4	IO_L83_4/No_Pair	BA14	
4	IO_L84N_4	AM15	
4	IO_L84P_4	AL15	
4	IO_L85N_4	AT16	
4	IO_L85P_4	AT15	
4	IO_L86N_4	AV15	
4	IO_L86P_4	AU15	
4	IO_L87N_4	AP15	
4	IO_L87P_4/VREF_4	AN15	
4	IO_L37N_4	AY16	
4	IO_L37P_4	AY15	
4	IO_L38N_4	BB15	
4	IO_L38P_4	BA15	

Table 14: FF1696 — XC2VP100

Bank	Pin Description	Pin Number	No Connects
			XC2VP100
N/A	GND	AF1	
N/A	GND	AC1	
N/A	GND	Y1	
N/A	GND	U1	
N/A	GND	N1	
N/A	GND	J1	
N/A	GND	E1	

Notes:

1. See [Table 4](#) for an explanation of the signals available on this pin.