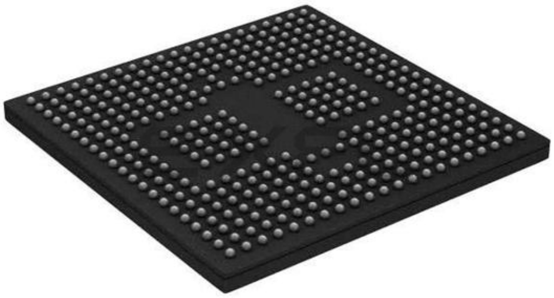
XMOS - XL224-1024-FB374-C40 Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	XCore
Core Size	32-Bit 24-Core
Speed	4000MIPS
Connectivity	-
Peripherals	-
Number of I/O	176
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	1M x 8
Voltage - Supply (Vcc/Vdd)	0.95V ~ 3.6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	374-LFBGA
Supplier Device Package	374-FBGA (18x18)
Purchase URL	https://www.e-xfl.com/product-detail/xmos/xl224-1024-fb374-c40

on events generated by hardware resources such as the I/O pins, communication channels and timers. Once triggered, a core runs independently and concurrently to other cores, until it pauses to wait for more events. Section 6.2

- **Channels and channel ends** Tasks running on logical cores communicate using channels formed between two channel ends. Data can be passed synchronously or asynchronously between the channel ends assigned to the communicating tasks. Section 6.5
- **xCONNECT Switch and Links** Between tiles, channel communications are implemented over a high performance network of xCONNECT Links and routed through a hardware xCONNECT Switch. Section 6.6
- **Ports** The I/O pins are connected to the processing cores by Hardware Response ports. The port logic can drive its pins high and low, or it can sample the value on its pins optionally waiting for a particular condition. Section 6.3
- **Clock blocks** xCORE devices include a set of programmable clock blocks that can be used to govern the rate at which ports execute. Section 6.4
- **Memory** Each xCORE Tile integrates a bank of SRAM for instructions and data, and a block of one-time programmable (OTP) memory that can be configured for system wide security features. Section 9
- **PLL** The PLL is used to create a high-speed processor clock given a low speed external oscillator. Section 7
- **JTAG** The JTAG module can be used for loading programs, boundary scan testing, in-circuit source-level debugging and programming the OTP memory. Section 10

1.1 Software

Devices are programmed using C, C++ or xC (C with multicore extensions). XMOS provides tested and proven software libraries, which allow you to quickly add interface and processor functionality such as USB, Ethernet, PWM, graphics driver, and audio EQ to your applications.

1.2 xTIMEcomposer Studio

The xTIMEcomposer Studio development environment provides all the tools you need to write and debug your programs, profile your application, and write images into flash memory or OTP memory on the device. Because xCORE devices operate deterministically, they can be simulated like hardware within xTIMEcomposer: uniquely in the embedded world, xTIMEcomposer Studio therefore includes a static timing analyzer, cycle-accurate simulator, and high-speed in-circuit instrumentation.

xTIMEcomposer can be driven from either a graphical development environment, or the command line. The tools are supported on Windows, Linux and MacOS X and available at no cost from xmos.com/downloads. Information on using the tools is provided in the xTIMEcomposer User Guide, X3766.

2 XL224-1024-FB374 Features

- **Multicore Microcontroller with Advanced Multi-Core RISC Architecture**
  - 24 real-time logical cores on 4 xCORE tiles
  - Cores share up to 2000 MIPS
    - Up to 4000 MIPS in dual issue mode
  - Each logical core has:
    - Guaranteed throughput of between 1/5 and 1/6 of tile MIPS
    - 16x32bit dedicated registers
  - 167 high-density 16/32-bit instructions
    - All have single clock-cycle execution (except for divide)
    - 32x32→64-bit MAC instructions for DSP, arithmetic and user-definable cryptographic functions
- **Programmable I/O**
  - 256 general-purpose I/O pins, configurable as input or output
    - Up to 56 x 1bit port, 22 x 4bit port, 13 x 8bit port, 6 x 16bit port, 4 x 32bit port
    - 8 xCONNECT links
  - Port sampling rates of up to 60 MHz with respect to an external clock
  - 128 channel ends (32 per tile) for communication with other cores, on or off-chip
- **Memory**
  - 1024KB internal single-cycle SRAM (max 256KB per tile) for code and data storage
  - 32KB internal OTP (max 8KB per tile) for application boot code
- **Hardware resources**
  - 24 clock blocks (6 per tile)
  - 40 timers (10 per tile)
  - 16 locks (4 per tile)
- **JTAG Module for On-Chip Debug**
- **Security Features**
  - Programming lock disables debug and prevents read-back of memory contents
  - AES bootloader ensures secrecy of IP held on external flash memory
- **Ambient Temperature Range**
  - Commercial qualification: 0 °C to 70 °C
  - Industrial qualification: -40 °C to 85 °C
- **Speed Grade**
  - 40: 2000 MIPS
- **Power Consumption**
  - 1140 mA (typical)
- **374-pin FBGA package 0.8 mm pitch**

Signal	Function						Type	Properties
X1D67	$X_0L2^1_{out}$					$32A^{16}$	I/O	IO, PD
X1D68	$X_0L2^2_{out}$					$32A^{17}$	I/O	IO, PD
X1D69	$X_0L2^3_{out}$					$32A^{18}$	I/O	IO, PD
X1D70	$X_0L2^4_{out}$					$32A^{19}$	I/O	IO, PD
X2D00		$1A^0$					I/O	IO, PD
X2D02			$4A^0$	$8A^0$	$16A^0$	$32A^{20}$	I/O	IO, PD
X2D03			$4A^1$	$8A^1$	$16A^1$	$32A^{21}$	I/O	IO, PD
X2D04			$4B^0$	$8A^2$	$16A^2$	$32A^{22}$	I/O	IO, PD
X2D05			$4B^1$	$8A^3$	$16A^3$	$32A^{23}$	I/O	IO, PD
X2D06			$4B^2$	$8A^4$	$16A^4$	$32A^{24}$	I/O	IO, PD
X2D07			$4B^3$	$8A^5$	$16A^5$	$32A^{25}$	I/O	IO, PD
X2D08			$4A^2$	$8A^6$	$16A^6$	$32A^{26}$	I/O	IO, PD
X2D09			$4A^3$	$8A^7$	$16A^7$	$32A^{27}$	I/O	IO, PD
X2D11		$1D^0$					I/O	IO, PD
X2D12		$1E^0$					I/O	IO, PD
X2D13		$1F^0$					I/O	IO, PD
X2D14			$4C^0$	$8B^0$	$16A^8$	$32A^{28}$	I/O	IO, PD
X2D15			$4C^1$	$8B^1$	$16A^9$	$32A^{29}$	I/O	IO, PD
X2D16	$X_2L4^4_{in}$		$4D^0$	$8B^2$	$16A^{10}$		I/O	IO, PD
X2D17	$X_2L4^3_{in}$		$4D^1$	$8B^3$	$16A^{11}$		I/O	IO, PD
X2D18	$X_2L4^2_{in}$		$4D^2$	$8B^4$	$16A^{12}$		I/O	IO, PD
X2D19	$X_2L4^1_{in}$		$4D^3$	$8B^5$	$16A^{13}$		I/O	IO, PD
X2D20			$4C^2$	$8B^6$	$16A^{14}$	$32A^{30}$	I/O	IO, PD
X2D21			$4C^3$	$8B^7$	$16A^{15}$	$32A^{31}$	I/O	IO, PD
X2D22		$1G^0$					I/O	IO, PD
X2D23		$1H^0$					I/O	IO, PD
X2D24	$X_2L7^0_{in}$	$1I^0$					I/O	IO, PD
X2D25	$X_2L7^0_{out}$	$1J^0$					I/O	IO, PD
X2D26	$X_2L7^3_{out}$		$4E^0$	$8C^0$	$16B^0$		I/O	IO, PD
X2D27	$X_2L7^4_{out}$		$4E^1$	$8C^1$	$16B^1$		I/O	IO, PD
X2D28			$4F^0$	$8C^2$	$16B^2$		I/O	IO, PD
X2D29			$4F^1$	$8C^3$	$16B^3$		I/O	IO, PD
X2D30			$4F^2$	$8C^4$	$16B^4$		I/O	IO, PD
X2D31			$4F^3$	$8C^5$	$16B^5$		I/O	IO, PD
X2D32			$4E^2$	$8C^6$	$16B^6$		I/O	IO, PD
X2D33			$4E^3$	$8C^7$	$16B^7$		I/O	IO, PD
X2D34	$X_2L7^1_{out}$	$1K^0$					I/O	IO, PD
X2D35	$X_2L7^2_{out}$	$1L^0$					I/O	IO, PD
X2D36		$1M^0$		$8D^0$	$16B^8$		I/O	IO, PD
X2D49	$X_2L5^4_{in}$					$32A^0$	I/O	IO, PD
X2D50	$X_2L5^3_{in}$					$32A^1$	I/O	IO, PD
X2D51	$X_2L5^2_{in}$					$32A^2$	I/O	IO, PD
X2D52	$X_2L5^1_{in}$					$32A^3$	I/O	IO, PD

(continued)

Signal	Function	Type	Properties
X3D32	$4E^2$ $8C^6$ $16B^6$	I/O	IOT, PD
X3D33	$4E^3$ $8C^7$ $16B^7$	I/O	IOT, PD
X3D40	$8D^4$ $16B^{12}$	I/O	IOT, PD
X3D41	$8D^5$ $16B^{13}$	I/O	IOT, PD
X3D42	$8D^6$ $16B^{14}$	I/O	IOT, PD
X3D43	$8D^7$ $16B^{15}$	I/O	IOT, PD

System pins (4)

Signal	Function	Type	Properties
CLK	PLL reference clock	Input	IO, PD, ST
DEBUG_N	Multi-chip debug	I/O	IO, PU
MODE0	Boot mode select	Input	PU
MODE1	Boot mode select	Input	PU

usb pins (10)

Signal	Function	Type	Properties
USB_2_DM	USB Serial Data Inverted, node 2	I/O	
USB_2_DP	USB Serial Data, node 2	I/O	
USB_2_ID	USB Device ID (OTG) - Reserved, node 2	I/O	
USB_2_RTUNE	USB resistor, node 2	I/O	
USB_2_VBUS	USB Power Detect Pin, node 2	I/O	
USB_DM	USB Serial Data Inverted	I/O	
USB_DP	USB Serial Data	I/O	
USB_ID	USB Device ID (OTG) - Reserved	I/O	
USB_RTUNE	USB resistor	I/O	
USB_VBUS	USB Power Detect Pin	I/O	

6 Product Overview

The XL224-1024-FB374 is a powerful device that consists of four xCORE Tiles, each comprising a flexible logical processing cores with tightly integrated I/O and on-chip memory.

6.1 Logical cores

Each tile has 6 active logical cores, which issue instructions down a shared five-stage pipeline. Instructions from the active cores are issued round-robin. If up to five logical cores are active, each core is allocated a fifth of the processing cycles. If more than five logical cores are active, each core is allocated at least $1/n$ cycles (for n cores). Figure 3 shows the guaranteed core performance depending on the number of cores used.

Figure 3: Logical core performance

Speed grade	MIPS	Frequency	Minimum MIPS per core (for *n* cores)							
			1	2	3	4	5	6		
20	2000 MIPS	500 MHz	100	100	100	100	100	83		

There is no way that the performance of a logical core can be reduced below these predicted levels (unless *priority threads* are used: in this case the guaranteed minimum performance is computed based on the number of priority threads as defined in the architecture manual). Because cores may be delayed on I/O, however, their unused processing cycles can be taken by other cores. This means that for more than five logical cores, the performance of each core is often higher than the predicted minimum but cannot be guaranteed.

The logical cores are triggered by events instead of interrupts and run to completion. A logical core can be paused to wait for an event.

6.2 xTIME scheduler

The xTIME scheduler handles the events generated by xCORE Tile resources, such as channel ends, timers and I/O pins. It ensures that all events are serviced and synchronized, without the need for an RTOS. Events that occur at the I/O pins are handled by the Hardware-Response ports and fed directly to the appropriate xCORE Tile. An xCORE Tile can also choose to wait for a specified time to elapse, or for data to become available on a channel.

Tasks do not need to be prioritised as each of them runs on their own logical xCORE. It is possible to share a set of low priority tasks on a single core using cooperative multitasking.

6.3 Hardware Response Ports

Hardware Response ports connect an xCORE tile to one or more physical pins and as such define the interface between hardware attached to the XL224-1024-FB374, and the software running on it. A combination of 1bit, 4bit, 8bit, 16bit and 32bit

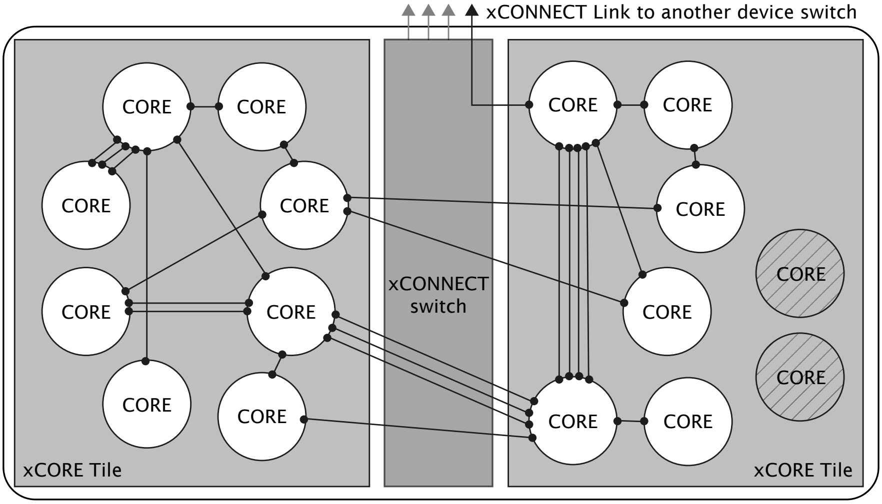


Figure 6: Switch, links and channel ends

and packet switched data can both be supported efficiently. Streams provide the fastest possible data rates between xCORE Tiles (up to 250 MBit/s), but each stream requires a single link to be reserved between switches on two tiles. All packet communications can be multiplexed onto a single link.

Information on the supported routing topologies that can be used to connect multiple devices together can be found in the XS1-L Link Performance and Design Guide, X2999.

7 PLL

The PLL creates a high-speed clock that is used for the switch, tile, and reference clock. The PLL multiplication value is selected through the two MODE pins, and can be changed by software to speed up the tile or use less power. The MODE pins are set as shown in Figure 7:

Figure 7: PLL multiplier values and MODE pins

Oscillator Frequency	MODE 1	MODE 0	Tile Frequency	PLL Ratio	PLL settings *OD*	PLL settings *F*	PLL settings *R*
3.25-10 MHz	0	0	130-400 MHz	40	1	159	0
9-25 MHz	1	1	144-400 MHz	16	1	63	0
25-50 MHz	1	0	167-400 MHz	8	1	31	0
50-100 MHz	0	1	196-400 MHz	4	1	15	0

Figure 7 also lists the values of OD, F and R, which are the registers that define the ratio of the tile frequency to the oscillator frequency:

$$F_{core} = F_{osc} \times \frac{F+1}{2} \times \frac{1}{R+1} \times \frac{1}{OD+1}$$

OD, F and R must be chosen so that $0 \leq R \leq 63$, $0 \leq F \leq 4095$, $0 \leq OD \leq 7$, and $260MHz \leq F_{osc} \times \frac{F+1}{2} \times \frac{1}{R+1} \leq 1.3GHz$. The OD, F, and R values can be modified by writing to the digital node PLL configuration register.

The MODE pins must be held at a static value during and after deassertion of the system reset.

If a different tile frequency is required (eg, 500 MHz), then the PLL must be reprogrammed after boot to provide the required tile frequency. The XMOS tools perform this operation by default. Further details on configuring the clock can be found in the xCORE-200 Clock Frequency Control document.

8 Boot Procedure

The device is kept in reset by driving RST_N low. When in reset, all GPIO pins have a pull-down enabled. When the device is taken out of reset by releasing RST_N the processor starts its internal reset process. After 15-150 μs (depending on the input clock) the processor boots.

Pin X2D06 must be pulled high with an external pull-up whilst the chip comes out of reset, to ensure that tile 2 will boot from link. X2D04, X2D05, and X2D07 should be kept low whilst the chip comes out of reset.

The xCORE Tile boot procedure is illustrated in Figure 8. If bit 5 of the security register (*see* §9.1) is set, the device boots from OTP. To get a high value, a 3K3 pull-up resistor should be strapped onto the pin. To assure a low value, a pull-down resistor is required if other external devices are connected to this port.

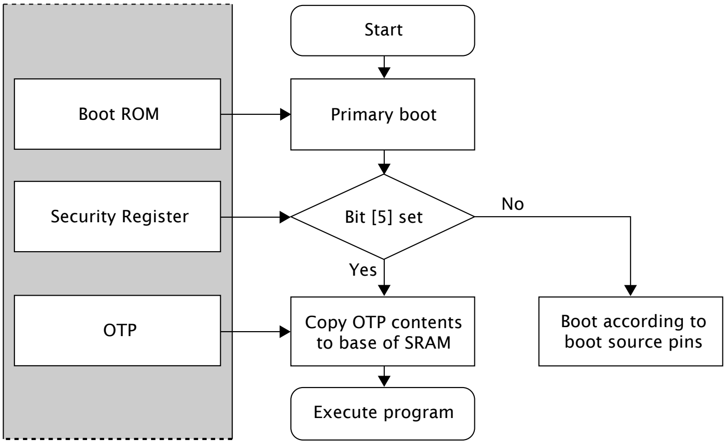


Figure 8: Boot procedure

The boot-rom on the core will then:

1. Allocate channel-end 0.
2. Input a word on channel-end 0. It will use this word as a channel to acknowledge the boot. Provide the null-channel-end 0x0000FF02 if no acknowledgment is required.
3. Input the boot image specified above, including the CRC.
4. Input an END control token.
5. Output an END control token to the channel-end received in step 2.
6. Free channel-end 0.
7. Jump to the loaded code.

8.5 Boot from OTP

If an xCORE tile is set to use secure boot (see Figure 8), the boot image is read from address 0 of the OTP memory in the tile's security module.

This feature can be used to implement a secure bootloader which loads an encrypted image from external flash, decrypts and CRC checks it with the processor, and discontinues the boot process if the decryption or CRC check fails. XMOS provides a default secure bootloader that can be written to the OTP along with secret decryption keys.

Each tile has its own individual OTP memory, and hence some tiles can be booted from OTP while others are booted from SPI or the channel interface. This enables systems to be partially programmed, dedicating one or more tiles to perform a particular function, leaving the other tiles user-programmable.

8.6 Security register

The security register enables security features on the xCORE tile. The features shown in Figure 13 provide a strong level of protection and are sufficient for providing strong IP security.

9 Memory

9.1 OTP

Each xCORE Tile integrates 8 KB one-time programmable (OTP) memory along with a security register that configures system wide security features. The OTP holds data in four sectors each containing 512 rows of 32 bits which can be used to implement secure bootloaders and store encryption keys. Data for the security register is loaded from the OTP on power up. All additional data in OTP is copied from the OTP to SRAM and executed first on the processor.

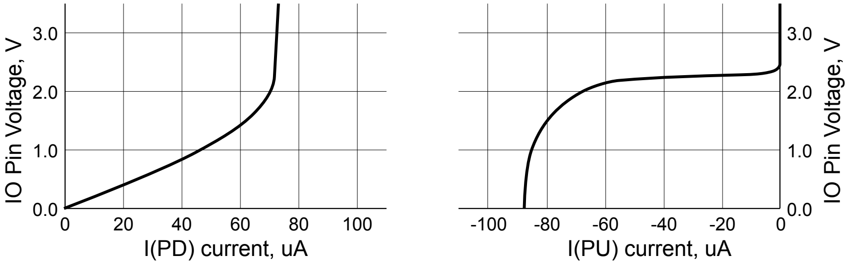


Figure 19: Typical internal pull-down and pull-up currents

12.3 ESD Stress Voltage

Figure 20: ESD stress voltage

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
HBM	Human body model	-2.00		2.00	KV	
CDM	Charged Device Model	-500		500	V	

12.4 Reset Timing

Figure 21: Reset timing

Symbol	Parameters	MIN	TYP	MAX	UNITS	Notes
T(RST)	Reset pulse width	5			µs	
T(INIT)	Initialization time			150	µs	A

A Shows the time taken to start booting after RST_N has gone high.

12.5 Power Consumption

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
I(DDCQ)	Quiescent VDD current		90		mA	A, B, C
PD	Tile power dissipation		325		µW/MIPS	A, D, E, F
IDD	Active VDD current		1140	1400	mA	A, G
I(ADDPLL)	PLL_AVDD current		5	7	mA	H
I(VDD33)	VDD33 current		53.4		mA	I
I(USB_VDD)	USB_VDD current		16.6		mA	J

Figure 22: xCORE Tile currents

A Use for budgetary purposes only.
B Assumes typical tile and I/O voltages with no switching activity.
C Includes PLL current.
D Assumes typical tile and I/O voltages with nominal switching activity.
E Assumes 1 MHz = 1 MIPS.
F PD(TYP) value is the usage power consumption under typical operating conditions.
G Measurement conditions: VDD = 1.0 V, VDDIO = 3.3 V, 25 °C, 500 MHz, average device resource usage.
H PLL_AVDD = 1.0 V
I HS mode transmitting while driving all 0’s data (constant JKJK on DP/DM). Loading of 10 pF. Transfers do not include any interpacket delay.
J HS receive mode; no traffic.

⚠ The tile power consumption of the device is highly application dependent and should be used for budgetary purposes only.

More detailed power analysis can be found in the XS1-L Power Consumption document,

12.6 Clock

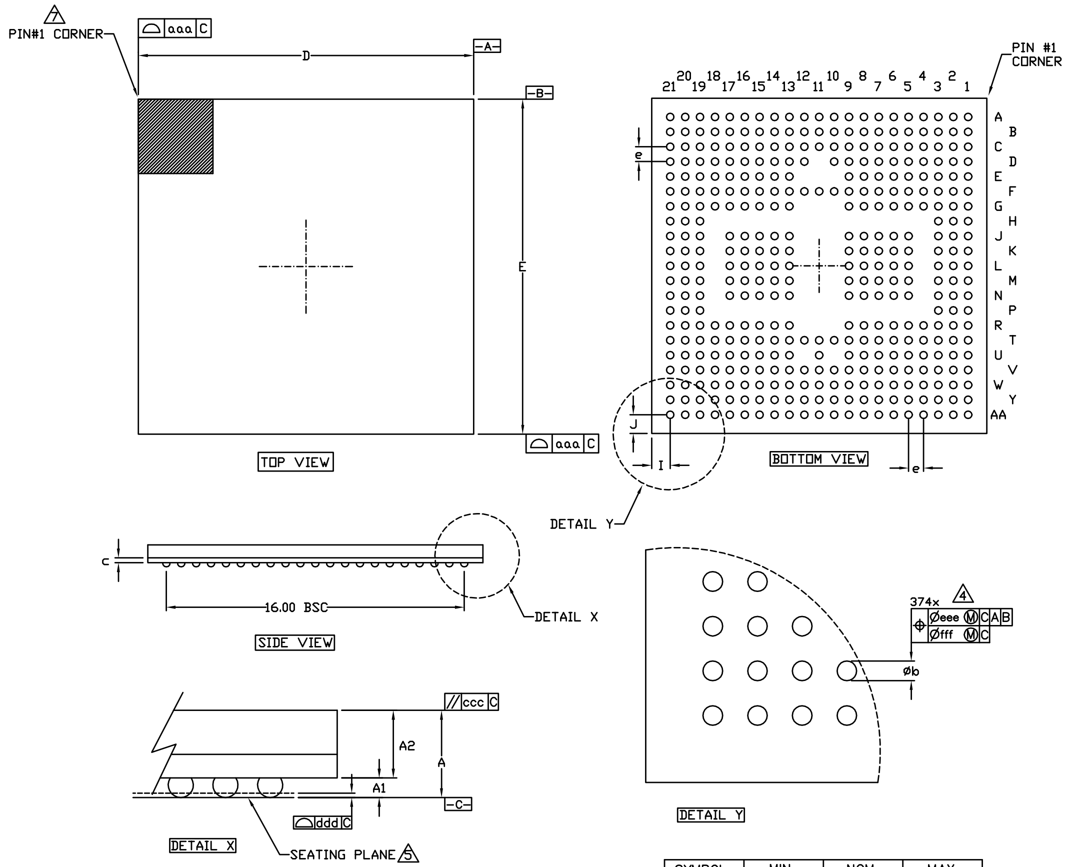
Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
f	Frequency	3.25	24	100	MHz	
SR	Slew rate	0.10			V/ns	
TJ(LT)	Long term jitter (pk-pk)			2	%	A
f(MAX)	Processor clock frequency			500	MHz	B

Figure 23: Clock

A Percentage of CLK period.
B Assumes typical tile and I/O voltages with nominal activity.

Further details can be found in the XS1-L Clock Frequency Control document,

13 Package Information



NOTE:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. "e" REPRESENTS THE BASIC SOLDER BALL GRID PITCH.
3. "M" REPRESENTS THE MAXIMUM SOLDER BALL MATRIX SIZE.
4. DIMENSIONS "b" IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER PARALLEL TO PRIMARY DATUM -C- .
5. PRIMARY DATUM -C- AND SEATING PLANE ARE DESIGNED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
6. DIMENSIONING AND TOLERANCING PER ASME Y14.5M–1994
7. A1 CORNER MUST BE IDENTIFIED BY LASER MARK.

SYMBOL	MIN.	NOM.	MAX.
A	1.16	1.26	1.36
A1	0.25	0.30	0.35
A2	0.91	0.96	1.01
D	17.90	18.00	18.10
E	17.90	18.00	18.10
I	1.00 REF.		
J	1.00 REF.		
M	21x21<DEPOPULATED>		
aaa			0.15
ccc			0.20
ddd			0.10
eee			0.15
fff			0.08
b	0.35	0.40	0.45
e	0.80 BSC.		
c	0.26 REF.		

A.2 Accessing an xCORE Tile configuration register

xCORE Tile configuration registers can be accessed through the interconnect using the functions `write_tile_config_reg(tileref, ...)` and `read_tile_config_reg(tileref, ...)`, where `tileref` is the name of the xCORE Tile, e.g. `tile[1]`. These functions implement the protocols described below.

Instead of using the functions above, a channel-end can be allocated to communicate with the xCORE tile configuration registers. The destination of the channel-end should be set to `0xnnnnC20C` where `nnnnnn` is the tile-identifier.

A write message comprises the following:

control-token 192	24-bit response channel-end identifier	16-bit register number	32-bit data	control-token 1

The response to a write message comprises either control tokens 3 and 1 (for success), or control tokens 4 and 1 (for failure).

A read message comprises the following:

control-token 193	24-bit response channel-end identifier	16-bit register number	control-token 1

The response to the read message comprises either control token 3, 32-bit of data, and control-token 1 (for success), or control tokens 4 and 1 (for failure).

A.3 Accessing node configuration

Node configuration registers can be accessed through the interconnect using the functions `write_node_config_reg(device, ...)` and `read_node_config_reg(device, ...)`, where `device` is the name of the node. These functions implement the protocols described below.

Instead of using the functions above, a channel-end can be allocated to communicate with the node configuration registers. The destination of the channel-end should be set to `0xnnnnC30C` where `nnnn` is the node-identifier.

A write message comprises the following:

control-token 192	24-bit response channel-end identifier	16-bit register number	32-bit data	control-token 1

The response to a write message comprises either control tokens 3 and 1 (for success), or control tokens 4 and 1 (for failure).

A read message comprises the following:

control-token 193	24-bit response channel-end identifier	16-bit register number	control-token 1

The response to a read message comprises either control token 3, 32-bit of data, and control-token 1 (for success), or control tokens 4 and 1 (for failure).

B.1 RAM base address: 0x00

This register contains the base address of the RAM. It is initialized to 0x00040000.

0x00: RAM base address

Bits	Perm	Init	Description
31:2	RW		Most significant 16 bits of all addresses.
1:0	RO	-	Reserved

B.2 Vector base address: 0x01

Base address of event vectors in each resource. On an interrupt or event, the 16 most significant bits of the destination address are provided by this register; the least significant 16 bits come from the event vector.

0x01: Vector base address

Bits	Perm	Init	Description
31:18	RW		The event and interrupt vectors.
17:0	RO	-	Reserved

B.3 xCORE Tile control: 0x02

Register to control features in the xCORE tile

Bits	Perm	Init	Description
31:26	RO	-	Reserved
25:18	RW	0	RGMII TX data delay value (in PLL output cycle increments)
17:9	RW	0	RGMII TX clock divider value. TX clk rises when counter (clocked by PLL output) reaches this value and falls when counter reaches (value»1). Value programmed into this field should be actual divide value required minus 1
8	RW	0	Enable RGMII interface periph ports
7:6	RO	-	Reserved
5	RW	0	Select the dynamic mode (1) for the clock divider when the clock divider is enabled. In dynamic mode the clock divider is only activated when all active threads are paused. In static mode the clock divider is always enabled.
4	RW	0	Enable the clock divider. This divides the output of the PLL to facilitate one of the low power modes.
3:0	RO	-	Reserved

0x02: xCORE Tile control

0x09: Ring Oscillator Value

Bits	Perm	Init	Description
31:16	RO	-	Reserved
15:0	RO	0	Ring oscillator Counter data.

B.10 Ring Oscillator Value: 0x0A

This register contains the current count of the Peripheral Wire ring oscillator. This value is not reset on a system reset.

0x0A: Ring Oscillator Value

Bits	Perm	Init	Description
31:16	RO	-	Reserved
15:0	RO	0	Ring oscillator Counter data.

B.11 RAM size: 0x0C

The size of the RAM in bytes

0x0C: RAM size

Bits	Perm	Init	Description
31:2	RO		Most significant 16 bits of all addresses.
1:0	RO	-	Reserved

B.12 Debug SSR: 0x10

This register contains the value of the SSR register when the debugger was called.

0x00: Device identification

Bits	Perm	Init	Description
31:24	CRO		Processor ID of this XCore.
23:16	CRO		Number of the node in which this XCore is located.
15:8	CRO		XCore revision.
7:0	CRO		XCore version.

C.2 xCORE Tile description 1: 0x01

This register describes the number of logical cores, synchronisers, locks and channel ends available on this xCORE tile.

0x01: xCORE Tile description 1

Bits	Perm	Init	Description
31:24	CRO		Number of channel ends.
23:16	CRO		Number of the locks.
15:8	CRO		Number of synchronisers.
7:0	RO	-	Reserved

C.3 xCORE Tile description 2: 0x02

This register describes the number of timers and clock blocks available on this xCORE tile.

0x02: xCORE Tile description 2

Bits	Perm	Init	Description
31:16	RO	-	Reserved
15:8	CRO		Number of clock blocks.
7:0	CRO		Number of timers.

C.4 Control PSwitch permissions to debug registers: 0x04

This register can be used to control whether the debug registers (marked with permission CRW) are accessible through the tile configuration registers. When this bit is set, write -access to those registers is disabled, preventing debugging of the xCORE tile over the interconnect.

0x07: Security configuration

Bits	Perm	Init	Description
31	CRO		Disables write permission on this register
30:15	RO	-	Reserved
14	CRO		Disable access to XCore's global debug
13	RO	-	Reserved
12	CRO		lock all OTP sectors
11:8	CRO		lock bit for each OTP sector
7	CRO		Enable OTP reduanacy
6	RO	-	Reserved
5	CRO		Override boot mode and read boot image from OTP
4	CRO		Disable JTAG access to the PLL/BOOT configuration registers
3:1	RO	-	Reserved
0	CRO		Disable access to XCore's JTAG debug TAP

C.8 Debug scratch: 0x20 .. 0x27

A set of registers used by the debug ROM to communicate with an external debugger, for example over the switch. This is the same set of registers as the Debug Scratch registers in the processor status.

0x20 .. 0x27: Debug scratch

Bits	Perm	Init	Description
31:0	CRW		Value.

C.9 PC of logical core 0: 0x40

Value of the PC of logical core 0.

0x40: PC of logical core 0

Bits	Perm	Init	Description
31:0	CRO		Value.

C.10 PC of logical core 1: 0x41

Value of the PC of logical core 1.

D.18 Static link configuration: 0xA0 .. 0xA7

These registers are used for static (ie, non-routed) links. When a link is made static, all traffic is forwarded to the designated channel end and no routing is attempted. The registers control links C, D, A, B, G, H, E, and F in that order.

0xA0 .. 0xA7: Static link configuration

Bits	Perm	Init	Description
31	RW	0	Enable static forwarding.
30:9	RO	-	Reserved
8	RW	0	The destination processor on this node that packets received in static mode are forwarded to.
7:5	RO	-	Reserved
4:0	RW	0	The destination channel end on this node that packets received in static mode are forwarded to.

F Schematics Design Check List

☑ This section is a checklist for use by schematics designers using the XL224-1024-FB374. Each of the following sections contains items to check for each design.

F.1 Power supplies

☐ VDDIO and OTP_VCC supply is within specification before the VDD (core) supply is turned on. Specifically, the VDDIO and OTP_VCC supply is within specification before VDD (core) reaches 0.4V (Section 11).

☐ The VDD (core) supply ramps monotonically (rises constantly) from 0V to its final value (0.95V - 1.05V) within 10ms (Section 11).

☐ The VDD (core) supply is capable of supplying 1400 mA (Section 11 and Figure 18).

☐ PLL_AVDD is filtered with a low pass filter, for example an RC filter, see Section 11 .

F.2 Power supply decoupling

☐ The design has multiple decoupling capacitors per supply, for example at least four0402 or 0603 size surface mount capacitors of 100nF in value, per supply (Section 11).

☐ A bulk decoupling capacitor of at least 10uF is placed on each supply (Section 11).

F.3 Power on reset

☐ The RST_N and TRST_N pins are asserted (low) during or after power up. The device is not used until these resets have taken place.

F.4 Clock

☐ The CLK input pin is supplied with a clock with monotonic rising edges and low jitter.

☐ Pins MODE0 and MODE1 are set to the correct value for the chosen oscillator frequency. The MODE settings are shown in the Oscillator section, Section 7. If you have a choice between two values, choose the value with the highest multiplier ratio since that will boot faster.

F.5 Boot

☐ The device is connected to a QSPI flash for booting, connected to X0D01, X0D04..X0D07, and X0D10 (Section 8). If not, you must boot the device through OTP or JTAG, or set it to boot from SPI and connect a SPI flash.

☐ The Flash that you have chosen is supported by **xflash**, or you have created a specification file for it.

F.6 JTAG, XScope, and debugging

☐ You have decided as to whether you need an XSYS header or not (Section E)

☐ If you have not included an XSYS header, you have devised a method to program the SPI-flash or OTP (Section E).

F.7 GPIO

☐ You have not mapped both inputs and outputs to the same multi-bit port.

☐ Pins X0D04, X0D05, X0D06, and X0D07 are output only and are, during and after reset, pulled high and low appropriately (Section 8)

☐ Pins X2D04, X2D05, X2D06 and X2D07 are output only and during and after reset, X2D06 is pulled high and X2D04, X2D05, and X2D07 are pulled low (Section 8)

F.8 Multi device designs

Skip this section if your design only includes a single XMOS device.

☐ One device is connected to a QSPI or SPI flash for booting.

☐ Devices that boot from link have, for example, X0D06 pulled high and have link XL0 connected to a device to boot from (Section 8).

H Associated Design Documentation

Document Title	Information	Document Number
Estimating Power Consumption For XS1-L Devices	Power consumption	X4271
Programming XC on XMOS Devices	Timers, ports, clocks, cores and channels	X9577
xTIMEcomposer User Guide	Compilers, assembler and linker/mapper Timing analyzer, xScope, debugger Flash and OTP programming utilities	X3766

I Related Documentation

Document Title	Information	Document Number
The XMOS XS1 Architecture	ISA manual	X7879
XS1 Port I/O Timing	Port timings	X5821
xCONNECT Architecture	Link, switch and system information	X4249
XS1-L Link Performance and Design Guidelines	Link timings	X2999
XS1-L Clock Frequency Control	Advanced clock control	X1433
XS1-L Active Power Conservation	Low-power mode during idle	X7411