



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

Product Status	Active
Core Processor	XCore
Core Size	32-Bit 6-Core
Speed	400MIPS
Connectivity	Configurable
Peripherals	-
Number of I/O	64
Program Memory Size	64KB (16K x 32)
Program Memory Type	SRAM
EEPROM Size	•
RAM Size	-
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	128-TQFP Exposed Pad
Supplier Device Package	128-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/xmos/xs1-l6a-64-tq128-c4

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

### **Table of Contents**

1	xCORE Multicore Microcontrollers
2	XS1-L6A-64-TQ128 Features
3	Pin Configuration
4	Signal Description
5	Product Overview
6	PLL
7	Boot Procedure
8	Memory
9	JTAG
10	Board Integration
11	DC and Switching Characteristics
12	Package Information
13	Ordering Information
App	endices
Α	Configuration of the XS1
В	Processor Status Configuration
С	Tile Configuration
D	Node Configuration
E	XMOS USB Interface
F	Device Errata
G	JTAG, xSCOPE and Debugging
Н	Schematics Design Check List
1	PCB Layout Design Check List
J	Associated Design Documentation
K	Related Documentation
L	Revision History

#### TO OUR VALUED CUSTOMERS

It is our intention to provide you with accurate and comprehensive documentation for the hardware and software components used in this product. To subscribe to receive updates, visit <a href="http://www.xmos.com/">http://www.xmos.com/</a>.

XMOS Ltd. is the owner or licensee of the information in this document and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. XMOS Ltd. makes no representation that the information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.

XMOS and the XMOS logo are registered trademarks of XMOS Ltd in the United Kingdom and other countries, and may not be used without written permission. Company and product names mentioned in this document are the trademarks or registered trademarks of their respective owners.



Figure 5: Switch, links and channel ends

> 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.

# 6 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 6:

	Oscillator	MODE		Tile	PLL Ratio	PLL	setting	gs
	Frequency	1	0	Frequency		OD	F	R
Figure 6:	5-13 MHz	0	0	130-399.75 MHz	30.75	1	122	0
PLL multiplier	13-20 MHz	1	1	260-400.00 MHz	20	2	119	0
values and	20-48 MHz	1	0	167-400.00 MHz	8.33	2	49	0
MODE pins	48-100 MHz	0	1	196-400.00 MHz	4	2	23	0

Figure 6 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:

-XMOS°

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

12

- A 32-bit program size *s* in words.
- Program consisting of  $s \times 4$  bytes.
- A 32-bit CRC, or the value 0x0D15AB1E to indicate that no CRC check should be performed.

The program size and CRC are stored least significant byte first. The program is loaded into the lowest memory address of RAM, and the program is started from that address. The CRC is calculated over the byte stream represented by the program size and the program itself. The polynomial used is 0xEDB88320 (IEEE 802.3); the CRC register is initialized with 0xFFFFFFFF and the residue is inverted to produce the CRC.

#### 7.1 Boot from SPI master

If set to boot from SPI master, the processor enables the four pins specified in Figure 9, and drives the SPI clock at 2.5 MHz (assuming a 400 MHz core clock). A READ command is issued with a 24-bit address 0x000000. The clock polarity and phase are 0 / 0.

Figure 9: SPI master pins

PinSignalDescriptionX0D00MISOMaster In Slave Out (Data)9:X0D01SSSlave Select2rX0D10SCLKClockSX0D11MOSIMaster Out Slave In (Data)

The xCORE Tile expects each byte to be transferred with the *least-significant bit first*. Programmers who write bytes into an SPI interface using the most significant bit first may have to reverse the bits in each byte of the image stored in the SPI device.

If a large boot image is to be read in, it is faster to first load a small boot-loader that reads the large image using a faster SPI clock, for example 50 MHz or as fast as the flash device supports.

The pins used for SPI boot are hardcoded in the boot ROM and cannot be changed. If required, an SPI boot program can be burned into OTP that uses different pins.

#### 7.2 Boot from xConnect Link

If set to boot from an xConnect Link, the processor enables Link B around 200 ns after the boot process starts. Enabling the Link switches off the pull-down on resistors X0D16..X0D19, drives X0D16 and X0D17 low (the initial state for the Link), and monitors pins X0D18 and X0D19 for boot-traffic. X0D18 and X0D19 must be low at this stage. If the internal pull-down is too weak to drain any residual charge, external pull-downs of 10K may be required on those pins.

-XM()S

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.

#### 7.3 Boot from OTP

If an xCORE tile is set to use secure boot (see Figure 7), 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.

#### 7.4 Security register

The security register enables security features on the xCORE tile. The features shown in Figure 10 provide a strong level of protection and are sufficient for providing strong IP security.

### 8 Memory

#### 8.1 OTP

The 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.

The OTP memory is programmed using three special I/O ports: the OTP address port is a 16-bit port with resource ID 0x100200, the OTP data is written via a 32-bit





*IPC/JEDEC Standard For Moisture/Reflow Sensitivity Classification For Nonhermetic Solid State Surface-Mount Devices J-STD-020* Revision D.

-XMOS

# 11 DC and Switching Characteristics

Symbol	Parameter	MIN	ТҮР	MAX	UNITS	Notes
VDD	Tile DC supply voltage	0.95	1.00	1.05	V	
VDDIO	I/O supply voltage	3.00	3.30	3.60	V	
PLL_AVDD	PLL analog supply	0.95	1.00	1.05	V	
OTP_VCC	OTP supply voltage	3.00	3.30	3.60	V	
OTP_VPP	OTP external programming voltage (optional program only)	6.18	6.50	6.83	V	
Cl	xCORE Tile I/O load capacitance			25	pF	
Та	Ambient operating temperature (Commercial)	0		70	°C	
	Ambient operating temperature (Industrial)	-40		85	°C	
Tj	Junction temperature			125	°C	
Tstg	Storage temperature	-65		150	°C	

### 11.1 Operating Conditions

Figure 16: Operating conditions

### 11.2 DC Characteristics

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
V(IH)	Input high voltage	2.00		3.60	V	A
V(IL)	Input low voltage	-0.30		0.70	V	A
V(OH)	Output high voltage	2.00			V	B, C
V(OL)	Output low voltage			0.60	V	B, C
R(PU)	Pull-up resistance		35K		Ω	D
R(PD)	Pull-down resistance		35K		Ω	D

Figure 17: DC characteristics

A All pins except power supply pins.

B Ports 1A, 1D, 1E, 1H, 1I, 1J, 1K and 1L are nominal 8 mA drivers, the remainder of the general-purpose I/Os are 4 mA.

C Measured with 4 mA drivers sourcing 4 mA, 8 mA drivers sourcing 8 mA.

D Used to guarantee logic state for an I/O when high impedance. The internal pull-ups/pull-downs should not be used to pull external circuitry.

### 11.3 ESD Stress Voltage

Figure 18 ESD stress voltage

8:	Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
SS	HBM	Human body model	-2.00		2.00	KV	
ge	MM	Machine model	-200		200	V	

-XMOS

#### 11.4 Reset Timing

Figure 19: Reset timing

Symbol	Parameters	MIN	ТҮР	MAX	UNITS	Notes
T(RST)	Reset pulse width	5			us	
T(INIT)	Initialization time			150	μs	А

A Shows the time taken to start booting after RST\_N has gone high.

#### 11.5 Power Consumption

Symbol	Parameter	MIN	ТҮР	MAX	UNITS	Notes
I(DDCQ)	Quiescent VDD current		14		mA	A, B, C
PD	Tile power dissipation		450		µW/MIPS	A, D, E, F
IDD	Active VDD current (Speed Grade 4)		160	300	mA	A, G
	Active VDD current (Speed Grade 5)		200	375	mA	А, Н
I(ADDPLL)	PLL_AVDD current			7	mA	I

Figure 20: 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.

-XM()S

G Measurement conditions: VDD = 1.0 V, VDDIO = 3.3 V, 25 °C, 400 MHz, average device resource usage.

H Measurement conditions: VDD = 1.0 V, VDDIO = 3.3 V, 25 °C, 500 MHz, average device resource usage.

I PLL\_AVDD = 1.0 V



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, X2999.

# Appendices

# A Configuration of the XS1

The device is configured through three banks of registers, as shown in Figure 27.



Figure 27: Registers

> The following communication sequences specify how to access those registers. Any messages transmitted contain the most significant 24 bits of the channel-end to which a response is to be sent. This comprises the node-identifier and the channel number within the node. if no response is required on a write operation, supply 24-bits with the last 8-bits set, which suppresses the reply message. Any multi-byte data is sent most significant byte first.

#### A.1 Accessing a processor status register

The processor status registers are accessed directly from the processor instruction set. The instructions GETPS and SETPS read and write a word. The register number should be translated into a processor-status resource identifier by shifting the register number left 8 places, and ORing it with 0x0C. Alternatively, the functions getps(reg) and setps(reg,value) can be used from XC.

### 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(tile  $\rightarrow$  ref, ...), 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 nnnnn is the tile-identifier.

A write message comprises the following:

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

-XM()S

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



 $-X \wedge ()S$ 

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).

29

#### B.1 RAM base address: 0x00

This register contains the base address of the RAM. It is initialized to 0x00010000.

**0x00**: RAM base address

<u>)0</u> .	Bits	Perm	Init	Description
se	31:2	RW		Most significant 16 bits of all addresses.
SS	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
•	31:16	RW		The most significant bits for all event and interrupt vectors.
	15:0	RO	-	Reserved

### B.3 xCORE Tile control: 0x02

Register to control features in the xCORE tile

Bits	Perm	Init	Description
31:6	RO	-	Reserved
5	RW	0	Set to 1 to select the dynamic mode for the clock divider when the clock divider is enabled. In dynamic mode the clock divider is only activated when all active logical cores are paused. In static mode the clock divider is always enabled.
4	RW	0	Set to 1 to enable the clock divider. This slows down the xCORE tile clock in order to use less power.
3:0	RO	-	Reserved

0x02: xCORE Tile control

### B.4 xCORE Tile boot status: 0x03

This read-only register describes the boot status of the xCORE tile.

-XMOS-

Bits	Perm	Init	Description
31:18	RO	-	Reserved
17:16	DRW		If the debug interrupt was caused by a hardware breakpoint or hardware watchpoint, this field contains the number of the breakpoint or watchpoint. If multiple breakpoints or watch- points trigger at once, the lowest number is taken.
15:8	DRW		If the debug interrupt was caused by a logical core, this field contains the number of that core. Otherwise this field is 0.
7:3	RO	-	Reserved
2:0	DRW	0	Indicates the cause of the debug interrupt 1: Host initiated a debug interrupt through JTAG 2: Program executed a DCALL instruction 3: Instruction breakpoint 4: Data watch point 5: Resource watch point

**0x15:** Debug interrupt type

### B.17 Debug interrupt data: 0x16

On a data watchpoint, this register contains the effective address of the memory operation that triggered the debugger. On a resource watchpoint, it countains the resource identifier.

0x16 Debug interrupt data

ox16: ebug	Bits	Perm	Init	Description
data	31:0	DRW		Value.

### B.18 Debug core control: 0x18

This register enables the debugger to temporarily disable logical cores. When returning from the debug interrupts, the cores set in this register will not execute. This enables single stepping to be implemented.

**0x18:** Debug core control

Bits	Perm	Init	Description
31:8	RO	-	Reserved
7:0	DRW		1-hot vector defining which logical cores are stopped when not in debug mode. Every bit which is set prevents the respective logical core from running.

XS1-L6A-64-TQ128

35

0x04: Control PSwitch permissions to debug registers

Bits	Perm	Init	Description
31:1	RO	-	Reserved
0	CRW		Set to 1 to restrict PSwitch access to all CRW marked registers to become read-only rather than read-write.

#### C.5 Cause debug interrupts: 0x05

This register can be used to raise a debug interrupt in this xCORE tile.

0x05: Cause debug interrupts

Bits	Perm	Init	Description
31:2	RO	-	Reserved
1	RO	0	Set to 1 when the processor is in debug mode.
0	CRW	0	Set to 1 to request a debug interrupt on the processor.

### C.6 xCORE Tile clock divider: 0x06

This register contains the value used to divide the PLL clock to create the xCORE tile clock. The divider is enabled under control of the tile control register

0x06: xCORE Tile clock divider

	Bits	Perm	Init	Description
e.	31:8	RO	-	Reserved
r	7:0	RW		Value of the clock divider minus one.

### C.7 Security configuration: 0x07

Copy of the security register as read from OTP.

0x07: Security configuration

ity	Bits	Perm	Init	Description
on	31:0	RO		Value.

### C.8 PLink status: 0x10 .. 0x13

Status of each of the four processor links; connecting the xCORE tile to the switch.

-XMOS-

### C.11 PC of logical core 1: 0x41

Ox41: PC of logical core 1 31:0

PermInitDescriptionROValue.

### C.12 PC of logical core 2: 0x42

<b>0x42:</b> PC of logical	Bits	Perm	Init	Description
core 2	31:0	RO		Value.

### C.13 PC of logical core 3: 0x43

0x42				
PC of logical	Bits	Perm	Init	Description
core 3	31:0	RO		Value.

### C.14 PC of logical core 4: 0x44

0x44: PC of logical core 4

Bits

31:0

Perm	Init	Description
RO		Value.

### C.15 PC of logical core 5: 0x45

0×45				
PC of logical	Bits	Perm	Init	Description
core 5	31:0	RO		Value.

XMOS

### C.16 SR of logical core 0: 0x60

Value of the SR of logical core 0

43

	Bits	Perm	Init	Description
	31:24	RO	-	Reserved
<b>0x01:</b> System switch	23:16	RO		Number of links on the switch.
	15:8	RO		Number of cores that are connected to this switch.
description	7:0	RO		Number of links per processor.

#### D.3 Switch configuration: 0x04

This register enables the setting of two security modes (that disable updates to the PLL or any other registers) and the header-mode.

Bits	Perm	Init	Description
31	RO	0	Set to 1 to disable any write access to the configuration registers in this switch.
30:9	RO	-	Reserved
8	RO	0	Set to 1 to disable updates to the PLL configuration register.
7:1	RO	-	Reserved
0	RO	0	Header mode. Set to 1 to enable 1-byte headers. This must be performed on all nodes in the system.

**0x04:** Switch configuration

### D.4 Switch node identifier: 0x05

This register contains the node identifier.

0x05 Switch node identifier

	Bits	Perm	Init	Description
•	31:16	RO	-	Reserved
: r	15:0	RW	0	The unique 16-bit ID of this node. This ID is matched most- significant-bit first with incoming messages for routing pur- poses.

#### D.5 PLL settings: 0x06

An on-chip PLL multiplies the input clock up to a higher frequency clock, used to clock the I/O, processor, and switch, see Oscillator. Note: a write to this register will cause the tile to be reset.

-XMOS

Bits	Perm	Init	Description
31:26	RO	-	Reserved
25:23	RW		OD: Output divider value The initial value depends on pins MODE0 and MODE1.
22:21	RO	-	Reserved
20:8	RW		F: Feedback multiplication ratio The initial value depends on pins MODE0 and MODE1.
7	RO	-	Reserved
6:0	RW		R: Oscilator input divider value The initial value depends on pins MODE0 and MODE1.

0x06: PLL settings

### D.6 System switch clock divider: 0x07

Sets the ratio of the PLL clock and the switch clock.

**0x07** System switch clock divider

17.	Bits	Perm	Init	Description
m	31:16	RO	-	Reserved
ck er	15:0	RW	0	Switch clock divider. The PLL clock will be divided by this value plus one to derive the switch clock.

### D.7 Reference clock: 0x08

Sets the ratio of the PLL clock and the reference clock used by the node.

**0x08:** Reference clock

Bits	Perm	Init	Description
31:16	RO	-	Reserved
15:0	RW	3	Architecture reference clock divider. The PLL clock will be divided by this value plus one to derive the 100 MHz reference clock.

### D.8 Directions 0-7: 0x0C

This register contains eight directions, for packets with a mismatch in bits 7..0 of the node-identifier. The direction in which a packet will be routed is goverened by the most significant mismatching bit.

-XMOS-

#### D.11 Debug source: 0x1F

Contains the source of the most recent debug event.

Bits	Perm	Init	Description
31:5	RO	-	Reserved
4	RW		If set, the external DEBUG_N pin is the source of the most recent debug interrupt.
3:1	RO	-	Reserved
0	RW		If set, the xCORE Tile is the source of the most recent debug interrupt.

0x1F: Debug source

### D.12 Link status, direction, and network: 0x20 .. 0x27

These registers contain status information for low level debugging (read-only), the network number that each link belongs to, and the direction that each link is part of. The registers control links C, D, A, B, G, H, E, and F in that order.

Bits	Perm	Init	Description
31:26	RO	-	Reserved
25:24	RO		If this link is currently routing data into the switch, this field specifies the type of link that the data is routed to: 0: plink 1: external link 2: internal control link
23:16	RO	0	If the link is routing data into the switch, this field specifies the destination link number to which all tokens are sent.
15:12	RO	-	Reserved
11:8	RW	0	The direction that this this link is associated with; set for rout- ing.
7:6	RO	-	Reserved
5:4	RW	0	Determines the network to which this link belongs, set for quality of service.
3	RO	-	Reserved
2	RO	0	Set to 1 if the current packet is junk and being thrown away. A packet is considered junk if, for example, it is not routable.
1	RO	0	Set to 1 if the switch is routing data into the link, and if a route exists from another link.
0	RO	0	Set to 1 if the link is routing data into the switch, and if a route is created to another link on the switch.

-XMOS

**0x20 .. 0x27:** Link status, direction, and network



	Bits	Perm	Init	Description
	31	RW	0	Write '1' to this bit to enable the link, write '0' to disable it. This bit controls the muxing of ports with overlapping links.
	30	RW	0	Set to 0 to operate in 2 wire mode or 1 to operate in 5 wire mode
	29:28	RO	-	Reserved
	27	RO	0	Set to 1 on error: an RX buffer overflow or illegal token encoding has been received. This bit clears on reading.
	26	RO	0	1 if this end of the link has issued credit to allow the remote end to transmit.
	25	RO	0	1 if this end of the link has credits to allow it to transmit.
	24	WO	0	Set to 1 to initialize a half-duplex link. This clears this end of the link's credit and issues a HELLO token; the other side of the link will reply with credits. This bit is self-clearing.
	23	WO	0	Set to 1 to reset the receiver. The next symbol that is detected will be assumed to be the first symbol in a token. This bit is self-clearing.
7.	22	RO	-	Reserved
87: nk on	21:11	RW	0	The number of system clocks between two subsequent transi- tions within a token
nd on	10:0	RW	0	The number of system clocks between two subsequent transmit tokens.

#### 0x80 .. 0x87: Link configuration and initialization

### D.15 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.

	Bits	Perm	Init	Description
	31	RW	0	Enable static forwarding.
:	30:5	RO	-	Reserved
( 1	4:0	RW	0	The destination channel end on this node that packets received in static mode are forwarded to.

-XMOS<sup>®</sup>

**0xA0 .. 0xA7** Static link configuration

# H Schematics Design Check List

✓ This section is a checklist for use by schematics designers using the XS1-L6A-64-TQ128. Each of the following sections contains items to check for each design.

#### H.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 10).
- The VDD (core) supply ramps monotonically (rises constantly) from 0V to its final value (0.95V 1.05V) within 10ms (Section 10).
- The VDD (core) supply is capable of supplying 300mA (Section 10).
- PLL\_AVDD is filtered with a low pass filter, for example an RC filter, see Section 10

#### H.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 10).
- □ A bulk decoupling capacitor of at least 10uF is placed on each supply (Section 10).

#### H.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. As the errata in the datasheets show, the internal pull-ups on these two pins can occasionally provide stronger than normal pull-up currents. For this reason, an RC type reset circuit is discouraged as behavior would be unpredictable. A voltage supervisor type reset device is recommended to guarantee a good reset. This also has the benefit of resetting the system should the relevant supply go out of specification.

#### H.4 Clock

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

-XM()S

# I PCB Layout Design Check List

This section is a checklist for use by PCB designers using the XS1-L6A-64-TQ128. Each of the following sections contains items to check for each design.

### I.1 Ground Plane

- □ Multiple vias (eg, 9) have been used to connect the center pad to the PCB ground plane. These minimize impedance and conduct heat away from the device. (Section 10.2).
- Other than ground vias, there are no (or only a few) vias underneath or closely around the device. This create a good, solid, ground plane.

#### I.2 Power supply decoupling

- $\Box$  The decoupling capacitors are all placed close to a supply pin (Section 10).
- $\Box$  The decoupling capacitors are spaced around the device (Section 10).
- The ground side of each decoupling capacitor has a direct path back to the center ground of the device.

#### I.3 PLL\_AVDD

The PLL\_AVDD filter (especially the capacitor) is placed close to the PLL\_AVDD pin (Section 10).

-XM()S

## L Revision History

Date	Description
2013-01-30	New datasheet - revised part numbering
2013-02-26	New multicore microcontroller introduction
	Moved configuration sections to appendices
2013-07-19	Updated Features list with available ports and links - Section 2
	Simplified link bits in Signal Description - Section 4
	New JTAG, xSCOPE and Debugging appendix - Section G
	New Schematics Design Check List - Section H
	New PCB Layout Design Check List - Section I
2013-09-16	Removed references to PCU. Pins set to GND - Section 3
2013-12-09	Added Industrial Ambient Temperature - Section 11.1
2014-07-08	Added PCU_GATE, PCU_CLK, PCU_VDD, PCU_VDDIO to Pin Configuration - Sec- tion 3
2015-04-14	Updated Introduction - Section 1; Pin Configuration - Section 3; Signal Descrip- tion - Section 4

# XMOS®

Copyright © 2015, All Rights Reserved.

Xmos Ltd. is the owner or licensee of this design, code, or Information (collectively, the "Information") and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. Xmos Ltd. makes no representation that the Information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.