



Welcome to [E-XFL.COM](https://www.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 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	-40°C ~ 85°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-i4

- ▶ **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 [5.5](#)
- ▶ **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 [5.6](#)
- ▶ **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 [5.3](#)
- ▶ **Clock blocks** xCORE devices include a set of programmable clock blocks that can be used to govern the rate at which ports execute. Section [5.4](#)
- ▶ **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 [8](#)
- ▶ **PLL** The PLL is used to create a high-speed processor clock given a low speed external oscillator. Section [6](#)
- ▶ **JTAG** The JTAG module can be used for loading programs, boundary scan testing, in-circuit source-level debugging and programming the OTP memory. Section [9](#)

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 XS1-L6A-64-TQ128 Features

► Multicore Microcontroller with Advanced Multi-Core RISC Architecture

- Six real-time logical cores
- Core share up to 500 MIPS
- Each logical core has:
 - Guaranteed throughput of between $\frac{1}{4}$ and $\frac{1}{6}$ of tile MIPS
 - 16x32bit dedicated registers
- 159 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

- 28 general-purpose I/O pins, configurable as input or output
 - Up to 16 x 1bit port, 6 x 4bit port, 4 x 8bit port, 2 x 16bit port, 1 x 32bit port
 - 4 xCONNECT links
- Port sampling rates of up to 60 MHz with respect to an external clock
- 32 channel ends for communication with other cores, on or off-chip

► Memory

- 64KB internal single-cycle SRAM for code and data storage
- 8KB internal OTP for application boot code

► Hardware resources

- 6 clock blocks
- 10 timers
- 4 locks

► 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

- 5: 500 MIPS
- 4: 400 MIPS

► Power Consumption

- Active Mode
 - 200 mA at 500 MHz (typical)
 - 160 mA at 400 MHz (typical)
- Standby Mode
 - 14 mA

► 128-pin TQ128 package 0.5 mm pitch

Signal	Function	Type	Properties
X0D41	8D ⁵ 16B ¹³	I/O	PD _S , R _U
X0D42	8D ⁶ 16B ¹⁴	I/O	PD _S , R _U
X0D43	8D ⁷ 16B ¹⁵	I/O	PU _S , R _U
X0D49	XLC _{out} ⁴ 32A ⁰	I/O	PD _S
X0D50	XLC _{out} ³ 32A ¹	I/O	PD _S
X0D51	XLC _{out} ² 32A ²	I/O	PD _S
X0D52	XLC _{out} ¹ 32A ³	I/O	PD _S
X0D53	XLC _{out} ⁰ 32A ⁴	I/O	PD _S
X0D54	XLC _{in} ⁰ 32A ⁵	I/O	PD _S
X0D55	XLC _{in} ¹ 32A ⁶	I/O	PD _S
X0D56	XLC _{in} ² 32A ⁷	I/O	PD _S
X0D57	XLC _{in} ³ 32A ⁸	I/O	PD _S
X0D58	XLC _{in} ⁴ 32A ⁹	I/O	PD _S
X0D61	XLD _{out} ⁴ 32A ¹⁰	I/O	PD _S
X0D62	XLD _{out} ³ 32A ¹¹	I/O	PD _S
X0D63	XLD _{out} ² 32A ¹²	I/O	PD _S
X0D64	XLD _{out} ¹ 32A ¹³	I/O	PD _S
X0D65	XLD _{out} ⁰ 32A ¹⁴	I/O	PD _S
X0D66	XLD _{in} ⁰ 32A ¹⁵	I/O	PD _S
X0D67	XLD _{in} ¹ 32A ¹⁶	I/O	PD _S
X0D68	XLD _{in} ² 32A ¹⁷	I/O	PD _S
X0D69	XLD _{in} ³ 32A ¹⁸	I/O	PD _S
X0D70	XLD _{in} ⁴ 32A ¹⁹	I/O	PD _S

5 Product Overview

The XS1-L6A-64-TQ128 is a powerful device that consists of a single xCORE Tile, which comprises a flexible logical processing cores with tightly integrated I/O and on-chip memory.

5.1 Logical cores

The tile has 6 active logical cores, which issue instructions down a shared four-stage pipeline. Instructions from the active cores are issued round-robin. If up to four logical cores are active, each core is allocated a quarter of the processing cycles. If more than four logical cores are active, each core is allocated at least $\frac{1}{n}$ cycles (for n cores). Figure 2 shows the guaranteed core performance depending on the number of cores used.

Figure 2:
Logical core
performance

Speed grade	MIPS	Frequency	Minimum MIPS per core (for n cores)							
			1	2	3	4	5	6		
4	400 MIPS	400 MHz	100	100	100	100	80	67		
5	500 MIPS	500 MHz	125	125	125	125	100	83		

There is no way that the performance of a logical core can be reduced below these predicted levels. 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 four 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.

5.2 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.

5.3 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 XS1-L6A-64-TQ128, and the software running on it. A combination of 1bit, 4bit, 8bit, 16bit and 32bit ports are available. All pins of a port provide either output or input. Signals in different directions cannot be mapped onto the same port.

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 $260\text{MHz} \leq F_{osc} \times \frac{F+1}{2} \times \frac{1}{R+1} \leq 1.3\text{GHz}$. 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 XS1-L Clock Frequency Control document, [X1433](#).

7 Boot Procedure

The device is kept in reset by driving RST_N low. When in reset, all GPIO pins are high impedance. 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), all GPIO pins have their internal pull-resistor enabled, and the processor boots at a clock speed that depends on MODE0 and MODE1.

The xCORE Tile boot procedure is illustrated in Figure 7. In normal usage, MODE[3:2] controls the boot source according to the table in Figure 8. If bit 5 of the security register (see §8.1) is set, the device boots from OTP.

Figure 7:
Boot
procedure

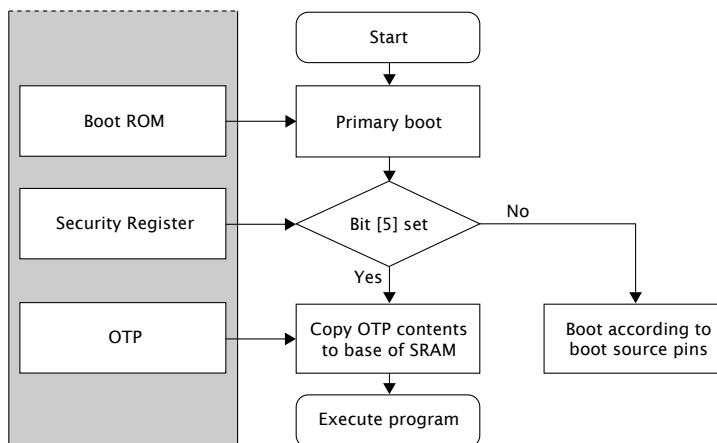


Figure 8:
Boot source
pins

MODE[3]	MODE[2]	Boot Source
0	0	None: Device waits to be booted via JTAG
0	1	Reserved
1	0	xConnect Link B
1	1	SPI

The boot image has the following format:

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

Feature	Bit	Description
Disable JTAG	0	The JTAG interface is disabled, making it impossible for the tile state or memory content to be accessed via the JTAG interface.
Disable Link access	1	Other tiles are forbidden access to the processor state via the system switch. Disabling both JTAG and Link access transforms an xCORE Tile into a “secure island” with other tiles free for non-secure user application code.
Secure Boot	5	The xCORE Tile is forced to boot from address 0 of the OTP, allowing the xCORE Tile boot ROM to be bypassed (<i>see</i> §7).
Redundant rows	7	Enables redundant rows in OTP.
Sector Lock 0	8	Disable programming of OTP sector 0.
Sector Lock 1	9	Disable programming of OTP sector 1.
Sector Lock 2	10	Disable programming of OTP sector 2.
Sector Lock 3	11	Disable programming of OTP sector 3.
OTP Master Lock	12	Disable OTP programming completely: disables updates to all sectors and security register.
Disable JTAG-OTP	13	Disable all (read & write) access from the JTAG interface to this OTP.
Disable Global Debug	14	Disables access to the DEBUG_N pin.
	21..15	General purpose software accessible security register available to end-users.
	31..22	General purpose user programmable JTAG UserID code extension.

Figure 10:
Security
register
features

port with resource ID 0x200100, and the OTP control is on a 16-bit port with ID 0x100300. Programming is performed through `libotp` and `xburn`.

8.2 SRAM

The xCORE Tile integrates a single 64KBSRAM bank for both instructions and data. All internal memory is 32 bits wide, and instructions are either 16-bit or 32-bit. Byte (8-bit), half-word (16-bit) or word (32-bit) accesses are supported and are executed within one tile clock cycle. There is no dedicated external memory interface, although data memory can be expanded through appropriate use of the ports.

9 JTAG

The JTAG module can be used for loading programs, boundary scan testing, in-circuit source-level debugging and programming the OTP memory.

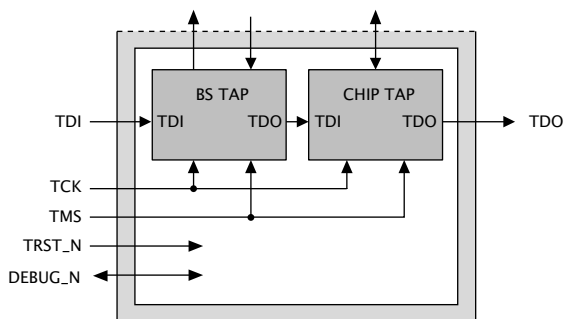


Figure 11:
JTAG chain
structure

The JTAG chain structure is illustrated in Figure 11. Directly after reset, two TAP controllers are present in the JTAG chain: the boundary scan TAP and the chip TAP. The boundary scan TAP is a standard 1149.1 compliant TAP that can be used for boundary scan of the I/O pins. The chip TAP provides access into the xCORE Tile, switch and OTP for loading code and debugging.

The TRST_N pin must be asserted low during and after power up for 100 ns. If JTAG is not required, the TRST_N pin can be tied to ground to hold the JTAG module in reset.

The DEBUG_N pin is used to synchronize the debugging of multiple xCORE Tiles. This pin can operate in both output and input mode. In output mode and when configured to do so, DEBUG_N is driven low by the device when the processor hits a debug break point. Prior to this point the pin will be tri-stated. In input mode and when configured to do so, driving this pin low will put the xCORE Tile into debug mode. Software can set the behavior of the xCORE Tile based on this pin. This pin should have an external pull up of 4K7-47KΩ or left not connected in single core applications.

The JTAG device identification register can be read by using the IDCODE instruction. Its contents are specified in Figure 12.

Figure 12:
IDCODE
return value

Bit31			Device Identification Register																												Bit0	
Version				Part Number																Manufacturer Identity												1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	1	0	0	1	1	
0				0				0				0				2				6				3				3				

The JTAG usercode register can be read by using the USERCODE instruction. Its contents are specified in Figure 13. The OTP User ID field is read from bits [22:31] of the security register, see §8.1 (all zero on unprogrammed devices).

Figure 13:
USERCODE
return value

Usercode Register																																Bit0
OTP User ID																Unused				Silicon Revision												
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
0				0				0				2				8				0				0				0				

11.4 Reset Timing

Figure 19:
Reset timing

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

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

11.5 Power Consumption

Figure 20:
xCORE Tile
currents

Symbol	Parameter	MIN	TYP	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	A, H
I(ADDPLL)	PLL_AVDD current			7	mA	I

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, 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](#).

11.6 Clock

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
f	Frequency	4.22	20	100	MHz	
SR	Slew rate	0.10			V/ns	
TJ(LT)	Long term jitter (pk-pk)			2	%	A
f(MAX)	Processor clock frequency (Speed Grade 4)			400	MHz	B
	Processor clock frequency (Speed Grade 5)			500	MHz	B

Figure 21:
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, [X1433](#).

The OTP may be programmed using its internal charge pump or by supplying a 6.5V VPP programming voltage on the OTP_VPP pin. Unless a programming cycle is underway the OTP_VPP pins should be left undriven.

11.7 xCORE Tile I/O AC Characteristics

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
T(XOVALID)	Input data valid window	8			ns	
T(XOINVALID)	Output data invalid window	9			ns	
T(XIFMAX)	Rate at which data can be sampled with respect to an external clock			60	MHz	

Figure 22:
I/O AC characteristics

The input valid window parameter relates to the capability of the device to capture data input to the chip with respect to an external clock source. It is calculated as the sum of the input setup time and input hold time with respect to the external clock as measured at the pins. The output invalid window specifies the time for which an output is invalid with respect to the external clock. Note that these parameters are specified as a window rather than absolute numbers since the device provides functionality to delay the incoming clock with respect to the incoming data.

Information on interfacing to high-speed synchronous interfaces can be found in the XS1 Port I/O Timing document, [X5821](#).

11.8 xConnect Link Performance

Figure 23:
Link
performance

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
B(2blinkP)	2b link bandwidth (packetized)			87	MBit/s	A, B
B(5blinkP)	5b link bandwidth (packetized)			217	MBit/s	A, B
B(2blinkS)	2b link bandwidth (streaming)			100	MBit/s	B
B(5blinkS)	5b link bandwidth (streaming)			250	MBit/s	B

A Assumes 32-byte packet in 3-byte header mode. Actual performance depends on size of the header and payload.

B 7.5 ns symbol time.

The asynchronous nature of links means that the relative phasing of CLK clocks is not important in a multi-clock system, providing each meets the required stability criteria.

11.9 JTAG Timing

Figure 24:
JTAG timing

Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
f(TCK_D)	TCK frequency (debug)			18	MHz	
f(TCK_B)	TCK frequency (boundary scan)			10	MHz	
T(SETUP)	TDO to TCK setup time	5			ns	A
T(HOLD)	TDO to TCK hold time	5			ns	A
T(DELAY)	TCK to output delay			15	ns	B

A Timing applies to TMS and TDI inputs.

B Timing applies to TDO output from negative edge of TCK.

All JTAG operations are synchronous to TCK apart from the global asynchronous reset TRST_N.

0x15:
Debug
interrupt type

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 watchpoints 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

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 contains the resource identifier.

0x16:
Debug
interrupt data

Bits	Perm	Init	Description
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.

0x10 .. 0x13:
PLink status

Bits	Perm	Init	Description
31:26	RO	-	Reserved
25:24	RO		00 - ChannelEnd, 01 - ERROR, 10 - PSCTL, 11 - Idle.
23:16	RO		Based on SRC_TARGET_TYPE value, it represents channelEnd ID or Idle status.
15:6	RO	-	Reserved
5:4	RO		Two-bit network identifier
3	RO	-	Reserved
2	RO		1 when the current packet is considered junk and will be thrown away.
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.

C.9 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.10 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	RO		Value.

0x60:
SR of logical
core 0

Bits	Perm	Init	Description
31:0	RO		Value.

C.17 SR of logical core 1: 0x61

0x61:
SR of logical
core 1

Bits	Perm	Init	Description
31:0	RO		Value.

C.18 SR of logical core 2: 0x62

0x62:
SR of logical
core 2

Bits	Perm	Init	Description
31:0	RO		Value.

C.19 SR of logical core 3: 0x63

0x63:
SR of logical
core 3

Bits	Perm	Init	Description
31:0	RO		Value.

C.20 SR of logical core 4: 0x64

0x64:
SR of logical
core 4

Bits	Perm	Init	Description
31:0	RO		Value.

C.21 SR of logical core 5: 0x65

0x65:
SR of logical
core 5

Bits	Perm	Init	Description
31:0	RO		Value.

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

0x04: Switch configuration	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.

D.4 Switch node identifier: 0x05

This register contains the node identifier.

0x05: Switch node identifier	Bits	Perm	Init	Description
	31:16	RO	-	Reserved
	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 purposes.

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.

0x06:
PLL settings

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: Oscillator input divider value The initial value depends on pins MODE0 and MODE1.

D.6 System switch clock divider: 0x07

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

0x07:
System
switch clock
divider

Bits	Perm	Init	Description
31:16	RO	-	Reserved
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 governed by the most significant mismatching bit.

0x0C:
Directions
0-7

Bits	Perm	Init	Description
31:28	RW	0	The direction for packets whose first mismatching bit is 7.
27:24	RW	0	The direction for packets whose first mismatching bit is 6.
23:20	RW	0	The direction for packets whose first mismatching bit is 5.
19:16	RW	0	The direction for packets whose first mismatching bit is 4.
15:12	RW	0	The direction for packets whose first mismatching bit is 3.
11:8	RW	0	The direction for packets whose first mismatching bit is 2.
7:4	RW	0	The direction for packets whose first mismatching bit is 1.
3:0	RW	0	The direction for packets whose first mismatching bit is 0.

D.9 Directions 8-15: 0x0D

This register contains eight directions, for packets with a mismatch in bits 15..8 of the node-identifier. The direction in which a packet will be routed is governed by the most significant mismatching bit.

0x0D:
Directions
8-15

Bits	Perm	Init	Description
31:28	RW	0	The direction for packets whose first mismatching bit is 15.
27:24	RW	0	The direction for packets whose first mismatching bit is 14.
23:20	RW	0	The direction for packets whose first mismatching bit is 13.
19:16	RW	0	The direction for packets whose first mismatching bit is 12.
15:12	RW	0	The direction for packets whose first mismatching bit is 11.
11:8	RW	0	The direction for packets whose first mismatching bit is 10.
7:4	RW	0	The direction for packets whose first mismatching bit is 9.
3:0	RW	0	The direction for packets whose first mismatching bit is 8.

D.10 DEBUG_N configuration: 0x10

Configures the behavior of the DEBUG_N pin.

0x10:
DEBUG_N
configuration

Bits	Perm	Init	Description
31:2	RO	-	Reserved
1	RW	0	Set to 1 to enable signals on DEBUG_N to generate DCALL on the core.
0	RW	0	When set to 1, the DEBUG_N wire will be pulled down when the node enters debug mode.

D.13 PLink status and network: 0x40 .. 0x43

These registers contain status information and the network number that each processor-link belongs to.

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.
15:6	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.

0x40 .. 0x43:
PLink status
and network

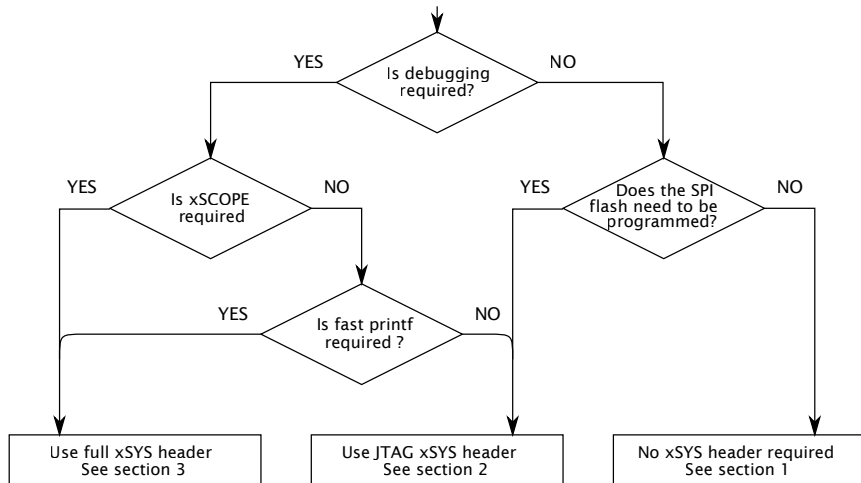
D.14 Link configuration and initialization: 0x80 .. 0x87

These registers contain configuration and debugging information specific to external links. The link speed and width can be set, the link can be initialized, and the link status can be monitored. The registers control links C, D, A, B, G, H, E, and F in that order.

G JTAG, xSCOPE and Debugging

If you intend to design a board that can be used with the XMOS toolchain and xTAG debugger, you will need an xSYS header on your board. Figure 32 shows a decision diagram which explains what type of xSYS connectivity you need. The three subsections below explain the options in detail.

Figure 32:
Decision
diagram for
the xSYS
header



G.1 No xSYS header

The use of an xSYS header is optional, and may not be required for volume production designs. However, the XMOS toolchain expects the xSYS header; if you do not have an xSYS header then you must provide your own method for writing to flash/OTP and for debugging.

G.2 JTAG-only xSYS header

The xSYS header connects to an xTAG debugger, which has a 20-pin 0.1" female IDC header. The design will hence need a male IDC header. We advise to use a boxed header to guard against incorrect plug-ins. If you use a 90 degree angled header, make sure that pins 2, 4, 6, ..., 20 are along the edge of the PCB.

Connect pins 4, 8, 12, 16, 20 of the xSYS header to ground, and then connect:

- ▶ TDI to pin 5 of the xSYS header
- ▶ TMS to pin 7 of the xSYS header
- ▶ TCK to pin 9 of the xSYS header
- ▶ DEBUG_N to pin 11 of the xSYS header

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 four 0402 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.