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	Obsolete
Core Processor	XCore
Core Size	32-Bit 6-Core
Speed	500MIPS
Connectivity	Configurable
Peripherals	-
Number of I/O	42
Program Memory Size	64KB (16K x 32)
Program Memory Type	SRAM
EEPROM Size	-
RAM Size	-
Voltage - Supply (Vcc/Vdd)	0.90V ~ 5.5V
Data Converters	A/D 4x12b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	96-LFBGA
Supplier Device Package	96-FBGA (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/xmos/xs1-a6a-64-fb96-c5

2 XS1-A6A-64-FB96 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 1/4 and 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
- **12b 1MSPS 4-channel SAR Analog-to-Digital Converter**
- **1 x LDO**
- **2 x DC-DC converters and Power Management Unit**
- **Watchdog Timer**
- **Onchip clocks/oscillators**
  - Crystal oscillator
  - 20MHz/31kHz silicon oscillators
- **Programmable I/O**
  - 42 general-purpose I/O pins, configurable as input or output
    - Up to 16 x 1bit port, 6 x 4bit port, 3 x 8bit port, 1 x 16bit port
    - 2 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
  - 128 bytes Deep Sleep Memory
- **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 (typical)**
  - 300 mW at 500 MHz (typical)
  - Sleep Mode: 500 µW
- **96-pin FBGA package 0.8 mm pitch**

3 Pin Configuration

	1	2	3	4	5	6	7	8	9	10	11	12
A	AVDD	ADC0	ADC2	NC	1A X0D00	4A X0D02	4B X0D04	4B X0D06	4A X0D08	1C X0D10	1E X0D12	4C X0D14
B	TDO	ADC1	ADC3	NC	1B X0D01	4A X0D03	4B X0D05	4B X0D07	4A X0D09	1D X0D11	1F X0D13	4C X0D15
C	TCK	RST_N									4D X0D17	4D X0D16
D	TMS	TDI									4D X0D19	4D X0D18
E	OSC_ EXT_N	DEBUG_ N			GND	GND	GND	GND			4C X0D21	4C X0D20
F	XI/ CLK	NC			AVSS	GND	GND	GND			1H X0D23	1G X0D22
G	XO	NC			GND	GND	GND	GND			ADC_ SAMPLE	32A X0D70
H	NC	VSUP			GND	GND	GND	GND			1J X0D25	1I X0D24
J	SW1	SW1									4E X0D27	4E X0D26
K	VDDCORE	VDDCORE									4F X0D29	4F X0D28
L	PGND	PGND	VDDIO	MODE[0]	MODE[1]	MODE[2]	X0D43/ WAKE	1L X0D35	1P X0D39	1N X0D37	4F X0D31	4F X0D30
M	VSUP	VSUP	VDDIO	PGND	VDD1V8	SW2	MODE[3]	1K X0D34	1O X0D38	1M X0D36	4E X0D33	4E X0D32

6 Product Overview

The XS1-A6A-64-FB96 comprises a digital and an analog node, as shown in Figure 3. The digital node comprises an xCORE Tile, a Switch, and a PLL (Phase-locked-loop). The analog node comprises a multi-channel ADC (Analog to Digital Converter), deep sleep memory, an oscillator, a real-time counter, and power supply control.

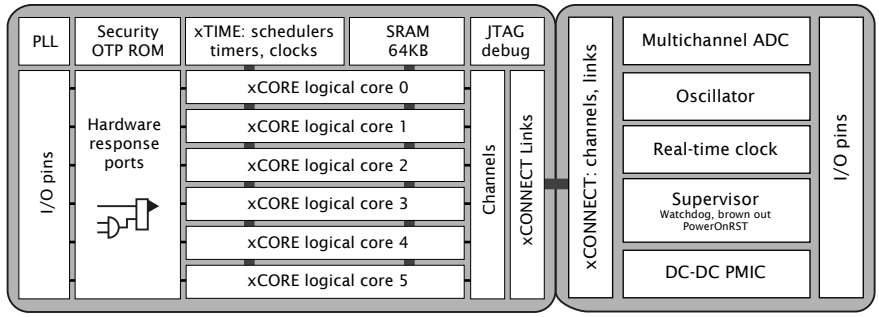

Figure 3: Block Diagram

All communication between the digital and analog node takes place over a link that is connected to the Switch of the digital node. As such, the analog node can be controlled from any node on the system. The analog functions can be configured using a set of node configuration registers, and a set of registers for each of the peripherals.

The device can be programmed using high-level languages such as C/C++ and the XMOS-originated XC language, which provides extensions to C that simplify the control over concurrency, I/O and timing, or low-level assembler.

6.1 XCore Tile

The xCORE Tile is a flexible multicore microcontroller component with tightly integrated I/O and on-chip memory. The tile contains multiple logical cores that run simultaneously, each of which is guaranteed a slice of processing power and can execute computational code, control software and I/O interfaces. The logical cores use channels to exchange data within a tile or across tiles. Multiple devices can be deployed and connected using an integrated switching network, enabling more resources to be added to a design. The I/O pins are driven using intelligent ports that can serialize data, interpret strobe signals and wait for scheduled times or events, making the device ideal for real-time control applications.

6.2 ADC and Power Management

Each XS1-A6A-64-FB96 device includes a set of analog components, including a 12b, 4-channel ADC, power management unit, watchdog timer, real-time counter and deep sleep memory. The device reduces the number of additional external components required and allows designs to be implemented using simple 2-layer boards.

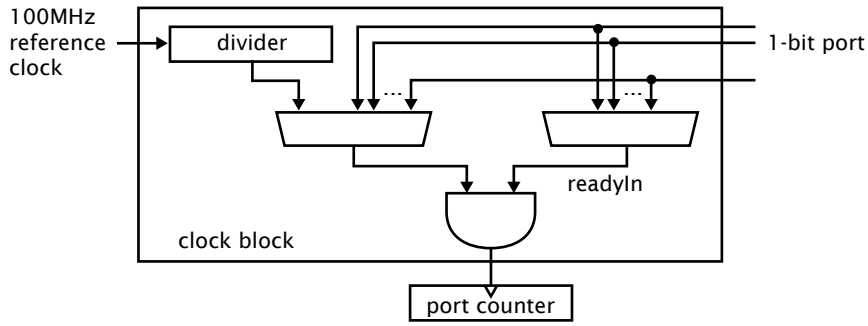

Figure 6: Clock block diagram

On reset, each port is connected to clock block 0, which runs from the processor reference clock.

7.5 Channels and Channel Ends

Logical cores communicate using point-to-point connections, formed between two channel ends. A channel-end is a resource on an xCORE tile, that is allocated by the program. Each channel-end has a unique system-wide identifier that comprises a unique number and their tile identifier. Data is transmitted to a channel-end by an output-instruction; and the other side executes an input-instruction. Data can be passed synchronously or asynchronously between the channel ends.

7.6 xCONNECT Switch and Links

XMOS devices provide a scalable architecture, where multiple xCORE devices can be connected together to form one system. Each xCORE device has an xCONNECT interconnect that provides a communication infrastructure for all tasks that run on the various xCORE tiles on the system.

The interconnect relies on a collection of switches and XMOS links. Each xCORE device has an on-chip switch that can set up circuits or route data. The switches are connected by xConnect Links. An XMOS link provides a physical connection between two switches. The switch has a routing algorithm that supports many different topologies, including lines, meshes, trees, and hypercubes.

The links operate in either 2 wires per direction or 5 wires per direction mode, depending on the amount of bandwidth required. Circuit switched, streaming and packet switched data can both be supported efficiently. Streams provide the fastest possible data rates between tiles , 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.

systems to be partially programmed, dedicating one or more tiles to perform a particular function, leaving the other tiles user-programmable.

9.4 Security register

The security register enables security features on the xCORE tile. The features shown in Figure 12 provide a strong level of protection and are sufficient for providing strong IP security.

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 processor is forced to boot from address 0 of the OTP, allowing the processor boot ROM to be bypassed (*see* §9).
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 accessable security register available to end-users.
	31..22	General purpose user programmable JTAG UserID code extension.

Figure 12: Security register features

10 Memory

10.1 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.

12 Supervisor Logic

An independent supervisor circuit provides power-on-reset, brown-out, and watchdog capabilities. This facilitates the design of systems that fail gracefully, whilst keeping BOM costs down.

The reset supervisor holds the chip in reset until all power supplies are good. This provides a power-on-reset (POR). An external reset is optional and the pin RST_N can be left not-connected.

If at any time any of the power supplies drop because of too little supply or too high a demand, the power supervisor will bring the chip into reset until the power supplies have been restored. This will reboot the system as if a cold-start has happened.

The 16-bit watchdog timer provides 1ms accuracy and runs independently of the real-time counter. It can be programmed with a time-out of between 1 ms and 65 seconds (Appendix E). If the watchdog is not set before it times out, the XS1-A6A-64-FB96 is reset. On boot, the program can read a register to test whether the reset was due to the watchdog. The watchdog timer is only enabled and clocked whilst the processor is in the AWAKE power state.

13 Energy management

XS1-A6A-64-FB96 devices can be powered by:

- An external 5v core and 3.3v I/O supply.
- A single 3.3v supply.

13.1 DC-DC

XS1-A6A-64-FB96 devices include two DC-DC buck converters which can be configured to take input voltages between 3.3-5V power supply and output circuit voltages (nominally 1.8V and 1.0V) required by the analog peripherals and digital node.

13.2 Power mode controller

The device transitions through multiple states during the power-up and powerdown process.

The device is quiescent in the ASLEEP state, and is running in the AWAKE state. The other states allow a controlled transition between AWAKE and ASLEEP.

A transition from AWAKE state to ASLEEP state is instigated by a write to the general control register. Sleep requests must only be made in the AWAKE state.

A transition from the ASLEEP state into the AWAKE state is instigated by a wakeup request triggered by an input, or a timer. The device only responds to a wakeup

accurate clock is required, even whilst asleep, then an external crystal or oscillator shall be provided that is used in both AWAKE and ASLEEP state.

The designer has to make a trade-off between accuracy of clocks when asleep and awake, costs, and deep-sleep power consumption. Four example designs are shown in Figure 15.

Clocks used		Power	BOM	Accuracy	
Awake	Asleep	Asleep	costs	Awake	Asleep
20 Mhz SiOsc	31,250 SiOsc	lowest	lowest	lowest	lowest
24 MHz Crystal	31,250 SiOsc	lowest	medium	highest	lowest
5 MHz ext osc	5 MHz ext osc	medium	highest	highest	highest
24 MHz Crystal	24 MHz crystal	highest	medium	highest	highest

Figure 15: Example trade-offs in oscillator selection

During deep-sleep, the program can store some state in 128 bytes of Deep Sleep Memory.

13.4 Requirements during sleep mode

Whilst in sleep mode, the device must still be powered as normal over 3V3 or 5V0 on VSUP, and 3V3 on VDDIO; however it will draw less power on both VSUP and VDDIO.

For best results (lowest power):

- The XTAL bias and XTAL oscillators should be switched off.
- The sleep register should be configured to
  - Disable all power supplies except DCDC2.
  - Set all power supplies to PFM mode
  - Mask the clock
  - Assert reset
- All GPIO and JTAG pins should be quiescent, and none should be driven against a pull-up or pull-down.
- 3V3 should be supplied as the input voltage to VSUP.

This will result in a power consumption of less than 100 uA on both VSUP and VDDIO.

If any power supply loses power-good status during the asleep-to-awake or awake-to-asleep transitions, a system reset is issued.

The JTAG usercode register can be read by using the USERCODE instruction. Its contents are specified in Figure 18. The OTP User ID field is read from bits [22:31] of the security register , *see* §10.1 (all zero on unprogrammed devices).

Figure 18: USERCODE return value

Bit31										Usercode Register																				Bit0
OTP User ID										Unused				Silicon Revision																
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0				0				0				2				C				0				0						

15 Board Integration

XS1-A6A-64-FB96 devices are optimized for layout on low cost PCBs using standard design rules. Careful layout is required to maximize the device performance. XMOS therefore recommends that the guidelines in this section are followed when laying out boards using the device.

The XS1-A6A-64-FB96 includes two DC-DC buck converters that take input voltages between 3.3-5V and output the 1.8V and 1.0V circuits required by the digital core and analogue peripherals. The DC-DC converters should have a 4.7uF X5R or X7R ceramic capacitor and a 100nF X5R or X7R ceramic capacitor on the VSUP input pins M1 and M2. These capacitors must be placed as close as possible to the those pins (within a maximum of 5mm), with the routing optimized to minimize the inductance and resistance of the traces.

The SW output pin must have an LC filter on the output with a 4.7uH inductor and 22uF X5R capacitor. The capacitor must have maximum ESR value of 0.015R, and the inductor should have a maximum DCR value of 0.07R, to meet the efficiency specifications of the DC-DC converter, although this requirement may be relaxed if a drop in efficiency is acceptable. A list of suggested inductors is in Figure 19.

Figure 19: Example 4.7 μH inductors

	Part number	Current	Max DCR	Package
Yuden	CBC2518T4R7M	680 mA	260 $m\Omega$	2518 (1007)
TDK	NLCV32T-4R7M-PFR	620 mA	200 $m\Omega$	3225 (1210)
Murata	LQM2HPN4R7MGC	800 mA	225 $m\Omega$	2520 (1008)
Sumida	0420CDMCBDS-4R7MC	3400 mA	80 $m\Omega$	4.7 x 4.3 mm
Wurth	744043004	1550 mA	70 $m\Omega$	4.8 x 4.8 mm
Murata	LQH55DN4R7M03L	2700 mA	57 $m\Omega$	5750 (2220)

The traces from the SW output pins to the inductor and from the output capacitor back to the VDD pins must be routed to minimize the coupling between them.

The power supplies must be brought up monotonically and input voltages must not exceed specification at any time.

The VDDIO supply to the XS1-A6A-64-FB96 requires a 100nF X5R or X7R ceramic decoupling capacitor placed as close as possible to the supply pins.

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 digital and analogue node configuration registers

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).

A.4 Accessing a register of an analogue peripheral

Peripheral registers can be accessed through the interconnect using the functions `write_periph_32(device, peripheral, ...)`, `read_periph_32(device, peripheral, ...)`, `write_periph_8(device, peripheral, ...)`, and `read_periph_8(device, peripheral, ...)`; where device is the name of the analogue device, and peripheral is the number of the peripheral. These functions implement the protocols described below.

A channel-end should be allocated to communicate with the configuration registers. The destination of the channel-end should be set to `0xnnnnpp02` where `nnnn` is the node-identifier and `pp` is the peripheral identifier.

A write message comprises the following:

control-token 36	24-bit response channel-end identifier	8-bit register number	8-bit size	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:

B Processor Status Configuration

The processor status control registers can be accessed directly by the processor using processor status reads and writes (use `getps(reg)` and `setps(reg,value)` for reads and writes).

Number	Perm	Description
0x00	RW	RAM base address
0x01	RW	Vector base address
0x02	RW	xCORE Tile control
0x03	RO	xCORE Tile boot status
0x05	RO	Security configuration
0x06	RW	Ring Oscillator Control
0x07	RO	Ring Oscillator Value
0x08	RO	Ring Oscillator Value
0x09	RO	Ring Oscillator Value
0x0A	RO	Ring Oscillator Value
0x10	DRW	Debug SSR
0x11	DRW	Debug SPC
0x12	DRW	Debug SSP
0x13	DRW	DGETREG operand 1
0x14	DRW	DGETREG operand 2
0x15	DRW	Debug interrupt type
0x16	DRW	Debug interrupt data
0x18	DRW	Debug core control
0x20 .. 0x27	DRW	Debug scratch
0x30 .. 0x33	DRW	Instruction breakpoint address
0x40 .. 0x43	DRW	Instruction breakpoint control
0x50 .. 0x53	DRW	Data watchpoint address 1
0x60 .. 0x63	DRW	Data watchpoint address 2
0x70 .. 0x73	DRW	Data breakpoint control register
0x80 .. 0x83	DRW	Resources breakpoint mask
0x90 .. 0x93	DRW	Resources breakpoint value
0x9C .. 0x9F	DRW	Resources breakpoint control register

Figure 41: Summary

B.8 Ring Oscillator Value: 0x08

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

0x08: Ring Oscillator Value

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

B.9 Ring Oscillator Value: 0x09

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

0x09: Ring Oscillator Value

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

B.11 Debug SSR: 0x10

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

0x10: Debug SSR

Bits	Perm	Init	Description
31:0	RO	-	Reserved

B.12 Debug SPC: 0x11

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

0x80 .. 0x83: Resources breakpoint mask

Bits	Perm	Init	Description
31:0	DRW		Value.

B.26 Resources breakpoint value: 0x90 .. 0x93

This set of registers contains the value for the four resource watchpoints.

0x90 .. 0x93: Resources breakpoint value

Bits	Perm	Init	Description
31:0	DRW		Value.

B.27 Resources breakpoint control register: 0x9C .. 0x9F

This set of registers controls each of the four resource watchpoints.

0x9C .. 0x9F: Resources breakpoint control register

Bits	Perm	Init	Description
31:24	RO	-	Reserved
23:16	DRW	0	A bit for each logical core in the tile allowing the breakpoint to be enabled individually for each logical core.
15:2	RO	-	Reserved
1	DRW	0	By default, resource watchpoints trigger when the resource id masked with the set Mask equals the Value. If set to 1, resource watchpoints trigger when the resource id masked with the set Mask is not equal to the Value.
0	DRW	0	When 1 the instruction breakpoint is enabled.

C.22 Chanend status: 0x80 .. 0x9F

These registers record the status of each channel-end on the tile.

0x80 .. 0x9F: Chanend 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.

D Digital Node Configuration

The digital node control registers can be accessed using configuration reads and writes (use `write_node_config_reg(device, ...)` and `read_node_config_reg(device, ...)` for reads and writes).

Number	Perm	Description
0x00	RO	Device identification
0x01	RO	System switch description
0x04	RW	Switch configuration
0x05	RW	Switch node identifier
0x06	RW	PLL settings
0x07	RW	System switch clock divider
0x08	RW	Reference clock
0x0C	RW	Directions 0-7
0x0D	RW	Directions 8-15
0x10	RW	DEBUG_N configuration
0x1F	RO	Debug source
0x20 .. 0x27	RW	Link status, direction, and network
0x40 .. 0x43	RW	PLink status and network
0x80 .. 0x87	RW	Link configuration and initialization
0xA0 .. 0xA7	RW	Static link configuration

Figure 43: Summary

D.1 Device identification: 0x00

This register contains version and revision identifiers and the mode-pins as sampled at boot-time.

Bits	Perm	Init	Description
31:24	RO	0x00	Chip identifier.
23:16	RO		Sampled values of pins MODE0, MODE1, ... on reset.
15:8	RO		SSwitch revision.
7:0	RO		SSwitch version.

0x00: Device identification

D.2 System switch description: 0x01

This register specifies the number of processors and links that are connected to this switch.

0x04: Node configuration register

Bits	Perm	Init	Description
31	RW	0	Set to 1 to disable further updates to the node configuration and link control and status registers.
30:1	RO	-	Reserved
0	RW	0	Header mode. 0: 3-byte headers; 1: 1-byte headers.

E.3 Node identifier: 0x05

0x05: Node identifier

Bits	Perm	Init	Description
31:16	RO	-	Reserved
15:0	RW	0	16-bit node identifier. This does not need to be set, and is present for compatibility with XS1-switches.

E.4 Reset and Mode Control: 0x50

The XS1-S has two main reset signals: a system-reset and an xCORE Tile-reset. System-reset resets the whole system including external devices, whilst xCORE Tile-reset resets the xCORE Tile(s) only. The resets are induced either by software (by a write to the register below) or by one of the following:

* External reset on RST_N (System reset)

* Brown out on one of the power supplies (System reset)

* Watchdog timer (System reset)

* Sleep sequence (xCORE Tile reset)

* Clock source change (xCORE Tile reset)

The minimum system reset duration is achieved when the fastest permissible clock is used. The reset durations will be proportionately longer when a slower clock is used. Note that the minimum system reset duration allows for all power rails except the VOUT2 to turn off, and decay.

The length of the system reset comes from an internal counter, counting 524,288 oscillator clock cycles which gives the maximum time allowable for the supply rails to discharge. The system reset duration is a balance between leaving a long time for the supply rails to discharge, and a short time for the system to boot. Example reset times are 44 ms with a 12 MHz oscillator or 5.5 ms with a 96 MHz oscillator.

E.8 Watchdog Disable: 0xD7

To enable the watchdog, write 0 to this register. To disable the watchdog, write the value 0x0D1SAB1E to this register.

0xD7: Watchdog Disable

Bits	Perm	Init	Description
31:0	RW	0x0D15AB1E	A value of 0x0D15AB1E written to this register resets and disables the watchdog timer.

F ADC Configuration

The device has a 12-bit Analogue to Digital Converter (ADC). It has multiple input pins, and on each positive clock edge on port 1I, it samples and converts a value on the next input pin. The data is transmitted to a channel-end that must be set on enabling the ADC input pin.

The *ADC* is peripheral 2. The control registers are accessed using 32-bit reads and writes (use `write_periph_32(device, 2, ...)` and `read_periph_32(device, 2, ...)` for reads and writes).

Figure 45: Summary

Number	Perm	Description
0x00	RW	ADC Control input pin 0
0x04	RW	ADC Control input pin 1
0x08	RW	ADC Control input pin 2
0x0C	RW	ADC Control input pin 3
0x20	RW	ADC General Control

F.1 ADC Control input pin 0: 0x00

Controls specific to ADC input pin 0.

0x00: ADC Control input pin 0

Bits	Perm	Init	Description
31:8	RW	0	The node and channel-end identifier to which data for this ADC input pin should be send to. This is the top 24 bits of the channel-end identifier as allocated on an xCORE Tile.
7:1	RO	-	Reserved
0	RW	0	Set to 1 to enable this input pin on the ADC.

0x18: Power supply states whilst AWAKE

Bits	Perm	Init	Description
31:15	RO	-	Reserved
14	RW	0	Set to 1 to disable clock to the xCORE Tile.
13:10	RO	-	Reserved
9	RW	0	Sets modulation used by DCDC2: 0: PWM modulation (max 475 mA) 1: PFM modulation (max 50 mA)
8	RW	0	Sets modulation used by DCDC1: 0: PWM modulation (max 700 mA) 1: PFM modulation (max 50 mA)
7:6	RO	-	Reserved
5	RW	1	Set to 1 to enable VOUT6 (IO supply).
4	RW	1	Set to 1 to enable LDO5 (core PLL supply).
3:2	RO	-	Reserved
1	RO	1	Set to 1 to enable DCDC2 (analogue supply).
0	RW	1	Set to 1 to enable DCDC1 (core supply).

J.8 Power supply states whilst SLEEPING1: 0x1C

This register controls what state the power control block should be in when in the SLEEPING1 state. It also defines the time that the system shall stay in this state.

M 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 51 shows a decision diagram which explains what type of xSYS connectivity you need. The three subsections below explain the options in detail.

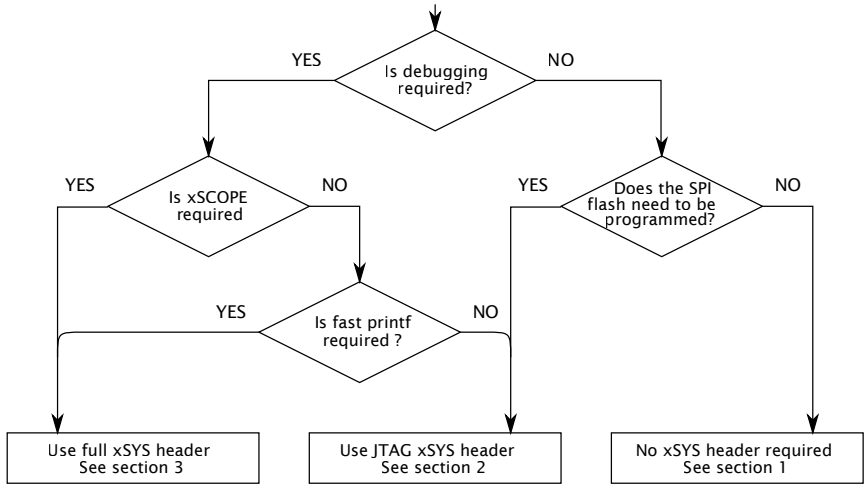

Figure 51: Decision diagram for the xSYS header

M.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.

M.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

- TDO to pin 13 of the xSYS header
- RST_N to pin 15 of the xSYS header
- If MODE2 is configured high, connect MODE2 to pin 3 of the xSYS header. Do not connect to VDDIO.
- If MODE3 is configured high, connect MODE3 to pin 3 of the xSYS header. Do not connect to VDDIO.

The RST_N net should be open-drain, active-low, and have a pull-up to VDDIO.

M.3 Full xSYS header

For a full xSYS header you will need to connect the pins as discussed in Section M.2, and then connect a 2-wire xCONNECT Link to the xSYS header. The links can be found in the Signal description table (Section 4): they are labelled XLA, XLB, etc in the function column. The 2-wire link comprises two inputs and outputs, labelled ${}^{1}_{out}$, ${}^{0}_{out}$, ${}^{0}_{in}$, and ${}^{1}_{in}$. For example, if you choose to use XLB of tile 0 for xSCOPE I/O, you need to connect up XLB^{1}_{out}, XLB^{0}_{out}, XLB^{0}_{in}, XLB^{1}_{in} as follows:

- XLB^{1}_{out} (X0D16) to pin 6 of the xSYS header with a 33R series resistor close to the device.
- XLB^{0}_{out} (X0D17) to pin 10 of the xSYS header with a 33R series resistor close to the device.
- XLB^{0}_{in} (X0D18) to pin 14 of the xSYS header.
- XLB^{1}_{in} (X0D19) to pin 18 of the xSYS header.

N Schematics Design Check List

☑ This section is a checklist for use by schematics designers using the XS1-A6A-64-FB96. Each of the following sections contains items to check for each design.

N.1 Clock

☐ Pins MODE0 and MODE1 are set to the correct value for the chosen frequency. The MODE settings are shown in the Oscillator section, Section 8. If you have a choice between two values, choose the value with the highest multiplier ratio since that will boot faster.

☐ OSC_EXT_N is tied to ground (for use with a crystal or oscillator) or tied to VDDIO (for use with the internal oscillator). If using the internal oscillator, set MODE0 and MODE1 to be for the 20-48 MHz range (Section 8).

☐ If you have used an oscillator, it is a 1V8 oscillator. (Section 16)

N.2 USB ULPI Mode

This section can be skipped if you do not have an external USB PHY.

☐ If using ULPI, the ULPI signals are connected to specific ports as shown in Section K.

☐ If using ULPI, the ports that are used internally are not connected, see Section K. (Note that this limitation only applies when the ULPI is enabled, they can still be used before or after the ULPI is being used.)

N.3 Boot

☐ The device is connected to a SPI flash for booting, connected to X0D0, X0D01, X0D10, and X0D11 (Section 9). If not, you must boot the device through OTP or JTAG.

☐ The device that is connected to flash has both MODE2 and MODE3 connected to pin 3 on the xSYS Header (MSEL). If no debug adapter connection is supported (not recommended) MODE2 and MODE3 are to be left NC (Section 9).

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