E·XFL



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 8-Core
Speed	1000MIPS
Connectivity	Configurable
Peripherals	-
Number of I/O	90
Program Memory Size	128KB (32K x 32)
Program Memory Type	SRAM
EEPROM Size	-
RAM Size	-
Voltage - Supply (Vcc/Vdd)	0.90V ~ 5.5V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	217-LFBGA
Supplier Device Package	217-FBGA (16x16)
Purchase URL	https://www.e-xfl.com/product-detail/xmos/xs1-a8a-128-fb217-i10

Email: info@E-XFL.COM

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

1 xCORE Multicore Microcontrollers

The XS1-A Series is a comprehensive range of 32-bit multicore microcontrollers that brings the low latency and timing determinism of the xCORE architecture to mainstream embedded applications. Unlike conventional microcontrollers, xCORE multicore microcontrollers execute multiple real-time tasks simultaneously and communicate between tasks using a high speed network. Because xCORE multicore microcontrollers are completely deterministic, you can write software to implement functions that traditionally require dedicated hardware.



Key features of the XS1-A8A-128-FB217 include:

- Tiles: Devices consist of one or more xCORE tiles. Each tile contains between four and eight 32-bit xCOREs with highly integrated I/O and on-chip memory.
- Logical cores Each logical core can execute tasks such as computational code, DSP code, control software (including logic decisions and executing a state machine) or software that handles I/O. Section 7.1
- ▶ **xTIME scheduler** The xTIME scheduler performs functions similar to an RTOS, in hardware. It services and synchronizes events in a core, so there is no requirement for interrupt handler routines. The xTIME scheduler triggers cores on events generated by hardware resources such as the I/O pins, communication channels and timers. Once triggered, a core runs independently and concurrently to other cores, until it pauses to wait for more events. Section 7.2

5 Example Application Diagram





9.1 Boot from SPI master

If set to boot from SPI master, the processor enables the four pins specified in Figure 11, 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 11: SPI master pins

X0D00 MISO Master In Slave Out (Data)	
AUDOU MISO Master in Slave Out (Data)	
X0D01 SS Slave Select	
X0D10 SCLK Clock	
X0D11 MOSI Master 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.

9.2 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.

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.

10 Memory

10.1 OTP

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

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 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.

10.2 SRAM

Each 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.

10.3 Deep Sleep Memory

The XS1-A8A-128-FB217 device includes 128 bytes of deep sleep memory for state storage during sleep mode. Deep sleep memory is volatile and if device input power is remove, the data will be lost.

11 Analog-to-Digital Converter

The device has a 12-bit 1MSample/second Successive Approximation Register (SAR) Analogue to Digital Converter (ADC). It has 8 input pins which are multiplexed into the ADC. The sampling of the ADC is controlled using the ADC_SAMPLE pin that should be wired to a GPIO pin, for example X0D24 (port 11). The sampling is triggered either by writing to the port, or by driving the pin externally. On each rising edge of the sample pin the ADC samples, holds and converts the data value from one of the analog input pins. Each of the 8 inputs can be enabled individually. Each of the enabled analog inputs is sampled in turn, on successive rising edges of the sample pin. The data is transmitted to the channel-end that the user configures during initialization of the ADC. Data is transmitted over the channel in individual packets, or in packets that contain multiple consecutive samples. The ADC uses an external reference voltage, nominally 3V3, which represents the full range of the ADC. The ADC configuration registers are documented in Appendix F.

The minimum latency for reading a value from the ADC into the xCORE register is shown in Figure 13:

-XMOS-

Figure 13:	Sample	Tile clock frequency	Start of packet	Subsequent samples
Minimum	32-bit	500 MHz	840 ns	710 ns
read sample	32-bit	400 MHz	870 ns	740 ns
from ADC to	16-bit	500 MHz	770 ns	640 ns
xCORE	16-bit	400 MHz	800 ns	670 ns

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 1 ms 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-A8A-128-FB217 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-A8A-128-FB217 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-A8A-128-FB217 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.

-XM()S

22

2. The 64-bit real-time counter reaches a value set by the program; or

When the chip is awake, the real-time counter counts the number of clock ticks on the oscillator. As such, the real-time counter will run at a fixed ratio, but synchronously with the 100 MHz timers on the xCORE Tile. When asleep, the real-time counter can be automatically switched to the 31,250 Hz silicon oscillator to save power (see Appendix H). To ensure that the real-time counter increases linearly over time, a programmable value is added to the counter on every 31,250 Hz clock-tick. This means that the clock will run at a granularity of 31,250 Hz but still maintain real-time in terms of the frequency of the main oscillator. If an 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.

Figure 15: Example trade-offs in oscillator selection

Clocks	used	Power	BOM	Accu	racy
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

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.



-XMOS[®]

XS1-A8A-128-FB217

17.9 External Oscillator Characteristics

Figure 32: Exter oscilla characte

xCORE

nal	Symbol	Parameter	MIN	ТҮР	MAX	UNITS	Notes
ıtor	F(EXT)	External Frequency			100	MHz	
eris-	V(IH)	Input high voltage	1.62		1.98	V	
tics	V(IL)	Input low voltage			0.4	V	

17.10 Power Consumption

	Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
	P(AWAKE)	Active Power for awake states (Speed Grade 10)	TBC	600	ТВС	mW	
Figure 33: CORE Tile		Active Power for awake states (Speed Grade 8)	TBC	480	ТВС	mW	
currents	P(SLEEP)	Power when asleep	TBC	500	TBC	μW	

17.11 Clock

	Symbol	Parameter	MIN	TYP	MAX	UNITS	Notes
Figure 34:	f(MAX)	Processor clock frequency (Speed Grade 10)			500	MHz	A
Figure 34: Clock		Processor clock frequency (Speed Grade 8)			400	MHz	A

A Assumes typical tile and I/O voltages with nominal activity.

17.12 Processor I/O AC Characteristics

Figure 35: I/O AC characteristics

Symbol	Parameter	MIN	ТҮР	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	

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.

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:

 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 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, \rightarrow ...), 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:

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

-XM()S

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	24-bit response	8-bit	8-bit	data	control-token
36	channel-end identifier	register number	size		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	24-bit response	8-bit	8-bit	control-token
37	channel-end identifier	register number	size	1

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



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	

-XMOS"

Figure 41: Summary



0x11: Debug SPC	Bits	Perm	Init	Description
	31:0	DRW		Value.

B.13 Debug SSP: 0x12

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

0x12: Debug SSP	Bits	Perm	Init	Description
	31:0	DRW		Value.

B.14 DGETREG operand 1: 0x13

The resource ID of the logical core whose state is to be read.

0x13: DGETREG	Bits	Perm	Init	Description
	31:8	RO	-	Reserved
operand 1	7:0	DRW		Thread number to be read

B.15 DGETREG operand 2: 0x14

Register number to be read by DGETREG

0x14: DGETREG operand 2

Bits	Perm	Init	Description
31:5	RO	-	Reserved
4:0	DRW		Register number to be read

B.16 Debug interrupt type: 0x15

Register that specifies what activated the debug interrupt.

·XMC

0x50 .. 0x53: Data watchpoint address 1

Data point ess 1	Bits	Perm	Init	Description		
	31:0	DRW		Value.		

B.23 Data watchpoint address 2: 0x60 .. 0x63

This set of registers contains the second address for the four data watchpoints.

0x60 .. 0x63: Data watchpoint address 2

a It	Bits	Perm	Init	Description
2	31:0	DRW		Value.

B.24 Data breakpoint control register: 0x70 .. 0x73

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

	Bits	Perm	Init	Description	
	31:24	RO	-	Reserved	
	23:16	DRW	0	A bit for each logical core in the tile allowing the breakpoint t be enabled individually for each logical core.	
	15:3	RO	-	Reserved	
	2	DRW	0	Set to 1 to enable breakpoints to be triggered on loads. Breakpoints always trigger on stores.	
3: a nt ol	1	DRW	0	By default, data watchpoints trigger if memory in the range [Address1Address2] is accessed (the range is inclusive of Address1 and Address2). If set to 1, data watchpoints trigger if memory outside the range (Address2Address1) is accessed (the range is exclusive of Address2 and Address1).	
er	0	DRW	0	When 1 the instruction breakpoint is enabled.	

0x70 .. 0x73: Data breakpoint control register

B.25 Resources breakpoint mask: 0x80 .. 0x83

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

-XMOS

0x80 .. 0x83: Resources breakpoint mask

nces point nask	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

es nt	Bits	Perm	Init	Description
ue	31:0	DRW		Value.

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

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

	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
0x9C 0x9F: Resources breakpoint control	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.
register	0	DRW	0	When 1 the instruction breakpoint is enabled.

-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	25 RO		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
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
: 	4:0	RW	0	The destination channel end on this node that packets received in static mode are forwarded to.

-XMOS[®]

0xA0 .. 0xA7 Static link configuration 0x18: ADC Control input pin 6

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.	

F.8 ADC Control input pin 7: 0x1C

Controls specific to ADC input pin 7.

0x1C: ADC Control input pin 7

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.	

-XMOS

F.9 ADC General Control: 0x20

General ADC control.

I Real time clock Configuration

The *Real time clock* is peripheral 5. The control registers are accessed using 32-bit reads and writes (use write_periph_32(device, 5, ...) and read_periph_32(device, \rightarrow 5, ...) for reads and writes).

	Number	Perm	Description
Figure 48: Summary	0x00	RW	Real time counter least significant 32 bits
	0x04	RW	Real time counter most significant 32 bits

I.1 Real time counter least significant 32 bits: 0x00

This registers contains the lower 32-bits of the real-time counter.

0x00: Real time counter least significant 32 bits					
	Bits	Perm	Init	Description	
	31:0	RO	0	Least significant 32 bits of real-time counter.	

I.2 Real time counter most significant 32 bits: 0x04

This registers contains the upper 32-bits of the real-time counter.



-XM()S

J Power control block Configuration

The *Power control block* is peripheral 6. The control registers are accessed using 32-bit reads and writes (use write_periph_32(device, 6, ...) and read_periph_32(\hookrightarrow device, 6, ...) for reads and writes).



Bits	Perm	Init	Description	
31:21	RO	-	Reserved	
20:16	RW	16	Log2 number of cycles to stay in this state: 0: 1 clock cycles 1: 2 clock cycles 2: 4 clock cycles 31: 2147483648 clock cycles	
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	0	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	0	Set to 1 to enable DCDC1 (core supply).	

0x1C: Power supply states whilst SLEEPING1

J.9 Power supply states whilst SLEEPING2: 0x20

-XMOS®

This register controls what state the power control block should be in when in the SLEEPING2 state. It also defines the time that the system shall stay in this state.

0x40: LDO5 level control

Bits	Perm	Init	Description
31:3	RO	-	Reserved
2:0	RW	pin	The required voltage in 100 mV steps: 0: 0.6V 1: 0.7V 2: 0.8V 6: 1.2V 7: 1.3V

88

