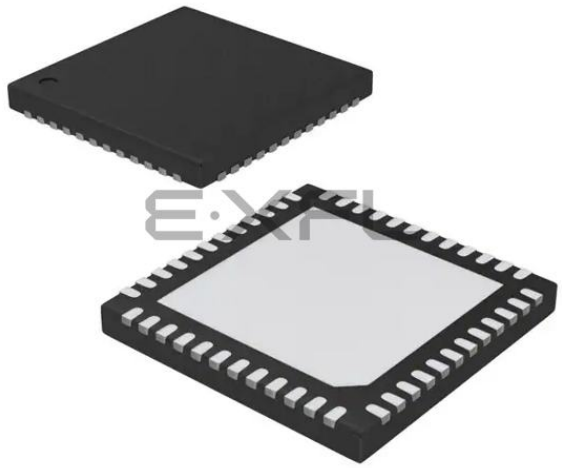
# XMOS - XUF208-128-QF48-C10 Datasheet



Welcome to E-XFL.COM

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | XCore |
| Core Size | 32-Bit 8-Core |
| Speed | 1000MIPS |
| Connectivity | USB |
| Peripherals | - |
| Number of I/O | 27 |
| Program Memory Size | 1MB (1M x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 128K x 8 |
| Voltage - Supply (Vcc/Vdd) | 0.95V ~ 3.6V |
| Data Converters | - |
| Oscillator Type | External |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 48-UFQFN Exposed Pad |
| Supplier Device Package | 48-UQFN (6x6) |
| Purchase URL | https://www.e-xfl.com/product-detail/xmos/xuf208-128-qf48-c10 |

- Ports The I/O pins are connected to the processing cores by Hardware Response ports. The port logic can drive its pins high and low, or it can sample the value on its pins optionally waiting for a particular condition. Section 6.3
- Clock blocks xCORE devices include a set of programmable clock blocks that can be used to govern the rate at which ports execute. Section 6.4
- Memory Each xCORE Tile integrates a bank of SRAM for instructions and data, and a block of one-time programmable (OTP) memory that can be configured for system wide security features. Section 9
- PLL The PLL is used to create a high-speed processor clock given a low speed external oscillator. Section 7
- USB The USB PHY provides High-Speed and Full-Speed, device, host, and on-the-go functionality. Data is communicated through ports on the digital node. A library is provided to implement USB device functionality. Section 10
- Flash The device has a built-in 2MBflash. Section 8
- JTAG The JTAG module can be used for loading programs, boundary scan testing, in-circuit source-level debugging and programming the OTP memory. Section 11

## 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.

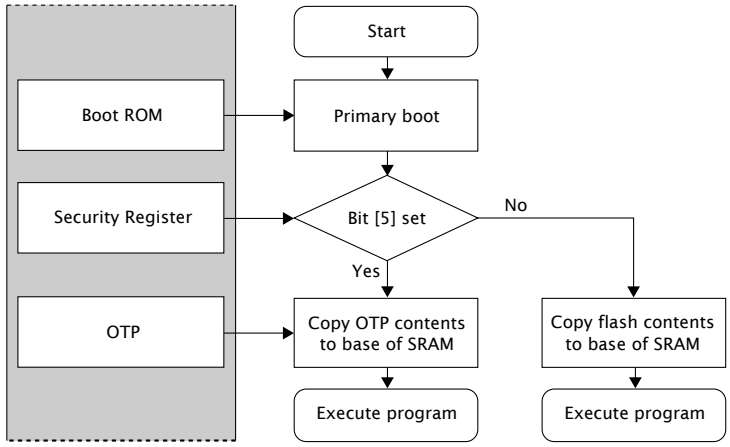



Figure 9: Boot procedure

802.3); the CRC register is initialized with 0xFFFFFFFF and the residue is inverted to produce the CRC.

### 8.1 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 su□cient for providing strong IP security.

## 9 Memory

### 9.1 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 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`.

### 9.2 SRAM

The xCORE Tile integrates a single 128KB SRAM 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

The PLL_AVDD supply should be separated from the other noisier supplies on the board. The PLL requires a very clean power supply, and a low pass filter (for example, a 4.7 Ω resistor and 100 nF multi-layer ceramic capacitor) is recommended on this pin.

The following ground pins are provided:

- GND for all supplies

All ground pins must be connected directly to the board ground.

The VDD and VDDIO supplies should be decoupled close to the chip by several 100 nF low inductance multi-layer ceramic capacitors between the supplies and GND (for example, 100nF 0402 for each supply pin). The ground side of the decoupling capacitors should have as short a path back to the GND pins as possible. A bulk decoupling capacitor of at least 10 uF should be placed on each of these supplies.

RST_N is an active-low asynchronous-assertion global reset signal. Following a reset, the PLL re-establishes lock after which the device boots up according to the boot mode (*see* §8). RST_N and must be asserted low during and after power up for 100 ns.

## 12.1 USB connections

USB_VBUS should be connected to the VBUS pin of the USB connector. A 2.2 uF capacitor to ground is required on the VBUS pin. A ferrite bead may be used to reduce HF noise.

For self-powered systems, a bleeder resistor may be required to stop VBUS from floating when no USB cable is attached.

USB_DP and USB_DN should be connected to the USB connector. USB_ID does not need to be connected.

## 12.2 USB signal routing and placement

The USB_DP and USB_DN lines are the positive and negative data polarities of a high speed USB signal respectively. Their high-speed di□erential nature implies that they must be coupled and properly isolated. The board design must ensure that the board traces for USB_DP and USB_DN are tightly matched. In addition, according to the USB 2.0 specification, the USB_DP and USB_DN di□erential impedance must be 90 Ω.

### 12.2.1 General routing and placement guidelines

The following guidelines will help to avoid signal quality and EMI problems on high speed USB designs. They relate to a four-layer (Signal, GND, Power, Signal) PCB.

For best results, most of the routing should be done on the top layer (assuming the USB connector and XS2-UF8A-128-QF48 are on the top layer) closest to GND.

Figure 15: USB trace separation showing a low speed signal, two di□erential pairs and a high-speed clock

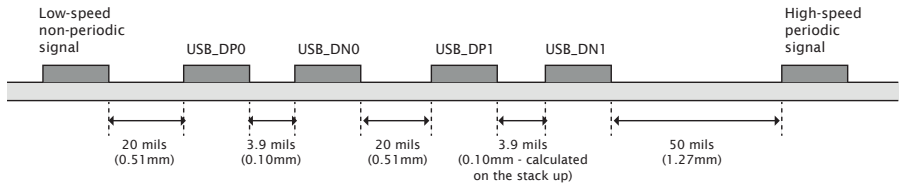
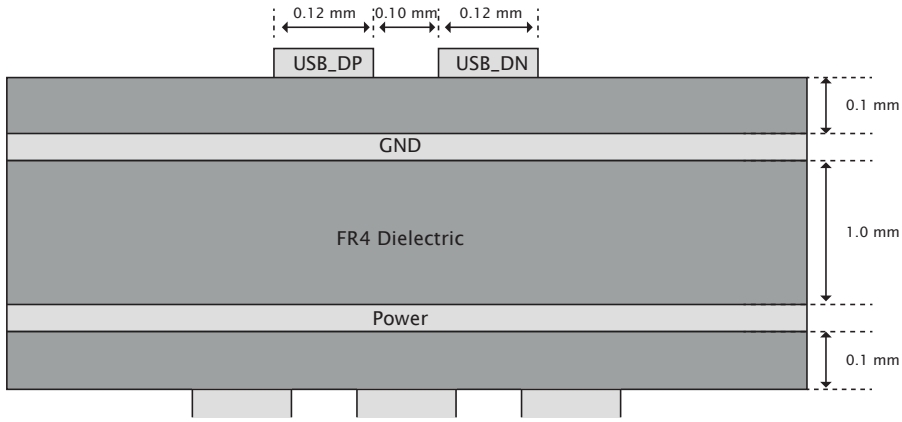



Figure 16: Example USB board stack




Reference planes should be below the transmission lines in order to maintain control of the trace impedance.

We recommend that the high-speed clock and high-speed USB di□erential pairs are routed first before any other routing. When routing high speed USB signals, the following guidelines should be followed:

- High speed di□erential pairs should be routed together.
- High-speed USB signal pair traces should be trace-length matched. Maximum trace-length mismatch should be no greater than 4mm.
- Ensure that high speed signals (clocks, USB di□erential pairs) are routed as far away from o□-board connectors as possible.
- High-speed clock and periodic signal traces that run parallel should be at least 1.27mm away from USB_DP/USB_DN (see Figure 15).
- Low-speed and non-periodic signal traces that run parallel should be at least 0.5mm away from USB_DP/USB_DN (see Figure 15).
- Route high speed USB signals on the top of the PCB wherever possible.

## 13.7 xCORE Tile I/O AC Characteristics

Figure 24: 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 | |

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.

## 13.8 xConnect Link Performance

Figure 25: 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.

## 13.9 JTAG Timing

Figure 26: 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.

# 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 | RW | 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 |
| 0x0C | RO | RAM size |
| 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 30: Summary

0x02: xCORE Tile control

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:26 | RO | - | Reserved |
| 25:18 | RW | 0 | RGMII TX data delay value (in PLL output cycle increments) |
| 17:9 | RW | 0 | RGMII TX clock divider value. TX clk rises when counter (clocked by PLL output) reaches this value and falls when counter reaches (value»1). Value programmed into this field should be actual divide value required minus 1 |
| 8 | RW | 0 | Enable RGMII interface periph ports |
| 7:6 | RO | - | Reserved |
| 5 | RW | 0 | Select the dynamic mode (1) for the clock divider when the clock divider is enabled. In dynamic mode the clock divider is only activated when all active threads are paused. In static mode the clock divider is always enabled. |
| 4 | RW | 0 | Enable the clock divider. This divides the output of the PLL to facilitate one of the low power modes. |
| 3 | RO | - | Reserved |
| 2 | RW | | Select between UTMI (1) and ULPI (0) mode. |
| 1 | RW | | Enable the ULPI Hardware support module |
| 0 | RO | - | Reserved |

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

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

0x03: xCORE Tile boot status

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:24 | RO | - | Reserved |
| 23:16 | RO | | Processor number. |
| 15:9 | RO | - | Reserved |
| 8 | RO | | Overwrite BOOT_MODE. |
| 7:6 | RO | - | Reserved |
| 5 | RO | | Indicates if core1 has been powered o□ |
| 4 | RO | | Cause the ROM to not poll the OTP for correct read levels |
| 3 | RO | | Boot ROM boots from RAM |
| 2 | RO | | Boot ROM boots from JTAG |
| 1:0 | RO | | The boot PLL mode pin value. |

## B.5 Security configuration: 0x05

Copy of the security register as read from OTP.

0x05: Security configuration

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

## B.6 Ring Oscillator Control: 0x06

There are four free-running oscillators that clock four counters. The oscillators can be started and stopped using this register. The counters should only be read when the ring oscillator has been stopped for at least 10 core clock cycles (this can be achieved by inserting two nop instructions between the SETPS and GETPS). The counter values can be read using four subsequent registers. The ring oscillators are asynchronous to the xCORE tile clock and can be used as a source of random bits.

0x06: Ring Oscillator Control

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:2 | RO | - | Reserved |
| 1 | RW | 0 | Core ring oscillator enable. |
| 0 | RW | 0 | Peripheral ring oscillator enable. |

## B.7 Ring Oscillator Value: 0x07

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

0x30 .. 0x33: Instruction breakpoint address

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:0 | DRW | | Value. |

## B.22 Instruction breakpoint control: 0x40 .. 0x43

This register controls which logical cores may take an instruction breakpoint, and under which condition.

0x40 .. 0x43: Instruction breakpoint control

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:24 | RO | - | Reserved |
| 23:16 | DRW | 0 | A bit for each thread in the machine allowing the breakpoint to be enabled individually for each thread. |
| 15:2 | RO | - | Reserved |
| 1 | DRW | 0 | When 0 break when PC == IBREAK_ADDR. When 1 = break when PC != IBREAK_ADDR. |
| 0 | DRW | 0 | When 1 the instruction breakpoint is enabled. |

## B.23 Data watchpoint address 1: 0x50 .. 0x53

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

0x50 .. 0x53: Data watchpoint address 1

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:0 | DRW | | Value. |

## B.24 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

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:0 | DRW | | Value. |

| | Bits | Perm | Init | Description |
|---|---|---|---|---|
| 0x9C .. 0x9F: Resources breakpoint control register | 31:24 | RO | - | Reserved |
| | 23:16 | DRW | 0 | A bit for each thread in the machine allowing the breakpoint to be enabled individually for each thread. |
| | 15:2 | RO | - | Reserved |
| | 1 | DRW | 0 | When 0 break when condition A is met. When 1 = break when condition B is met. |
| | 0 | DRW | 0 | When 1 the instruction breakpoint is enabled. |

# C Tile Configuration

The xCORE Tile control registers can be accessed using configuration reads and writes (use `write_tile_config_reg(tileref, ...)` and `read_tile_config_reg(tileref, ...)` for reads and writes).

Figure 31: Summary

| Number | Perm | Description |
|---|---|---|
| 0x00 | CRO | Device identification |
| 0x01 | CRO | xCORE Tile description 1 |
| 0x02 | CRO | xCORE Tile description 2 |
| 0x04 | CRW | Control PSwitch permissions to debug registers |
| 0x05 | CRW | Cause debug interrupts |
| 0x06 | CRW | xCORE Tile clock divider |
| 0x07 | CRO | Security configuration |
| 0x20 .. 0x27 | CRW | Debug scratch |
| 0x40 | CRO | PC of logical core 0 |
| 0x41 | CRO | PC of logical core 1 |
| 0x42 | CRO | PC of logical core 2 |
| 0x43 | CRO | PC of logical core 3 |
| 0x44 | CRO | PC of logical core 4 |
| 0x45 | CRO | PC of logical core 5 |
| 0x46 | CRO | PC of logical core 6 |
| 0x47 | CRO | PC of logical core 7 |
| 0x60 | CRO | SR of logical core 0 |
| 0x61 | CRO | SR of logical core 1 |
| 0x62 | CRO | SR of logical core 2 |
| 0x63 | CRO | SR of logical core 3 |
| 0x64 | CRO | SR of logical core 4 |
| 0x65 | CRO | SR of logical core 5 |
| 0x66 | CRO | SR of logical core 6 |
| 0x67 | CRO | SR of logical core 7 |

## C.1 Device identification: 0x00

This register identifies the xCORE Tile

## D.2 System switch description: 0x01

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

0x01: System switch description

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:24 | RO | - | Reserved |
| 23:16 | RO | | Number of SLinks on the SSwitch. |
| 15:8 | RO | | Number of processors on the SSwitch. |
| 7:0 | RO | | Number of processors on the device. |

## 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 | RW | 0 | 0 = SSCTL registers have write access. 1 = SSCTL registers can not be written to. |
| 30:9 | RO | - | Reserved |
| 8 | RW | 0 | 0 = PLL_CTL_REG has write access. 1 = PLL_CTL_REG can not be written to. |
| 7:1 | RO | - | Reserved |
| 0 | RW | 0 | 0 = 2-byte headers, 1 = 1-byte headers (reset as 0). |

## 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 ID of this node. |

## 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 | RW | | If set to 1, the chip will not be reset |
| 30 | RW | | If set to 1, the chip will not wait for the PLL to re-lock. Only use this if a gradual change is made to the PLL |
| 29 | DW | | If set to 1, set the PLL to be bypassed |
| 28 | DW | | If set to 1, set the boot mode to boot from JTAG |
| 27:26 | RO | - | Reserved |
| 25:23 | RW | | Output divider value range from 1 (8'h0) to 250 (8'hF9). P value. |
| 22:21 | RO | - | Reserved |
| 20:8 | RW | | Feedback multiplication ratio, range from 1 (8'h0) to 255 (8'hFE). M value. |
| 7 | RO | - | Reserved |
| 6:0 | RW | | Oscilator input divider value range from 1 (8'h0) to 32 (8'h0F). N value. |

## 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 | SSwitch clock generation |

## 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 | Software ref. clock divider |

0x1F: Debug source

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:5 | RO | - | Reserved |
| 4 | RW | | Reserved. |
| 3:2 | RO | - | Reserved |
| 1 | RW | | If set, XCore1 is the source of last GlobalDebug event. |
| 0 | RW | | If set, XCore0 is the source of last GlobalDebug event. |

## D.15 Link status, direction, and network: 0x20 .. 0x28

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 0..7.

0x20 .. 0x28: Link status, direction, and network

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:26 | RO | - | Reserved |
| 25:24 | RO | | Identify the SRC_TARGET type 0 - SLink, 1 - PLink, 2 - SSCTL, 3 - Undefine. |
| 23:16 | RO | | When the link is in use, this is the destination link number to which all packets are sent. |
| 15:12 | RO | - | Reserved |
| 11:8 | RW | 0 | The direction that this link operates in. |
| 7:6 | RO | - | Reserved |
| 5:4 | RW | 0 | Determines the network to which this link belongs, reset as 0. |
| 3 | RO | - | Reserved |
| 2 | RO | | 1 when the current packet is considered junk and will be thrown away. |
| 1 | RO | | 1 when the dest side of the link is in use. |
| 0 | RO | | 1 when the source side of the link is in use. |

## D.16 PLink status and network: 0x40 .. 0x47

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

## 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 System clock frequency: 0x51

0x51: System clock frequency

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:7 | RO | - | Reserved |
| 6:0 | RW | 25 | Oscillator clock frequency in MHz rounded up to the nearest integer value. Only values between 5 and 100 MHz are valid - writes outside this range are ignored and will be NACKed. This field must be set on start up of the device and any time that the input oscillator clock frequency is changed. It must contain the system clock frequency in MHz rounded up to the nearest integer value. |

## E.5 Link Control and Status: 0x80

0x80: Link Control and Status

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:28 | RO | - | Reserved |
| 27 | RO | | Rx bu□er overflow or illegal token encoding received. |
| 26 | RO | 0 | This end of the xlink has issued credit to allow the remote end to transmit |
| 25 | RO | 0 | This end of the xlink has credit to allow it to transmit. |
| 24 | WO | | Clear this end of the xlink's credit and issue a HELLO token. |
| 23 | WO | | Reset the receiver. The next symbol that is detected will be the first symbol in a token. |
| 22 | RO | - | Reserved |
| 21:11 | RW | 1 | Specify min. number of idle system clocks between two continuous symbols witin a transmit token -1. |
| 10:0 | RW | 1 | Specify min. number of idle system clocks between two continuous transmit tokens -1. |

0x10: UIFM on-the-go control

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:8 | RO | - | Reserved |
| 7 | RW | 0 | Set to 1 to switch UIFM to EXTVBUSIND mode. |
| 6 | RW | 0 | Set to 1 to switch UIFM to DRVVBUSEXT mode. |
| 5 | RO | - | Reserved |
| 4 | RW | 0 | Set to 1 to switch UIFM to UTMI+ CHRGVBUS mode. |
| 3 | RW | 0 | Set to 1 to switch UIFM to UTMI+ DISCHRGVBUS mode. |
| 2 | RW | 0 | Set to 1 to switch UIFM to UTMI+ DMPULLDOWN mode. |
| 1 | RW | 0 | Set to 1 to switch UIFM to UTMI+ DPPULLDOWN mode. |
| 0 | RW | 0 | Set to 1 to switch UIFM to IDPULLUP mode. |

## F.6 UIFM on-the-go flags: 0x14

Status flags used for on-the-go negotiation

0x14: UIFM on-the-go flags

| Bits | Perm | Init | Description |
|---|---|---|---|
| 31:6 | RO | - | Reserved |
| 5 | RO | 0 | Value of UTMI+ Bvalid flag. |
| 4 | RO | 0 | Value of UTMI+ IDGND flag. |
| 3 | RO | 0 | Value of UTMI+ HOSTDIS flag. |
| 2 | RO | 0 | Value of UTMI+ VBUSVLD flag. |
| 1 | RO | 0 | Value of UTMI+ SESSVLD flag. |
| 0 | RO | 0 | Value of UTMI+ SESSEND flag. |

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

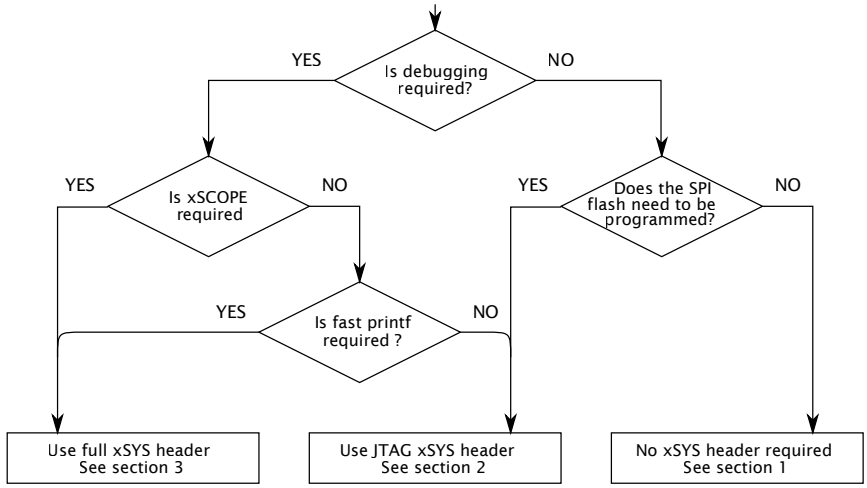



Figure 35: Decision diagram for the xSYS header

## 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
- TDO to pin 13 of the xSYS header

# I PCB Layout Design Check List

☑ This section is a checklist for use by PCB designers using the XS2-UF8A-128-QF48. 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 12.4).

☐ 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

☐ The decoupling capacitors are all placed close to a supply pin (Section 12).

☐ The decoupling capacitors are spaced around the device (Section 12).

☐ 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 12).