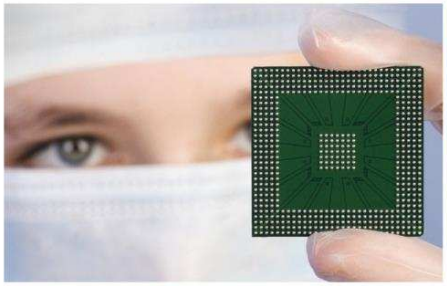
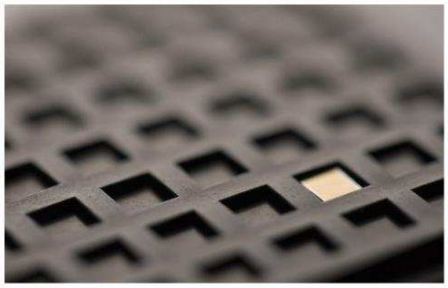
# E64G401 EPIPHANY™ 64-CORE MICROPROCESSOR Datasheet



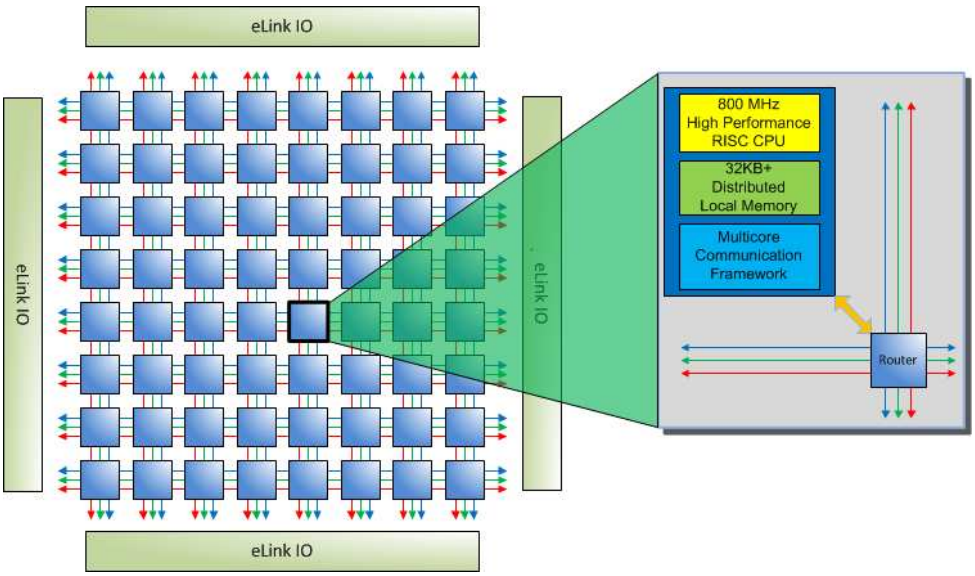


**Epiphany - The world leader in microprocessor energy efficiency**

The Epiphany is a scalable multicore architecture with up to 4,095 processors sharing a common 32 bit memory space. The Epiphany combines fully-featured floating point C/C++ programmable RISC processors, a high bandwidth distributed memory system, a low latency Network-On-Chip, and low overhead off-chip IO to bring an unprecedented level of processing to power constrained systems.

**E16G401 Feature Summary**

- 64 high performance RISC CPU cores
- C/C++ and OpenCL programmable
- 32-bit IEEE floating point support
- 2MB on-chip distributed shared memory
- 128 independent DMA channels
- Up to 800MHz operating frequency
- 102 GFLOPS peak performance
- 1.6 TB/s local memory bandwidth
- 102 GB/s Network-On-Chip bisection bandwidth
- Four 1.8 GB/s off-chip sub LVDS interfaces
- 1.875ns network per-hop latency
- <2 Watt maximum chip power consumption
- 324-ball 15×15mm 0.8mm pitch BGA



| Revision | Changes |
|---|---|
| **3.13.06.14** | Initial Revision |
| **14.03.11** | General: Added electrical specs disclaimer notes<br>Section 3.5: Added section on unused pins<br>Section 6.4: Removed unused signals in pin mapping<br>Section 6.4:Corrected incorrect pin map table<br>Section 2.2: Added Link Debug, Chip Start, and Chip Halt register.<br>Section 2.2: Fixed the link register offsets.<br>Section 2: Added official register names to be used by SDK<br>Added IO register access explanation<br>Section3: Added eLink details<br>Section 5: Added AC/DC timing specifications<br>Added power consumption measurements<br>Section 6: Added thermal data |
| | |
| | |
| | |
| | |
| | |
| | |

## Table of Contents

# Preface

This document describes the *E64G401* chip product. The document is written for system designers with a fundamental understanding of processor architectures and chip specifications.

## Related Documents

- *Epiphany Architecture Reference*: The complete reference for the Epiphany multi-core computer architecture.
- *Epiphany SDK Reference*: The development tools and run-time library for the Epiphany architecture

# 1 Introduction

## *1.1 Overview*

The E64G401 is a 64-core System-On-Chip implemented in a 65nm based on the 3rd generation of the Epiphany multicore IP. The Epiphany™ architecture is a scalable, distributed-shared-memory computing fabric comprised of a 2D array of processor nodes connected by a low-latency mesh network-on-chip. Here is a brief summary of the key components of the E64G401:

### Processor:

The E64G401 includes 64 superscalar floating point RISC CPUs (eCore), each one capable of two floating point operations per clock cycle and one integer calculation per clock cycle. The CPU has an efficient general-purpose instruction set that excels at compute intensive applications while being efficiently programmable in C/C++.

### Memory System:

The Epiphany memory architecture is based on a flat shared memory map in which each compute node has up to 1MB of local memory as a unique addressable slice of the total 32-bit address space. A processor can access its own local memory and other processors' memory through regular load/store instructions. The local memory system is comprised of 4 separate sub-banks, allowing for simultaneous memory accesses by the instruction fetch engine, local load-store instructions, and by memory transactions initiated by the DMA engine other processors within system.

### Network-On-Chip:

The Epiphany Network-on-Chip (eMesh) is a 2D mesh network that handles all on-chip and off-chip communication. The eMesh network uses atomic 32-bit memory transactions and operates without the need for any special programming. The network consists of three separate and orthogonal mesh structures, each serving different types of transaction traffic: one network for on-chip write traffic, one network for off chip write traffic, and one network for all read traffic.

### Off-Chip IO:

The eMesh network and memory architecture extends off-chip using source synchronous dual data rate LVDS links ("elinks"). Each E64G401 has 4 independent off-chip elinks, one in each physical

direction (north, east, west and south). The off chip links allows for glueless connection of multiple E64G401chips on a board and for interfacing to an FPGA.

For more detailed information about the Epiphany architecture, please refer to the *Epiphany Architecture Reference Manual.*

## 1.2 System Examples

The E64G401 product can be used in a number of different system configurations, some of which are shown in this section.

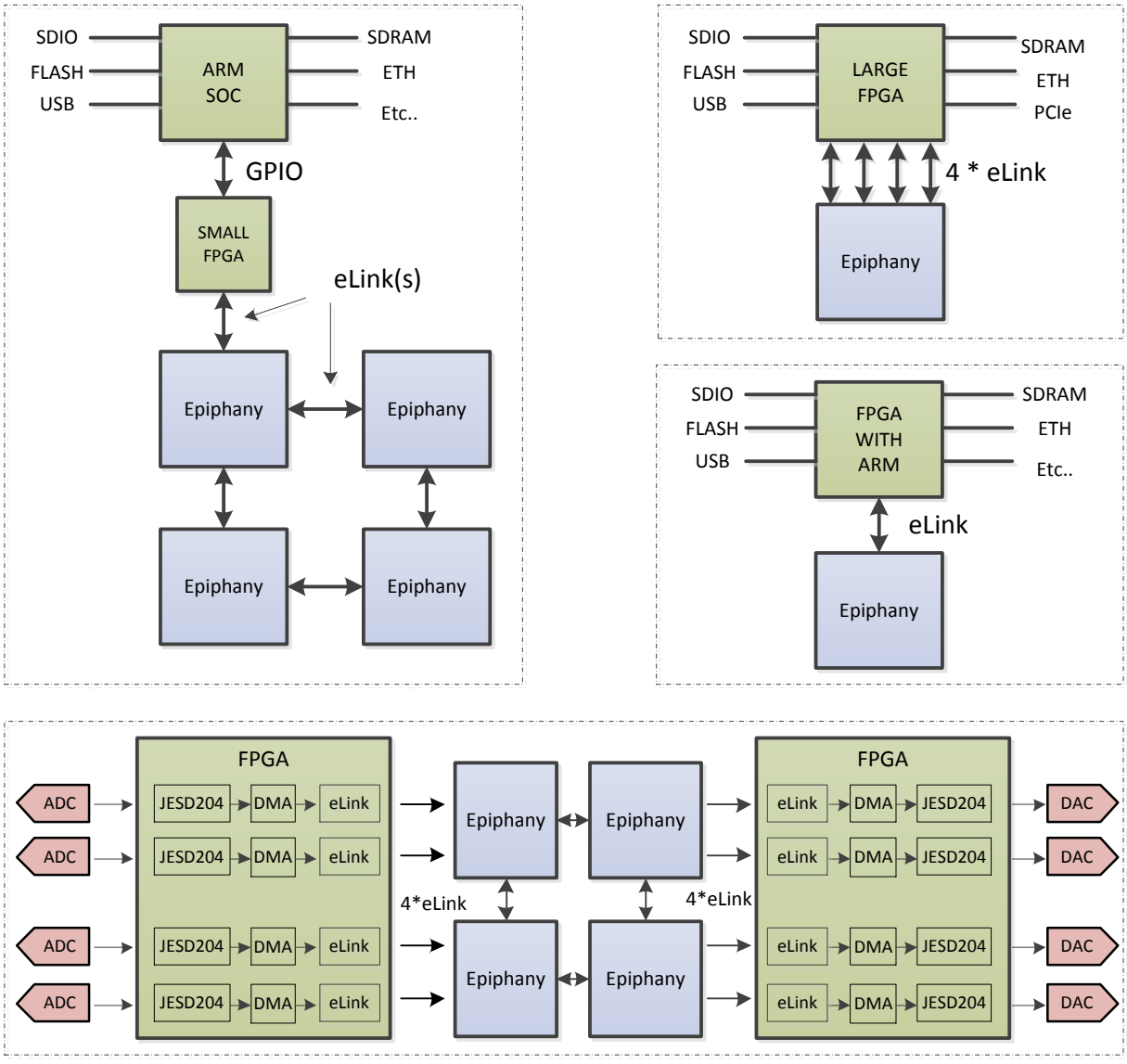



**Figure 1: Epiphany System Architecture**

## 1.3 Applications

The following list shows some applications for which the E64G401 is particularly well suited:

**Consumer:**

- Smart-phones and tablet application acceleration
- High end audio
- Computational photography
- Speech Recognition
- Face detection/recognition

**Computing Infrastructure:**

- Super Computers
- Big Data Analytics
- Software Defined Networking

**Mil/Aero:**

- Radar/Sonar
- Extremely Large Sensor Imaging
- Hyperspectral Imaging
- Military Radios

**Medical:**

- Ultrasound
- CT

**Communication:**

- Communication test-bed
- Software defined radio
- Adaptive Pre-distortion

**Embedded Vision**

- Machine Vision
- Autonomous Robots/Navigation
- Automotive Safety

**Other:**

- Compression
- Security Cameras
- Video Transcoding

# 2 Memory Architecture

## 2.1 Global Memory Map

The memory map configuration of the E64G401 within the 32 bit memory map is controlled with the ROWID[2:0] and COLID[2:0] chip input pins. The ROWID and COLID chip pins are sampled on the rising edge of RESET_N and are used to set the offset of the core's memory map. Figure 2 shows the distributed memory scheme of the Epiphany architecture. Table 1 shows the distribution of the SRAM within a specific E64G401 chip without the specific chip-ID offset. The complete core memory map of the local cores can be found in the Epiphany Architecture Reference Manual.

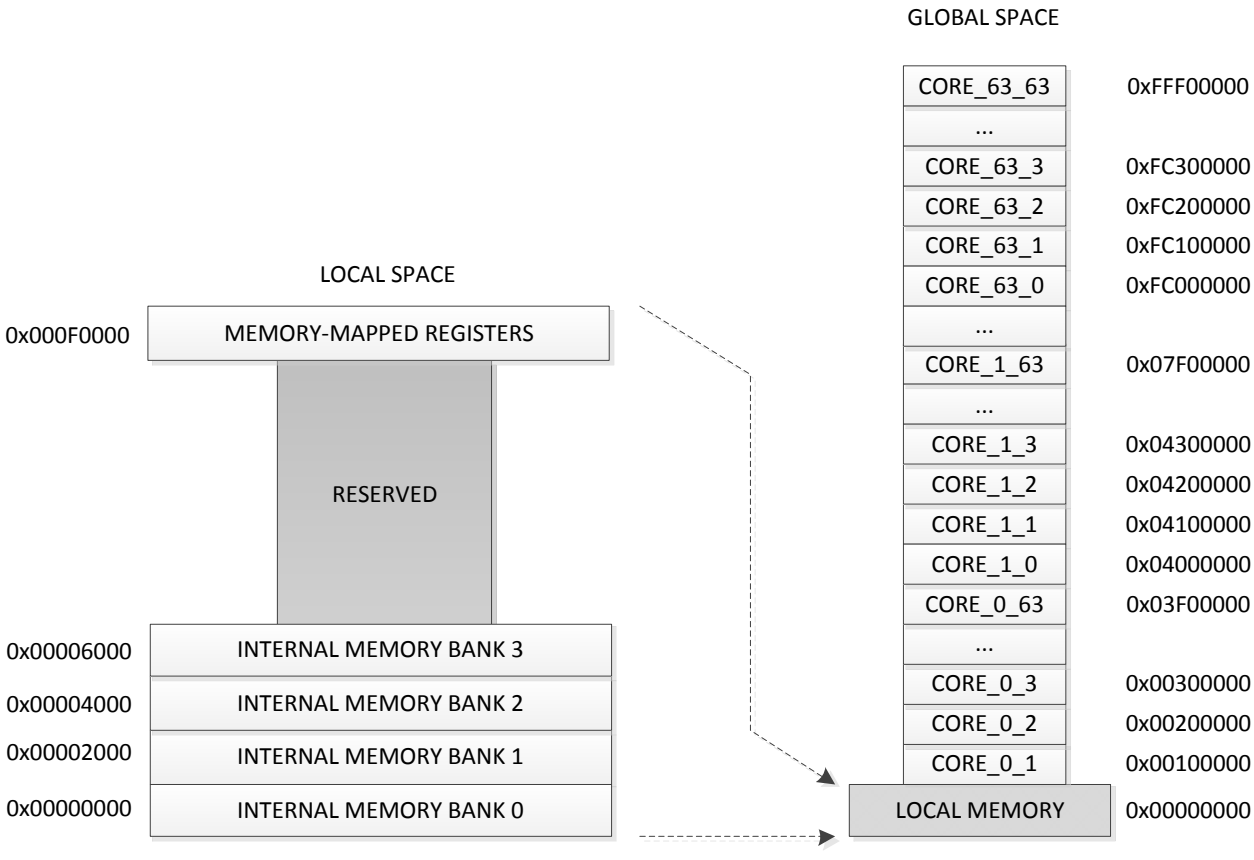



**Figure 2: Epiphany Memory Architecture**

| Chip Core Number | Start Address | End Address | Size |
|---|---|---|---|
| **(0,0)** | 00000000 | 00007FFF | 32KB |
| **(0,1)** | 00100000 | 00107FFF | 32KB |
| **(0,2)** | 00200000 | 00207FFF | 32KB |
| **(0,3)** | 00300000 | 00307FFF | 32KB |
| **(1,0)** | 04000000 | 04007FFF | 32KB |
| **(1,1)** | 04100000 | 04107FFF | 32KB |
| **(1,2)** | 04200000 | 04207FFF | 32KB |
| **(1,3)** | 04300000 | 04307FFF | 32KB |
| **(2,0)** | 08000000 | 08007FFF | 32KB |
| **(2,1)** | 08100000 | 08107FFF | 32KB |
| **(2,2)** | 08200000 | 08207FFF | 32KB |
| **(2,3)** | 08300000 | 08307FFF | 32KB |
| **(3,0)** | 0C000000 | 0C007FFF | 32KB |
| **(3,1)** | 0C100000 | 0C107FFF | 32KB |
| **(3,2)** | 0C200000 | 0C207FFF | 32KB |
| **(3,3)** | 0C300000 | 0C307FFF | 32KB |

**Table 1: Relative Chip Memory Map**

## 2.2 Memory Mapped Registers

The E64G401 has a set of chip IO registers used to configure the operating mode of the product. The memory locations of these configuration registers and their respective functions are shown in Table 2. These registers are cleared by asserting the chip RESET_N pin. The address entries below are relative to the ROWID and COLID chip address settings. In the table, the link registers additionally have an offset that is dependent on the link in question as follows:

- North link offset: **0x00600000** (Note: different from E16G301!)
- East link offset: **0x08700000** (Note: different from E16G301!)
- South link offset: **0x1c200000** (Note: different from E16G301!)
- West link offset: **0x08400000** (Note: different from E16G301!)

| Register Name | Register Address | Bits | Description |
|---|---|---|---|
| **ELINKCLKSO**<br>**ELINKCLKNO**<br>**ELINKCLKEA**<br>**ELINKCLKWE** | 0xF0300 + link offset | [3:0] | LCLK Transmit Frequency control<br>0=Divide cclk by 2<br>1=Divide cclk by 4<br>2=Divide cclk by 8 |
| **ELINKTXSO**<br>**ELINKTXNO**<br>**ELINKTXEA**<br>**ELINKTXWE** | 0xF0304 + link offset | [11:0] | Transmitter low power mode<br>0xFFF=turned off<br>0x000=turned on |
| **ELINKRXSO**<br>**ELINKRXNO**<br>**ELINKRXEA**<br>**ELINKRXWE** | 0xF0308 + link offset | [11:0] | Receiver lo power mode<br>0xFFF=turned off<br>0x000=turned on |
| **ELINKDEBUGSO**<br>**ELINKDEBUGNO**<br>**ELINKDEBUGEA**<br>**ELINKDEBUGWE** | 0xF0328 + link offset | [0]<br>[1]<br>[13:2]<br>[14] | Set constant on link TX<br>Loopback RX to TX<br>Constant to drive on TX<br>Force a chip transaction match on RX transactions |
| **IOFLAG** | 0x006F0318 | [5:0] | 0=Sets MONITOR pin to 0<br>1=Sets MONITOR pin to 1 |
| **CHIPRESET** | 0x083F0324 | | Writing to register creates a three clock cycle long pulse that resets the rest of the chip. |
| **CHIPSYNC** | 0x083F031C | | Writing to register creates a chip wide “SYNC” interrupt |
| **CHIPHALT** | 0x083F0320 | | Writing to register puts all cores in the HALT debug state. |

**Table 2: Memory Mapped Registers**

In order to write to these memory mapped registers, the store transaction must be configured with a special control mode that allows the transaction to bypass the regular eMesh routing protocol. The special routing mode is controlled through bits [15:12] of the CONFIG register of the core initiating the write transaction to the IO registers and should be set according to Table 3.

| Register Name | CONFIG[15:12] |
|---|---|
| **NORTH ELINK REGISTERS** | 0001 |
| **EAST ELINK REGISTERS** | 0101 |
| **SOUTH ELINK REGISTERS** | 1000 |
| **WEST ELINK REGISTERS** | 1101 |
| **IOFLAG, CHIPRESET, CHIPSYNC, CHIPHALT** | 1101 |

**Table 3: CONFIG Register Routing Mode Selection**

The CONFIG[15:12] register bits of a processor node are added to each outgoing on-chip write transactions as a CTRLMODE tag, overriding the default address matching algorithm of the cMesh. By default, when a write transaction arrives at a processor node, the write transaction is accepted into the node if the ROWID and COLID of the write transaction matches the node's coordinates exactly. By specifying a different matching direction in CONFIG[15:12], a matching write transaction can be directed to the south, north, east, or west instead of into the processor. To return the routing behavior to normal mode, CONFIG[15:12] should be reset to 0000. The safe method of writing to chip level IO register should use the following tightly coupled write sequence:

1. Set the CONFIG[15:12] appropriately according to Table 3 using a MOVTS instruction.
2. Write to the chip level register address as specified in Table 2
3. Reset CONFIG[15:12] to 0000 using a MOVTS instruction.

# 3 I/O Interfaces

## 3.1 Overview

The E64G401 includes the following basic signal groups:

- 1 core supply (VDD)
- 1 IO supply (DVDD)
- 1 differential input clock signal
- 1 single ended reset signal
- 8 single ended configuration strap pins
- 24 differential pairs for each one of the four different links (east/west/north/south)

NOTE: Input pins do NOT have on-chip pull-down resistors. To avoid erratic behavior, the input clock, reset and all eLink frame signals should not be left floating.

## 3.2 Supplies

The chip needs two separate and independent supplies for proper operation.

| Signal Name | Signal Description | Direction | Signaling Type |
|---|---|---|---|
| **DVDD** | IO Supply | Input | Supply |
| **VDD** | Core Supply | Input | Supply |
| **VSS** | Common Ground | Input | Supply |

**Table 4: Chip Supplies**

## 3.3 Reset and Clock

The E64G401 does not have an on-chip PLL and instead receives a high speed LVDS clock directly from a differential LVDS input signal.

| Signal Name | Signal Description | Direction | Signaling Type |
|---|---|---|---|
| **RESET_N** | Active Low Reset | Input | LVCMOS |
| **RXI_WE_CCLK_{P,N}** | Chip Clock Input | Input | LVDS |

**Table 5: Chip Clock and Rest**

NOTE: For correct reset sequencing, the RXI_WE_CCLK_{P/N} signal should NOT be toggling during the rising edge of RESET_N.

## 3.4 Monitor Signals

The flag pin is a general purpose output pin that can be connected to an LED or routed to an FPGA or controller to be used as an interrupt or indicator. The pin is controlled by writing a 1 or 0 to the "Monitor Register" as described in Chapter 2.2.

| Signal Name | Signal Description | Direction | Signaling Type |
|---|---|---|---|
| **FLAG** | Monitor Signal | Output | LVCMOS |
| **MVDD** | On-chip core voltage sensing pin | Output | Analog |
| **MVSS** | On-chip ground voltage sensing pin | Output | Analog |

**Table 6: Monitor Signals**

## 3.5 Unused Input Signals

The following signal pairs should have the P/N pair pulled to a safe constant differential value. (i.e. if the _N signal is pulled low then the _P signal should be pulled high).

| Signal Name | Signal Description | Direction | Signaling Type |
|---|---|---|---|
| **RXI_EA_CCLK_{P,N}** | Unimplemented signal | Input | LVDS |
| **RXI_SO_CCLK_{P,N}** | Unimplemented signal | Input | LVDS |
| **RXI_NO_CCLK_{P,N}** | Unimplemented signal | Input | LVDS |
| **COLID[3]** | Unimplemented signal | Input | LVDS |
| **ROWID[3]** | Unimplemented signal | Input | LVDS |

**Table 7: Unused Signals**

## 3.6 Chip Coordinate Signals

There are eight coordinate signals that are sampled at the rising edge of RESET_N to set the coordinate ID of the E64G401. The COLID[2:0] and ROWID[2:0] set bits [31:29] and [25:23] respectively of the chip's address map. The bits are needed to differentiate between chips in systems containing multiple E64G401 chips. Figure 3 shows an example system with four E16G401 chips with different COLID and ROWID input pin values.

| Signal Name | Signal Description | Direction | Signaling Type |
|---|---|---|---|
| **COLID[0]** | Bit[29] of chip address map | Input | LVCMOS |
| **COLID[1]** | Bit[30] of chip address map | Input | LVCMOS |
| **COLID[2]** | Bit[31] of chip address map | Input | LVCMOS |
| **ROWID[0]** | Bit[23] of chip address map | Input | LVCMOS |
| **ROWID[1]** | Bit[24] of chip address map | Input | LVCMOS |
| **ROWID[2]** | Bit[25] of chip address map | Input | LVCMOS |

**Table 8: Chip Coordinate Settings**

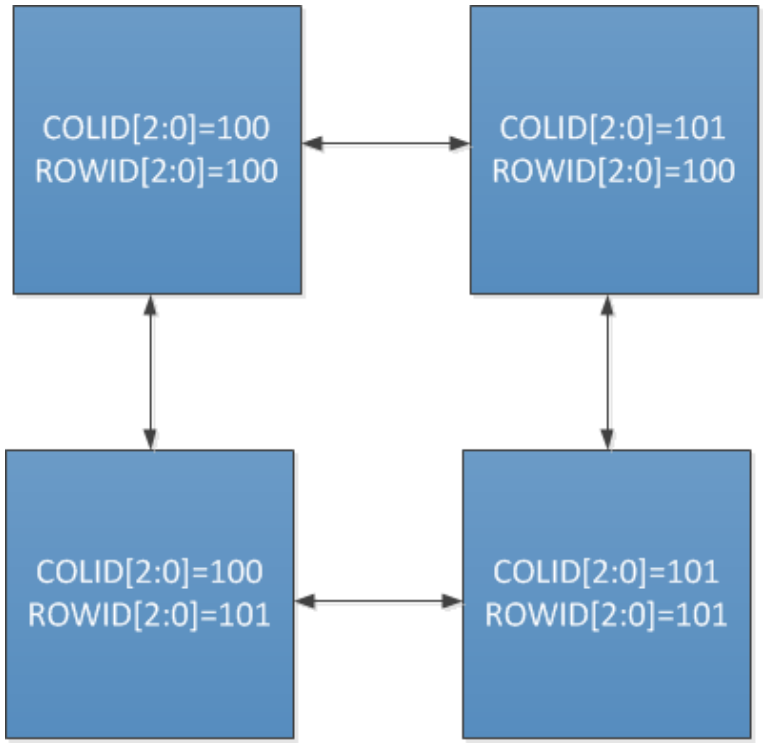



**Figure 3: Chip coordinate configuration example**

## 3.7 Chip-To-Chip Link Interface

The E64G401 has four identical source-synchronous bidirectional off chip LVDS links (eLink) that can be used to connect the E64G401 to other E64G401 chips, FPGAs, and/or ASICs. Interfacing the E64G401 with an FPGA should be done by instantiating the eLink HDL open source HDL code provided by Adapteva. The eLink enables glue-less chip to chip connections as illustrated in Figure 4.

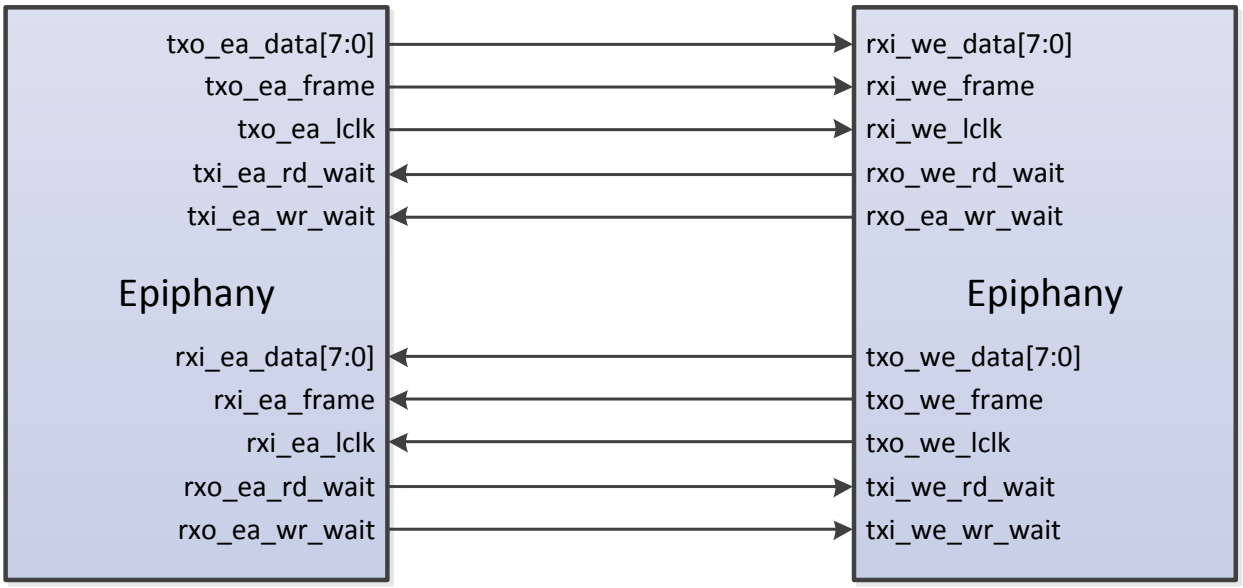



**Figure 4: eLink Chip Interface**

The eLink aggregates all read and write transactions from the eMesh Network-On Chip inside the E64G401 and sends them out as 104 bit memory mapped atomic transactions with the following sub-fields:

- Read indicator
- Write indicator
- Data size (2 bits)
- Control mode (4 bits)
- Destination address (32 bits)
- Lower data field (32 bits)
- Upper data field/source address (32 bits)

NOTE: The following eLink description is included to provide clarity. For a complete description, please refer to the open source eLink implementation at: http://github.com/parallella/parallella-hw

The eLink interface protocol is illustrated in Figure 5. The eLink data interface is 8 bits wide with a protocol specified in Table 9. The number of bytes to be received is determined by the data of the first "valid" byte (byte0) and the level of the FRAME signal. The data captured on the rising edge of the LCLK is considered to be byte0 if the FRAME control captured at the same cycle (rising edge) is high but was low at the rising edge of the previous LCLK cycle.

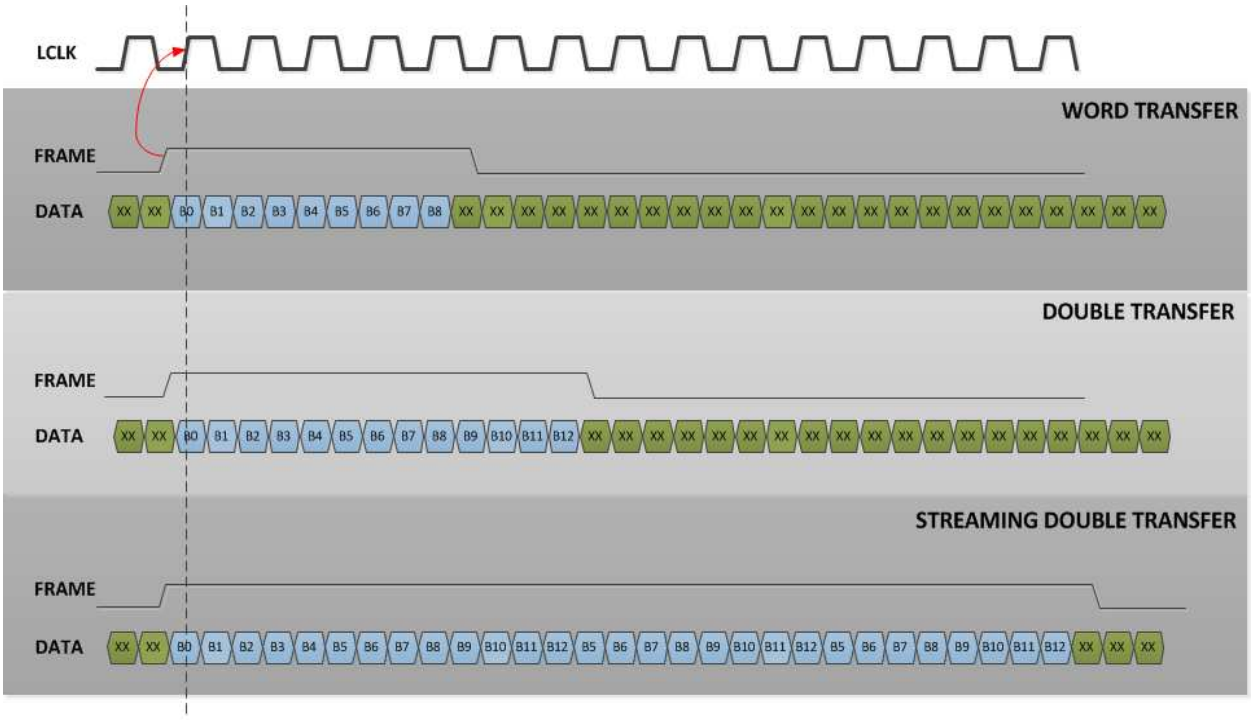



**Figure 5: eLink Interface Illustration**

The cycle after the last byte of the transaction (byte8 or byte12) will determine if the receiver should go into data streaming mode based on the level of the FRAME control signal. If the FRAME signal is low, the transaction is complete. If the FRAME control signal stays high, the eLink goes into "streaming mode", meaning that the last byte of the previous transaction (byte8 or byte12) will be followed by byte5 of the new transaction.

| Byte | Content | | | | |
|---|---|---|---|---|---|
| **Byte0** | {control mode[3:0], datamode[1:0], write, read} | | | | |
| **Byte1** | dstaddr[7:0] | | | | |
| **Byte 2** | dstaddr[15:8] | | | | |
| **Byte 3** | dstadr[23:16] | | | | |
| **Byte 4** | dstaddr[31:24] | | | | |
| **{datamode[1:0], read, write}** | 0X01 | 1001 | 1101 | XX10 | XX11 |
| **Byte 5** | data[7:0] | data[7:0] | data[7:0] | srcaddr[7:0] | data[7:0] |
| **Byte6** | data[15:8] | data[15:8] | data[15:8] | srcaddr[15:8] | data[15:8] |
| **Byte7** | data[23:16] | data[23:16] | data[23:16] | srcaddr[23:16] | data[23:16] |
| **Byte8** | data[31:24] | data[31:24] | data[31:24] | srcaddr[31:24] | data[31:24] |
| **Byte9** | N/A | N/A | data[32:39] | N/A | srcaddr[7:0] |
| **Byte10** | N/A | N/A | data[47:40] | N/A | srcaddr[15:8] |
| **Byte11** | N/A | N/A | data[56:48] | N/A | srcaddr[23:16] |
| **Byte12** | N/A | N/A | data[63:57] | N/A | srcaddr[31:24] |

**Table 9: eLink Data Format**

NOTE: Optimal eLink bandwidth utilization is achieved by transmitting a sequence of 64-bit write transactions with increasing address order (e.g 0x80800000, 0x80800008, 0x80800010, ). Read transactions, non 64-bit transactions, and non-sequential 64-bit write transactions will result in a max bandwidth of 1/4$^{th}$ of peak.

The WAIT_RD and WAIT_WR signals are used to stall transmission when a receiver is unable to accept more transactions. This mechanism is shown in Figure 6. The receiver will raise its WAIT output signal on the second rising edge of LCLK input following the capturing rising edge of the last transaction byte (byte8 or byte12) but will be ready to accept one more full transaction (byte0 through byte8/byte12). The WAIT signal seen by the transmitter is assumed to be of the "unspecified" phase delay (while still of the LCLK clock period) and therefore has to be sampled with the two-cycle synchronizer. Once synchronized to the transmitter's LCLK clock domain, the WAIT control signals will prevent new transaction from being transmitted. If the transaction is in the middle of the transmission when the synchronized WAIT control goes high, the transmission process is to be completed without interruption.

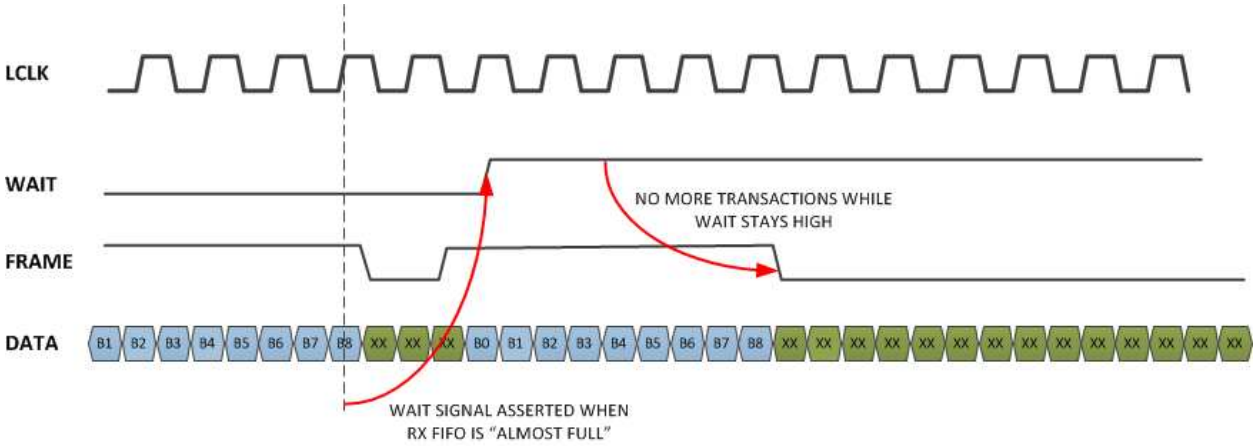



**Figure 6: Transmitter Pushback Mechanism**

The txo_* interface driven out from the E64G401 uses a divided version of the core cock frequency (RXI_WE_CCLK_{P,N}). The transmit clock is automatically aligned in the middle of the data eye by the eLink on chip transmit logic. The receiver logic assumes the clock is aligned at the center of the receiver data eye. The "wait" signals are used to indicate to the transmit logic that no more transactions can be received because the receiver buffer full.

For the sake of brevity the signal descriptions of the four links are merged in a single table below with each link having a unique modifier depending on the direction of the link. The modifier is: NO for north, SO for south, EA for east, and WE for west. The suffix P/N indicates positive or negative leg of the differential pair.

| Signal Name | Direction | Signal Description |
|---|---|---|
| **RXI_{NO,SO,EA,WE}_DATA_{P,N}[7:0]** | Input | Receiver data |
| **RXI_{NO,SO,EA,WE}_FRAME_{P,N}** | Input | Receiver packet framing signal |
| **RXI_{NO,SO,EA,WE}_LCLK_{P,N}** | Input | Receiver clock |
| **RXO_{NO,SO,EA,WE}_WR_WAIT_{P,N}** | Output | Asynchronous push-back for transmitter indicating that device must hold off on sending another write packet. |
| **RXO_{NO,SO,EA,WE}_RD_WAIT_{P,N}** | Output | Asynchronous push-back for transmitter indicating that device must hold off on sending another read packet. |
| **TXO_{NO,SO,EA,WE}_DATA_{P,N}[7:0]** | Output | Transmitter data |
| **TXO_{NO,SO,EA,WE}_FRAME_{P,N}** | Output | Transmitter packet framing signal |
| **TXO_{NO,SO,EA,WE}_LCLK_{P,N}** | Output | Transmitter clock |
| **TXI_{NO,SO,EA,WE}_WR_WAIT_{P,N}** | Input | Asynchronous push-back from receiver indicating that transmitter must hold off on sending another write packet. |
| **TXI_{NO,SO,EA,WE}_RD_WAIT_{P,N}** | Input | Asynchronous push-back from transmitter indicating that transmitter must hold off on sending another read packet. |

**Table 10: eLink Signals**

# 4 Electrical Specifications

NOTE: Absolute ratings are based on simulation results, process information, and initial testing. Final product qualification data not yet available, information provided without warranty.

## 4.1 Maximum Absolute Ratings

| Parameter | Description | Value | Units |
|---|---|---|---|
| $V_{DVDD}$ | IO Supply Voltage range | -0.5 to 1.98 | V |
| $V_{VDD}$ | Core supply voltage range | -0.3 to 1.155 | V |
| $V_{PAD}$ | Voltage range | -0.5 to (VDD+0.5) | V |
| $T_{MAX}$ | Storage temperature | -40C to 150 | °C |
| $T_J$ | Junction operating temperature | -40 to 125 | °C |

**Table 11: Maximum Absolute Ratings**

Charged devices and circuit boards can discharge without detection. Although this product features robust ESD protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

## 4.2 Recommended Operating Conditions

| Parameter | Description | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{DVDD}$ | IO Supply Voltage Range | 1.62 | 1.8 | 1.98 | V |
| $V_{VDD}$ | Core Supply Voltage Range | 0.90 | 1.0 | 1.1 | V |
| $V_{PAD}$ | Voltage at IO | -0.3 | | $V_{DVDD}$ + 0.3 | V |
| $V_{IH}$ | High-level input voltage at IO | 0.7 * $V_{DVDD}$ | | $V_{DVDD}$ + 0.3 | V |
| $V_{IL}$ | Low-level input voltage at IO | $V_{DVSS}$ - 0.3 | | $V_{DVDD}$ + 0.3 | V |
| $T_J$ | Junction Temp | -40 | 25 | 125 | °C |
| $I_{VDD}$ | Core supply current at 800MHz | | TBD | | mA |
| $I_{VDVDD}$ | IO supply current | | TBD | | mA |

**Table 12: Recommended Operating Conditions**

## 4.3 Control Signal AC/DC Characteristics

The FLAG, ROWID, COLDID, and RESET_N pins use full swing LVCMOS signaling with AC/DC specifications given in the following table.

| Parameter | Description | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $t_{rise}$ | Transmitter rise time (20pF load) | 0.94 | 1.43 | 2.31 | nS |
| $t_{fall}$ | Transmitter fall time (20pF load) | 0.96 | 1.39 | 2.19 | nS |
| $I_{OH}$ | Output drive current | 16.4 | 24.5 | 35.7 | mA |
| $I_{OL}$ | Output drive current | 16.9 | 27.0 | 39.1 | mA |
| $I_{SCH}$ | Output high short circuit | -- | -- | 70 | mA |
| $I_{SCL}$ | Output low short circuit | -- | -- | 72 | mA |
| $I_{L}$ | Input leakage | -- | -- | 1 | uA |

**Table 13: LVCMOS AC/DC Characteristics**

## 4.4 eLink AC/DC Specifications

The E64G401 eLinks conform to the sub-LVDS standard but has shown to be compatible with many LVDS receivers and drivers as well per (IEEE Std 1596.3-1996, Low Voltage Differential Signaling Standard). The driver design has all the necessary components for transmission of sub-LVDS data and a temperature stable internal reference for setting of the LVDS signaling voltage and common mode level.

| Parameter | Description | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{INPUT}$ | Common Mode Input Voltage | 0.4 | 0.9 | 1.4 | V |
| $V_{IDT}$ | Input differential threshold | 50 | | | mV |
| $V_{ID}$ | Input differential voltage | 50 | | $V_{DVDD}$ | mV |
| $V_{IA,IB}$ | Input voltage | 0 | 0.9 | $V_{DVDD}$ | V |
| $I_{IA,IA}$ | Input leakage current | | | 1 | uA |
| $t_{psupply}$ | Power supply sensitivity | | 2 | 0.3 | ps/mV |
| $t_{dcd}$ | Input duty cycle distortion | | 50 | | ps |
| $t_{rise}$ | Output differential rise time | 120 | | 250 | ps |
| $t_{fall}$ | Output differential fall time | 120 | | 250 | ps |
| $v_{OH}$ | Output voltage high | | 975 | 1100 | mV |
| $v_{OL}$ | Output voltage low | 700 | 825 | | mV |
| $v_{OD}$ | Differential output voltage | 100 | 150 | 200 | mV |
| $v_{OS}$ | Output offset voltage | 800 | 900 | 1000 | mV |
| $I_{SA,SB}$ | Output short circuit current | -12 | 5 | 12 | mA |

**Table 14:** LVDS Electrical Specifications

## 4.5 Power Consumption

TBD

**Figure 7: Power Consumption**

# 5 Timing Specifications

## 5.1 Reset Sequence

The CCLK_N/CCLK_P should toggle at least 10 times while RESET_N is held low and should be held constant during the rising edge of RESET_N.

## 5.2 eLINK Timing

The following table gives recommended eLink timing constraints for E64G401 system integration.

| Parameter | Value |
|---|---|
| Maximum skew between all rxi_* signals on one link | 250ps |
| Maximum skew between all txo_* signals on one link | 250ps |
| Maximum skew between nets of a p/n pair signal | 50ps |
| Receiver clock min setup to data edge | 250ps |
| Receiver clock min hold from data edge | 250ps |
| Minimum receiver LCLK period | 2ns |

**Table 15: eLink Timing Constraints**

# 6 Device Package

## 6.1 Overview

The E64G401 uses a 324 ball 0.8 mm pitch wire-bond BGA package that measures 15×15mm.

## 6.2 Graphical Pin Mapping

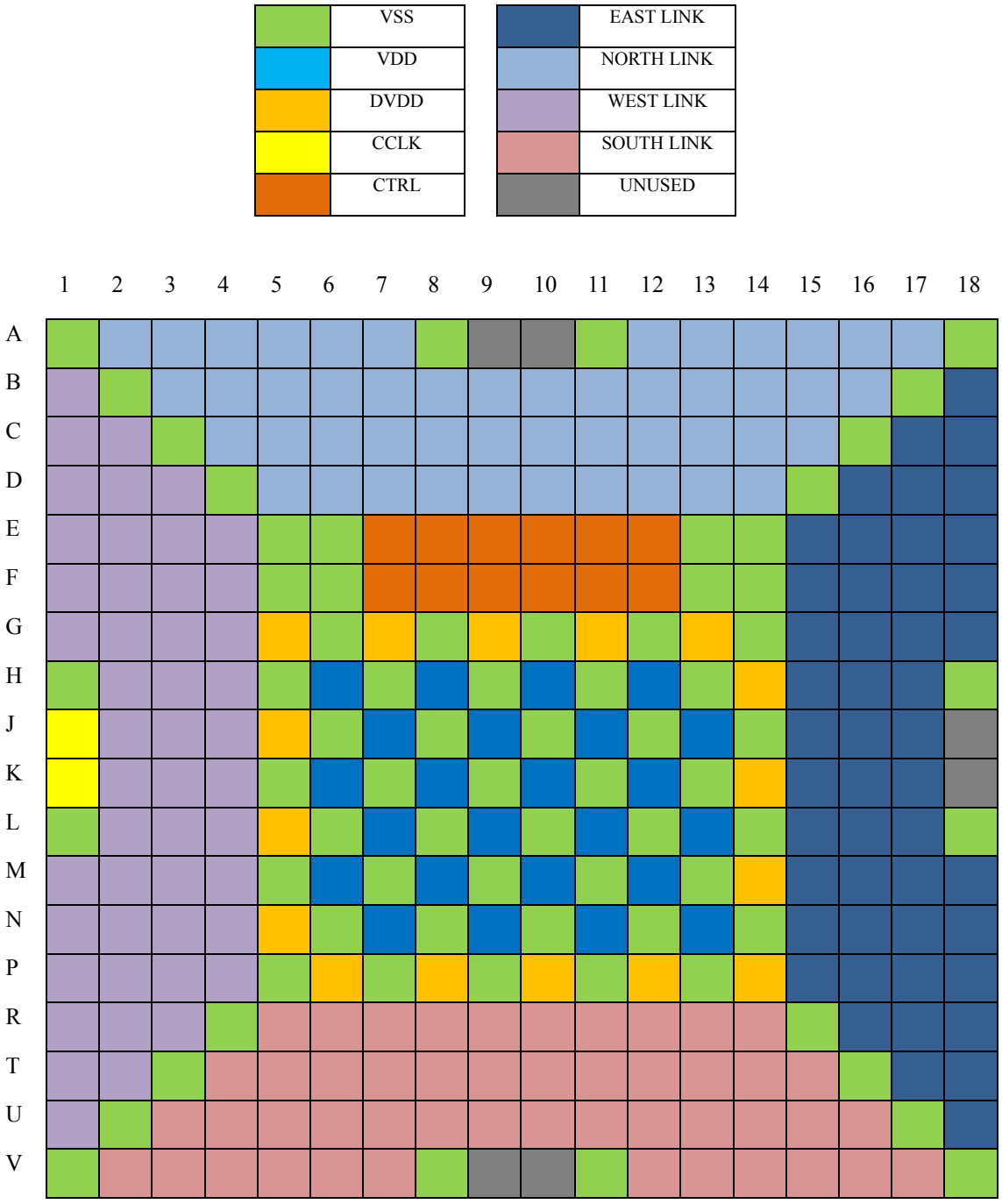
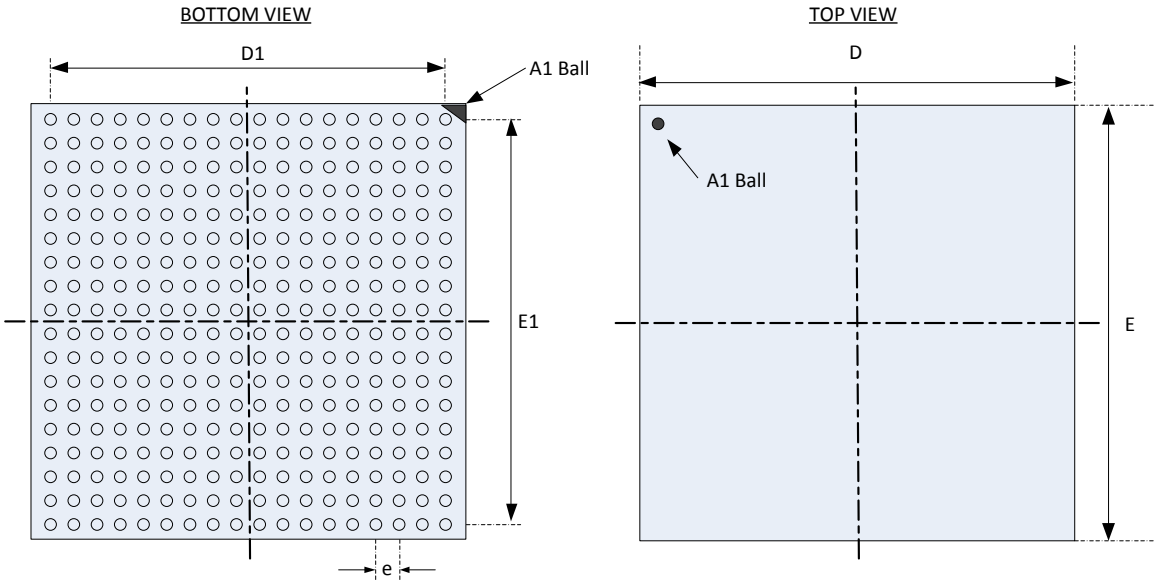



**Figure 8: Pin Mapping Diagram**

## 6.3 Mechanical Drawing




NOTES:
1. All dimensions and tolerances conform to ASME Y14.5M-1994
2. Symbol "M" is the pin matrix size
3. There shall be a minimum clearance of 0.25mm between edge of solder ball and body edge
4. Conforms to JEDEC MO-275-KKAB-1

| SYMBOL | MIN (mm) | NOM (mm) | MAX (mm) |
|---|---|---|---|
| A | 1.00 | ~ | 1.50 |
| A1 | 0.25 | ~ | 0.40 |
| A2 | 0.75 | 0.90 | 1.10 |
| D/E | | 15.00 | |
| D1/E1 | | 13.60 | |
| e | | 0.80 | |
| b | 0.40 | 0.45 | 0.50 |
| M | | 18 | |

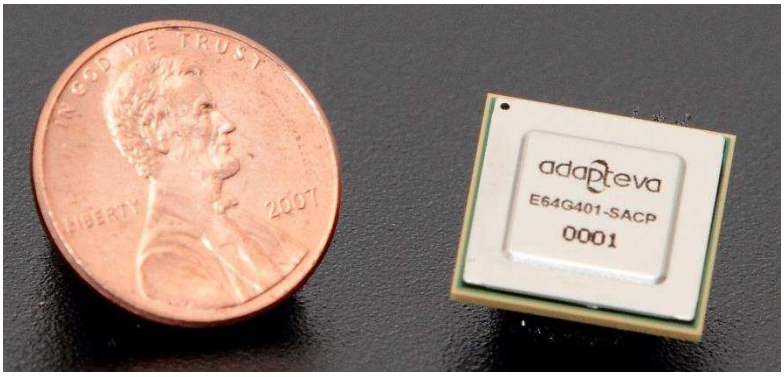



**Figure 9: Mechanical Drawing**

## 6.4 Thermal Characteristics

TBD

**Table 16: Thermal Characteristics of Device Package**

## 6.5 Complete Package Pin-out

| PIN | BGA BALL |
|---|---|
| **VSS (72)** | A1,A8,A11,A18,<br>B2,B17, C3,C16,D4,D15,<br>E5,E6,E13,E14,<br>F5,F6,F13,F14,<br>G6,G8,G10,G12,G14,<br>H1,H5,H7,H9,H11,H13,H18,<br>J6,J8,J10,J12,J14,<br>K5,K7,K9,K11,K13,<br>L1,L6,L8,L10,L12,L14,L18,<br>M5,M7,M9,M11,M13,<br>N6,N8,N10,N12,N14,<br>P5,P7, P9, P11,P13,<br>R4,R15, T3,T16,<br>U2,U17, V1, V8, V11,V18 |
| **VDD (24)** | H6,H8,H10,H12,J7,J9,J11,J13,K6,K8,K10,K12,L7,L9,L11,L13,M6<br>,M8,M10,M12,N7,N9,N11,N13 |
| **DVDD (16)** | G5,G7,G9,G11,G13,H14,J5,K14,L5,M14,N5,P6,P8,P10,P12,P14 |

| PIN | BGA BALL |
|---|---|
| **COLID[0]** | F12 |
| **COLID[1]** | E8 |
| **COLID[2]** | F10 |
| **COLID[3]** | E7 |
| **ROWID[0]** | E10 |
| **ROWID[1]** | E11 |
| **ROWID[2]** | E9 |
| **ROWID[3]** | F9 |
| **FLAG** | E12 |
| **MVDD** | F8 |
| **MVSS** | F7 |

| | |
|---|---|
| **RESET_N** | F11 |
| **RXI_EA_CCLK_P** | J18 |
| **RXI_EA_CCLK_N** | K18 |
| **RXI_EA_DATA_P[0]** | T18 |
| **RXI_EA_DATA_P[1]** | R17 |
| **RXI_EA_DAA_P[2]** | P16 |
| **RXI_EA_DATA_P[3]** | N15 |
| **RXI_EA_DATA_P[4]** | P18 |
| **RXI_EA_DATA_P[5]** | N17 |
| **RXI_EA_DATA_P[6]** | M16 |
| **RXI_EA_DATA_P[7]** | L15 |
| **RXI_EA_DATA_N[0]** | U18 |
| **RXI_EA_DATA_N[1]** | T17 |
| **RXI_EA_DATA_N[2]** | R16 |
| **RXI_EA_DATA_N[3]** | P15 |
| **RXI_EA_DATA_N[4]** | R18 |
| **RXI_EA_DATA_N[5]** | P17 |
| **RXI_EA_DATA_N[6]** | N16 |
| **RXI_EA_DATA_N[7]** | M15 |
| **RXI_EA_FRAME_P** | L17 |
| **RXI_EA_FRAME_N** | M17 |
| **RXI_EA_LCLK_P** | M18 |
| **RXI_EA_LCLK_N** | N18 |
| **TXI_NO_WR_WAIT_P** | A9 |
| **TXI_NO_WR_WAIT_N** | A10 |
| **TXO_NO_DATA_P[7]** | A2 |

| | |
|---|---|
| **TXO_NO_DATA_P[6]** | B3 |
| **TXO_NO_DATA_P[5]** | C4 |
| **TXO_NO_DATA_P[4]** | D5 |
| **TXO_NO_DATA_P[3]** | A4 |
| **TXO_NO_DATA_P[2]** | B5 |
| **TXO_NO_DATA_P[1]** | C6 |
| **TXO_NO_DATA_P[0]** | D7 |
| **TXO_NO_DATA_N[7]** | A3 |
| **TXO_NO_DATA_N[6]** | B4 |
| **TXO_NO_DATA_N[5]** | C5 |
| **TXO_NO_DATA_N[4]** | D6 |
| **TXO_NO_DATA_N[3]** | A5 |
| **TXO_NO_DATA_N[2]** | B6 |
| **TXO_NO_DATA_N[1]** | C7 |
| **TXO_NO_DATA_N[0]** | D8 |
| **TXI_NO_RD_WAIT_P** | B7 |
| **TXI_NO_RD_WAIT_N** | B8 |
| **TXO_NO_LCLK_P** | A6 |
| **TXO_NO_LCLK_N** | A7 |
| **RXO_SO_WR_WAIT_N** | V10 |
| **RXO_SO_WR_WAIT_P** | V9 |
| **RXI_SO_DATA_N[7]** | V3 |
| **RXI_SO_DATA_N[6]** | U4 |
| **RXI_SO_DATA_N[5]** | T5 |
| **RXI_SO_DATA_N[4]** | R6 |
| **RXI_SO_DATA_N[3]** | V5 |

| | |
|---|---|
| **RXI_SO_DATA_N[2]** | U6 |
| **RXI_SO_DATA_N[1]** | T7 |
| **RXI_SO_DATA_N[0]** | R8 |
| **RXI_SO_DATA_P[7]** | V2 |
| **RXI_SO_DATA_P[6]** | U3 |
| **RXI_SO_DATA_P[5]** | T4 |
| **RXI_SO_DATA_P[4]** | R5 |
| **RXI_SO_DATA_P[3]** | V4 |
| **RXI_SO_DATA_P[2]** | U5 |
| **RXI_SO_DATA_P[1]** | T6 |
| **RXI_SO_DATA_P[0]** | R7 |
| **RXO_SO_RD_WAIT_N** | U8 |
| **RXO_SO_RD_WAIT_P** | U7 |
| **RXI_SO_LCLK_N** | V7 |
| **RXI_SO_LCLK_P** | V6 |
| **RXI_WE_CCLK_N** | K1 |
| **RXI_WE_CCLK_P** | J1 |
| **TXO_WE_DATA_N[0]** | U1 |
| **TXO_WE_DATA_N[1]** | T2 |
| **TXO_WE_DATA_N[2]** | R3 |
| **TXO_WE_DATA_N[3]** | P4 |
| **TXO_WE_DATA_N[4]** | R1 |
| **TXO_WE_DATA_N[5]** | P2 |
| **TXO_WE_DATA_N[6]** | N3 |
| **TXO_WE_DATA_N[7]** | M4 |
| **TXO_WE_DATA_P[0]** | T1 |

| | |
|---|---|
| **TXO_WE_DATA_P[1]** | R2 |
| **TXO_WE_DATA_P[2]** | P3 |
| **TXO_WE_DATA_P[3]** | N4 |
| **TXO_WE_DATA_P[4]** | P1 |
| **TXO_WE_DATA_P[5]** | N2 |
| **TXO_WE_DATA_P[6]** | M3 |
| **TXO_WE_DATA_P[7]** | L4 |
| **TXO_WE_FRAME_N** | M2 |
| **TXO_WE_FRAME_P** | L2 |
| **TXO_WE_LCLK_N** | N1 |
| **TXO_WE_LCLK_P** | M1 |
| **RXO_EA_RD_WAIT_P** | J17 |
| **RXO_EA_RD_WAIT_N** | K17 |
| **RXO_EA_WR_WAIT_P** | K16 |
| **RXO_EA_WR_WAIT_N** | L16 |
| **RXO_NO_WR_WAIT_P** | B9 |
| **RXO_NO_WR_WAIT_N** | B10 |
| **RXO_NO_RD_WAIT_P** | C8 |
| **RXO_NO_RD_WAIT_N** | C9 |
| **TXI_SO_WR_WAIT_N** | U10 |
| **TXI_SO_WR_WAIT_P** | U9 |
| **TXI_SO_RD_WAIT_N** | T9 |
| **TXI_SO_RD_WAIT_P** | T8 |
| **TXI_WE_RD_WAIT_N** | K2 |
| **TXI_WE_RD_WAIT_P** | J2 |
| **TXI_WE_WR_WAIT_N** | L3 |

| | |
|---|---|
| **TXI_WE_WR_WAIT_P** | K3 |
| **TXI_EA_RD_WAIT_P** | G17 |
| **TXI_EA_RD_WAIT_N** | H17 |
| **TXI_EA_WR_WAIT_P** | H16 |
| **TXI_EA_WR_WAIT_N** | J16 |
| **RXI_NO_FRAME_P** | B11 |
| **RXI_NO_FRAME_N** | B12 |
| **RXI_NO_CCLK_P** | C10 |
| **RXI_NO_CCLK_N** | C11 |
| **TXO_SO_FRAME_N** | U12 |
| **TXO_SO_FRAME_P** | U11 |
| **RXI_SO_CCLK_N** | T11 |
| **RXI_SO_CCLK_P** | T10 |
| **RXO_WE_RD_WAIT_N** | H2 |
| **RXO_WE_RD_WAIT_P** | G2 |
| **RXO_WE_WR_WAIT_N** | J3 |
| **RXO_WE_WR_WAIT_P** | H3 |
| **TXO_EA_DATA_P[0]** | G15 |
| **TXO_EA_DATA_P[1]** | F16 |
| **TXO_EA_DATA_P[2]** | E17 |
| **TXO_EA_DATA_P[3]** | D18 |
| **TXO_EA_DATA_P[4]** | E15 |
| **TXO_EA_DATA_P[5]** | D16 |
| **TXO_EA_DATA_P[6]** | C17 |
| **TXO_EA_DATA_P[7]** | B18 |
| **TXO_EA_DATA_N[0]** | H15 |

| | |
|---|---|
| **TXO_EA_DATA_N[1]** | G16 |
| **TXO_EA_DATA_N[2]** | F17 |
| **TXO_EA_DATA_N[3]** | E18 |
| **TXO_EA_DATA_N[4]** | F15 |
| **TXO_EA_DATA_N[5]** | E16 |
| **TXO_EA_DATA_N[6]** | D17 |
| **TXO_EA_DATA_N[7]** | C18 |
| **TXO_EA_FRAME_P** | J15 |
| **TXO_EA_FRAME_N** | K15 |
| **TXO_EA_LCLK_P** | F18 |
| **TXO_EA_LCLK_N** | G18 |
| **RXI_NO_DATA_P[7]** | D11 |
| **RXI_NO_DATA_P[6]** | C12 |
| **RXI_NO_DATA_P[5]** | B13 |
| **RXI_NO_DATA_P[4]** | A14 |
| **RXI_NO_DATA_P[3]** | D13 |
| **RXI_NO_DATA_P[2]** | C14 |
| **RXI_NO_DATA_P[1]** | B15 |
| **RXI_NO_DATA_P[0]** | A16 |
| **RXI_NO_DATA_N[7]** | D12 |
| **RXI_NO_DATA_N[6]** | C13 |
| **RXI_NO_DATA_N[5]** | B14 |
| **RXI_NO_DATA_N[4]** | A15 |
| **RXI_NO_DATA_N[3]** | D14 |
| **RXI_NO_DATA_N[2]** | C15 |
| **RXI_NO_DATA_N[1]** | B16 |

| | |
|---|---|
| **RXI_NO_DATA_N[0]** | A17 |
| **TXO_NO_FRAME_P** | D9 |
| **TXO_NO_FRAME_N** | D10 |
| **RXI_NO_LCLK_P** | A12 |
| **RXI_NO_LCLK_N** | A13 |
| **TXO_SO_DATA_N[7]** | R12 |
| **TXO_SO_DATA_N[6]** | T13 |
| **TXO_SO_DATA_N[5]** | U14 |
| **TXO_SO_DATA_N[4]** | V15 |
| **TXO_SO_DATA_N[3]** | R14 |
| **TXO_SO_DATA_N[2]** | T15 |
| **TXO_SO_DATA_N[1]** | U16 |
| **TXO_SO_DATA_N[0]** | V17 |
| **TXO_SO_DATA_P[7]** | R11 |
| **TXO_SO_DATA_P[6]** | T12 |
| **TXO_SO_DATA_P[5]** | U13 |
| **TXO_SO_DATA_P[4]** | V14 |
| **TXO_SO_DATA_P[3]** | R13 |
| **TXO_SO_DATA_P[2]** | T14 |
| **TXO_SO_DATA_P[1]** | U15 |
| **TXO_SO_DATA_P[0]** | V16 |
| **RXI_SO_FRAME_N** | R10 |
| **RXI_SO_FRAME_P** | R9 |
| **TXO_SO_LCLK_N** | V13 |
| **TXO_SO_LCLK_P** | V12 |
| **RXI_WE_DATA_N[0]** | H4 |

| | |
|---|---|
| **RXI_WE_DATA_N[1]** | G3 |
| **RXI_WE_DATA_N[2]** | F2 |
| **RXI_WE_DATA_N[3]** | E1 |
| **RXI_WE_DATA_N[4]** | F4 |
| **RXI_WE_DATA_N[5]** | E3 |
| **RXI_WE_DATA_N[6]** | D2 |
| **RXI_WE_DATA_N[7]** | C1 |
| **RXI_WE_DATA_P[0]** | G4 |
| **RXI_WE_DATA_P[1]** | F3 |
| **RXI_WE_DATA_P[2]** | E2 |
| **RXI_WE_DATA_P[3]** | D1 |
| **RXI_WE_DATA_P[4]** | E4 |
| **RXI_WE_DATA_P[5]** | D3 |
| **RXI_WE_DATA_P[6]** | C2 |
| **RXI_WE_DATA_P[7]** | B1 |
| **RXI_WE_FRAME_N** | K4 |
| **RXI_WE_FRAME_P** | J4 |
| **RXI_WE_LCLK_N** | G1 |
| **RXI_WE_LCLK_P** | F1 |

# 7 System Integration

## 7.1 FPGA/ASIC Interfacing

The E64G401 can be directly interfaced to an FPGA or ASIC by instantiating the eLink interface provided by Adapteva. The eLink interface block is used to convert the high speed serial link I/O interface to a lower speed parallel interface. To the system, the eLink interface looks like a simple memory mapped interface.

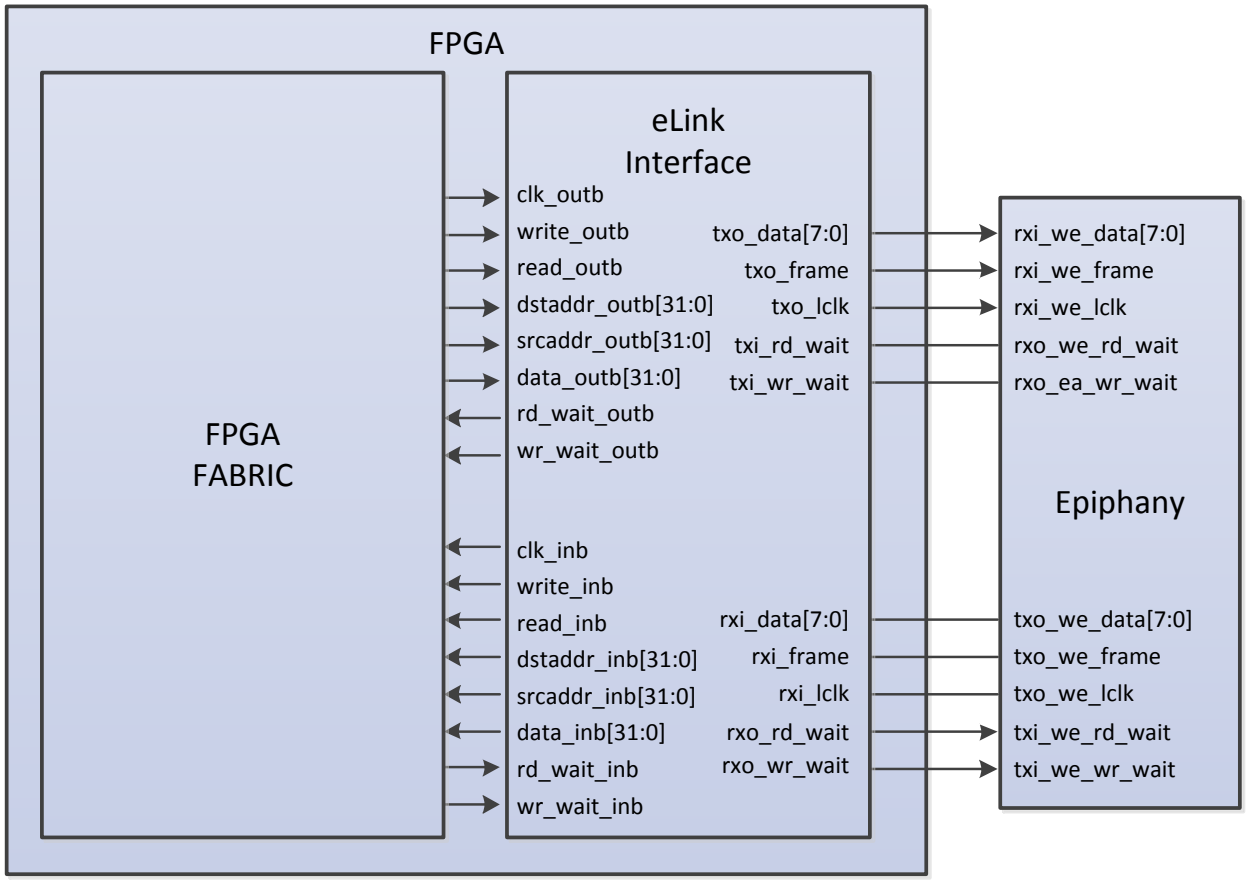



**Figure 10: FPGA eLink Integration Example**

The following table documents the signal description of the eLink interface as seen by the FPGA internal logic.

| Signal Name | Width | Direction | Description |
|---|---|---|---|
| **clk_inb** | 1 | In | Clock used to write to transaction FIFO in eMesh interface |
| **access_inb** | 1 | In | Assert high and stays keep until read or write transaction has completed |
| **write_inb** | 1 | In | Asserted high to indicate a write transaction |
| **datamode_inb** | 2 | In | Datasize of transaction.<br>00=8-bit, 01=16-bit, 10=32-bit, 11=64-bit |
| **ctrlmode_inb** | 4 | In | Reserved, should be tied to 0 |
| **dstaddr_inb** | 32 | In | Address of memory mapped transaction |
| **data_inb** | 32 | In | For write transaction: Data to be written<br>For read transaction: Value ignored |
| **srcaddr_inb** | 32 | In | For write transaction: Upper data of 64 bit transaction in case of double write transaction, otherwise ignored<br>For read transaction: Returning address to send data to once the data has been read from the address in the dstaddr field. |
| **wr_wait_inb** | 1 | Out | Pushback indicating that eMesh write transaction receiving FIFO is full |
| **rd_wait_inb** | 1 | Out | Pushback indicating that eMesh read transaction receiving FIFO is full |

**Table 17: Incoming Transaction (TO FPGA/ASIC)**

| Signal Name | Width | Direction | Description |
|---|---|---|---|
| **clk_outb** | 1 | Out | Clock to be used to sample transaction from eMesh |
| **access_outb** | 1 | Out | Asserted high and stays keep until read or write transaction has completed |
| **write_outb** | 1 | Out | Asserted high to indicate a write transaction |
| **datamode_outb** | 2 | Out | Data size of transaction.<br>00=8-bit, 01=16-bit, 10=32-bit, 11=64-bit |
| **ctrlmode_outb** | 4 | Out | Reserved, should be tied to 0 |
| **dstaddr_outb** | 32 | Out | Address of memory mapped transaction |
| **data_outb** | 32 | Out | For write transaction: Data to write to dstaddr_out<br>For read transaction: Value can be ignored |
| **srcaddr_outb** | 32 | Out | For write transaction: Upper data of 64 bit transaction in case of double write transaction, otherwise ignored<br>For read transaction: Returning address to send data to once the data has been read from the address in the dstaddr field. |
| **wr_wait_outb** | 1 | In | Driven high to stop eMesh from sending more write transactions |
| **rd_wait_outb** | 1 | In | Driven high to stop eMesh from sending more read transactions |

**Table 18: Outgoing Transaction (FROM FPGA/ASIC)**

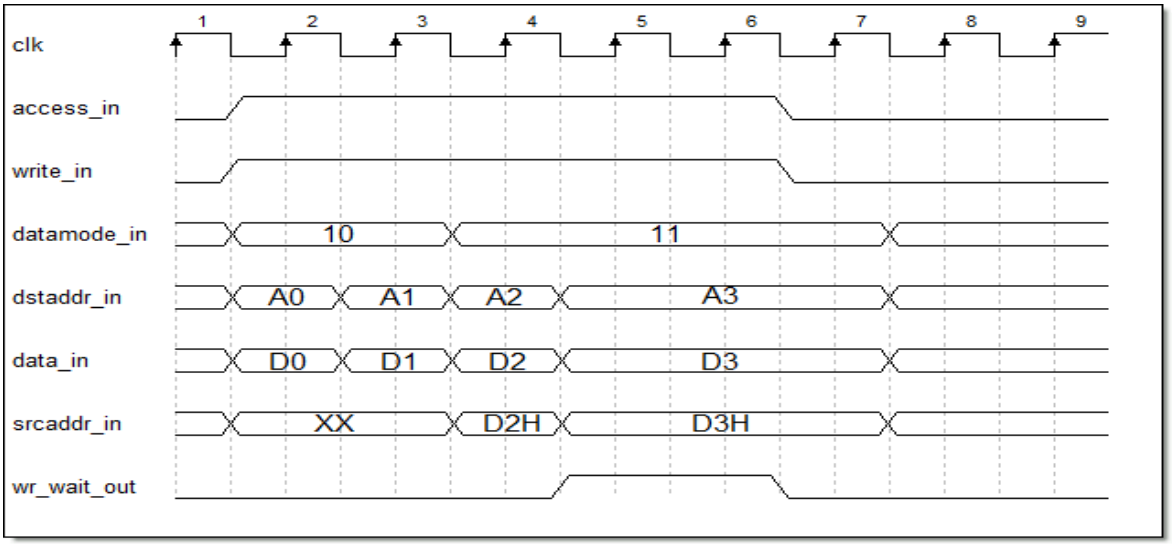



**Figure 11: eLink Write Transaction Timing Diagram**

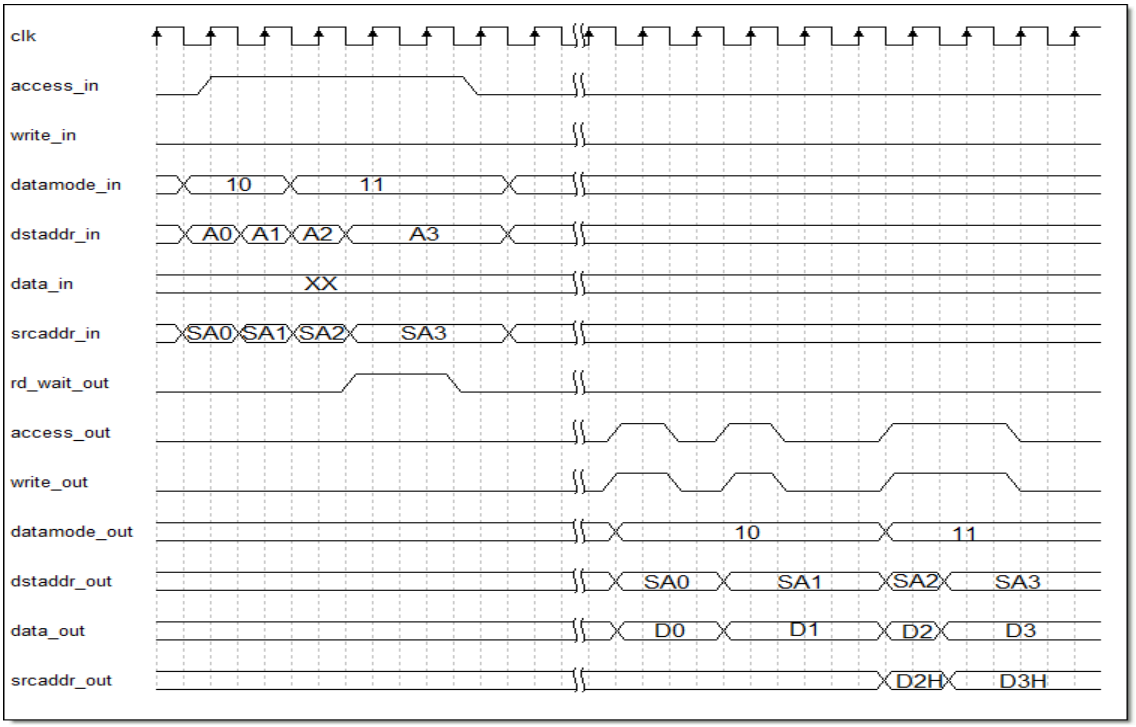



**Figure 12: eLink Read Transaction Timing Diagram**

# 8 Ordering Information

## 8.1 Part Number Naming Methodology

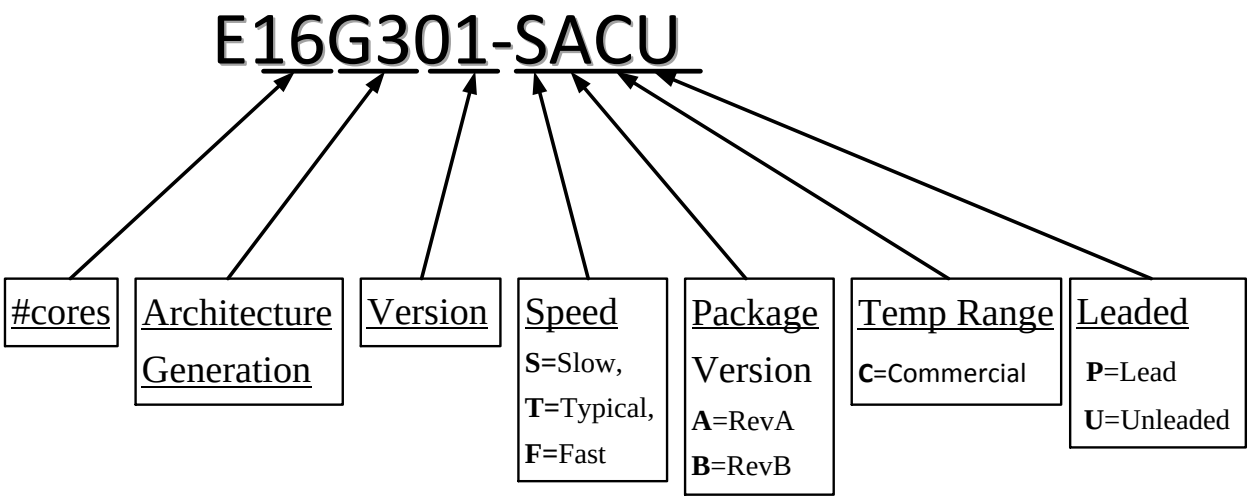



**Figure 13: Part Numbering Methodology**

## 8.2 Product Availability

| Part Number | Temp Range | Speed Grade | Comment |
|---|---|---|---|
| **E64G401-SACP** | –40°C to +85°C | 700MHz | Sampling |
| **E64G401-TACU** | –40°C to +85°C | 800MHz | Planned for Q4, 2014 |

**Table 19: Product Ordering Table**

# 9 Errata

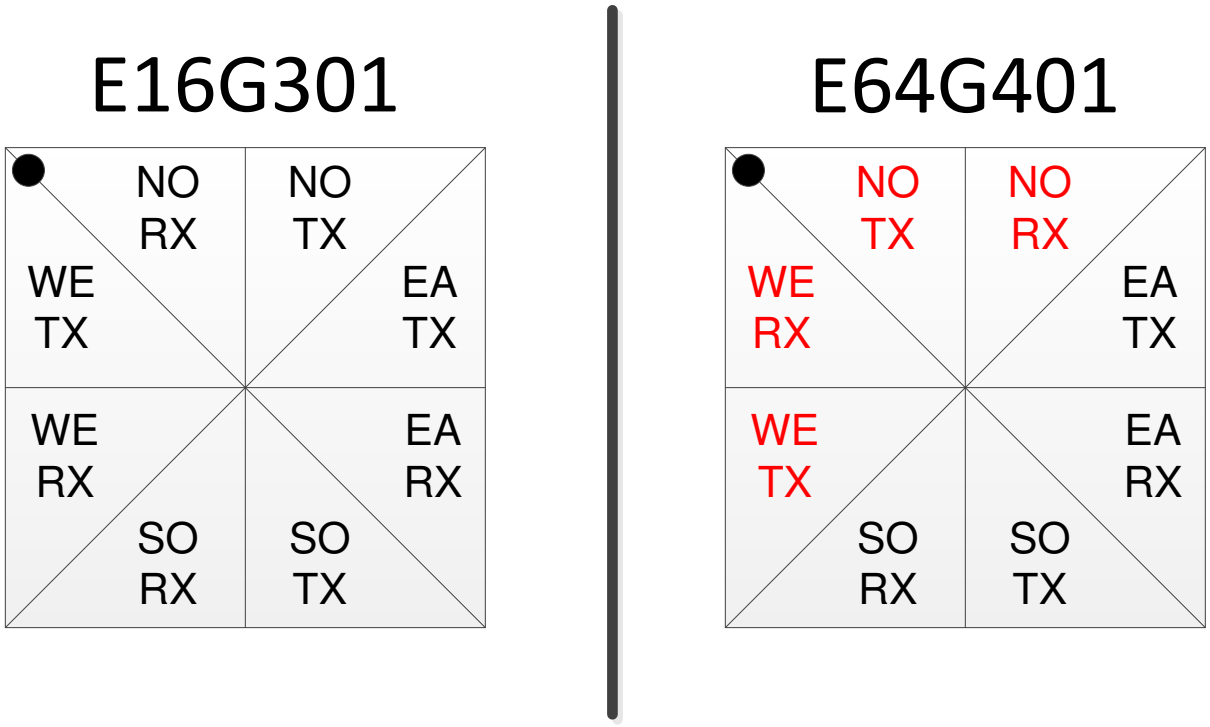
This table contains all the known errata for the E64G401 product. System designers and software developers should consider these items as a part of the current product but they should also beware that that the behavior of the errata items is likely to change in future versions of the Epiphany products.

| Errata # | Errata Item | Chip Version | Type | Explanation |
|---|---|---|---|---|
| **0** | Duplicate IO Transaction | E64G401 | Functional | All incoming transactions (read or write) to a link port destined for row 2 (east/west off chip transaction) or column 2 (south/north off chip transaction) are duplicated at link interface. This means instruction fetches and data loads from external memory is not possible for these cores but DMA transfers to these cores will work correctly. The latency between the first and second duplicate transaction is short, but software needs to take into account that a second write transaction will occur. |
| **2** | NOC FIFO Full | E64G401 | Performance | The FIFO interface between the compute node and the Network-On-Chip currently indicates FIFO full too early, causing a degradation in peak outgoing transfer bandwidth from the Epiphany processor node to the eMesh NOC. |

**Table 20: Errata List**

# 10 Migration from E16G301 to the E64G401

The 64-core E16G401 package design is similar to the E16G301 and with careful system design it is possible to produce boards that can accommodate both chip products. As illustrated in the figurebelow, the E64G401 uses an improved pin-out whereby the RX and TX links of neighboring Epiphany chips line up more favorably. Unfortunately, this means that multi-Epiphany board products with direct chip to chip Epiphany connections will not be able to accommodate both the E16G301 and E64G401. Products such as the Parallella boards which only connect an FPGA to an Epiphany chip can be designed to accommodate both products. **NOTE:** Some FPGAs have hard coded pin constraints which limit the use of certain pins as either inputs OR outputs, making it difficult and/or impossible to share board designs between the two Epiphany products.




## *10.1 Chip Supplies*

The nominal IO voltage for the E16301 is 2.5V while the nominal voltage for the E64G401 is 1.8V. Common power system design is possible because the E16G301 can operate down to 1.8V supply voltage with a slightly reduced link operating frequency. All ground and supply pins are in identical positions on the two products.

## 10.2 Chip Coordinates (ROWID/COLID)

It is recommended that the E64G401 chip coordinate pins on the board be configurable and/or software programmable through an FPGA to maximize flexibility.

## 10.3 Link Registers

The address locations of the south and east links moved locations as described in Section 2.2

## 10.4 Migration Pin Mapping

The following table shows the ball map differences between the E16G301 and E64G401. Balls that are not included in the table have identical mapping on the two products.

| BALL | E16G301 NET | E64G401 NET | Comment |
|---|---|---|---|
| J18 | RXI_EA_CCLK_N | RXI_EA_CCLK_P | INVERTED |
| K18 | RXI_EA_CCLK_P | RXI_EA_CCLK_N | INVERTED |
| T18 | RXI_EA_DATA_N[0] | RXI_EA_DATA_P[0] | INVERTED |
| R17 | RXI_EA_DATA_N[1] | RXI_EA_DATA_P[1] | INVERTED |
| P16 | RXI_EA_DATA_N[2] | RXI_EA_DATA_P[2] | INVERTED |
| N15 | RXI_EA_DATA_N[3] | RXI_EA_DATA_P[3] | INVERTED |
| P18 | RXI_EA_DATA_N[4] | RXI_EA_DATA_P[4] | INVERTED |
| N17 | RXI_EA_DATA_N[5] | RXI_EA_DATA_P[5] | INVERTED |
| M16 | RXI_EA_DATA_N[6] | RXI_EA_DATA_P[6] | INVERTED |
| L15 | RXI_EA_DATA_N[7] | RXI_EA_DATA_P[7] | INVERTED |
| U18 | RXI_EA_DATA_P[0] | RXI_EA_DATA_N[0] | INVERTED |
| T17 | RXI_EA_DATA_P[1] | RXI_EA_DATA_N[1] | INVERTED |
| R16 | RXI_EA_DATA_P[2] | RXI_EA_DATA_N[2] | INVERTED |
| P15 | RXI_EA_DATA_P[3] | RXI_EA_DATA_N[3] | INVERTED |
| R18 | RXI_EA_DATA_P[4] | RXI_EA_DATA_N[4] | INVERTED |
| P17 | RXI_EA_DATA_P[5] | RXI_EA_DATA_N[5] | INVERTED |
| N16 | RXI_EA_DATA_P[6] | RXI_EA_DATA_N[6] | INVERTED |

| | | | |
|---|---|---|---|
| M15 | RXI_EA_DATA_P[7] | RXI_EA_DATA_N[7] | INVERTED |
| L17 | RXI_EA_FRAME_N | RXI_EA_FRAME_P | INVERTED |
| M17 | RXI_EA_FRAME_P | RXI_EA_FRAME_N | INVERTED |
| M18 | RXI_EA_LCLK_N | RXI_EA_LCLK_P | INVERTED |
| N18 | RXI_EA_LCLK_P | RXI_EA_LCLK_N | INVERTED |
| A9 | RXI_NO_CCLK_N | TXI_NO_WR_WAIT_P | SCRAMBLED |
| A10 | RXI_NO_CCLK_P | TXI_NO_WR_WAIT_N | SCRAMBLED |
| A2 | RXI_NO_DATA_N[0] | TXO_NO_DATA_P[7] | SCRAMBLED-REVERSED |
| B3 | RXI_NO_DATA_N[1] | TXO_NO_DATA_P[6] | SCRAMBLED-REVERSED |
| C4 | RXI_NO_DATA_N[2] | TXO_NO_DATA_P[5] | SCRAMBLED-REVERSED |
| D5 | RXI_NO_DATA_N[3] | TXO_NO_DATA_P[4] | SCRAMBLED-REVERSED |
| A4 | RXI_NO_DATA_N[4] | TXO_NO_DATA_P[3] | SCRAMBLED-REVERSED |
| B5 | RXI_NO_DATA_N[5] | TXO_NO_DATA_P[2] | SCRAMBLED-REVERSED |
| C6 | RXI_NO_DATA_N[6] | TXO_NO_DATA_P[1] | SCRAMBLED-REVERSED |
| D7 | RXI_NO_DATA_N[7] | TXO_NO_DATA_P[0] | SCRAMBLED-REVERSED |
| A3 | RXI_NO_DATA_P[0] | TXO_NO_DATA_N[7] | SCRAMBLED-REVERSED |
| B4 | RXI_NO_DATA_P[1] | TXO_NO_DATA_N[6] | SCRAMBLED-REVERSED |
| C5 | RXI_NO_DATA_P[2] | TXO_NO_DATA_N[5] | SCRAMBLED-REVERSED |
| D6 | RXI_NO_DATA_P[3] | TXO_NO_DATA_N[4] | SCRAMBLED-REVERSED |
| A5 | RXI_NO_DATA_P[4] | TXO_NO_DATA_N[3] | SCRAMBLED-REVERSED |
| B6 | RXI_NO_DATA_P[5] | TXO_NO_DATA_N[2] | SCRAMBLED-REVERSED |
| C7 | RXI_NO_DATA_P[6] | TXO_NO_DATA_N[1] | SCRAMBLED-REVERSED |
| D8 | RXI_NO_DATA_P[7] | TXO_NO_DATA_N[0] | SCRAMBLED-REVERSED |
| B7 | RXI_NO_FRAME_N | TXI_NO_RD_WAIT_P | SCRAMBLED-REVERSED |
| B8 | RXI_NO_FRAME_P | TXI_NO_RD_WAIT_N | SCRAMBLED-REVERSED |
| A6 | RXI_NO_LCLK_N | TXO_NO_LCLK_P | SCRAMBLED-REVERSED |
| A7 | RXI_NO_LCLK_P | TXO_NO_LCLK_N | SCRAMBLED-REVERSED |
| V10 | RXI_SO_CCLK_N | RXO_SO_WR_WAIT_N | SCRAMBLED-REVERSED |
| V9 | RXI_SO_CCLK_P | RXO_SO_WR_WAIT_P | SCRAMBLED-REVERSED |
| V3 | RXI_SO_DATA_N[0] | RXI_SO_DATA_N[7] | SCRAMBLED |
| U4 | RXI_SO_DATA_N[1] | RXI_SO_DATA_N[6] | SCRAMBLED |

| | | | |
|---|---|---|---|
| T5 | RXI_SO_DATA_N[2] | RXI_SO_DATA_N[5] | SCRAMBLED |
| R6 | RXI_SO_DATA_N[3] | RXI_SO_DATA_N[4] | SCRAMBLED |
| V5 | RXI_SO_DATA_N[4] | RXI_SO_DATA_N[3] | SCRAMBLED |
| U6 | RXI_SO_DATA_N[5] | RXI_SO_DATA_N[2] | SCRAMBLED |
| T7 | RXI_SO_DATA_N[6] | RXI_SO_DATA_N[1] | SCRAMBLED |
| R8 | RXI_SO_DATA_N[7] | RXI_SO_DATA_N[0] | SCRAMBLED |
| V2 | RXI_SO_DATA_P[0] | RXI_SO_DATA_P[7] | SCRAMBLED |
| U3 | RXI_SO_DATA_P[1] | RXI_SO_DATA_P[6] | SCRAMBLED |
| T4 | RXI_SO_DATA_P[2] | RXI_SO_DATA_P[5] | SCRAMBLED |
| R5 | RXI_SO_DATA_P[3] | RXI_SO_DATA_P[4] | SCRAMBLED |
| V4 | RXI_SO_DATA_P[4] | RXI_SO_DATA_P[3] | SCRAMBLED |
| U5 | RXI_SO_DATA_P[5] | RXI_SO_DATA_P[2] | SCRAMBLED |
| T6 | RXI_SO_DATA_P[6] | RXI_SO_DATA_P[1] | SCRAMBLED |
| R7 | RXI_SO_DATA_P[7] | RXI_SO_DATA_P[0] | SCRAMBLED |
| U8 | RXI_SO_FRAME_N | RXO_SO_RD_WAIT_N | SCRAMBLED-REVERSED |
| U7 | RXI_SO_FRAME_P | RXO_SO_RD_WAIT_P | SCRAMBLED-REVERSED |
| U1 | RXI_WE_DATA_N[0] | TXO_WE_DATA_N[0] | REVERSED |
| T2 | RXI_WE_DATA_N[1] | TXO_WE_DATA_N[1] | REVERSED |
| R3 | RXI_WE_DATA_N[2] | TXO_WE_DATA_N[2] | REVERSED |
| P4 | RXI_WE_DATA_N[3] | TXO_WE_DATA_N[3] | REVERSED |
| R1 | RXI_WE_DATA_N[4] | TXO_WE_DATA_N[4] | REVERSED |
| P2 | RXI_WE_DATA_N[5] | TXO_WE_DATA_N[5] | REVERSED |
| N3 | RXI_WE_DATA_N[6] | TXO_WE_DATA_N[6] | REVERSED |
| M4 | RXI_WE_DATA_N[7] | TXO_WE_DATA_N[7] | REVERSED |
| T1 | RXI_WE_DATA_P[0] | TXO_WE_DATA_P[0] | REVERSED |
| R2 | RXI_WE_DATA_P[1] | TXO_WE_DATA_P[1] | REVERSED |
| P3 | RXI_WE_DATA_P[2] | TXO_WE_DATA_P[2] | REVERSED |
| N4 | RXI_WE_DATA_P[3] | TXO_WE_DATA_P[3] | REVERSED |
| P1 | RXI_WE_DATA_P[4] | TXO_WE_DATA_P[4] | REVERSED |
| N2 | RXI_WE_DATA_P[5] | TXO_WE_DATA_P[5] | REVERSED |
| M3 | RXI_WE_DATA_P[6] | TXO_WE_DATA_P[6] | REVERSED |

| | | | |
|---|---|---|---|
| L4 | RXI_WE_DATA_P[7] | TXO_WE_DATA_P[7] | REVERSED |
| M2 | RXI_WE_FRAME_N | TXO_WE_FRAME_N | REVERSED |
| L2 | RXI_WE_FRAME_P | TXO_WE_FRAME_P | REVERSED |
| N1 | RXI_WE_LCLK_N | TXO_WE_LCLK_N | REVERSED |
| M1 | RXI_WE_LCLK_P | TXO_WE_LCLK_P | REVERSED |
| J17 | RXO_EA_RD_WAIT_N | RXO_EA_RD_WAIT_P | INVERTED |
| K17 | RXO_EA_RD_WAIT_P | RXO_EA_RD_WAIT_N | INVERTED |
| K16 | RXO_EA_WR_WAIT_N | RXO_EA_WR_WAIT_P | INVERTED |
| L16 | RXO_EA_WR_WAIT_P | RXO_EA_WR_WAIT_N | INVERTED |
| B9 | RXO_NO_RD_WAIT_N | RXO_NO_WR_WAIT_P | SCRAMBLED |
| B10 | RXO_NO_RD_WAIT_P | RXO_NO_WR_WAIT_N | SCRAMBLED |
| C8 | RXO_NO_WR_WAIT_N | RXO_NO_RD_WAIT_P | SCRAMBLED |
| C9 | RXO_NO_WR_WAIT_P | RXO_NO_RD_WAIT_N | SCRAMBLED |
| U10 | RXO_SO_RD_WAIT_N | TXI_SO_WR_WAIT_N | SCRAMBLED-REVERSED |
| U9 | RXO_SO_RD_WAIT_P | TXI_SO_WR_WAIT_P | SCRAMBLED-REVERSED |
| T9 | RXO_SO_WR_WAIT_N | TXI_SO_RD_WAIT_N | SCRAMBLED-REVERSED |
| T8 | RXO_SO_WR_WAIT_P | TXI_SO_RD_WAIT_P | SCRAMBLED-REVERSED |
| K2 | RXO_WE_RD_WAIT_N | TXI_WE_RD_WAIT_N | SCRAMBLED-REVERSED |
| J2 | RXO_WE_RD_WAIT_P | TXI_WE_RD_WAIT_P | SCRAMBLED-REVERSED |
| L3 | RXO_WE_WR_WAIT_N | TXI_WE_WR_WAIT_N | REVERSED |
| K3 | RXO_WE_WR_WAIT_P | TXI_WE_WR_WAIT_P | REVERSED |
| G17 | TXI_EA_RD_WAIT_N | TXI_EA_RD_WAIT_P | INVERTED |
| H17 | TXI_EA_RD_WAIT_P | TXI_EA_RD_WAIT_N | INVERTED |
| H16 | TXI_EA_WR_WAIT_N | TXI_EA_WR_WAIT_P | INVERTED |
| J16 | TXI_EA_WR_WAIT_P | TXI_EA_WR_WAIT_N | INVERTED |
| B11 | TXI_NO_RD_WAIT_N | RXI_NO_FRAME_P | SCRAMBLED |
| B12 | TXI_NO_RD_WAIT_P | RXI_NO_FRAME_N | SCRAMBLED |
| C10 | TXI_NO_WR_WAIT_N | RXI_NO_CCLK_P | SCRAMBLED |
| C11 | TXI_NO_WR_WAIT_P | RXI_NO_CCLK_N | SCRAMBLED |
| U12 | TXI_SO_RD_WAIT_N | TXO_SO_FRAME_N | SCRAMBLED-REVERSED |
| U11 | TXI_SO_RD_WAIT_P | TXO_SO_FRAME_P | SCRAMBLED-REVERSED |

| | | | |
|---|---|---|---|
| T11 | TXI_SO_WR_WAIT_N | RXI_SO_CCLK_N | SCRAMBLED-REVERSED |
| T10 | TXI_SO_WR_WAIT_P | RXI_SO_CCLK_P | SCRAMBLED-REVERSED |
| H2 | TXI_WE_RD_WAIT_N | RXO_WE_RD_WAIT_N | SCRAMBLED-REVERSED |
| G2 | TXI_WE_RD_WAIT_P | RXO_WE_RD_WAIT_P | SCRAMBLED-REVERSED |
| J3 | TXI_WE_WR_WAIT_N | RXO_WE_WR_WAIT_N | SCRAMBLED-REVERSED |
| H3 | TXI_WE_WR_WAIT_P | RXO_WE_WR_WAIT_P | SCRAMBLED-REVERSED |
| G15 | TXO_EA_DATA_N[0] | TXO_EA_DATA_P[0] | INVERTED |
| F16 | TXO_EA_DATA_N[1] | TXO_EA_DATA_P[1] | INVERTED |
| E17 | TXO_EA_DATA_N[2] | TXO_EA_DATA_P[2] | INVERTED |
| D18 | TXO_EA_DATA_N[3] | TXO_EA_DATA_P[3] | INVERTED |
| E15 | TXO_EA_DATA_N[4] | TXO_EA_DATA_P[4] | INVERTED |
| D16 | TXO_EA_DATA_N[5] | TXO_EA_DATA_P[5] | INVERTED |
| C17 | TXO_EA_DATA_N[6] | TXO_EA_DATA_P[6] | INVERTED |
| B18 | TXO_EA_DATA_N[7] | TXO_EA_DATA_P[7] | INVERTED |
| H15 | TXO_EA_DATA_P[0] | TXO_EA_DATA_N[0] | INVERTED |
| G16 | TXO_EA_DATA_P[1] | TXO_EA_DATA_N[1] | INVERTED |
| F17 | TXO_EA_DATA_P[2] | TXO_EA_DATA_N[2] | INVERTED |
| E18 | TXO_EA_DATA_P[3] | TXO_EA_DATA_N[3] | INVERTED |
| F15 | TXO_EA_DATA_P[4] | TXO_EA_DATA_N[4] | INVERTED |
| E16 | TXO_EA_DATA_P[5] | TXO_EA_DATA_N[5] | INVERTED |
| D17 | TXO_EA_DATA_P[6] | TXO_EA_DATA_N[6] | INVERTED |
| C18 | TXO_EA_DATA_P[7] | TXO_EA_DATA_N[7] | INVERTED |
| J15 | TXO_EA_FRAME_N | TXO_EA_FRAME_P | INVERTED |
| K15 | TXO_EA_FRAME_P | TXO_EA_FRAME_N | INVERTED |
| F18 | TXO_EA_LCLK_N | TXO_EA_LCLK_P | INVERTED |
| G18 | TXO_EA_LCLK_P | TXO_EA_LCLK_N | INVERTED |
| D11 | TXO_NO_DATA_N[0] | RXI_NO_DATA_P[7] | SCRAMBLED-REVERSED |
| C12 | TXO_NO_DATA_N[1] | RXI_NO_DATA_P[6] | SCRAMBLED-REVERSED |
| B13 | TXO_NO_DATA_N[2] | RXI_NO_DATA_P[5] | SCRAMBLED-REVERSED |
| A14 | TXO_NO_DATA_N[3] | RXI_NO_DATA_P[4] | SCRAMBLED-REVERSED |
| D13 | TXO_NO_DATA_N[4] | RXI_NO_DATA_P[3] | SCRAMBLED-REVERSED |

| | | | |
|---|---|---|---|
| C14 | TXO_NO_DATA_N[5] | RXI_NO_DATA_P[2] | SCRAMBLED-REVERSED |
| B15 | TXO_NO_DATA_N[6] | RXI_NO_DATA_P[1] | SCRAMBLED-REVERSED |
| A16 | TXO_NO_DATA_N[7] | RXI_NO_DATA_P[0] | SCRAMBLED-REVERSED |
| D12 | TXO_NO_DATA_P[0] | RXI_NO_DATA_N[7] | SCRAMBLED-REVERSED |
| C13 | TXO_NO_DATA_P[1] | RXI_NO_DATA_N[6] | SCRAMBLED-REVERSED |
| B14 | TXO_NO_DATA_P[2] | RXI_NO_DATA_N[5] | SCRAMBLED-REVERSED |
| A15 | TXO_NO_DATA_P[3] | RXI_NO_DATA_N[4] | SCRAMBLED-REVERSED |
| D14 | TXO_NO_DATA_P[4] | RXI_NO_DATA_N[3] | SCRAMBLED-REVERSED |
| C15 | TXO_NO_DATA_P[5] | RXI_NO_DATA_N[2] | SCRAMBLED-REVERSED |
| B16 | TXO_NO_DATA_P[6] | RXI_NO_DATA_N[1] | SCRAMBLED-REVERSED |
| A17 | TXO_NO_DATA_P[7] | RXI_NO_DATA_N[0] | SCRAMBLED-REVERSED |
| D9 | TXO_NO_FRAME_N | TXO_NO_FRAME_P | INVERTED |
| D10 | TXO_NO_FRAME_P | TXO_NO_FRAME_N | INVERTED |
| A12 | TXO_NO_LCLK_N | RXI_NO_LCLK_P | INVERTED |
| A13 | TXO_NO_LCLK_P | RXI_NO_LCLK_N | INVERTED |
| R12 | TXO_SO_DATA_N[0] | TXO_SO_DATA_N[7] | SCRAMBLED |
| T13 | TXO_SO_DATA_N[1] | TXO_SO_DATA_N[6] | SCRAMBLED |
| U14 | TXO_SO_DATA_N[2] | TXO_SO_DATA_N[5] | SCRAMBLED |
| V15 | TXO_SO_DATA_N[3] | TXO_SO_DATA_N[4] | SCRAMBLED |
| R14 | TXO_SO_DATA_N[4] | TXO_SO_DATA_N[3] | SCRAMBLED |
| T15 | TXO_SO_DATA_N[5] | TXO_SO_DATA_N[2] | SCRAMBLED |
| U16 | TXO_SO_DATA_N[6] | TXO_SO_DATA_N[1] | SCRAMBLED |
| V17 | TXO_SO_DATA_N[7] | TXO_SO_DATA_N[0] | SCRAMBLED |
| R11 | TXO_SO_DATA_P[0] | TXO_SO_DATA_P[7] | SCRAMBLED |
| T12 | TXO_SO_DATA_P[1] | TXO_SO_DATA_P[6] | SCRAMBLED |
| U13 | TXO_SO_DATA_P[2] | TXO_SO_DATA_P[5] | SCRAMBLED |
| V14 | TXO_SO_DATA_P[3] | TXO_SO_DATA_P[4] | SCRAMBLED |
| R13 | TXO_SO_DATA_P[4] | TXO_SO_DATA_P[3] | SCRAMBLED |
| T14 | TXO_SO_DATA_P[5] | TXO_SO_DATA_P[2] | SCRAMBLED |
| U15 | TXO_SO_DATA_P[6] | TXO_SO_DATA_P[1] | SCRAMBLED |
| V16 | TXO_SO_DATA_P[7] | TXO_SO_DATA_P[0] | SCRAMBLED |

| | | | |
|---|---|---|---|
| R10 | TXO_SO_FRAME_N | RXI_SO_FRAME_N | REVERSED DIRECTION |
| R9 | TXO_SO_FRAME_P | RXI_SO_FRAME_P | REVERSED DIRECTION |
| H4 | TXO_WE_DATA_N[0] | RXI_WE_DATA_N[0] | REVERSED DIRECTION |
| G3 | TXO_WE_DATA_N[1] | RXI_WE_DATA_N[1] | REVERSED DIRECTION |
| F2 | TXO_WE_DATA_N[2] | RXI_WE_DATA_N[2] | REVERSED DIRECTION |
| E1 | TXO_WE_DATA_N[3] | RXI_WE_DATA_N[3] | REVERSED DIRECTION |
| F4 | TXO_WE_DATA_N[4] | RXI_WE_DATA_N[4] | REVERSED DIRECTION |
| E3 | TXO_WE_DATA_N[5] | RXI_WE_DATA_N[5] | REVERSED DIRECTION |
| D2 | TXO_WE_DATA_N[6] | RXI_WE_DATA_N[6] | REVERSED DIRECTION |
| C1 | TXO_WE_DATA_N[7] | RXI_WE_DATA_N[7] | REVERSED DIRECTION |
| G4 | TXO_WE_DATA_P[0] | RXI_WE_DATA_P[0] | REVERSED DIRECTION |
| F3 | TXO_WE_DATA_P[1] | RXI_WE_DATA_P[1] | REVERSED DIRECTION |
| E2 | TXO_WE_DATA_P[2] | RXI_WE_DATA_P[2] | REVERSED DIRECTION |
| D1 | TXO_WE_DATA_P[3] | RXI_WE_DATA_P[3] | REVERSED DIRECTION |
| E4 | TXO_WE_DATA_P[4] | RXI_WE_DATA_P[4] | REVERSED DIRECTION |
| D3 | TXO_WE_DATA_P[5] | RXI_WE_DATA_P[5] | REVERSED DIRECTION |
| C2 | TXO_WE_DATA_P[6] | RXI_WE_DATA_P[6] | REVERSED DIRECTION |
| B1 | TXO_WE_DATA_P[7] | RXI_WE_DATA_P[7] | REVERSED DIRECTION |
| K4 | TXO_WE_FRAME_N | RXI_WE_FRAME_N | REVERSED DIRECTION |
| J4 | TXO_WE_FRAME_P | RXI_WE_FRAME_P | REVERSED DIRECTION |
| G1 | TXO_WE_LCLK_N | RXI_WE_LCLK_N | REVERSED DIRECTION |
| F1 | TXO_WE_LCLK_P | RXI_WE_LCLK_P | REVERSED DIRECTION |

## 10.5 Guide for sharing PCB design between E16G301 & E64G401

With the Parallella-I we demonstrated that it's possible to use the same PCB for both the E16G301 and E64G401 if the following guidelines are followed:

- Use a 1.8V power supply for the DVDD
- Leave pull-up and pull-down resistor foot prints on all COLID and ROWID inputs
- Connect the east eLink interface to an FPGA interface
- Connect the south, north, and west eLinks to off-chip connectors only (or leave unconnected)
- Drive the reset signal from an FPGA

# 11 Copyright Information



Adapteva Inc.

1666 Massachusetts Ave, Suite 14

Lexington, MA 02420

USA

# 12 Disclaimers

THERE IS NO WARRANTY FOR THE DESIGN MATERIALS DESCRIBED IN THIS MANUAL, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE DESIGN MATERIALS “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE DESIGN MATERIALS IS WITH YOU. SHOULD THE DESIGN MATERIALS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION

**Agreement to Defend, Indemnify and Hold Harmless**: You agree to defend, indemnify and hold the Suppliers, its licensors and their representatives harmless from and against any and all claims, damages, losses, expenses, costs and liabilities (collectively, "Claims") arising out of or in connection with any use of the Adapteva products that is not in accordance with the terms of the agreement. This obligation shall apply whether Claims arise under law of tort or contract or any other legal theory, and even if the Adapteva product fails to perform as described or expected.

**Safety-Critical or Life-Critical Applications**: If you intend to evaluate the components for possible use in safety critical applications (such as life support) where a failure of the Supplier’s product would reasonably be expected to cause severe personal injury or death, such as devices which are classified as FDA Class III or similar classification, then you must specifically notify Suppliers of such intent and enter into a separate Assurance and Indemnity Agreement.

# 13 Warranty

Adapteva, Inc (Supplier) provides Epiphany products under the following conditions:

- The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies Supplier from all claims arising from the handling or use of the goods.
- Should the Adapteva product not meet the specifications indicated in this document, the product may be returned within 90 days from the date of delivery to the distributor of purchase for a full refund.

THE FOREGOING LIMITED WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY SELLER TO BUYER AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.