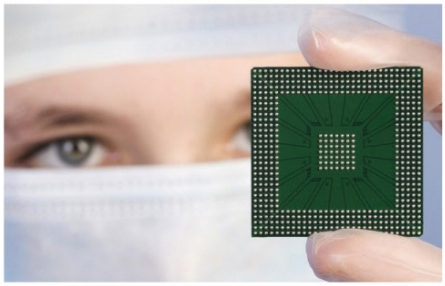
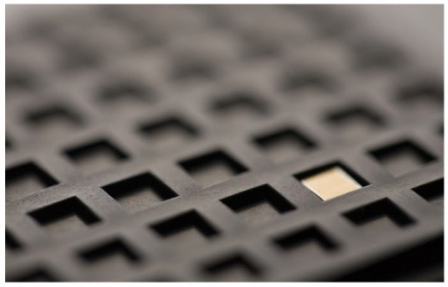
E16G301 EPIPHANY™ 16-CORE MICROPROCESSOR Datasheet



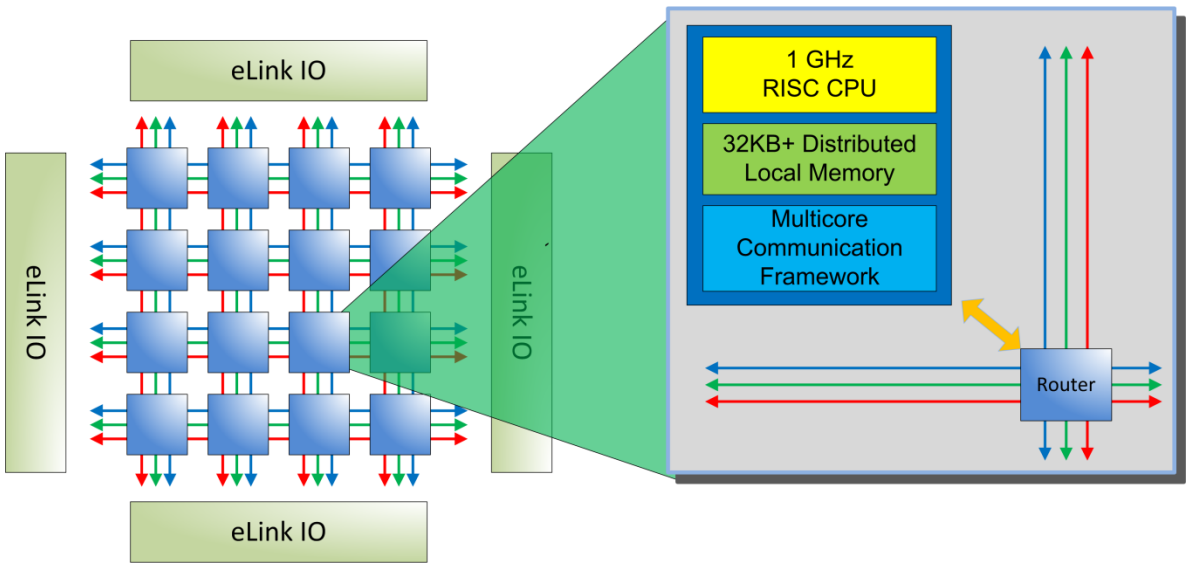


Epiphany - The world leader in microprocessor energy efficiency

The Epiphany is a scalable multicore architecture with up to 4,096 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 real-time processing to performance and power constrained mobile devices like smartphones and tablet computers, as well as improving performance levels for an array of other parallel computing platforms.

E16G301 Features

- 16 High Performance RISC CPU Cores
- Up to 1GHz Operating Frequency
- 32 GFLOPS Peak Performance
- 0.5 MB On-Chip Distributed Shared Memory
- 512 GB/s Local Memory Bandwidth
- 64 GB/s Network-On-Chip Bisection Bandwidth
- 8 GB/s Off-Chip Bandwidth
- 2 Watt Maximum Chip Power Consumption
- Fully-featured ANSI-C/C++ and OpenCL programming environments
- Source synchronous LVDS off chip links for host or direct chip-to-chip interfacing.
- 324-ball 15×15mm 0.8mm pitch BGA



Revision	Changes
3.13.2.13	Initial Revision
3.13.6.14	**General:** Added electrical specs disclaimer notes **Section 3.5:** Added section on unused pins **Section 6.3:** Added new mechanical drawing (RevB) **Section 6.4:** Removed unused signals in pin mapping **Section 6.4:**Corrected incorrect pin map table **Errata**: Added Errata Section **Ordering:** Added part number methodology and ordering information **Electrical:** Changed minimum DVDD voltage to 1.8V

Table of Contents

Preface

This document describes the *E16G301* 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, available from Adapteva.
- *Epiphany Quick-Start Guide*: A tutorial on how to get up and running quickly with the Epiphany development tools, available from Adapteva.

1 Introduction

1.1 Overview

The E16G301 is a 16-core processor based on the 3rd generation of the Epiphany multicore IP. The Epiphany™ architecture defines a multicore, scalable, shared-memory computing fabric comprised of a 2D array of compute nodes connected by a low-latency mesh network-on-chip. Here is a brief summary of the key components of the E16G301:

Processor:

The E16G301 includes 16 superscalar floating point RISC CPUs (eCore), each providing two floating point operations 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 memory map in which each compute node has 32KB 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 network is based on 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. Each E16G301 has 4 independent off-chip links, one in each physical

direction (north, east, west and south). The off chip links allows for glueless connection of multiple E16G301chips 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 E16G301 product can be used in a number of different system configurations, some of which are shown in this section.

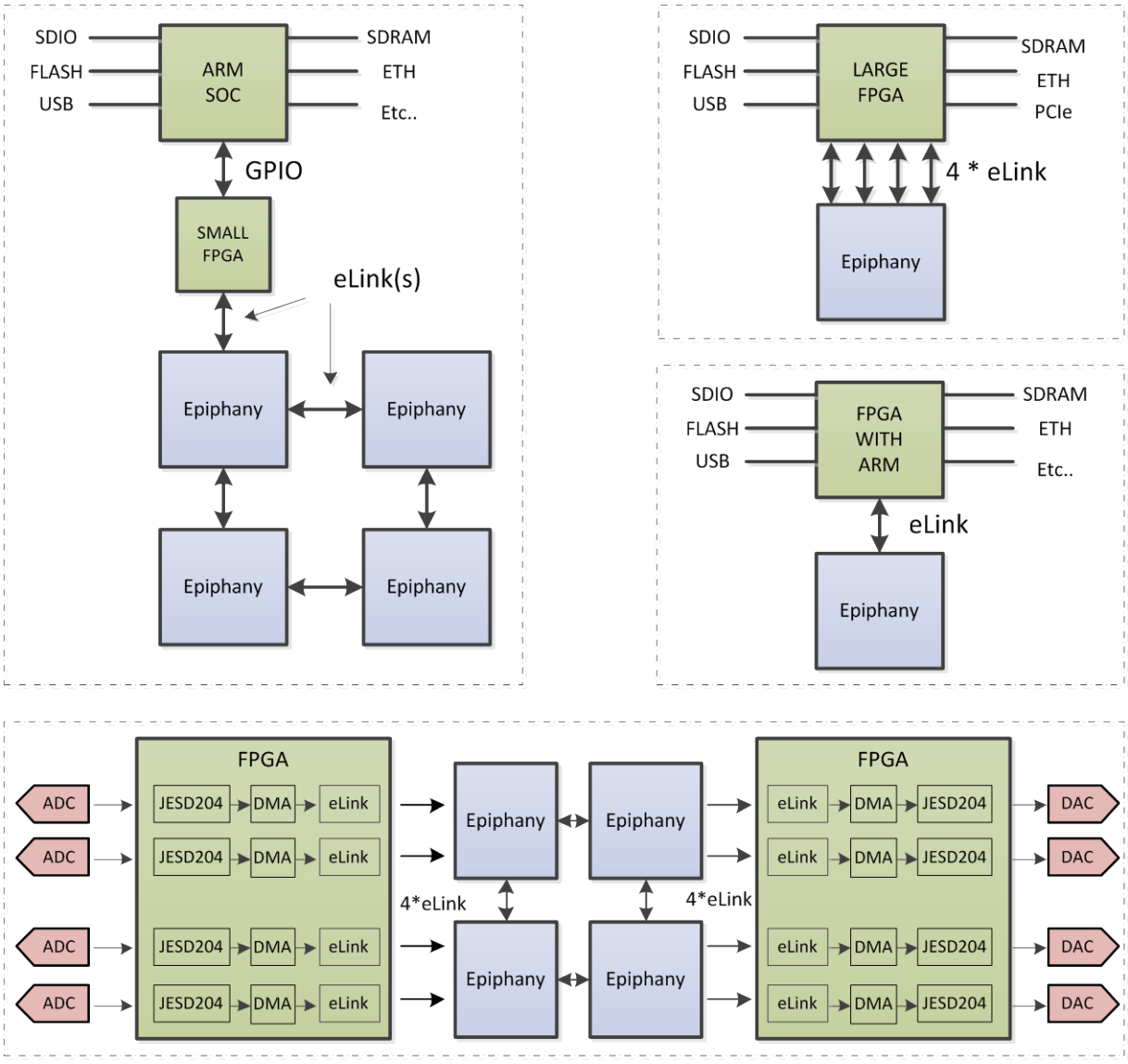


Figure 1: Epiphany System Architecture

1.3 Applications

The following list shows some applications for which the E16G301 is 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 E16G301 within the 32 bit memory map is controlled with the ROWID[3:0] and COLID[3:0] 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 1 shows the distributed memory scheme of the Epiphany architecture. Table 1 shows the distribution of the SRAM within a specific E16G301 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.

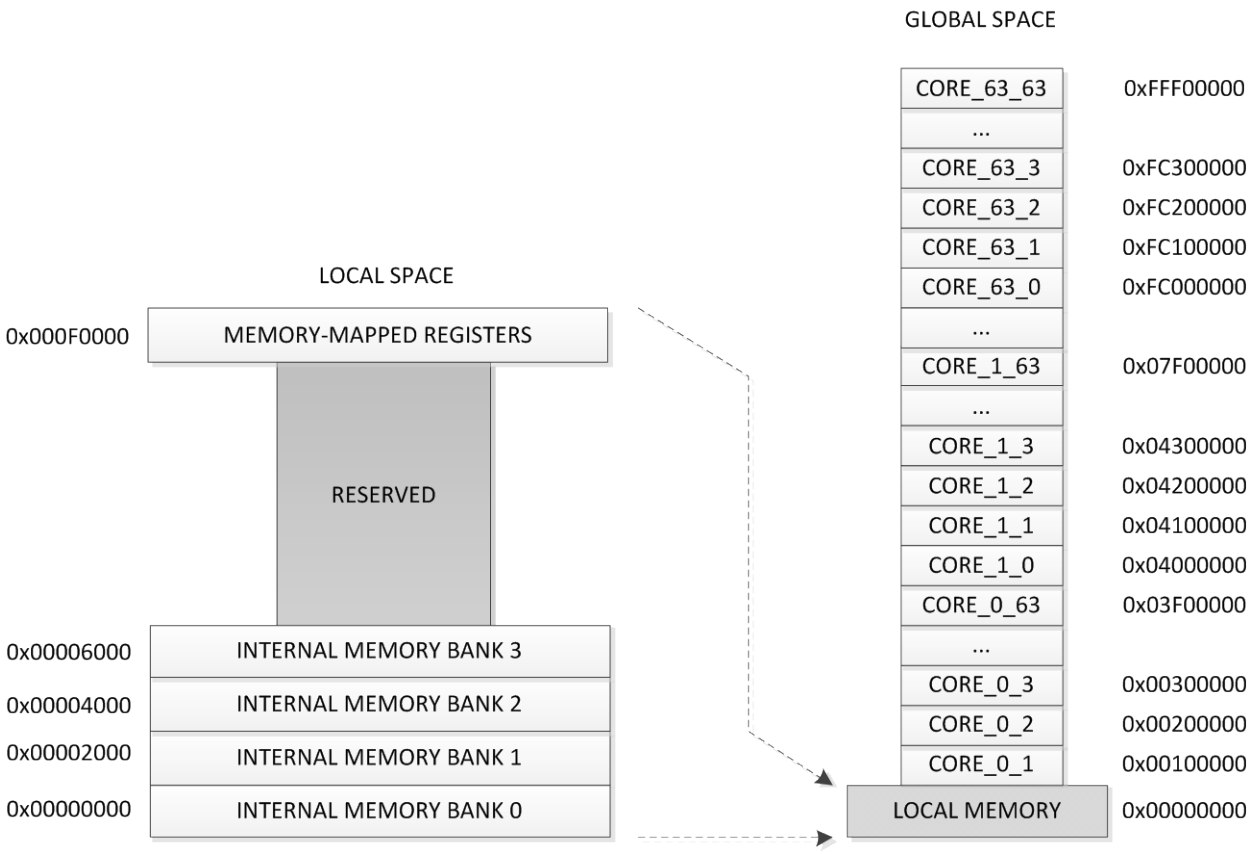


Figure 2: Epiphany Memory Architecture

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 E16G301 has a set of chip 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. 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. For the north link the offset is 0x00600000, for the east link it is 0x08300000, for the south link it is 0x0C600000 and for the west link it is 0x08400000. In all cases, the chip ID (consisting of COLID and ROWID) offset should be added to the entries in the table to derive the complete global address. These registers are cleared by asserting the RESET_N pin.

Register Name	Register Address	Bits	Description
Link Clock Configuration	0xF0300 + link offset	[3:0]	LCLK Transmit Frequency control 0=Divide by 2 1=Divide by 4 2=Divide by 8
Link Transmit Power-down	0xF0304 + link offset	[11:0]	Transmitter power down, 0xFFF=turned off 0x000=turned on
Link Receiver Power-down	0xF0308 + link offset	[11:0]	Receiver power down. 0xFFF=turned off 0x000=turned on
Monitor Register	0x006F0318	[5:0]	0=Sets MONITOR pin to 0 1=Sets MONITOR pin to 1
Chip Reset Register	0x083F0324	[0]	Writing to register creates a three clock cycle long pulse that resets the rest of the chip.

Table 2: Memory Mapped Registers

3 I/O Interfaces

3.1 Overview

The E16G301 includes the following basic signal groups:

- 2 supply domains
- 1 differential input clock
- 1 reset signal
- 7 static control signals
- 24 signal pairs for each one of the four different links (east/west/north/south)

NOTE: Chip input pins do NOT have on-chip pull-down resistors and must be properly driven at the board level at all times.

3.2 Supplies

The chip needs two separate and independent supplies for proper operation. Systems requiring minimal power consumption should operate the supply voltages at their minimally allowed values.

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

Table 3: Chip Supplies

3.3 Reset and Clock

The E16G301 does not have an on-chip PLL and instead receives a high speed LVDS clock directly from a differential LVDS input signal. For high speed operation, the system designer should select one of the high frequency and low-jitter clock drivers available on the market. For lower speed operations, the clock signal can be fed directly by any device (such as an FPGA) that supports standard LVDS electrical signaling.

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 4: Chip Clock and Rest

NOTE: For correct reset wakeup sequencing, the RXI_WE_CCLK_{P/N} signal must be forced held stable for the duration of 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 5: Monitor Signals

3.5 Unused Input Signals

The following signal pairs should have the "_P" pin pulled down to ground and the "_N" pin pulled up to DVDD through resistors.

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

Table 6: 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 E16G301. The COLID[3:0] and ROWID[3:0] set bits [31:28] and [25:22] respectively of the chip's address map. The bits are needed to differentiate between chips in systems containing multiple E16G301 chips.

Signal Name	Signal Description	Direction	Signaling Type
COLID[0]	Bit[28] of chip address map	Input	2.5V CMOS
COLID[1]	Bit[29] of chip address map	Input	2.5V CMOS
COLID[2]	Bit[30] of chip address map	Input	2.5V CMOS
COLID[3]	Bit[31] of chip address map	Input	2.5V CMOS
ROWID[0]	Bit[22] of chip address map	Input	2.5V CMOS
ROWID[1]	Bit[23] of chip address map	Input	2.5V CMOS
ROWID[2]	Bit[24] of chip address map	Input	2.5V CMOS
ROWID[3]	Bit[25] of chip address map	Input	2.5V CMOS

Table 7: Chip Coordinate Settings

3.7 Chip-To-Chip Link Interface

The E16G301 has four identical source-synchronous bidirectional off chip LVDS links (eLink) that can be used for connecting a chip to other E16G301 chips, FPGAs, and/or ASICs. Interfacing the E16G301 with an FPGA is done by instantiating the eLink HDL source code provided by Adapteva. The following two figures demonstrate how the eLink can be used to connect to an FPGA module and to connect multiple E16G301 chips together.

The eLink is non-blocking and allows interleaving of independent data packets as small as single byte packets. Optimal bandwidth is achieved when a large number of incrementally numbered 64 bit data packets are sent consecutively.

The txo_* interface driven out from the E16G301 is a divided version of the core cock frequency (CCLK). The transmit clock is automatically aligned in the middle of the data eye by the eLink transmitter 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.

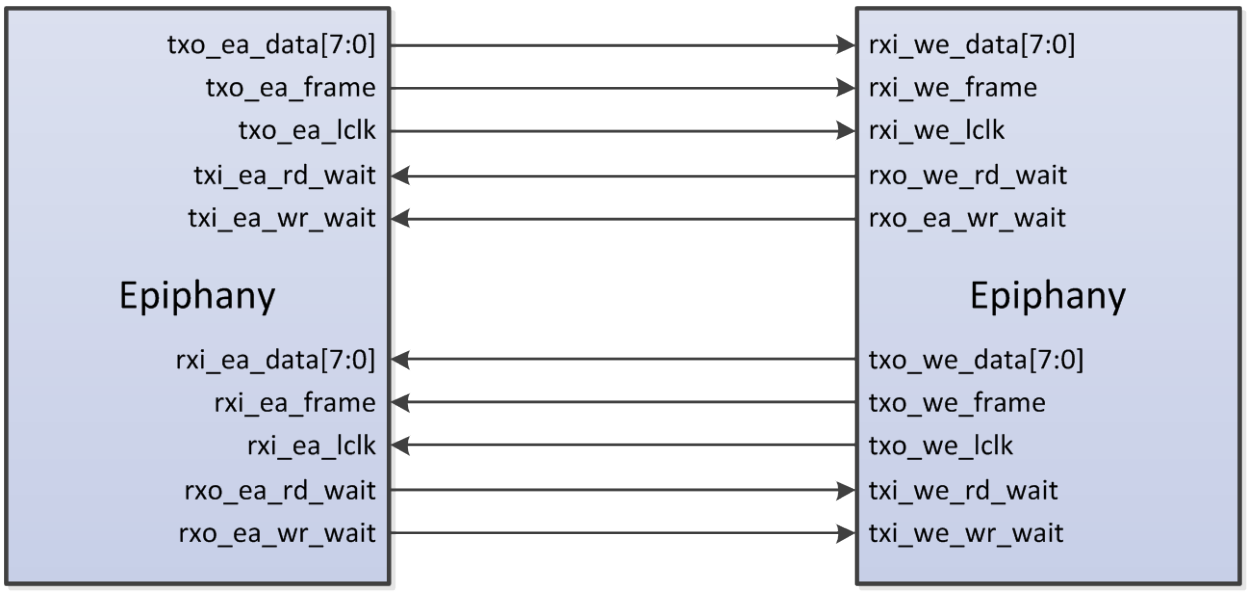


Figure 3: eLink Chip Interface

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	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	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	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	Push-back from transmitter indicating that transmitter must hold off on sending another read packet.

Table 8: eLink Signals

4 System Integration

4.1 FPGA/ASIC Interfacing

The E16G301 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.

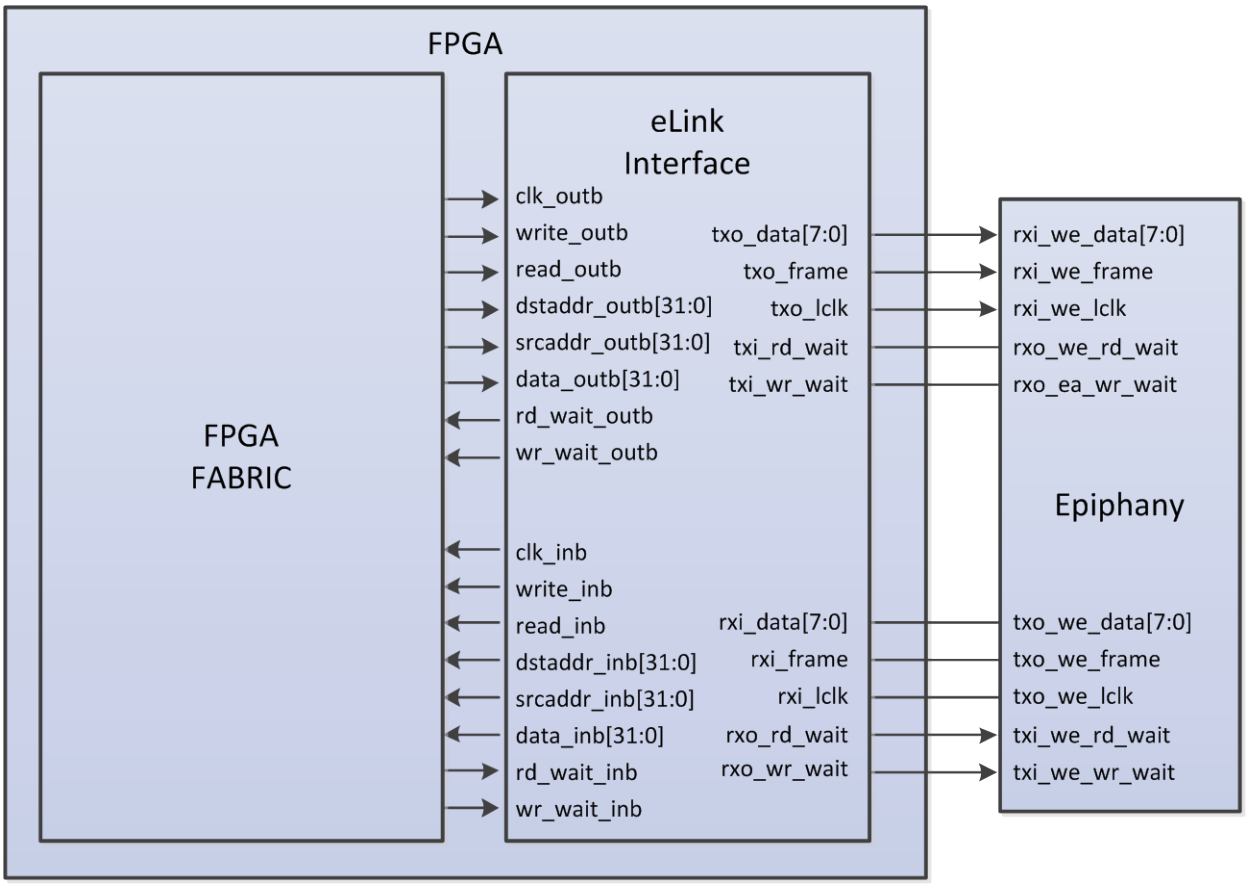


Figure 4: FPGA eLink Integration Example

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. 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 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 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 9: 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. 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 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 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 10: Outgoing Transaction (FROM FPGA/ASIC)

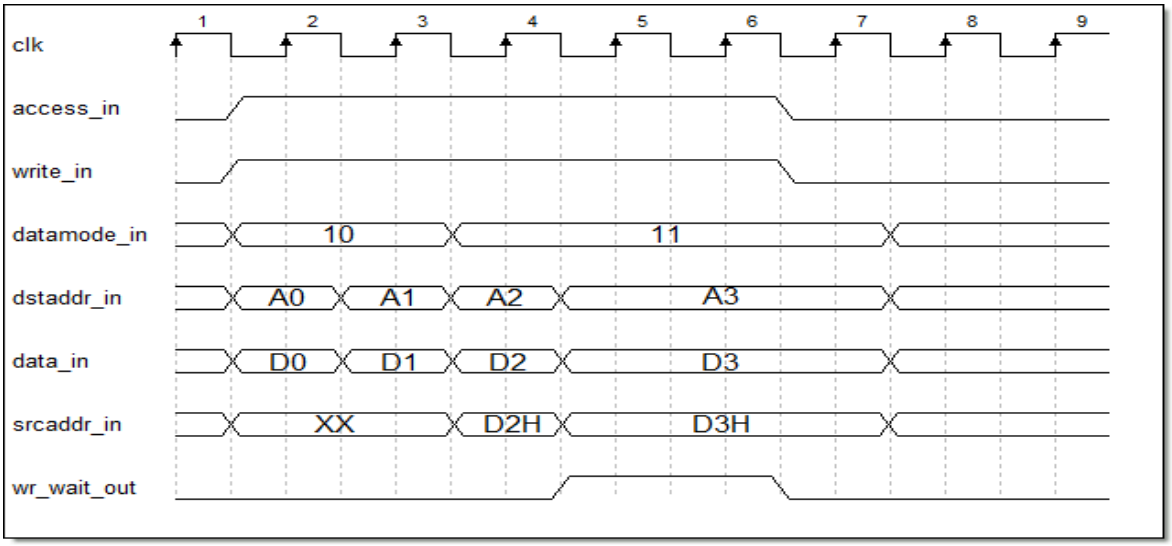


Figure 5: eLink Write Transaction Timing Diagram

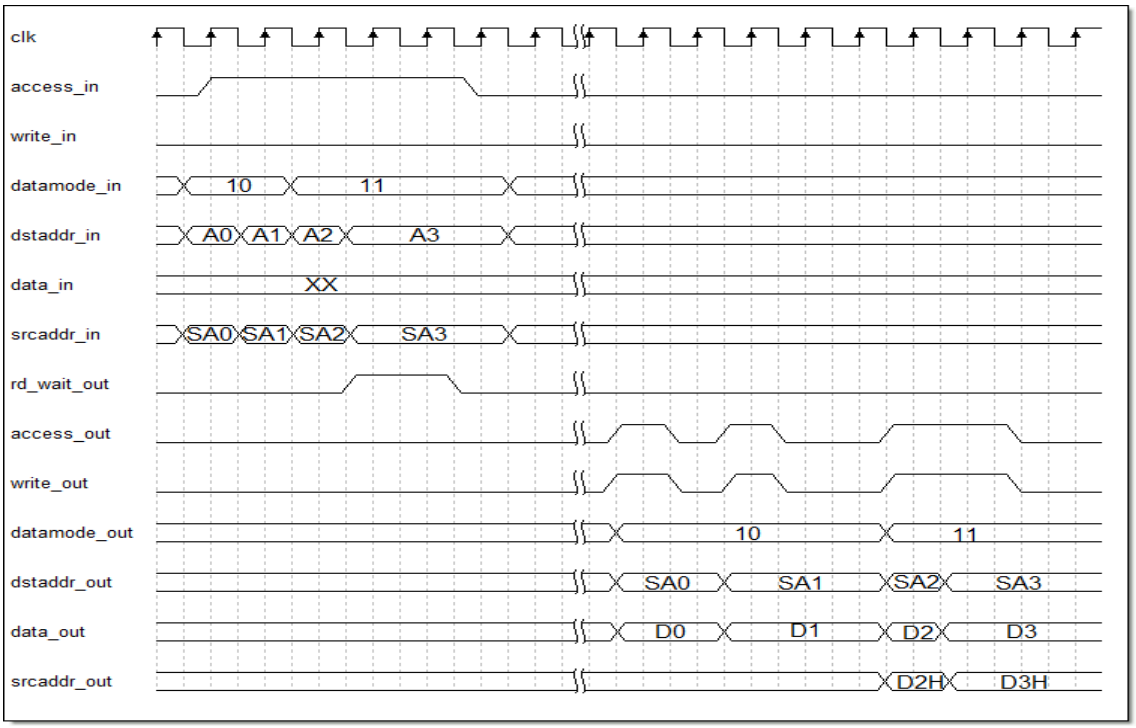


Figure 6: eLink Read Transaction Timing Diagram

5 Electrical Specifications

5.1 Maximum Absolute Ratings

Parameter	Description	Min	Typical	Max	Unit
DVDD	IO Supply Voltage Range	1.8	2.5	2.75	V
VDD	Core Supply Voltage Range	0.90	1.0	1.2	V
V_{IO}	High Level Input Voltage	-0.3		DVDD+0.3	V
T_J	Junction Temp	-40		125	°C

Table 11: Maximum Absolute Ratings

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

5.2 CMOS Signaling

The monitor, address map, and reset inputs use LVCMOS signaling levels and should be considered static pins.

5.3 LVDS Signaling

The E16G301 conforms to the standard LVDS specifications (IEEE Std 1596.3-1996, Low Voltage Differential Signaling). The driver design has all the necessary components for transmission of LVDS data and a temperature stable internal reference for setting of the LVDS signaling voltage and common mode level. The LVDS receiver does not comply with the hysteresis requirements of the TIA and IEEE standards for LVDS differential signaling at the specified rates.

Parameter	Description	Min	Typical	Max	Unit
V_{INPUT}	Common Mode Input Voltage	0	1.2	DVDD-0.1	V
V_{IDT}	Input differential threshold	75			mV
V_{ID}	Input differential voltage	100		200	mV
$V_{IA,IB}$	Input voltage	0	1.2	DVDD-0.1	V
$I_{IA,IA}$	Input leakage current			5	uA
$t_{psupply}$	Power supply sensitivity		0.15	0.3	ps/mV
t_{jps}	Total receiver pk-pk jitter		100	250	Ps
t_{rise}	Output differential rise time	150		300	Ps
t_{fall}	Output differential fall time	150		300	Ps
t_{OH}	Output voltage high		1365	1485	mV
t_{OL}	Output voltage low	960	1035		mV
t_{OD}	Differential output voltage	250	325	410	mV
t_{OS}	Output offset voltage	1170	1200	1230	mV
$I_{SA,SB}$	Output short circuit current	-40	10	40	mA

Table 12: LVDS Electrical Specifications

NOTE: LVDS electrical specifications are based on simulation results. Final product qualification data not yet available, information provided without warranty.

5.4 Power Consumption

The following table shows typical power consumption of the E16G301 running at 700MHz with DVDD=1.8V, VDD=1.0V.

Device Feature	Power Consumption
Single Core Max Power	TBD
Single eLink Transmitter	TBD
Single eLink Receiver	TBD
Maximum Chip Power	TBD
Minimal Standby Power	TBD

Table 13: Power Consumption Measurements

6 Device Package

6.1 Overview

The E16G301 uses a 324 ball 0.8 mm pitch wire-bond BGA package that measures 15×15mm.

6.2 Graphical Pin Mapping

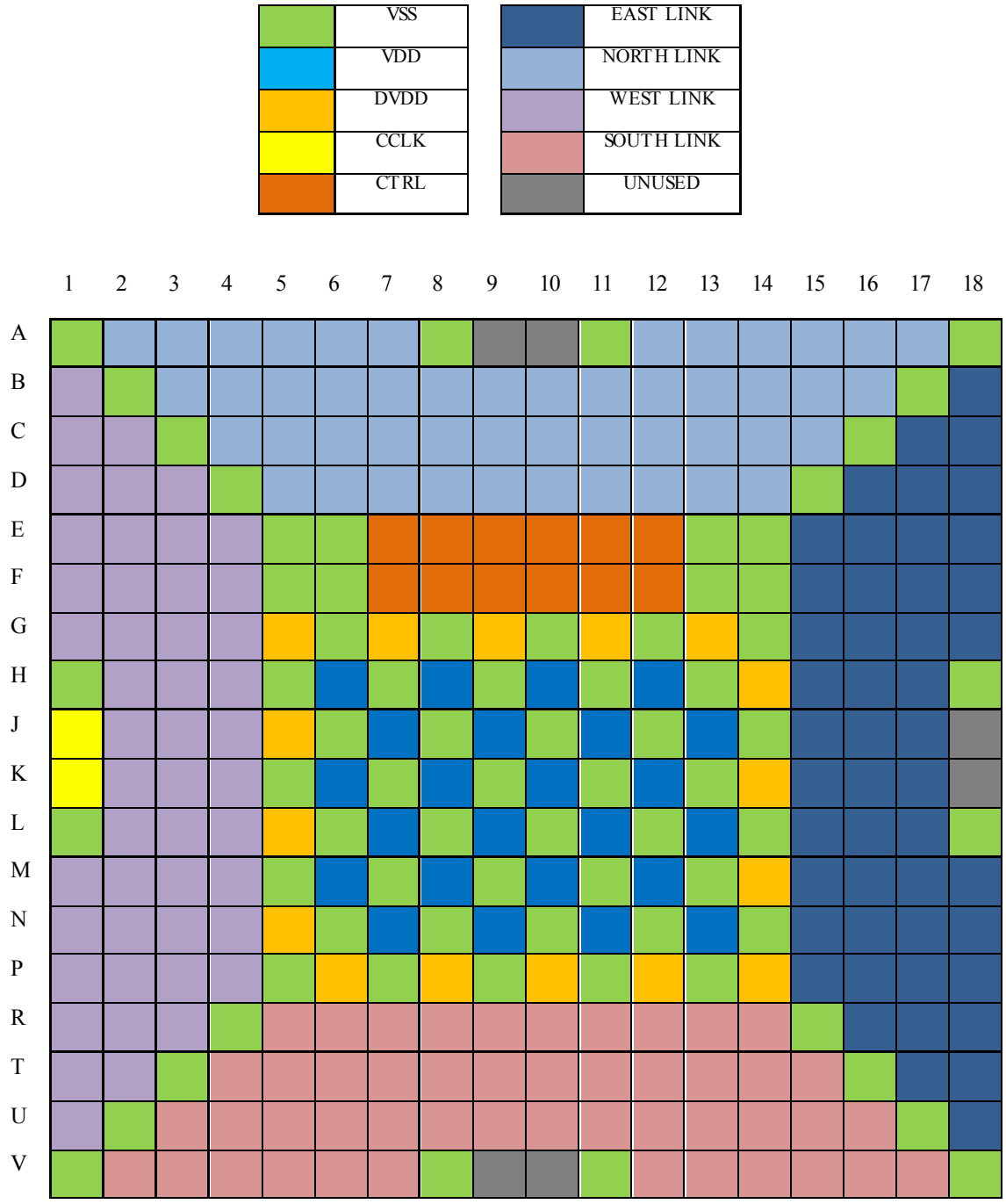
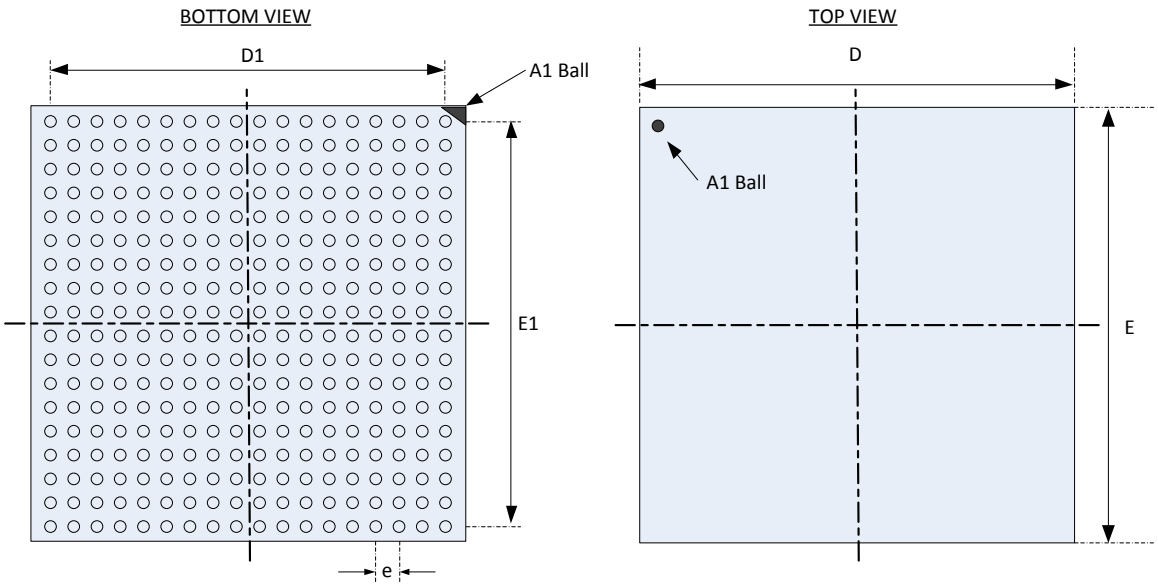
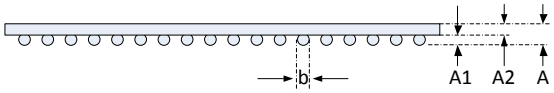


Figure 7: Pin Mapping Diagram

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		

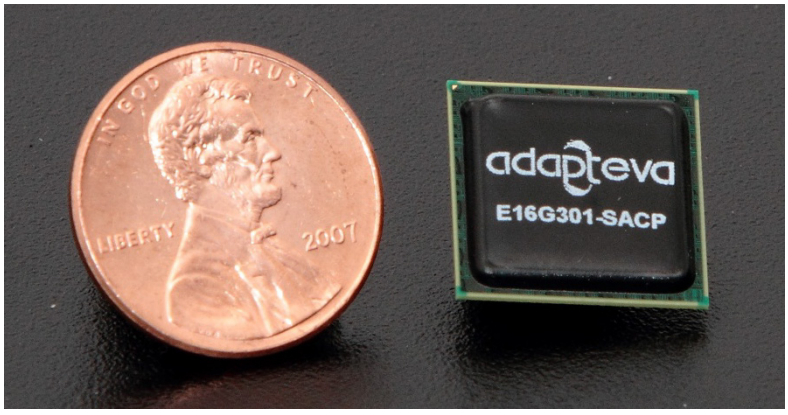
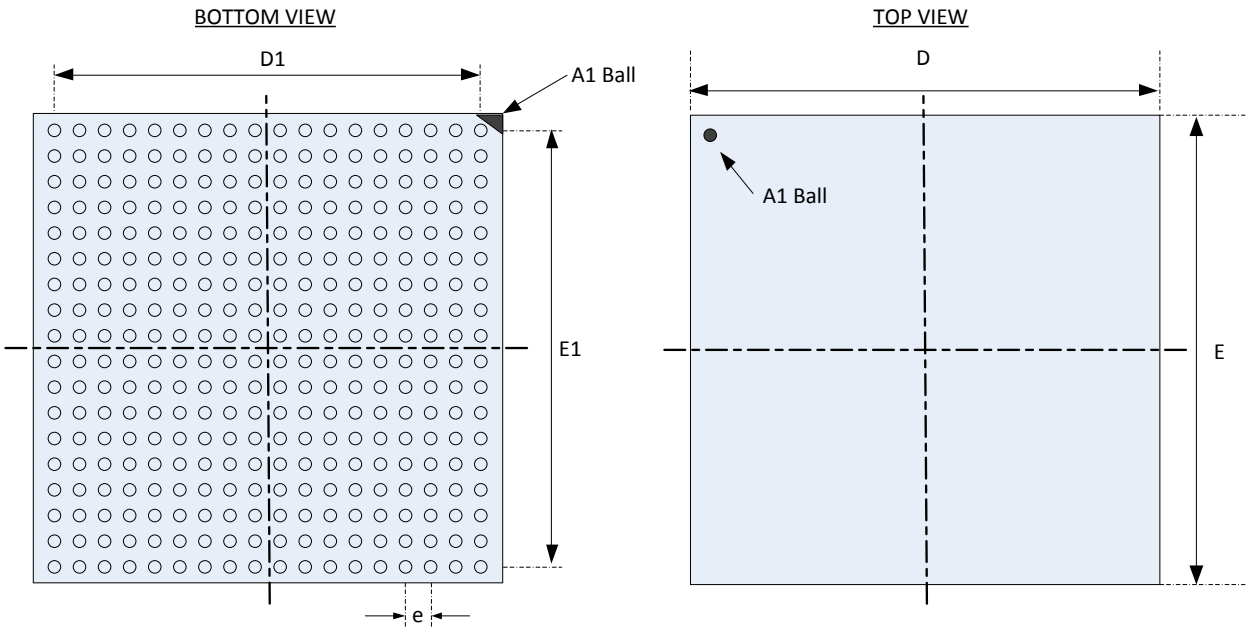
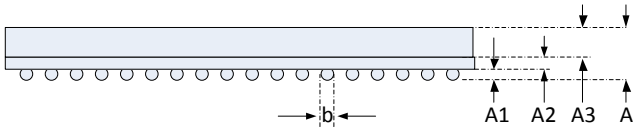


Figure 8: Rev-A Package





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)
Total Thickness	A	~	~	1.50
Standoff	A1	0.25	~	0.40
Substrate Height	A2	~	0.26	~
Mold+Heatslug	A3	~	0.8	~
Body Size	D/E	15.00		
Ball Array Size	D1/E1	13.60		
Ball Pitch	e	0.80		
Ball Size	b	0.40	0.45	0.50

Figure 9: Rev-B Package

6.4 Complete Package Pin-out

PIN	BGA BALL
VSS (72)	A1,A8,A11,A18,B2,B17,C3,C16,D4,D15,E5,E6,E13,E14,F5,F6,F13,F14,G6,G8,G10,G12,G14,H1,H13,H18,J6,J8,J10,J12,J14,K5,K7,K9,K11,K13,L1,L6,L8,L10,L12,L14,L18,M5,M7,M9,M11,M13,N6,N8,N10,N12,N14,P5,P7,P9,P11,P13,R4,R15,T3,T16,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,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_N	J18
RXI_EA_CCLK_P	K18
RXI_EA_DATA_N[0]	T18
RXI_EA_DATA_N[1]	R17
RXI_EA_DATA_N[2]	P16
RXI_EA_DATA_N[3]	N15
RXI_EA_DATA_N[4]	P18
RXI_EA_DATA_N[5]	N17
RXI_EA_DATA_N[6]	M16
RXI_EA_DATA_N[7]	L15

RXI_EA_DATA_P[0]	U18
RXI_EA_DATA_P[1]	T17
RXI_EA_DATA_P[2]	R16
RXI_EA_DATA_P[3]	P15
RXI_EA_DATA_P[4]	R18
RXI_EA_DATA_P[5]	P17
RXI_EA_DATA_P[6]	N16
RXI_EA_DATA_P[7]	M15
RXI_EA_FRAME_N	L17
RXI_EA_FRAME_P	M17
RXI_EA_LCLK_N	M18
RXI_EA_LCLK_P	N18
RXI_EA_RD_WAIT_N	J17
RXI_EA_RD_WAIT_P	K17
RXI_EA_WR_WAIT_N	K16
RXI_EA_WR_WAIT_P	L16
RXI_NO_CCLK_N	A9
RXI_NO_CCLK_P	A10
RXI_NO_DATA_N[0]	A2
RXI_NO_DATA_N[1]	B3
RXI_NO_DATA_N[2]	C4
RXI_NO_DATA_N[3]	D5
RXI_NO_DATA_N[4]	A4
RXI_NO_DATA_N[5]	B5
RXI_NO_DATA_N[6]	C6
RXI_NO_DATA_N[7]	D7
RXI_NO_DATA_P[0]	A3
RXI_NO_DATA_P[1]	B4
RXI_NO_DATA_P[2]	C5
RXI_NO_DATA_P[3]	D6
RXI_NO_DATA_P[4]	A5
RXI_NO_DATA_P[5]	B6
RXI_NO_DATA_P[6]	C7
RXI_NO_DATA_P[7]	D8
RXI_NO_FRAME_N	B7
RXI_NO_FRAME_P	B8
RXI_NO_LCLK_N	A6
RXI_NO_LCLK_P	A7

RXI_NO_RD_WAIT_N	B9
RXI_NO_RD_WAIT_P	B10
RXI_NO_WR_WAIT_N	C8
RXI_NO_WR_WAIT_P	C9
RXI_SO_CCLK_N	V10
RXI_SO_CCLK_P	V9
RXI_SO_DATA_N[0]	V3
RXI_SO_DATA_N[1]	U4
RXI_SO_DATA_N[2]	T5
RXI_SO_DATA_N[3]	R6
RXI_SO_DATA_N[4]	V5
RXI_SO_DATA_N[5]	U6
RXI_SO_DATA_N[6]	T7
RXI_SO_DATA_N[7]	R8
RXI_SO_DATA_P[0]	V2
RXI_SO_DATA_P[1]	U3
RXI_SO_DATA_P[2]	T4
RXI_SO_DATA_P[3]	R5
RXI_SO_DATA_P[4]	V4
RXI_SO_DATA_P[5]	U5
RXI_SO_DATA_P[6]	T6
RXI_SO_DATA_P[7]	R7
RXI_SO_FRAME_N	U8
RXI_SO_FRAME_P	U7
RXI_SO_LCLK_N	V7
RXI_SO_LCLK_P	V6
RXI_SO_RD_WAIT_N	U10
RXI_SO_RD_WAIT_P	U9
RXI_SO_WR_WAIT_N	T9
RXI_SO_WR_WAIT_P	T8
RXI_WE_CCLK_N	K1
RXI_WE_CCLK_P	J1
RXI_WE_DATA_N[0]	U1
RXI_WE_DATA_N[1]	T2
RXI_WE_DATA_N[2]	R3
RXI_WE_DATA_N[3]	P4
RXI_WE_DATA_N[4]	R1
RXI_WE_DATA_N[5]	P2

RXI_WE_DATA_N[6]	N3
RXI_WE_DATA_N[7]	M4
RXI_WE_DATA_P[0]	T1
RXI_WE_DATA_P[1]	R2
RXI_WE_DATA_P[2]	P3
RXI_WE_DATA_P[3]	N4
RXI_WE_DATA_P[4]	P1
RXI_WE_DATA_P[5]	N2
RXI_WE_DATA_P[6]	M3
RXI_WE_DATA_P[7]	L4
RXI_WE_FRAME_N	M2
RXI_WE_FRAME_P	L2
RXI_WE_LCLK_N	N1
RXI_WE_LCLK_P	M1
RXI_WE_RD_WAIT_N	K2
RXI_WE_RD_WAIT_P	J2
RXI_WE_WR_WAIT_N	L3
RXI_WE_WR_WAIT_P	K3
TXO_EA_DATA_N[0]	G15
TXO_EA_DATA_N[1]	F16
TXO_EA_DATA_N[2]	E17
TXO_EA_DATA_N[3]	D18
TXO_EA_DATA_N[4]	E15
TXO_EA_DATA_N[5]	D16
TXO_EA_DATA_N[6]	C17
TXO_EA_DATA_N[7]	B18
TXO_EA_DATA_P[0]	H15
TXO_EA_DATA_P[1]	G16
TXO_EA_DATA_P[2]	F17
TXO_EA_DATA_P[3]	E18
TXO_EA_DATA_P[4]	F15
TXO_EA_DATA_P[5]	E16
TXO_EA_DATA_P[6]	D17
TXO_EA_DATA_P[7]	C18
TXO_EA_FRAME_N	J15
TXO_EA_FRAME_P	K15
TXO_EA_LCLK_N	F18
TXO_EA_LCLK_P	G18

TXO_EA_RD_WAIT_N	G17
TXO_EA_RD_WAIT_P	H17
TXO_EA_WR_WAIT_N	H16
TXO_EA_WR_WAIT_P	J16
TXO_NO_DATA_N[0]	D11
TXO_NO_DATA_N[1]	C12
TXO_NO_DATA_N[2]	B13
TXO_NO_DATA_N[3]	A14
TXO_NO_DATA_N[4]	D13
TXO_NO_DATA_N[5]	C14
TXO_NO_DATA_N[6]	B15
TXO_NO_DATA_N[7]	A16
TXO_NO_DATA_P[0]	D12
TXO_NO_DATA_P[1]	C13
TXO_NO_DATA_P[2]	B14
TXO_NO_DATA_P[3]	A15
TXO_NO_DATA_P[4]	D14
TXO_NO_DATA_P[5]	C15
TXO_NO_DATA_P[6]	B16
TXO_NO_DATA_P[7]	A17
TXO_NO_FRAME_N	D9
TXO_NO_FRAME_P	D10
TXO_NO_LCLK_N	A12
TXO_NO_LCLK_P	A13
TXO_NO_RD_WAIT_N	B11
TXO_NO_RD_WAIT_P	B12
TXO_NO_WR_WAIT_N	C10
TXO_NO_WR_WAIT_P	C11
TXO_SO_DATA_N[0]	R12
TXO_SO_DATA_N[1]	T13
TXO_SO_DATA_N[2]	U14
TXO_SO_DATA_N[3]	V15
TXO_SO_DATA_N[4]	R14
TXO_SO_DATA_N[5]	T15
TXO_SO_DATA_N[6]	U16
TXO_SO_DATA_N[7]	V17
TXO_SO_DATA_P[0]	R11
TXO_SO_DATA_P[1]	T12

TXO_SO_DATA_P[2]	U13
TXO_SO_DATA_P[3]	V14
TXO_SO_DATA_P[4]	R13
TXO_SO_DATA_P[5]	T14
TXO_SO_DATA_P[6]	U15
TXO_SO_DATA_P[7]	V16
TXO_SO_FRAME_N	R10
TXO_SO_FRAME_P	R9
TXO_SO_LCLK_N	V13
TXO_SO_LCLK_P	V12
TXO_SO_RD_WAIT_N	U12
TXO_SO_RD_WAIT_P	U11
TXO_SO_WR_WAIT_N	T11
TXO_SO_WR_WAIT_P	T10
TXO_WE_DATA_N[0]	H4
TXO_WE_DATA_N[1]	G3
TXO_WE_DATA_N[2]	F2
TXO_WE_DATA_N[3]	E1
TXO_WE_DATA_N[4]	F4
TXO_WE_DATA_N[5]	E3
TXO_WE_DATA_N[6]	D2
TXO_WE_DATA_N[7]	C1
TXO_WE_DATA_P[0]	G4
TXO_WE_DATA_P[1]	F3
TXO_WE_DATA_P[2]	E2
TXO_WE_DATA_P[3]	D1
TXO_WE_DATA_P[4]	E4
TXO_WE_DATA_P[5]	D3
TXO_WE_DATA_P[6]	C2
TXO_WE_DATA_P[7]	B1
TXO_WE_FRAME_N	K4
TXO_WE_FRAME_P	J4
TXO_WE_LCLK_N	G1
TXO_WE_LCLK_P	F1
TXO_WE_RD_WAIT_N	H2
TXO_WE_RD_WAIT_P	G2
TXO_WE_WR_WAIT_N	J3
TXO_WE_WR_WAIT_P	H3

7 Ordering Information

7.1 Part Number Naming Methodology

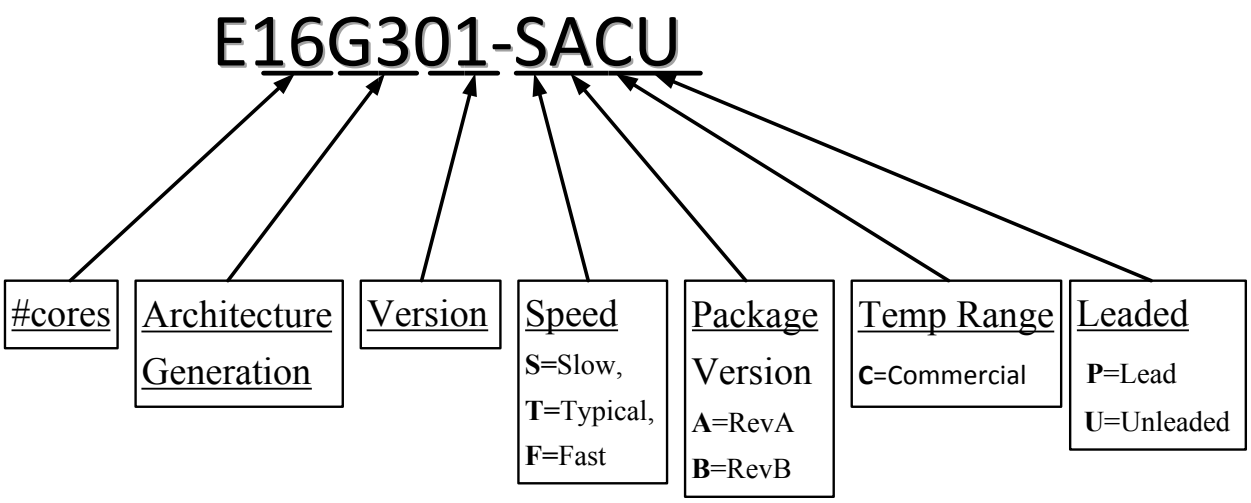


Figure 10: Part Numbering Methodology

7.2 Product Availability

Part Number	Temp Range	Speed Grade	Comment
E16G301-SACP	–40°C to +85°C	700MHz	End-Of-Life
E16G301-SBCU	–40°C to +85°C	700MHz	Sampling Q3, 2013
E16G301-TBCU	–40°C to +85°C	800MHz	Planned for Q4, 2013
E16G301-FBCU	–40°C to +85°C	1 GHz	Planned for Q4, 2013

Table 14: Power Consumption Measurements

8 Errata

This table contains all the known errata for the E16G301 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	Reset sensitivity	E16G301	Functional	To guarantee a correct and repeatable reset wakeup sequencing, the RXI_WE_CCLK_{P/N} signal must be held stable for the duration of the rising edge of RESET_N.
1	DMA Throttle	E16G301	Performance	The DMA engine bandwidth per channel is stuck at 50% throttle, meaning that each DMA channel can transfer at most 1 double word every two clock cycles.
2	NOC FIFO Full	E16G301	Performance	The FIFO interface between the compute note and the Network On Chip currently indicates FIFO full too early, causing a degradation in peak outgoing transfer bandwidth from the Epiphany processor node on the mesh NOC.

Table 15: Errata Table

Adapteva Inc.

1666 Massachusetts Ave, Suite 14

Lexington, MA 02420

USA