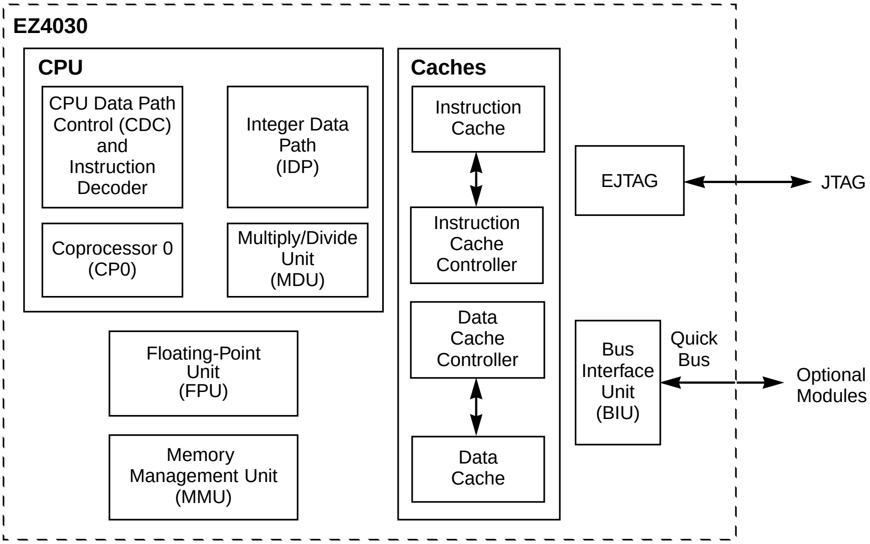
MiniRISC® EZ4030 EasyMACRO™ Microprocessor

Preliminary Datasheet

The MiniRISC EZ4030 EasyMACRO Microprocessor is a compact, high-performance, 64-bit microprocessor subsystem developed by LSI Logic Corporation. The EZ4030 uses CoreWare® system-on-a-chip methodology and executes the MIPS III Instruction Set Architecture (ISA). It is ideal for high-performance, cost-sensitive embedded processor applications. As shown in Figure 1, the EZ4030 is composed of the following components:

- CPU consisting of system coprocessor, integer data path, and the CPU data path control and instruction decoder
- Separate instruction and data caches
- Memory management unit
- Floating-point coprocessor
- Bus interface unit (BIU) implementing the Quick Bus protocol
- Enhanced JTAG (EJTAG) module with Joint Test Action Group (JTAG) interface

Figure 1 EZ4030 Block Diagram



Features and Benefits

- High-Performance RISC CPU
  - Single issue, 5-stage pipeline
  - 250 native MIPS, 275 Dhrystone MIPS, and 160 Whetstone MIPS at 250 MHz
  - 250 MHz operation at WCABS (Tj = 125 °C, VDD = 1.71 V, WC process)
  - Both big- and little-endian support for load and store operations
  - R4000 standard, 32-bit timer/counter
  - MIPS CPU standard interrupt exceptions (one nonmaskable interrupt (NMI), one timer, five hardware, two software)
- High-Performance Memory System
  - Harvard cache architecture with integrated instruction and data caches
  - 16 Kbyte, 2-way, set-associative instruction and data caches
    - ◊ Least recently used (LRU) algorithm for replacement
    - ◊ Line level lock for instruction RAM and scratchpad memory
    - ◊ Data cache has write-through or write-back update policy, programmable on a page basis
    - ◊ Nonblocking operation
  - Instruction and data streaming
  - Instruction prefetching
  - Unaligned data access support
- MIPS III Instruction Set Architecture
  - MIPS III ISA supporting 64-bit operations
  - Thirty-two 64-bit, general-purpose registers
  - Thirty-two 64-bit, floating point registers
  - R4000-style status register and exception processing
  - Wait for Interrupt (WAITI) instruction for power saving
  - Supports SPECIAL2 Multiply-Accumulate extensions
- Integrated Floating-Point Coprocessor (CP1)
  - Single- and double-precision arithmetic
  - IEEE Std 754-compliant
- Integrated Integer Multiply-and-Divide unit
  - High-performance, 8-bit/cycle multiplier

Multiply Type (Bits)	Latency Cycles	Repeat Rate Cycles
32	5	5
64	9	9

  - Compact, 1-bit/cycle divider

Divide Type (Bits)	Latency Cycles	Repeat Rate Cycles
32	34	34
64	66	66

Features and Benefits (Cont.)

- System Interface
  - 64-bit split transaction Quick Bus
  - 1x, 1/2x, 1/3x processor frequency
  - AMBA AHB support through external bridge
- Advanced Debug Support
  - MIPS EJTAG 2.0
  - Instruction and data breakpoints
  - Real-time Program Counter (PC) trace
  - Processor single step and software debug breakpoints
- MMU supports complex operating systems such as Linux and Windows CE
- Technology
  - Size: 12.5 mm^2
  - Power: 3.0 mW/MHz (estimated)
  - 1.8 V Core VDD
  - G12™ CMOS technology (0.18 μ L-drawn, 0.15 μ L-effective)

Application Examples

The following applications are ideal for the EZ4030 microprocessor subsystem:

- Broadband
- Networking
- Digital Consumer
- Printing Systems
- Storage Systems

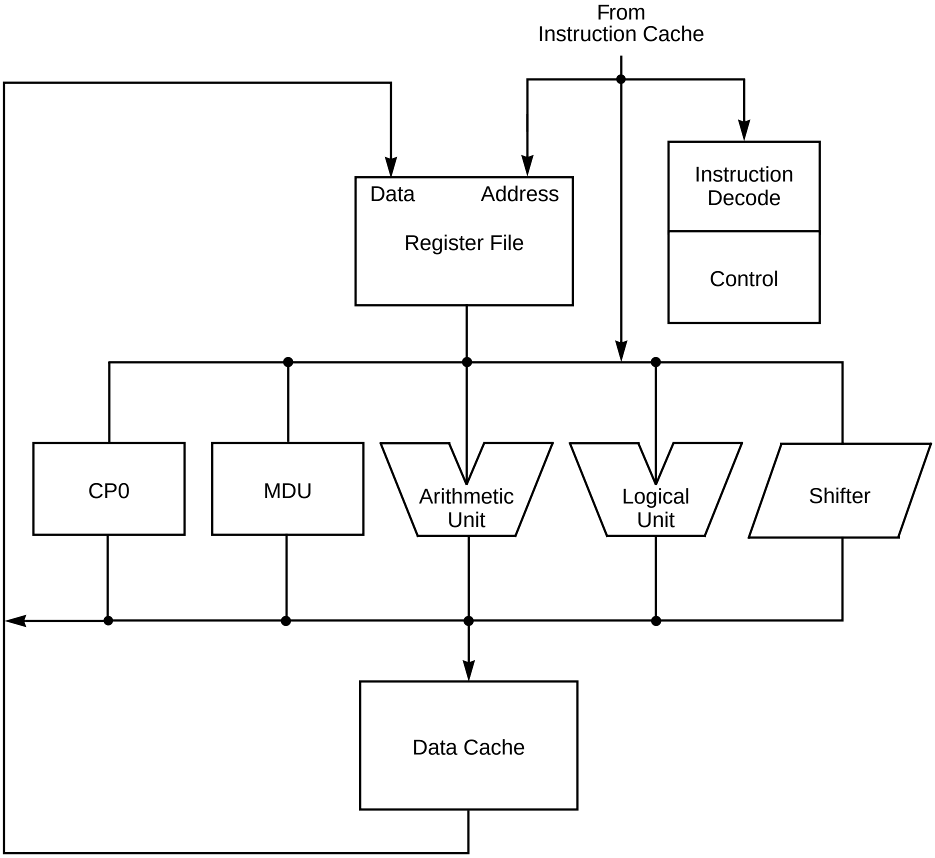
Functional Description

The EZ4030 is a high-performance RISC microprocessor core that includes the following functional units:

- Central Processing Unit (CPU)

  The CPU (see Figure 2) uses a single-issue, 5-stage pipeline that operates at 250 MHz. It includes an integer data path module, a control module, and system coprocessor 0 (CP0). The 64-bit integer data path includes a 32 x 64-bit register file and an integrated MDU. The control module provides integer data path control and instruction decoding. CP0 provides exception processing support using the MIPS R4000 exception model and processor state control, which includes three operating modes: kernel, user, and supervisor. It also offers a debugging capability with support for a software debug breakpoint instruction and single-step debugging.

Figure 2 CPU Block Diagram



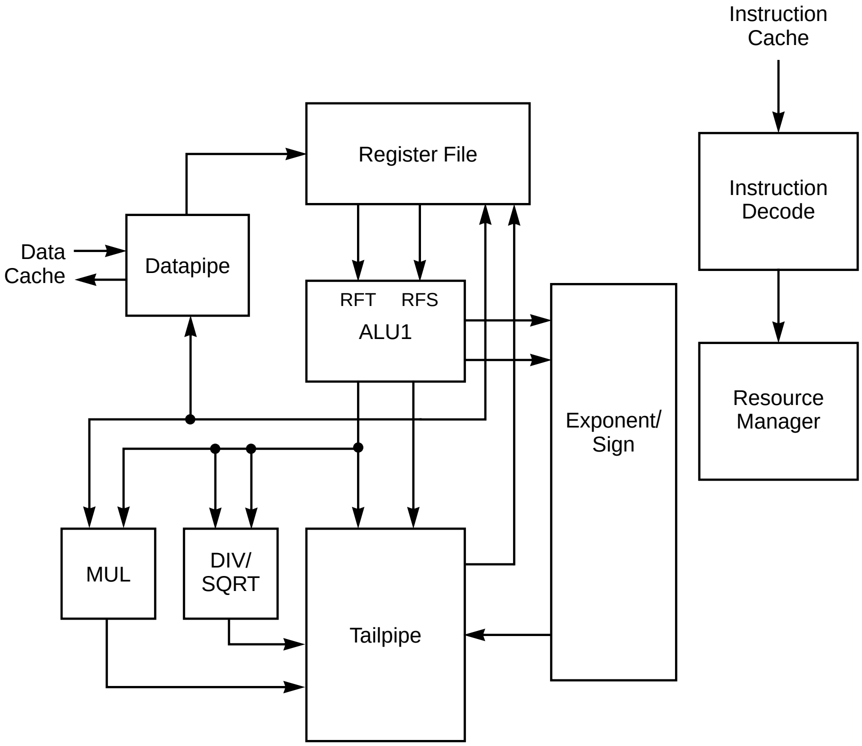
- Memory Management Unit (MMU)

  The MMU performs virtual-to-physical address translation using a 32-entry, joint translation lookaside buffer (TLB). The MMU supports a variety of page sizes from 4 Kbytes to 16 Mbytes.

- Floating-Point Unit (FPU)

  The FPU (see Figure 3) extends the CPU instruction set to perform arithmetic operations on floating-point values. Like the CPU, the FPU instruction set is load- and store-based. The FPU, along with its associated system software, fully conforms to the ANSI/IEEE 754-1985 Standard for Binary Floating-Point Arithmetic. Together the CPU and FPU form a tightly-coupled unit with seamless integration of floating-point and fixed-point operations.

Figure 3 FPU Block Diagram



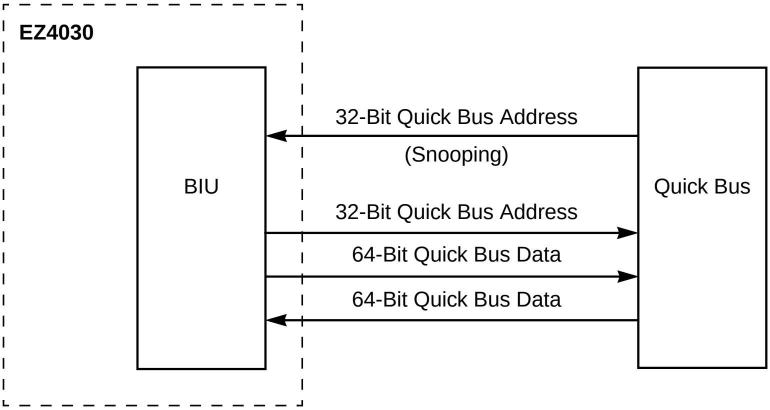
- Cache Memory

  The EZ4030 has separate instruction and data caches (I-Cache and D-Cache, respectively). Each cache is organized as a 2-way, set associative, 16 Kbyte cache with a fixed cache block (line) size of 8 words (32 bytes). The caches are virtually indexed and physically tagged.

- Bus Interface Unit (BIU)

  The BIU (see Figure 4) passes address/data between the CPU and the Quick Bus and arbitrates CPU-to-Quick Bus access. It provides a 64-bit incoming data path from the Quick Bus to the CPU and a 64-bit path for outgoing data. Depending on the customer design, outgoing Quick Bus data can go to a variety of destinations outside the EZ4030.

Figure 4 BIU I/O Block Diagram



- On-Chip Quick Bus Interface

  The Quick Bus is a split-transaction bus that allows the overlap of memory requests (fetch/load/store) and data returns. A Quick Bus design operating at up to one-half the processor frequency can achieve a peak bandwidth of 1.0 Gbytes per second.

  Optionally, you can add an external Quick Bus-to-AHB bridge to the EZ4030 that allows for easy connection of industry-standard AMBA peripherals.

- Debug Support

  The EZ4030 includes a JTAG interface and supports EJTAG functions. EJTAG is a debug feature of MIPS-based processors. You can use EJTAG to debug standalone processors as well as 32- or 64-bit processors that are embedded in a system like the EZ4030.

Pipeline Architecture

The EZ4030 is a 5-stage pipelined processor. The function of each pipeline stage is as follows.

Instruction Fetch (F) – The EZ4030 fetches the instruction during this first stage.

Register Read (R) – In this stage, the CPU reads any required operands from the Register file and finishes decoding the instruction.

Execute (X) – Computational and logical instructions execute during this stage. The CPU resolves conditional branches during this stage, and does the address calculations for load and store instructions.

Memory Access (M) – In this stage, the CPU accesses the cache for load and store instructions. Data returns to the register bypass logic at the end of this stage.

Write Back (W) – The CPU writes results into the Register file in this stage.

EZ4030 Instruction Set Summary

Table 1 summarizes the EZ4030 instruction set, except for the floating-point instructions, which are provided separately in Table 2.

Table 1 EZ4030 CPU Instruction Set Summary

Instruction	Description	Instruction	Description
Normal CPU Load/Store Instructions			
LB	Load Byte – MIPS I	LWU	Load Word Unsigned – MIPS III
LBU	Load Byte Unsigned – MIPS I	SB	Store Byte – MIPS I
LD	Load Doubleword – MIPS III	SD	Store Doubleword – MIPS III
LH	Load Halfword – MIPS I	SH	Store Halfword – MIPS I
LHU	Load Halfword Unsigned – MIPS I	SW	Store Word – MIPS I
LW	Load Word – MIPS I		
Unaligned CPU Load/Store Instructions			
LDL	Load Doubleword Left – MIPS III	SDL	Store Doubleword Left – MIPS III
LDR	Load Doubleword Right – MIPS III	SDR	Store Doubleword Right – MIPS III
LWL	Load Word Left – MIPS I	SWL	Store Word Left – MIPS I
LWR	Load Word Right – MIPS I	SWR	Store Word Right – MIPS I
Atomic Update CPU Load/Store Instructions			
LL	Load Linked Word – MIPS II	SC	Store Conditional Word – MIPS II
LLD	Load Linked Doubleword – MIPS III	SCD	Store Conditional Doubleword – MIPS III
Coprocessor Load/Store Instructions			
LDCz	Load Doubleword to Coprocessor – MIPS II	SDCz	Store Doubleword from Coprocessor – MIPS II
LWCz	Load Word to Coprocessor – MIPS I	SWCz	Store Word from Coprocessor – MIPS I
ALU Instructions with Immediate Operand			
ADDI	Add Immediate – MIPS I	LUI	Load Upper Immediate – MIPS I
ADDIU	Add Immediate Unsigned – MIPS I	ORI	OR Immediate – MIPS I
ANDI	AND Immediate – MIPS I	SLTI	Set on Less than Immediate – MIPS I
(Sheet 1 of 5)			

Table 1 EZ4030 CPU Instruction Set Summary (Cont.)

Instruction	Description	Instruction	Description
DADDI	Doubleword Add Immediate – MIPS III	SLTIU	Set on Less than Immediate Unsigned – MIPS I
DADDIU	Doubleword Add Immediate Unsigned – MIPS III	XORI	Exclusive OR Immediate – MIPS I
3-Operand ALU Instructions			
ADD	Add – MIPS I	NOR	Logical NOR – MIPS I
ADDU	Add Unsigned – MIPS I	OR	Logical OR – MIPS I
AND	Logical AND – MIPS I	SLT	Set on Less than – MIPS I
DADD	Doubleword Add – MIPS III	SLTU	Set on Less than Unsigned – MIPS I
DADDU	Doubleword Add Unsigned – MIPS III	SUB	Subtract – MIPS I
DSUB	Doubleword Subtract – MIPS III	SUBU	Subtract Unsigned – MIPS I
DSUBU	Doubleword Subtract Unsigned – MIPS III	XOR	Exclusive Logical OR – MIPS I
Shift Instructions			
DSLL	Doubleword Shift Left Logical – MIPS III	DSRAV	Doubleword Shift Right Arithmetic Variable – MIPS III
DSLL32	Doubleword Shift Left Logical Plus 32 – MIPS III	SLL	Shift Left Logical – MIPS I
DSRA	Doubleword Shift Right Arithmetic – MIPS III	SLLV	Shift Left Logical Variable – MIPS I
DSRA32	Doubleword Shift Right Arithmetic Plus 32 – MIPS III	SRA	Shift Right Arithmetic – MIPS I
DSRL	Doubleword Shift Right Logical – MIPS III	SRAV	Shift Right Arithmetic Variable – MIPS I
DSRL32	Doubleword Shift Right Logical Plus 32 – MIPS III	SRL	Shift Right Logical – MIPS I
DSLLV	Doubleword Shift Left Logical Variable – MIPS III	SRLV	Shift Right Logical Variable – MIPS I
DSRLV	Doubleword Shift Right Logical Variable – MIPS III		
(Sheet 2 of 5)			

Table 1 EZ4030 CPU Instruction Set Summary (Cont.)

Instruction	Description	Instruction	Description
Multiply/Divide Instructions			
DDIV	Doubleword Divide – MIPS III	MFHI	Move from HI – MIPS I
DDIVU	Doubleword Divide Unsigned – MIPS III	MFLO	Move from LO – MIPS I
DIV	Divide – MIPS I	MTHI	Move to HI – MIPS I
DIVU	Divide Unsigned – MIPS I	MTLO	Move to LO – MIPS I
DMULT	Doubleword Multiply – MIPS III	MULT	Multiply – MIPS I
DMULTU	Doubleword Multiply Unsigned – MIPS III	MULTU	Multiply Unsigned – MIPS I
Jump Instructions			
J	Jump – MIPS I	JALR	Jump and Link Register – MIPS I
JAL	Jump and Link – MIPS I	JR	Jump Register – MIPS I
PC-Relative Conditional Branch Instructions			
BEQ	Branch on Equal – MIPS I	BLEZ	Branch on Less than or Equal to Zero – MIPS I
BGEZ	Branch on Greater than or Equal to Zero – MIPS I	BLTZ	Branch on Less than Zero – MIPS I
BGEZAL	Branch on Greater than or Equal to Zero and Link – MIPS I	BLTZAL	Branch on Less than Zero and Link – MIPS I
BGTZ	Branch on Greater than Zero – MIPS I	BNE	Branch on Not Equal – MIPS I
PC-Relative Conditional Branch Likely Instructions			
BEQL	Branch on Equal Likely – MIPS II	BLEZL	Branch on Less than or Equal to Zero Likely – MIPS II
BGEZALL	Branch on Greater than or Equal to Zero and Link Likely – MIPS II	BLTZALL	Branch on Less than Zero and Link Likely – MIPS II
BGEZL	Branch on Greater than or Equal to Zero Likely – MIPS II	BLTZL	Branch on Less than Zero Likely – MIPS II
BGTZL	Branch on Greater than Zero Likely – MIPS II	BNEL	Branch on Not Equal Likely – MIPS II
(Sheet 3 of 5)			

Table 1 EZ4030 CPU Instruction Set Summary (Cont.)

Instruction	Description	Instruction	Description
Breakpoint and System Call Instructions			
BREAK	Breakpoint – MIPS I	SYSCALL	System Call – MIPS I
Trap-on-Condition Instructions			
TEQ	Trap if Equal – MIPS II	TLT	Trap if Less than – MIPS II
TEQI	Trap if Equal Immediate – MIPS II	TLTI	Trap if Less than Immediate – MIPS II
TGE	Trap if Greater than or Equal – MIPS II	TLTIU	Trap if Less than Immediate Unsigned – MIPS II
TGEI	Trap if Greater than or Equal Immediate – MIPS II	TLTU	Trap if Less than Unsigned – MIPS II
TGEIU	Trap if Greater than or Equal Immediate Unsigned – MIPS II	TNE	Trap if Not Equal – MIPS II
TGEU	Trap if Greater than or Equal Unsigned – MIPS II	TNEI	Trap If Not Equal Immediate – MIPS II
Serialization Instructions			
SYNC	Synchronize Shared Memory – MIPS II		
Coprocessor Data Movement and Conditional Branch Instructions			
BCzF	Branch on Coprocessor z False	DMFCz	Doubleword Move from Coprocessor z
BCzT	Branch on Coprocessor z True	DMTCz	Doubleword Move to Coprocessor z
CFCz	Move Control from Coprocessor z	MFCz	Move from Coprocessor z
COPz	Coprocessor z Operation	MTCz	Move to Coprocessor z
CTCz	Move Control to Coprocessor z		
System Control Coprocessor (CP0) Instructions			
ERET	Exception Return	TLBR	Read Indexed TLB Entry
MFC0	Move from CP0	TLBWI	Write Indexed TLB Entry
MTC0	Move to CP0	TLBWR	Write Random TLB Entry
(Sheet 4 of 5)			

Table 1 EZ4030 CPU Instruction Set Summary (Cont.)

Instruction	Description	Instruction	Description
TLBP	Probe TLB For Matching Entry		
Cache Maintenance Instruction			
CACHE	Cache Maintenance		
General 32-Bit Instruction Extensions			
MADD	Multiply Add	MSUB	Multiply Subtract
MADDU	Multiply Add Unsigned	MSUBU	Multiply Subtract Unsigned
MUL	Multiply	SDBBP	Software Debug Breakpoint
CP0 Instruction Extensions			
DERET	Debug Exception Return	WAITI	Wait for Interrupt
(Sheet 5 of 5)			

Table 2 EZ4030 Floating-Point Instruction Set Summary

Instruction	Description	Instruction	Description
Floating-Point Load, Store, and Move Instructions			
CFC1	Move Control Word from Coprocessor 1 – MIPS I	LWC1	Load Word to Coprocessor 1 – MIPS I
CTC1	Move Control Word to Coprocessor 1 – MIPS I	MFC1	Move Word from Coprocessor 1 – MIPS I
DMFC1	Doubleword Move from Coprocessor 1 – MIPS III	MTC1	Move Word to Coprocessor 1 – MIPS I
DMTC1	Doubleword Move to Coprocessor 1 – MIPS III	SDC1	Store Doubleword from Coprocessor 1 – MIPS II
LDC1	Load Doubleword to Coprocessor 1 – MIPS II	SWC1	Store Word from Coprocessor 1 – MIPS I
Floating-Point Conversion Instructions			
CEIL.L.fmt	Floating-Point Ceiling to 64-Bit Fixed Point – MIPS III	FLOOR.L.fmt	Floating-Point Floor to 64-Bit Fixed Point – MIPS III
(Sheet 1 of 2)			

Table 2 EZ4030 Floating-Point Instruction Set Summary (Cont.)

Instruction	Description	Instruction	Description
CEIL.W.fmt	Floating-Point Ceiling to 32-Bit Fixed Point – MIPS II	FLOOR.W.fmt	Floating-Point Floor to 32-Bit Fixed Point – MIPS II
CVT.D.fmt	Floating-Point Convert to Double FP – MIPS I	ROUND.L.fmt	Floating-Point Round to 64-Bit Fixed Point – MIPS II
CVT.L.fmt	Floating-Point Convert to 64-Bit Fixed Point – MIPS III	ROUND.W.fmt	Floating-Point Round to 32-Bit Fixed Point – MIPS II
CVT.S.fmt	Floating-Point Convert to Single FP – MIPS I	TRUNC.L.fmt	Floating-Point Truncate to 64-Bit Fixed Point – MIPS III
CVT.W.fmt	Floating-Point Convert to 32-Bit Fixed Point – MIPS I	TRUNC.W.fmt	Floating-Point Truncate to 32-Bit Fixed Point – MIPS II
Floating-Point Computational Instructions			
ABS.fmt	Floating-Point Absolute Value – MIPS I	MOV.fmt	Floating-Point Move – MIPS I
ADD.fmt	Floating-Point Add – MIPS I	NEG.fmt	Floating-Point Negate – MIPS I
DIV.fmt	Floating-Point Divide – MIPS I	SQRT.fmt	Floating-Point Square Root – MIPS II
MUL.fmt	Floating-Point Multiply – MIPS I	SUB.fmt	Floating-Point Subtract – MIPS I
Floating-Point Compare and Branch Instructions			
BC1F	Branch on Coprocessor 1 False – MIPS I	BC1TL	Branch on Coprocessor 1 True Likely – MIPS II
BC1FL	Branch on Coprocessor 1 False Likely – MIPS II	C.cond.fmt	Floating-Point Compare – MIPS I
BC1T	Branch on FPU True – MIPS I		
(Sheet 2 of 2)			

Signal Descriptions

Table 3 defines all the signals that interface with the EZ4030. In the table, the signals are divided into the following categories:

- Quick Bus Interface
- Interrupt, Clock, and Reset
- EJTAG and PC Trace
- Global Test Mode
- RAM BIST
- Miscellaneous

Mnemonics for signals that are active LOW end with an “N”, and mnemonics for signals that are active HIGH end with a “P”.

Table 3 EZ4030 EasyMACRO Module Signals

Signal	I/O	Description
Quick Bus Interface Signals		
BC_D_QB_BREQP	Output	Bus Request for Data Access
BC_E_QB_BREQP	Output	Bus Request for EJTAG DMA
BC_I_QB_BREQP	Output	Bus Request for Instruction Fetch
BC_QB_ADDRP[31:3]	Output	Address
BC_QB_BURSTREQP	Output	Burst Request
BC_QB_BYTEP[7:0]	Output	Byte Enable Signals
BC_QB_CMDLOCKP	Output	Command Lock
BC_QB_RDACKP	Output	Read Data Acknowledge
BC_QB_READP	Output	Read Request
BC_QB_WRDATAP[63:0]	Output	Write Data
BC_QB_WRITEP	Output	Write Request
(Sheet 1 of 5)		

Table 3 EZ4030 EasyMACRO Module Signals (Cont.)

Signal	I/O	Description
QB_ADDRP[31:3]	Input	Address (Snooping)
QB_BADADDRP	Input	Bad Address Error
QB_BOFFP	Input	Back Off (Snooping)
QB_BURSTACKP	Input	Burst Request Acknowledge
QB_CMDRDYP	Input	Command Ready
QB_GRANT_BC_DP	Input	Data Access Command Bus Grant
QB_GRANT_BC_EP	Input	EJTAG DMA Command Bus Grant
QB_GRANT_BC_IP	Input	Instruction Fetch Command Bus Grant
QB_RDDATAP[63:0]	Input	Read Data
QB_RDERRP	Input	Read Data Error
QB_RDRDY_BC_DP	Input	Read Data Ready for Data Read
QB_RDRDY_BC_EP	Input	Read Data Ready for EJTAG Read
QB_RDRDY_BC_IP	Input	Read Data Ready for Instruction Fetch
QB_SLRDY_BC_DP	Input	Slave Ready for Data Access
QB_SLRDY_BC_EP	Input	Slave Ready for EJTAG DMA
QB_SLRDY_BC_IP	Input	Slave Ready for Instruction Fetch
QB_WRITEP	Input	Snoop Request (Snooping)
Interrupt, Clock, and Reset Signals		
CG_RESETP	Input	Clock Generator Reset
EZ_SCLK_GATEP	Output	System/Secondary Clock Gate
INTP[4:0]	Input	External Hardware Interrupts
NMIP	Input	Nonmaskable Interrupt
PCLKP	Input	Primary CPU Clock
RESETP	Input	System Cold Reset
(Sheet 2 of 5)		

Table 3 EZ4030 EasyMACRO Module Signals (Cont.)

Signal	I/O	Description
SCLKP_DIVP[1:0]	Input	SCLKP Divide Ratio
EJTAG and PC Trace Signals		
BC_DMP	Output	Debug Mode
DCLKP	Output	EJTAG PC Trace Clock
DJ_DBGBRKP	Input	Debug Break
DJ_DBRKDEMUXP	Output	Disable DTIP_DINTN Break
DJ_EJTAGIRBITS[1:0]	Input	EIR Extension Width
DJ_JTAGALSOP	Input	Parallel JTAG Present
DJ_JTAGTDOP	Input	Parallel JTAG TDO
DJ_PERRSTN	Output	Peripheral Reset
DJ_PON[19:0]	Input	Part Number
DJ_PURETDO_DRN	Output	Pure TDO Output Enable
DJ_PURETDOP	Output	Pure TDO
DJ_TCKP	Input	EJTAG Test Clock
DJ_TDIP_DINTN	Input	Test Data Input/Debug Interrupt
DJ_TDO_DRIVEN	Output	TDO Output Enable
DJ_TDOP_TPCP	Output	Test Data Output/Target PC Output
DJ_TMSP	Input	Test Mode Select
DJ_TRSTN	Input	Test Reset
DT_PCST1[2:0]	Output	PC Trace Status Information 1
DT_PCST2[2:0]	Output	PC Trace Status Information 2
DT_PCTEN	Input	PC Trace Enable
DT_TPC1P	Output	Bit 1 of the TPC (TPC[1])
DT_TPCPLP[6:0]	Output	TPC Plus
(Sheet 3 of 5)		

Table 3 EZ4030 EasyMACRO Module Signals (Cont.)

Signal	I/O	Description
MMU_CNTALWYSP	Input	MMU Increment Count Reg in Debug Mode
Global Test Mode Signals		
GSCAN_ENABLEP	Input	Core Scan Enable
GSCAN_IN1P	Input	Core Scan Chain 1 Input
GSCAN_IN2P	Input	Core Scan Chain 2 Input
GSCAN_IN3P	Input	Core Scan Chain 3 Input
GSCAN_IN4P	Input	Core Scan Chain 4 Input
GSCAN_IN5P	Input	Core Scan Chain 5 Input
GSCAN_IN6P	Input	Core Scan Chain 6 Input
GSCAN_MODEP	Input	Global Test Mode
GSCAN_OUT1P	Output	Core Scan Chain 1 Output
GSCAN_OUT2P	Output	Core Scan Chain 2 Output
GSCAN_OUT3P	Output	Core Scan Chain 3 Output
GSCAN_OUT4P	Output	Core Scan Chain 4 Output
GSCAN_OUT5P	Output	Core Scan Chain 5 Output
GSCAN_OUT6P	Output	Core Scan Chain 6 Output
GSCAN_RAMCLKP	Input	Scan Test RAM Clock and BIST Clock
RAM BIST Signals		
BIST_DIAG_EN	Input	BIST Diag Enable
BIST_HOLD	Input	BIST Hold Command
BIST_SETUP[1:0]	Input	BIST Setup
BIST_SHIFT	Input	BIST Shift Command
BIST_SI	Input	BIST Shift Data In
BIST_SO	Output	BIST Shift Data Out
(Sheet 4 of 5)		

Table 3 EZ4030 EasyMACRO Module Signals (Cont.)

Signal	I/O	Description
MBIST_DONE	Output	BIST Test Completion
MBIST_EN	Input	BIST Controller Enable
MBIST_GO	Output	BIST Test Failure Indication
TCK	Input	BIST TAP Controller Clock
TCK_MODE	Input	BIST TCK Mode
Miscellaneous Signals		
BIG_ENDIANP	Input	Big Endian Mode
CACHE_TESTP	Input	Cache Test Mode
COP_CP0COND	Input	CP0 Branch Condition
CPU_EC_WAITI	Output	Wait for Interrupt
DCACHE_ENABLEP	Input	Data Cache Enable
DHQ_SCR1[31:0]	Output	System Configuration Register
ICACHE_ENABLEP	Input	Instruction Cache Enable
LOADSCHED_ENABLEP	Input	Load Scheduling Enable
MMU_ENABLEP	Input	MMU Enable
PREFETCH_ENABLEP	Input	Instruction Prefetch Enable
PRID_REV[3:0]	Input	Processor Revision Identifier
PSTALLP	Output	EZ4030 Global Stall
READPRI_ENABLEP	Input	Read Priority Enable
SNOOP_ENABLEP	Input	Data Cache Snoop Enable
WRITEBUF_ENABLEP	Input	Write Buffer Enable
(Sheet 5 of 5)		

Headquarters

LSI Logic Corporation
North American Headquarters
Milpitas CA
Tel: 408.433.8000
Fax: 408.433.8989

LSI Logic Europe Ltd
European Headquarters
Bracknell England
Tel: 44.1344.426544
Fax: 44.1344.481039

LSI Logic K.K.
Headquarters
Tokyo Japan
Tel: 81.3.5463.7821
Fax: 81.3.5463.7820

To receive product literature, visit us at http://www.lsilogic.com.
For a current list of our distributors, sales offices, and design resource centers, view our web page located at http://www.lsilogic.com/contacts/na_salesoffices.html.

ISO 9000 Certified

This document is preliminary. As such, it contains data derived from functional simulations and performance estimates. LSI Logic has not verified the functional descriptions or electrical and mechanical specifications using production parts.

LSI Logic Corporation reserves the right to make changes to any products and services herein at any time without notice. LSI Logic does not assume any responsibility or liability arising out of the application or use of any product or service described herein, except as expressly agreed to in writing by LSI Logic; nor does the purchase, lease, or use of a product or service from LSI Logic convey a license under any patent rights, copyrights, trademark rights, or any other of the intellectual property rights of LSI Logic or of third parties.

BM
Doc. No. DB08-000163-00