For: dwight
Printed on: Fri, Jul 12, 1991 10:41:30
From book: c_oview
Document: chapter
Last saved on: Fri, Jul 12, 1991 10:37:51

1. Architectural Overview

The iWarp component (iWM) is the basic building block for the iWarp Distributed Memory Multicomputer. The iWarp system is a modular, high-performance parallel architecture that is expandable to thousands of processors. A single iWM plus memory forms a complete processor, called a cell, of an iWarp array. The iWM supports one-dimensional and two-dimensional systems. Figure 1 illustrates a system configuration.

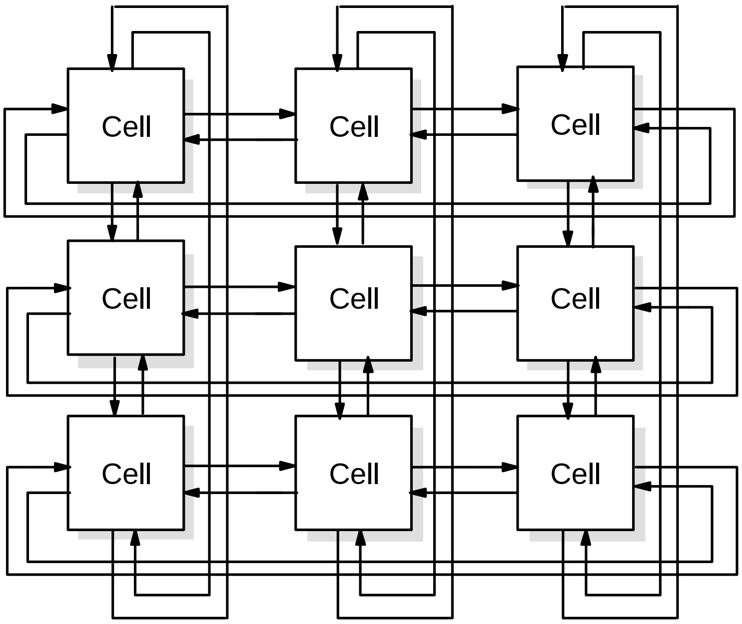


Figure 1: iWarp System Configuration

The iWM is divided into a Computation Agent and a Communication Agent. The Computation Agent and Communication Agent function independently, allowing communication activities to proceed with minimum disruption of the computation activities.

The Communication Agent supports four input and four output buses that connect directly to the corresponding buses of adjacent cells in an array. Figure 2 illustrates the communication pathways.

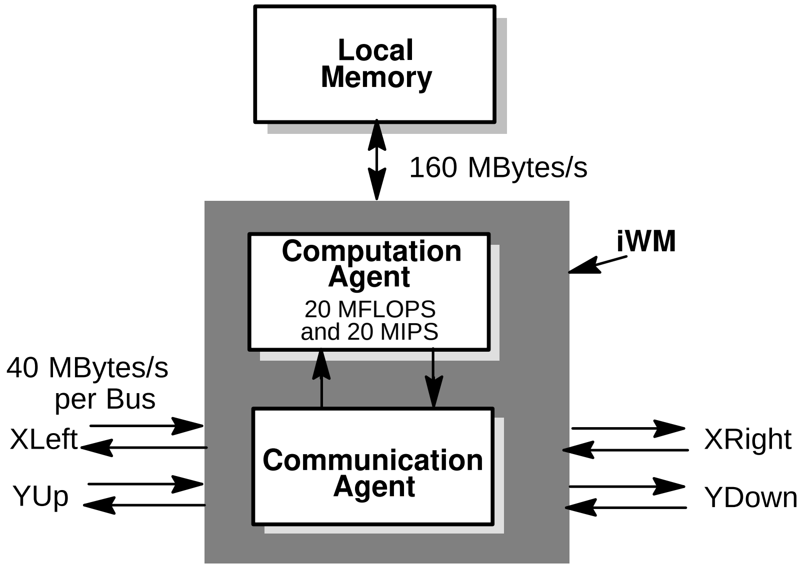


Figure 2: iWarp Communication Pathways

Each bus can sustain a 40 MBytes/s communication data rate with a combined data rate of 320 MBytes/s for each iWM in the array. Messages are routed through the array using a self-routing (''worm-hole routing'') scheme that synchronizes communication on a word-by-word basis across multiple cell connections. The Communication Agent of an iWM can supply data to the Computation Agent and memory at up to 160 MBytes/s.

The Computation Agent contains five functional units: the Register File Unit, Floating-Point Arithmetic Unit, Integer Logic Unit, Program Store Unit, and Streaming/Spooling Unit.

The heart of the iWM architecture is the 15-port Register File Unit with 128 32-bit registers addressable as bytes, half-words, words, or double words. Nine read and six write operations are supported in a single clock, providing a combined access bandwidth of 720 MBytes/s. Communication is performed as a side-effect of register accesses. This provides a low latency access path between the processor elements of the Computation Agent, the communication pathway, and memory.

The Program Store Unit of the iWM contains a 1 KByte instruction cache and an 8 KByte instruction ROM. On power-up, code in the ROM initializes the processor and performs self-test operations. The instruction cache has four sectors, with each sector containing four 64-byte blocks. A single-function or multi-function instruction can be fetched from the cache each internal clock. Prefetch on the next cache block within a sector from external memory is concurrent with the instruction fetches. The mapping of sectors is fully associative, with blocks in a sector containing contiguous addresses.

The Floating-Point Adder performs scalar add and subtract, and the Floating-Point Multiplier performs non-pipelined multiply in two clocks. The Floating-Point Adder and Floating-Point Multiplier can operate in parallel, yielding a peak performance for each cell of 20 MFLOPS on 32-bit operations and 10 MFLOPS for 64-bit operations. Bypass paths support chaining of computations without the added latency of register file updates. The Floating-Point Unit fully conforms to the ANSI/IEEE standard 754-1985 for floating-point arithmetic. The Integer Logic Unit operates in parallel with the Floating-Point Unit, performing two address operations during a floating-point add/multiply combination.

The Streaming/Spooling Unit transfers data between the Computation Agent or cell memory and the Communication Agent. Spool operations transfer blocks of data between the Communication Agent and memory through eight DMA channels without processor intervention. Spooling is primarily used to support memory-to-memory message passing between cells and allows removal of blocked messages from the pathway, relieving pathway congestion. Stream operations transfer data between the Communication Agent and the Computation Agent through the stream gates of the Register File Unit. Streaming is primarily used to support program-to-program variable passing, and allows direct receiving and sending of data between the Communication Agent and the Computation Agent on a word-by-word basis. This direct access of data under program control provides support for systolic applications across multiple cells.

For a complete description of hardware and software for iWarp systems, refer to the publication, *Introduction to iWarp*.

2. Instruction Set

The iWM uses a general-purpose 32-bit RISC-like single-function instruction set which includes a 96-bit multi-function Compute and Access instruction. The multi-function Compute and Access instruction has the benefits of a long instruction word architecture, which allows multiple operations in parallel. This instruction can perform floating-point add, floating-point multiply, two memory address computations and memory access operations, plus loop decrement and branch evaluation in two clocks for single precision data.

Word three of the Compute and Access instruction can specify either a 32-bit general-purpose instruction or perform two memory address operations. The 32-bit instruction option is useful when a general-purpose ALU operation is needed during a compute and access cycle.

The execution time for the Compute and Access instruction is the maximum of the following:

- the time to execute a Floating-Point Adder operation
- the time to execute a Floating-Point Multiplier operation
- the time to execute a third word instruction or one to two three-clock memory accesses

Figure 3 illustrates the format of the Compute and Access instruction.

Word 1

	-(2)-	-(4)-	-(4)-	-(7)-	-(7)-	-(7)-
J	1 1	Data Mode	FADD	B operand reg	A operand reg	K operand reg
		specifies data types for operands and result	subtract add max min compare scale copy_sign move change_sign logb classify convert	B source register for FADD	A source register for FADD	K destination register for FADD

Word 2

-(9)-	-(2)-	-(7)-	-(7)-	-(7)-
Memory Control	FMUL	M operand reg	N operand reg	R operand reg
read - write single - double pre - post address compute access memory	multiply neg_mult move	M source register for FMUL	N source register for FMUL	R destination register for FMUL

Word 3 (Option 1)

-(2)-	-(7)-	-(7)-	-(2)-	-(7)-	-(7)-
OP1	Offset 1	Base 1	OP2	Offset 2	Base 2
move add sub literal	source register for 1st address offset	source and destination register for 1st base address	move add sub literal	source register for 2nd address offset	source and destination register for 2nd base address

Word 3 (Option 2)

-(32)-
General–purpose ILU Instruction

Refer to Table 1 for a list of ILU instructions

Figure 3: Compute and Access Instruction Format

The single-function instruction set supports general control functions such as timer operations, pathway control, event handling, and flow control. The Floating-Point Adder instructions include add, subtract, compare, maximum, minimum, and binary log. The Floating-Point Multiplier instructions include multiply, divide, square root, and remainder.

The general-purpose instruction set also includes data conversion operations and a full range of integer arithmetic and logical operations. The instruction set also supports byte, half-word, full-word, and double-word memory operations, as well as transparent read-modify-write in hardware for byte and half-word operations. Table 1 summarizes the iWM 32-bit instruction set by instruction type. The *iWarp Macroarchitecture Specification* provides a detailed description of each instruction.

Table 1: 32-bit Instruction Set by Instruction Type

INSTRUCTION TYPE	INSTRUCTIONS	
Floating-Point	FPAdder • add • subtract • compare • max • min • copy • convert	FPMultiplier • multiply • divide • copy • remainder • square root
Integer/Logical	Logical Op • AND • OR • NAND • XOR • XNOR • compare • invert • copy • move High Bit Find Rotate	Arithmetic Op • add • add with carry • subtract • subtract with borrow • compare with borrow Bit Op • set • reset
Data Conversion	IntegerConvert	FPConvert
Literal Loads	Load Literal	
Memory Access	Access Memory Access Byte Memory	
Flow Control	Call Pop Debug Break Enter Loop Release Stack Absolute Call Indirect Call	Push Branch Return Allocate Stack Supervisor Call Absolute Branch Indirect Branch
General Control	NOP Move Register Move Control Space Register	Indirect Pointer Pathway Control

Table 2 lists the clock counts required for each iWM single-function instruction. Each clock is 50ns based on a 40 MHz external clock.

Table 2: Instruction Clock Count Summary

Instruction	Execution time (clocks)
Absolute Branch literal target address register target address	 2 3
Absolute Call literal target address register target address	 2 3
Access Byte Memory word double word byte/half word	 2-3 3-4 4-5
Access Memory word double word	 2-3 3-4
Allocate Stack	3
Arithmetic Op	1
Bit Op	2-3
Branch condition = *false* condition = *true* branch with loop termination	 1-2 2-3 4
Call condition = *false* condition = *true* call with loop termination	 1-2 2-3 4
Debug Break Control Register debug mode not set Control Register debug mode set: condition = *false* condition = *true*	 1 3-4 3-4

Table 2: Instruction Clock Count Summary (con't)

Instruction			Execution time (clocks)
Enter Loop			
	instruction		3
	with *terminate loop* field set:		
		loop repetition	2-3
		loop termination	3
FP Adder			
	single precision		2
	double precision		4
	convert to double		4
	convert to single		5
FP Convert			
	single precision:	wrap/unwrap	2
		fix/trunc/float	2
	double precision	wrap/unwrap	4
		fix/trunc/float	4
FP Multiplier			
	single precision		2
	double precision		4
	integer multiply		3
	single precision:	divide	16
		square root	31
		remainder	5-162
	double precision:	divide	31
		square root	61
		remainder	5-1087
High Bit Find			1
Indirect Branch			4
Indirect Call			4
Indirect Pointer			1
Integer Convert			1

Table 2: Instruction Clock Count Summary (con't)

Instruction	**Execution time (clocks)**
Load Literal	1
Logical Op	1
Move Control Space Register	1
Move Register	3
NOP	1
Pathway Control	3
Pop single word double word	 2 3
Push single word double word	 2 3
Release Stack	4
Return	2
Rotate	1
Supervisor Call	4

3. Pin Description

The iWM is a 271-pin PGA package. Figure 4 provides the pin-side view pinout of the iWM, and Figure 5 provides the top-side view pinout.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
U	DAT38	DAT35	DAT20	DAT17	DAT14	VSS	VSS	VSS	VSS	VSS	VSS	RST#	VSS	XRI1	XRI3	VSS	XRINQ0	XRICLK	YDI2
T	PAR2	DAT34	DAT22	DAT21	DAT13	VCC	DAT10	PAR0	DAT7	DAT4	DAT0	VREF	VCC	XRI4	XRI6	VCC	XRODQ0	YDODQ1	YDI1
S	DAT40	DAT33	DAT27	DAT23	DAT18	DAT15	DAT12	DAT8	DAT5	DAT3	CLK2	XRI2	XRI7	XRIP	YDICLK	YDIP	YDINQ0	YDIQ	XRONQ1
R	DAT41	RPTO	DAT30	DAT26	DAT25	DAT24	DAT19	DAT11	DAT6	DAT1	XRIQ	XRINQ1	XRODQ1	YDI6	YDINQ1	YDI7	XROP	XRONQ0	XROCLK
Q	VSS	VCC	DAT32	RDY#	DAT28	DAT29	PAR1	DAT16	DAT9	DAT2	XRI5	YDODQ0	YDI4	YDI3	YDI5	XRIDQ1	XRO6	XRO4	YDIDQ1
P	VSS	DAT42	DAT37	DAT31											XRIDQ0	XRO5	XRO3	XRO0	VSS
N	VSS	DAT43	RPTI	DAT36											XR07	XRO2	YDIDQ0	YDOP	VSS
M	VSS	DAT45	DAT46	DAT39											XRO1	YDONQ1	YDONQ0	YDO7	YDO5
L	VSS	DAT48	DAT47	DAT44											YDO6	YDOCLK	YDO3	YDO1	VSS
K	VSS	VCC	DAT49	DAT51											YUIDQ1	YDO0	YDO2	YDO4	VSS
J	VSS	VCC	DAT54	DAT59											YUOCLK	YUO5	YUONQ0	YUIDQ0	VSS
H	VSS	VCC	DAT60	FAIL#											XLONQ1	YUO0	YUO6	YUONQ1	VSS
G	VSS	VCC	DAT61	RD0#											XLOCLK	XLONQ0	YUO2	YUO7	YUOP
F	DAT53	DAT55	DAT63	RD1#											YUI2	XLO2	XLOP	VCC	VSS
E	DAT50	DAT52	STAT1	A3	A6	A8	A12	A16	A20	A22	XLI2	XLICLK	YUIP	YUI6	YUI1	XLO0	XLO5	YUO1	YUO4
D	VSS	VCC	WR0#	A5	A9	A14	A17	A23	A19	A21	XLI3	XLI7	XLINQ1	XLODQ1	YUI5	YUI0	XLO3	XLIDQ0	YUO3
C	DAT56	DAT58	WR1#	A7	A13	A15	A18	EVENT#	XLI1	XLI0	XLI4	XLI6	XLIP	XLODQ0	YUICLK	YUI7	XLO1	XLO4	XLIDQ1
B	PAR3	DAT57	VSS	VSS	A11	VCC	A24	A25	VCC	VCC	VCC	XLI5	VSS	VSS	VSS	YUODQ0	XLINQ0	XLO7	XLO6
A	DAT62	STAT0	A4	VCC	A10	VSS	VSS	VSS	VSS	VSS	VSS	XLINQ0	VCC	VCC	VCC	YUODQ1	YUINQ1	YUI4	YUI3

Figure 4: Component pinout (pin-side view)

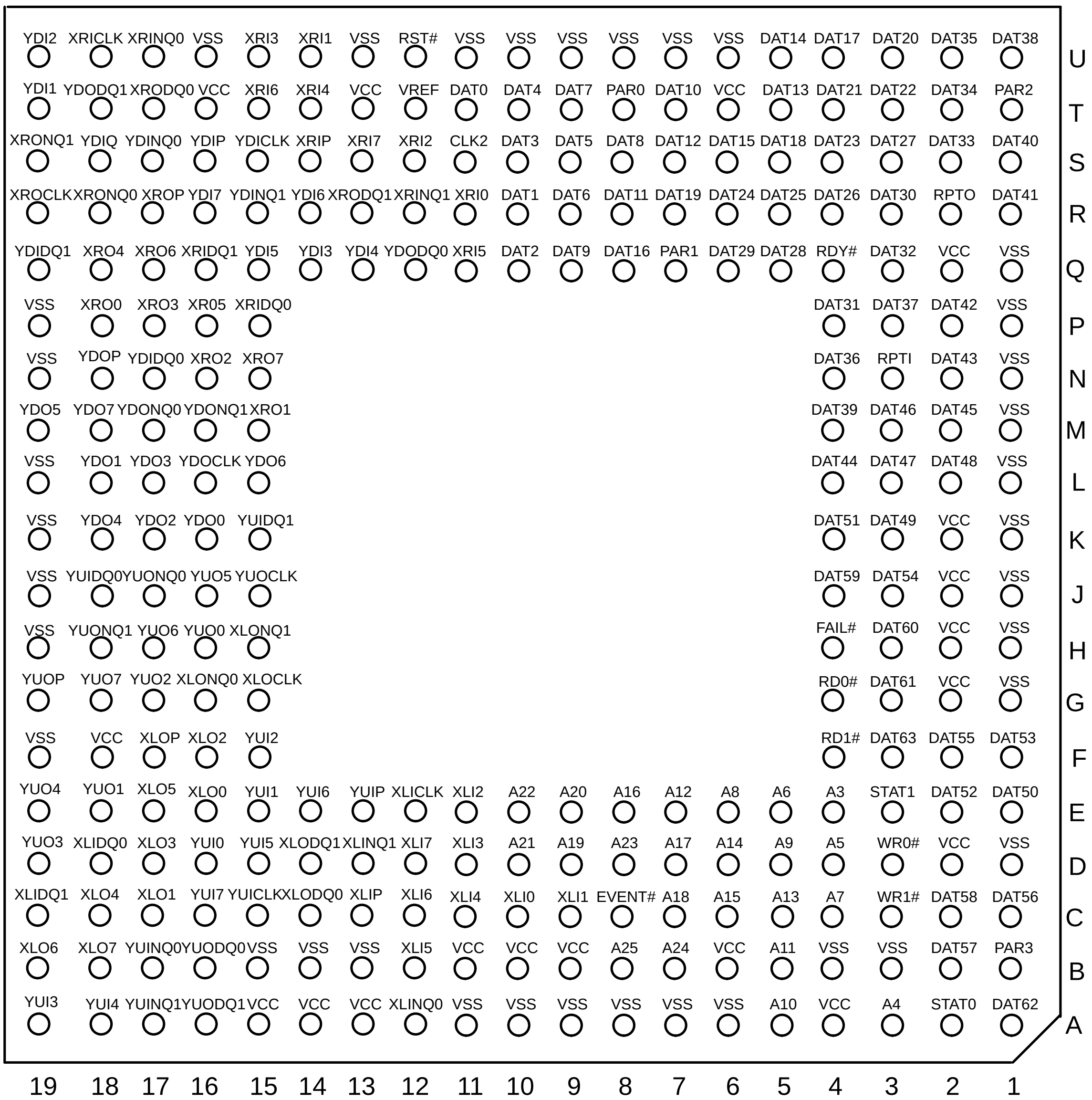


Figure 5: Component pinout (top-side view)

Table 3 provides a cross reference of the pins by pin name. Refer to Tables 5 through 9 for complete descriptions of the pins.

Note: XLIDQ, XRIDQ, YUIDQ, and YDIDQ are output pins. XLODQ, XRODQ, YUODQ, and YDODQ are input pins.

Table 3: Pin Cross Reference by Pin Name

Pin Name	Location	Pin Name	Location	Pin Name	Location
A3	E4	**DAT6**	R9	**DAT36**	N4
A4	A3	**DAT7**	T9	**DAT37**	P3
A5	D4	**DAT8**	S8	**DAT38**	U1
A6	E5	**DAT9**	Q9	**DAT39**	M4
A7	C4	**DAT10**	T7	**DAT40**	S1
A8	E6	**DAT11**	R8	**DAT41**	R1
A9	D5	**DAT12**	S7	**DAT42**	P2
A10	A5	**DAT13**	T5	**DAT43**	N2
A11	B5	**DAT14**	U5	**DAT44**	L4
A12	E7	**DAT15**	S6	**DAT45**	M2
A13	C5	**DAT16**	Q8	**DAT46**	M3
A14	D6	**DAT17**	U4	**DAT47**	L3
A15	C6	**DAT18**	S5	**DAT48**	L2
A16	E8	**DAT19**	R7	**DAT49**	K3
A17	D7	**DAT20**	U3	**DAT50**	E1
A18	C7	**DAT21**	T4	**DAT51**	K4
A19	D9	**DAT22**	T3	**DAT52**	E2
A20	E9	**DAT23**	S4	**DAT53**	F1
A21	D10	**DAT24**	R6	**DAT54**	J3
A22	D10	**DAT25**	R5	**DAT55**	F2
A23	D8	**DAT26**	R4	**DAT56**	C1
A24	B7	**DAT27**	S3	**DAT57**	B2
A25	B8	**DAT28**	Q5	**DAT58**	C2
CLK2	S11	**DAT29**	Q6	**DAT59**	J4
DAT0	T11	**DAT30**	R3	**DAT60**	H3
DAT1	R10	**DAT31**	P4	**DAT61**	G3
DAT2	Q10	**DAT32**	Q3	**DAT62**	A1
DAT3	S10	**DAT33**	S2	**DAT63**	F3
DAT4	T10	**DAT34**	T2	**EVENT#**	C8
DAT5	S9	**DAT35**	U2	**FAIL#**	H4

Table 3: Pin Cross Reference by Pin Name (con't)

Pin Name	Location	Pin Name	Location	Pin Name	Location
PAR0	T8	**VREF**	T12	**VSS**	U7
PAR1	Q7	**VSS**	A6	**VSS**	U8
PAR2	T1	**VSS**	A7	**VSS**	U9
PAR3	B1	**VSS**	A8	**VSS**	U10
RD0#	G4	**VSS**	A9	**VSS**	U11
RD1#	F4	**VSS**	A10	**VSS**	U13
RDY#	Q4	**VSS**	A11	**VSS**	U16
RPTI	N3	**VSS**	B3	**WR0#**	D3
RPTO	R2	**VSS**	B4	**WR1#**	C3
RST#	U12	**VSS**	B13	**XLI0**	C10
STAT0	A2	**VSS**	B14	**XLI1**	C9
STAT1	E3	**VSS**	B15	**XLI2**	E11
VCC	A4	**VSS**	D1	**XLI3**	D11
VCC	A13	**VSS**	F19	**XLI4**	C11
VCC	A14	**VSS**	G1	**XLI5**	B12
VCC	A15	**VSS**	H1	**XLI6**	C12
VCC	B6	**VSS**	H19	**XLI7**	D12
VCC	B9	**VSS**	J1	**XLICLK**	E12
VCC	B10	**VSS**	J19	**XLIDQ0**	D18
VCC	B11	**VSS**	K1	**XLIDQ1**	C19
VCC	D2	**VSS**	K19	**XLINQ0**	A12
VCC	F18	**VSS**	L1	**XLINQ1**	D13
VCC	G2	**VSS**	L19	**XLIP**	C13
VCC	H2	**VSS**	M1	**XLO0**	E16
VCC	J2	**VSS**	N1	**XLO1**	C17
VCC	K2	**VSS**	N19	**XLO2**	F16
VCC	Q2	**VSS**	P1	**XLO3**	D17
VCC	T6	**VSS**	P19	**XLO4**	C18
VCC	T13	**VSS**	Q1	**XLO5**	E17
VCC	T16	**VSS**	U6	**XLO6**	B19

Table 3: Pin Cross Reference by Pin Name (con't)

Pin Name	Location	Pin Name	Location	Pin Name	Location
XLO7	B18	**XRODQ0**	T17	**YUONQ0**	J17
XLOCLK	G15	**XRODQ1**	R13	**YUONQ1**	H18
XLODQ0	C14	**XRONQ0**	R18	**YUOP**	G19
XLODQ1	D14	**XRONQ1**	S19	**YDI0**	S18
XLONQ0	G16	**XROP**	R17	**YDI1**	T19
XLONQ1	H15	**YUI0**	D16	**YDI2**	U19
XLOP	F17	**YUI1**	E15	**YDI3**	Q14
XRI0	R11	**YUI2**	F15	**YDI4**	Q13
XRI1	U14	**YUI3**	A19	**YDI5**	Q15
XRI2	S12	**YUI4**	A18	**YDI6**	R14
XRI3	U15	**YUI5**	D15	**YDI7**	R16
XRI4	T14	**YUI6**	E14	**YDICLK**	S15
XRI5	Q11	**YUI7**	C16	**YDIDQ0**	N17
XRI6	T15	**YUICLK**	C15	**YDIDQ1**	Q19
XRI7	S13	**YUIDQ0**	J18	**YDINQ0**	S17
XRICLK	U18	**YUIDQ1**	K15	**YDINQ1**	R15
XRIDQ0	P15	**YUINQ0**	B17	**YDIP**	S16
XRIDQ1	Q16	**YUINQ1**	A17	**YDO0**	K16
XRINQ0	U17	**YUIP**	E13	**YDO1**	L18
XRINQ1	R12	**YUO0**	H16	**YDO2**	K17
XRIP	S14	**YUO1**	E18	**YDO3**	L17
XRO0	P18	**YUO2**	G17	**YDO4**	K18
XRO1	M15	**YUO3**	D19	**YDO5**	M19
XRO2	N16	**YUO4**	E19	**YDO6**	L15
XRO3	P17	**YUO5**	J16	**YDO7**	M18
XRO4	Q18	**YUO6**	H17	**YDOCLK**	L16
XRO5	P16	**YUO7**	G18	**YDODQ0**	Q12
XRO6	Q17	**YUOCLK**	J15	**YDODQ1**	T18
XRO7	N15	**YUODQ0**	B16	**YDONQ0**	M17
XROCLK	R19	**YUODQ1**	A16	**YDONQ1**	M16
				YDOP	N18

Table 4 provides a cross reference of the pins by pin location.

Table 4: Pin Cross Reference by Location

Location	Pin Name	Location	Pin Name	Location	Pin Name
A1	DAT62	**B12**	XLI5	**D4**	A5
A2	STAT0	**B13**	VSS	**D5**	A9
A3	A4	**B14**	VSS	**D6**	A14
A4	VCC	**B15**	VSS	**D7**	A17
A5	A10	**B16**	YUODQ0	**D8**	A23
A6	VSS	**B17**	YUINQ0	**D9**	A19
A7	VSS	**B18**	XLO7	**D10**	A21
A8	VSS	**B19**	XLO6	**D11**	XLI3
A9	VSS	**C1**	DAT56	**D12**	XLI7
A10	VSS	**C2**	DAT58	**D13**	XLINQ1
A11	VSS	**C3**	WR1#	**D14**	XLODQ1
A12	XLINQ0	**C4**	A7	**D15**	YUI5
A13	VCC	**C5**	A13	**D16**	YUI0
A14	VCC	**C6**	A15	**D17**	XLO3
A15	VCC	**C7**	A18	**D18**	XLIDQ0
A16	YUODQ1	**C8**	EVENT#	**D19**	YUO3
A17	YUINQ1	**C9**	XLI1	**E1**	DAT50
A18	YUI4	**C10**	XLI0	**E2**	DAT52
A19	YUI3	**C11**	XLI4	**E3**	STAT1
B1	PAR3	**C12**	XLI6	**E4**	A3
B2	DAT57	**C13**	XLIP	**E5**	A6
B3	VSS	**C14**	XLODQ0	**E6**	A8
B4	VSS	**C15**	YUICLK	**E7**	A12
B5	A11	**C16**	YUI7	**E8**	A16
B6	VCC	**C17**	XLO1	**E9**	A20
B7	A24	**C18**	XLO4	**E10**	A22
B8	A25	**C19**	XLIDQ1	**E11**	XLI2
B9	VCC	**D1**	VSS	**E12**	XLICLK
B10	VCC	**D2**	VCC	**E13**	YUIP
B11	VCC	**D3**	WR0#	**E14**	YUI6

Table 4: Pin Cross Reference by Location (con't)

Location	Pin Name	Location	Pin Name	Location	Pin Name
E15	YUI1	**H18**	YUONQ1	**M2**	DAT45
E16	XLO0	**H19**	VSS	**M3**	DAT46
E17	XLO5	**J1**	VSS	**M4**	DAT39
E18	YUO1	**J2**	VCC	**M15**	XRO1
E19	YUO4	**J3**	DAT54	**M16**	YDONQ1
F1	DAT53	**J4**	DAT59	**M17**	YDONQ0
F2	DAT55	**J15**	YUOCLK	**M18**	YDO7
F3	DAT63	**J16**	YUO5	**M19**	YDO5
F4	RD1#	**J17**	YUONQ0	**N1**	VSS
F15	YUI2	**J18**	YUIDQ0	**N2**	DAT43
F16	XLO2	**J19**	VSS	**N3**	RPTI
F17	XLOP	**K1**	VSS	**N4**	DAT36
F18	VCC	**K2**	VCC	**N15**	XRO7
F19	VSS	**K3**	DAT49	**N16**	XRO2
G1	VSS	**K4**	DAT51	**N17**	YDIDQ0
G2	VCC	**K15**	YUIDQ1	**N18**	YDOP
G3	DAT61	**K16**	YDO0	**N19**	VSS
G4	RD0#	**K17**	YDO2	**P1**	VSS
G15	XLOCLK	**K18**	YDO4	**P2**	DAT42
G16	XLONQ0	**K19**	VSS	**P3**	DAT37
G17	YUO2	**L1**	VSS	**P4**	DAT31
G18	YUO7	**L2**	DAT48	**P15**	XRIDQ0
G19	YUOP	**L3**	DAT47	**P16**	XRO5
H1	VSS	**L4**	DAT44	**P17**	XRO3
H2	VCC	**L15**	YDO6	**P18**	XRO0
H3	DAT60	**L16**	YDOCLK	**P19**	VSS
H4	FAIL#	**L17**	YDO3	**Q1**	VSS
H15	XLONQ1	**L18**	YDO1	**Q2**	VCC
H16	YUOQ	**L19**	VSS	**Q3**	DAT32
H17	YUO6	**M1**	VSS	**Q4**	RDY#

Table 4: Pin Cross Reference by Location (con't)

Location	Pin Name	Location	Pin Name	Location	Pin Name
Q5	DAT28	**R16**	YDI7	**T8**	PAR0
Q6	DAT29	**R17**	XROP	**T9**	DAT7
Q7	PAR1	**R18**	XRONQ0	**T10**	DAT4
Q8	DAT16	**R19**	XROCLK	**T11**	DAT0
Q9	DAT9	**S1**	DAT40	**T12**	VREF
Q10	DAT2	**S2**	DAT33	**T13**	VCC
Q11	XRI5	**S3**	DAT27	**T14**	XRI4
Q12	YDODQ0	**S4**	DAT23	**T15**	XRI6
Q13	YDI4	**S5**	DAT18	**T16**	VCC
Q14	YDI3	**S6**	DAT15	**T17**	XRODQ0
Q15	YDI5	**S7**	DAT12	**T18**	YDODQ1
Q16	XRIDQ1	**S8**	DAT8	**T19**	YDI1
Q17	XRO6	**S9**	DAT5	**U1**	DAT38
Q18	XRO4	**S10**	DAT3	**U2**	DAT35
Q19	YDIDQ1	**S11**	CLK2	**U3**	DAT20
R1	DAT41	**S12**	XRI2	**U4**	DAT17
R2	RPTO	**S13**	XRI7	**U5**	DAT14
R3	DAT30	**S14**	XRIP	**U6**	VSS
R4	DAT26	**S15**	YDICLK	**U7**	VSS
R5	DAT25	**S16**	YDIP	**U8**	VSS
R6	DAT24	**S17**	YDINQ0	**U9**	VSS
R7	DAT19	**S18**	YDI0	**U10**	VSS
R8	DAT11	**S19**	XRONQ1	**U11**	VSS
R9	DAT6	**T1**	PAR2	**U12**	RST#
R10	DAT1	**T2**	DAT34	**U13**	VSS
R11	XRI0	**T3**	DAT22	**U14**	XRI1
R12	XRINQ1	**T4**	DAT21	**U15**	XRI3
R13	XRODQ1	**T5**	DAT13	**U16**	VSS
R14	YDI6	**T6**	VCC	**U17**	XRINQ0
R15	YDINQ1	**T7**	DAT10	**U18**	XRICLK
				U19	YDI2

3.1 Pathway Bus Interfaces

The iWM has four Pathway input bus interfaces: XLeft input (XLI), YUp input (YUI), XRight input (XRI), and YDown input (YDI). The iWM also has four Pathway output bus interfaces: XLeft output (XLO), YUp output (YUO), XRight output (XRO), and YDown output (YDO). These eight buses are independent of each other. Time on the buses is divided into fixed intervals consisting of four CLK2 intervals after the disassertion of RST#. Timing alignment of the intervals is offset to allow efficient multiplexing. The XLI, XLO, XRI, and XRO buses differ from the YUI, YUO, YDI, and YDO buses by two CLK2 periods.

The X and Y bus pins contain time-multiplexed information as defined in Figure 6.

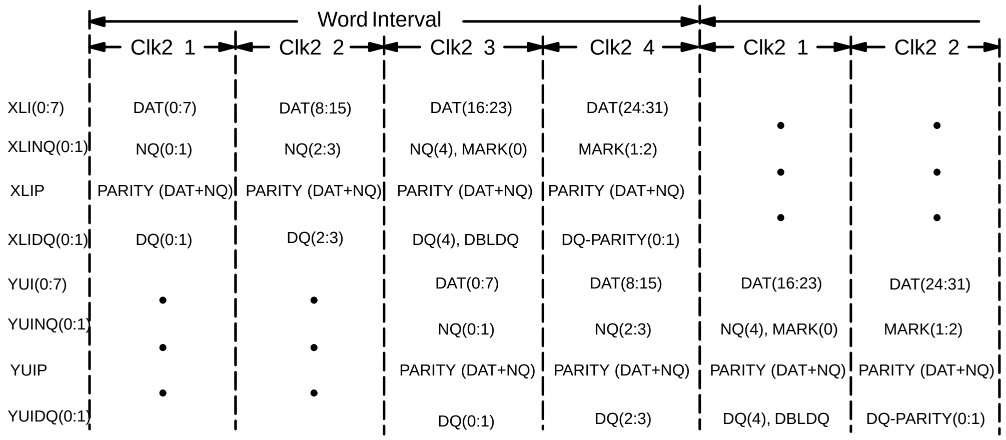


Figure 6: Multiplexing on X and Y buses

Input Buses

The XLI input bus consists of the following pins:

- XLI(0:7)
- XLINQ(0:1)
- XLIP
- XLIDQ(0:1)
- XLICLK

Table 5 defines the XLI input bus pins. In the TYPE column, I indicates input and O indicates output.

Table 5: XLI Input Bus Pins

<table>
<tr><th>PINS</th><th>TYPE</th><th colspan="2">FUNCTION</th></tr>
<tr><td>XLI(0:7)</td><td>I</td><td colspan="2">XLI0-XLI7 are the eight data input lines of the XLI input bus interface. Only 32-bit words are sent over these input lines, and each word is received as four sequential, adjacent, and indivisible 8-bit records. The XLI input bus is driven from an adjacent iWM's output bus.</td></tr>
<tr><td rowspan="3">XLINQ(0:1)</td><td rowspan="3">I</td><td colspan="2">XLINQ0 and XLINQ1 receive multiplexed NQ(0:4) and MARK(0:2) information. NQ selects the PCT buffer that receives the data in the current word interval. If the value of NQ is 11111B, no data is present in the current word interval. The MARK information denotes the following boundaries:</td></tr>
<tr><td>MARK value</td><td>Definition</td></tr>
<tr><td>000
001
010
011
100
101

110

111</td><td>Data Word
Software Mark
End Message
Begin Message + ID Word
End Connection
Begin Connection + ID
Word
End Message and
Connection
Begin Connection and
Message + ID Word</td></tr>
<tr><td>XLIP</td><td>I</td><td colspan="2">XLIP contains odd parity computed across both DAT(0:7) and XLINQ(0:1) during each CLK2 period.</td></tr>
</table>

Table 5: XLI Input Bus Pins (con't)

PINS	TYPE	FUNCTION
XLIDQ(0:1)	O	XLIDQ0 and XLIDQ1 return multiplexed DQ(0:4), DBLDQ, and DQ-PARITY(0:1) information. DQ is an index that indicates a word or double word has been dequeued from the indexed PCT buffer in this component. The sending component can then track how full the receiving buffers are. DBLDQ indicates whether a single or double word was dequeued. DBLDQ = 1 indicates a double word. DQ-PARITY(0:1) is odd parity computed on the information of the previous three clocks on the XLIDQ(0:1) pins.
XLICLK	I	XLICLK clocks in the XLI input information.

The YUI, XRI, and YDI input bus pins have the same pin descriptions as the corresponding XLI pins defined in Table 5.

The YUI input bus consists of the following pins:

- YUI(0:7)
- YUINQ(0:1)
- YUIP
- YUIDQ(0:1)
- YUICLK

The XRI input bus consists of the following pins:

- XRI(0:7)
- XRINQ(0:1)
- XRIP
- XRIDQ(0:1)
- XRICLK

The YDI input bus consists of the following pins:

- YDI(0:7)
- YDINQ(0:1)
- YDIP

- YDIDQ(0:1)
- YDICLK

Output Buses

The XLO output bus consists of the following pins:

- XLO(0:7)
- XLONQ(0:1)
- XLOP
- XLODQ(0:1)
- XLOCLK

Table 6 defines the XLO output bus pins.

Table 6: XLO Output Bus Pins

PINS	TYPE	FUNCTION	
XLO(0:7)	O	XLO0-XLO7 are the eight data output lines of the XLO output bus interface. Only 32-bit words are sent over these output lines, and each word is sent as four sequential, adjacent, and indivisible 8-bit records. The XLO output bus drives an adjacent iWM's input bus.	
XLONQ(0:1)	O	XLONQ0 and XLONQ1 send multiplexed NQ(0:4) and MARK(0:2) information. NQ selects the PCT buffer that receives the data in the current word interval. If the value of NQ is 11111B, no data is present in the current word interval. The MARK information denotes the following boundaries:	
		MARK value	Definition
		000	Data Word
		001	Software Mark
		010	End Message
		011	Begin Message + ID Word
		100	End Connection
		101	Begin Connection + ID Word
		110	End Message and Connection
		111	Begin Connection and Message + ID Word

Table 6: XLO Output Bus Pins (con't)

PINS	TYPE	FUNCTION
XLOP	O	XLOP contains odd parity computed across both DAT(0:7) and XLONQ(0:1) during each CLK2 period.
XLODQ(0:1)	I	XLODQ0 and XLODQ1 receive multiplexed DQ(0:4), DBLDQ, and DQ-PARITY(0:1) information. DQ is an index that indicates a word or double word has been dequeued from the indexed PCT buffer in the sending component. DBLDQ indicates whether a single or double word was dequeued. DBLDQ = 1 indicates a double word. DQ-PARITY(0:1) is odd parity computed on the information of the previous three clocks on the XLODQ(0:1) pins.
XLOCLK	O	XLOCLK clocks out the XLO output information.

The YUO, XRO, and YDO output bus pins have the same pin descriptions as the corresponding XLO pins defined in Table 6.

The YUO output bus consists of the following pins:

- YUO(0:7)
- YUONQ(0:1)
- YUOP
- YUODQ(0:1)
- YUOCLK

The XRO output bus consists of the following pins:

- XRO(0:7)
- XRONQ(0:1)
- XROP
- XRODQ(0:1)
- XROCLK

The YDO output bus consists of the following pins:

- YDO(0:7)
- YDONQ(0:1)
- YDOP
- YDODQ(0:1)
- YDOCLK

3.2 Local Memory Interface

The Local Memory Interface connects the iWM to its local memory and provides an interface to a host system or other external interface. Table 7 defines the Local Memory Interface pins.

Table 7: Local Memory Interface pins

PINS	TYPE	FUNCTION
DAT(0:63)	I/O	The bidirectional data path to and from local memory.
A(3:25)	O	The address bus to the local memory array. The addresses are word addresses, so local memory can access up to 64 Mbytes.
WR0#	O	The memory write control signal for the low order word of the 64-bit array. WR0# is active low.
WR1#	O	The memory write control signal for the high order word of the 64-bit array. WR1# is active low.
RD0#	O	The memory read control signal for the low order word of the 64-bit array. RD0# is active low.
RD1#	O	The memory read control signal for the high order word of the 64-bit array. RD1# is active low.
RDY#	I	Signals the processor that a memory operation is complete. RDY# allows slower memories to be incorporated into the local memory array. RDY# is active low.
PARITY(0:3)	I/O	Provides even parity error checking on local memory accesses. Each parity bit checks parity on two bytes of data. These pins are bidirectional.
EVENT#	I	Signals an external event to the Computation Agent. EVENT# is active low.

3.3 System Reporting

Table 8 provides descriptions of the system reporting pins.

Table 8: System Reporting Pins

PINS	TYPE	FUNCTION
RPTO	O	Reports system-wide event messages. These messages are composed of a start bit, message type, source ID, and parity bit.
RPTI	I	Receives the OR of all system-wide event messages. RPTO and RPTI support a priority arbitration scheme that allows the highest priority simultaneous message to supersede lower priority messages.

3.4 System Support

Table 9 provides descriptions of the system support pins.

Table 9: System Support Pins

PINS	TYPE	FUNCTION
CLK2	I	External system clock input. CLK2 is internally divided to produce two non-overlapping active-high phase clocks. 20 Mhz operation requires a 40 Mhz CLK2.
RST#	I	Initializes the component to a known initial state. RST# is active low.
VREF	I	A voltage reference input that sets the input pin threshold.

Table 9: System Support Pins (con't)

PINS	TYPE	FUNCTION	
STAT0 STAT1 FAIL#	I/O	These pins combine to perform the following three separate functions: • input three bits of configuration information during reset • serve as two software controlled status open-drain outputs and one fail status output after reset • serve as test control inputs when test mode is enabled	
		The configuration information is defined as follows, with the order of bits, STAT1 STAT0 FAIL#:	
		000	interface cell, skip self-test
		001	interface cell, run self-test
		010	array cell, skip self-test
		011	array cell, run self-test
		100	test mode, skip self-test
		101	test mode, run self-test
		110	set ROM patch mask
		111	burn-in

4. Electrical Characteristics

Sections 4.1 through 4.3 describe the electrical characteristics of the iWM.

4.1 Maximum Ratings

Table 10 represents a stress rating only, and functional operation at the maximums is not guaranteed. Sections 4.2 and 4.3 specify functional operating conditions.

Extended exposure to the maximum ratings may affect device reliability. Although the iWM contains protective circuitry to resist damage from static electric discharge, always take precautions to avoid high static voltages or electric fields.

Table 10: Absolute Maximum Ratings

Parameter	Maximum Rating
Storage Temperature	-65°C to +150°C
Case Temperature under Bias	-65°C to +110°C
Supply Voltage with respect to V_{SS}	-0.5V to +6.5V
Voltage on Other Pins	-0.5V to V_{CC} + 0.5V
Power Dissipation	7 Watts

4.2 D.C. Specifications

Table 11 provides the D.C. characteristics of the iWM.

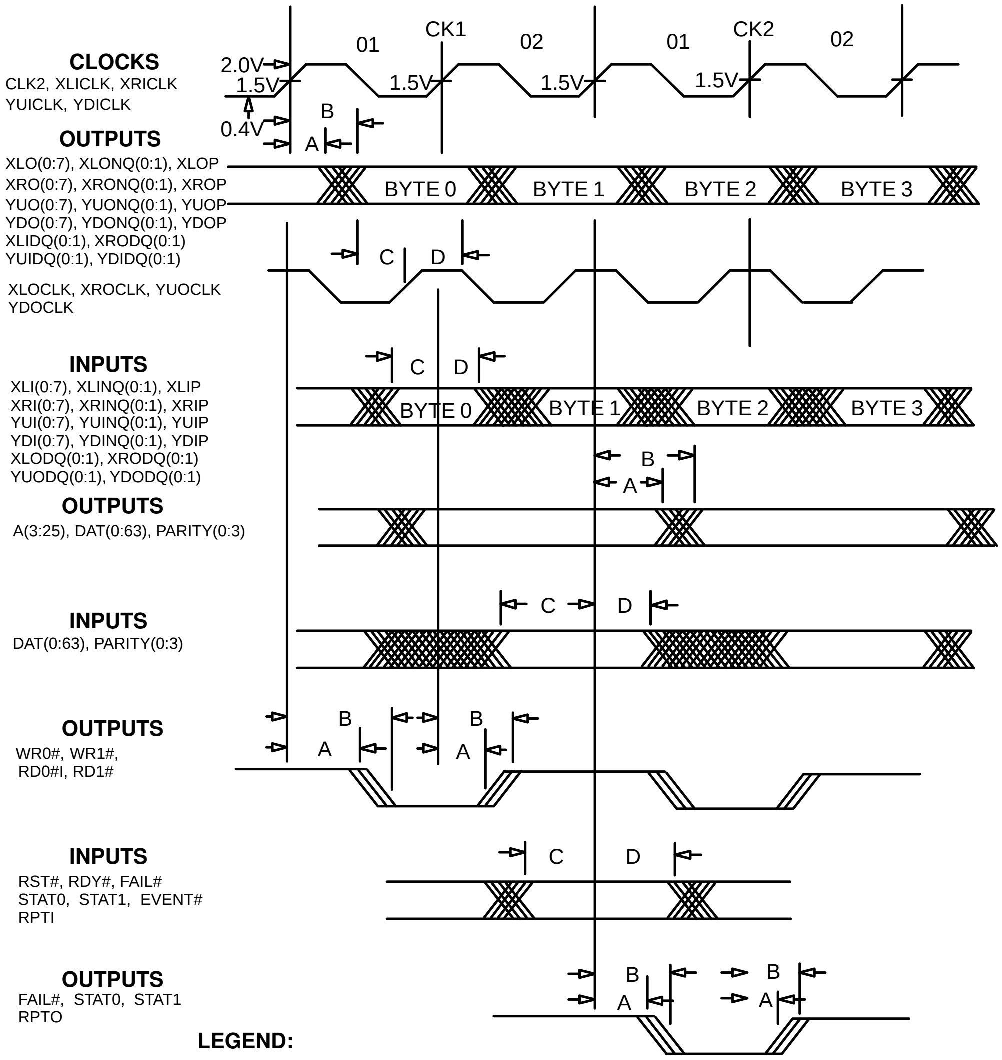
Functional Operating Range: V_{CC} = 5V ± 5%; T_{CASE} = 0°C to 85°C

Table 11: D.C. Characteristics

Symbol	Parameter	Min	Max	Unit	Notes
V_{IL}	Input Low Voltage	-0.3	1.3	V	
V_{IH}	Input High Voltage	2.0	V_{CC}+0.3	V	
V_{REF}	Reference Voltage	1.0	2.0		
V_{OL}	Output Low Voltage I_{OL} = 4ma		0.4	V	
V_{OH}	Output High Voltage I_{OH} = 1ma	2.4		V	
I_{LI}	Input Leakage Current		±10	µA	$0V \leq V_{IN} \leq V_{CC}$
I_{LO}	Output Leakage Current		±10	µA	$0.4V \leq V_{OUT} \leq V_{CC}$
I_{VREF}	Reference Leakage Current		±10	µA	
I_{CC}	Supply Current CLK2 = 40MHz		1000	mA	
C_{IN}	Input Capacitance		10	pF	F_C = 1MHz
C_{OUT}	Output or I/O Capacitance		10	pF	F_C = 1MHz

4.3 A.C. Specifications

Figure 7 defines A.C. specification measurement. Inputs must be driven to the voltage levels indicated by Figure 7 when measuring A.C. specifications. Specifications for ouput delays include minimum and maximum limits, and Figure 7 illustrates how these limits are measured. The minimum delay times are hold times provided to external circuitry. The specifications for setup and hold times are minimums, defining the smallest acceptable sampling window. Within the sampling window, synchronous input signals must be stable for the iWM to operate correctly.



LEGEND:

A - minimum output delay spec
B - maximum output delay spec
C - minimum input setup spec
D - minimum input hold spec

Figure 7: A.C. Specification Drive Levels and Measurement Points

Table 12 provides the A.C. specifications for output delays, input setup requirements, and input hold requirements. All A.C. specifications are relative to the rising edge of the CLK2 signal crossing the 1.5V level.

Table 12: iWM A.C. Specifications

Symbol	Parameter	Min	Max	Units	Figure	Notes
	Operating Frequency	4	20	MHz		0.5 X F_{CLK2}
t_1	CLK2 Period	25	125	ns	8	
t_{2a}	CLK2 XLICLK XRICLK YUICLK YDICLK High Time	6		ns	8	at 2V
t_{2b}	CLK2 XLICLK XRICLK YUICLK YDICLK Low Time	6		ns	8	at 0.8V
t_3	A3-A25 Valid Delay	4	11	ns	9	C_L = 150pF
t_4	WR0#, WR1# RD0#, RD1# Valid Delay	11	18	ns	9	C_L = 75pF
t_5	WR0#, WR1# RD0#, RD1# Invalid Delay	46	53	ns	9	C_L = 75pF
t_6	DAT0-DAT63 PAR0-PAR3 Valid Delay	22	34	ns	9	C_L = 50 pF
t_7	DAT0-DAT63 PAR0-PAR3 Float Delay	3	8	ns	9	C_L = 50pF

Table 12: iWM A.C. Specifications (con't)

Symbol	Parameter	Min	Max	Units	Figure	Notes
t_{8a}	DAT0-DAT63 PAR0-PAR3 Setup Time	3		ns	9	
t_{8b}	DAT0-DAT63 PAR0-PAR3 Hold Time	1		ns	9	
t_{9a}	RDY# EVENT# Setup Time	3		ns	9	
t_{9b}	RDY# EVENT# Hold Time	1		ns	9	
t_{10a}	XRO0-XRO7 XLO0-XLO7 YDO0-YDO7 YUO0-YUO7 XRONQ0-XRONQ1 XLONQ0-XLONQ1 YDONQ0-YDONQ1 YUONQ0-YUONQ1 XRIDQ0-XRIDQ1 XLIDQ0-XLIDQ1 YDIDQ0-YDIDQ1 YUIDQ0-YUIDQ1 XROP XLOP YDOP YUOP Valid Delay	4	10	ns	10	$C_L = 50pF$
t_{10b}	XROCLK XLOCLK YDOCLK YUOCLK Valid Delay	11	14	ns	10	$C_L = 50pF$

Table 12: iWM A.C. Specifications (con't)

Symbol	Parameter	Min	Max	Units	Figure	Notes
t_{11}	XRI0-XRI7 XLI0-XLI7 YDI0-YDI7 YUI0-YUI7 XRINQ0-XRINQ1 XLINQ0-XLINQ1 YDINQ0-YDINQ1 YUINQ0-YUINQ1 XRODQ0-XRODQ1 XLODQ0-XLODQ1 YDODQ0-YDODQ1 YUODQ0-YUODQ1 XRIP XLIP YDIP YUIP Setup Time	3		ns	10	
t_{12}	XRI0-XRI7 XLI0-XLI7 YDI0-YDI7 YUI0-YUI7 XRINQ0-XRINQ1 XLINQ0-XLINQ1 YDINQ0-YDINQ1 YUINQ0-YUINQ1 XRODQ0-XRODQ1 XLODQ0-XLODQ1 YDODQ0-YDODQ1 YUODQ0-YUODQ1 XRIP XLIP YDIP YUIP Hold Time	1		ns	10	

Table 12: iWM A.C. Specifications (con't)

Symbol	Parameter	Min	Max	Units	Figure	Notes
t_{13a}	RST# Setup Time	6		ns	11	
t_{13b}	RST# Pulse Width	8 x t1			11	
t_{14}	RST# Hold Time	0		ns	11	
t_{15}	FAIL# STAT0 STAT1 RPTI Setup Time	3		ns	11, 12	
t_{16}	FAIL# STAT0 STAT1 RPTI Hold Time	1		ns	11, 12	
t_{17}	FAIL# STAT0 STAT1 RPTO Valid Delay	4	10	ns	12	C_L = 50pF

5. Timing Characteristics

Figures 8 through 12 illustrate timings for the iWM.

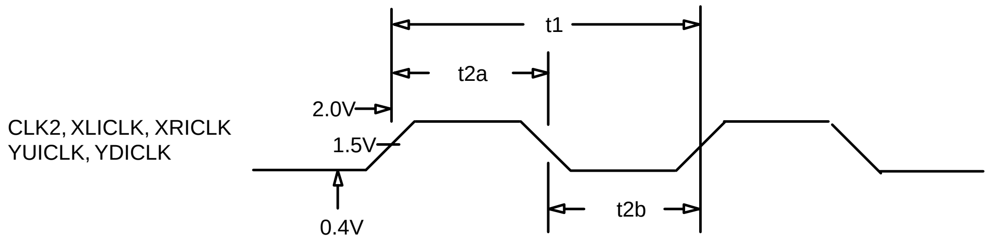


Figure 8: Clock Timing

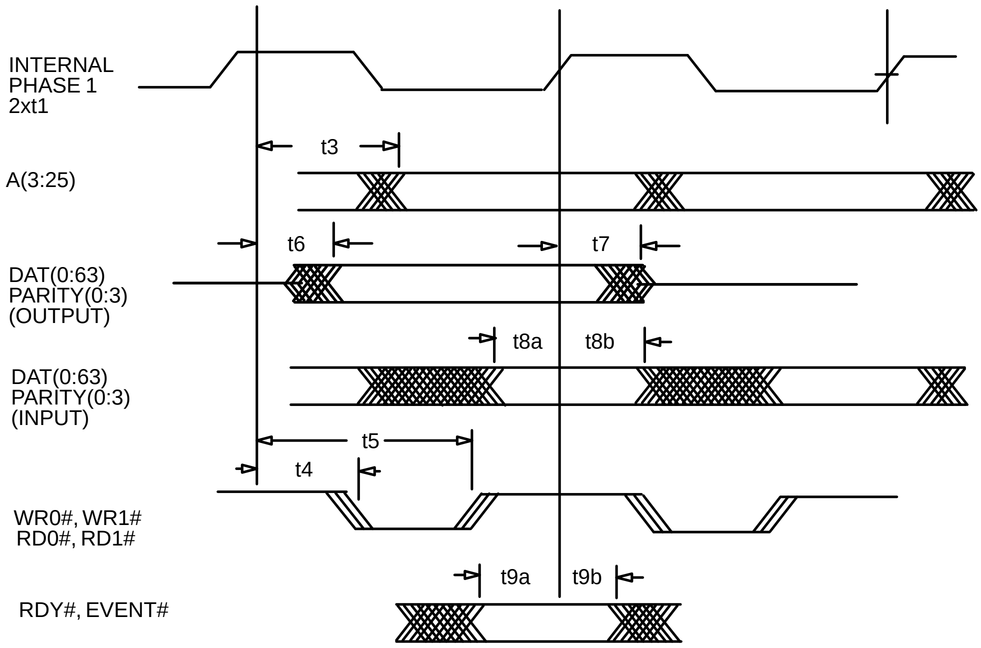


Figure 9: Local Memory Timing

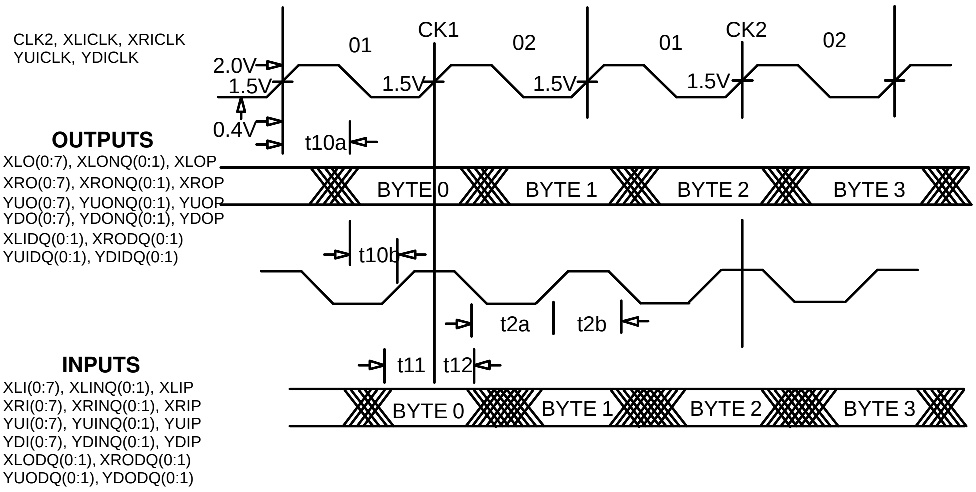


Figure 10: Pathway Timing

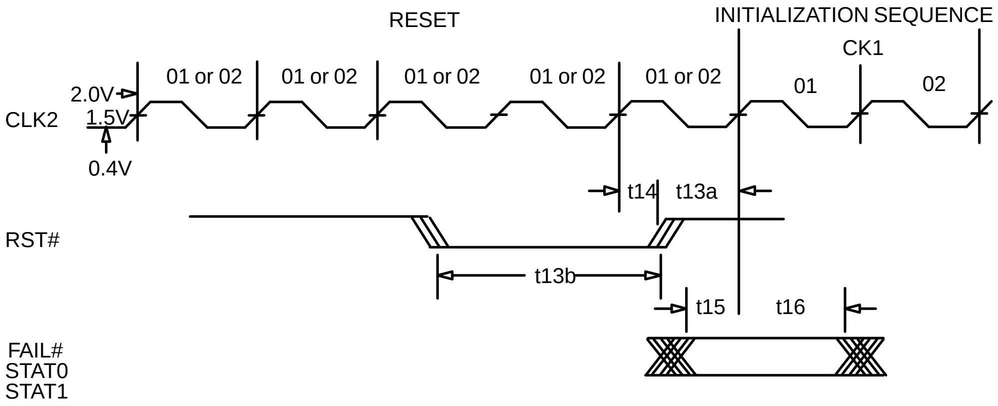


Figure 11: Reset Timing

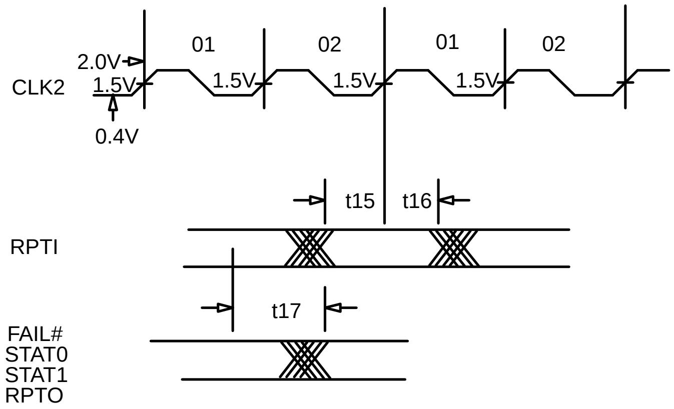


Figure 12: System Reporting and Support Timing