

GND	101	GND	102
VCC	103	VCC	103
VCC	104	VCC	104
+12V	105	+12V	105
-5V	106	-5V	106
GND	107	GND	107

BCLK/	113	INIT/	114
BPRN/	115	BPRO/	116
BUSY/	117	BREQ/	118
MRDC/	119	MWTC/	120
IORC/	121	IOWC/	122
XACK/	123		

BHEN/	127	AD10/	128
CCLK/	131	AD11/	132
		AD12/	133
		AD13/	134

INT6/	135	INT7/	136
INT4/	137	INT5/	138
INT2/	139	INT3/	140
INT0/	141	INT1/	142

ADRE/	143	ADRF/	144
ADRC/	145	ADRD/	146
ADRA/	147	ADRB/	148
ADR8/	149	ADR9/	150
ADR6/	151	ADR7/	152
ADR4/	153	ADR5/	154
ADR2/	155	ADR3/	156
ADR0/	157	ADR1/	158

DATE/	159	DATF/	160
DATC/	161	DATD/	162
DATA/	163	DATB/	164
DAT8/	165	DAT9/	166
DAT6/	167	DAT7/	168
DAT4/	169	DAT5/	170
DAT2/	171	DAT3/	172
DAT0/	173	DAT1/	174

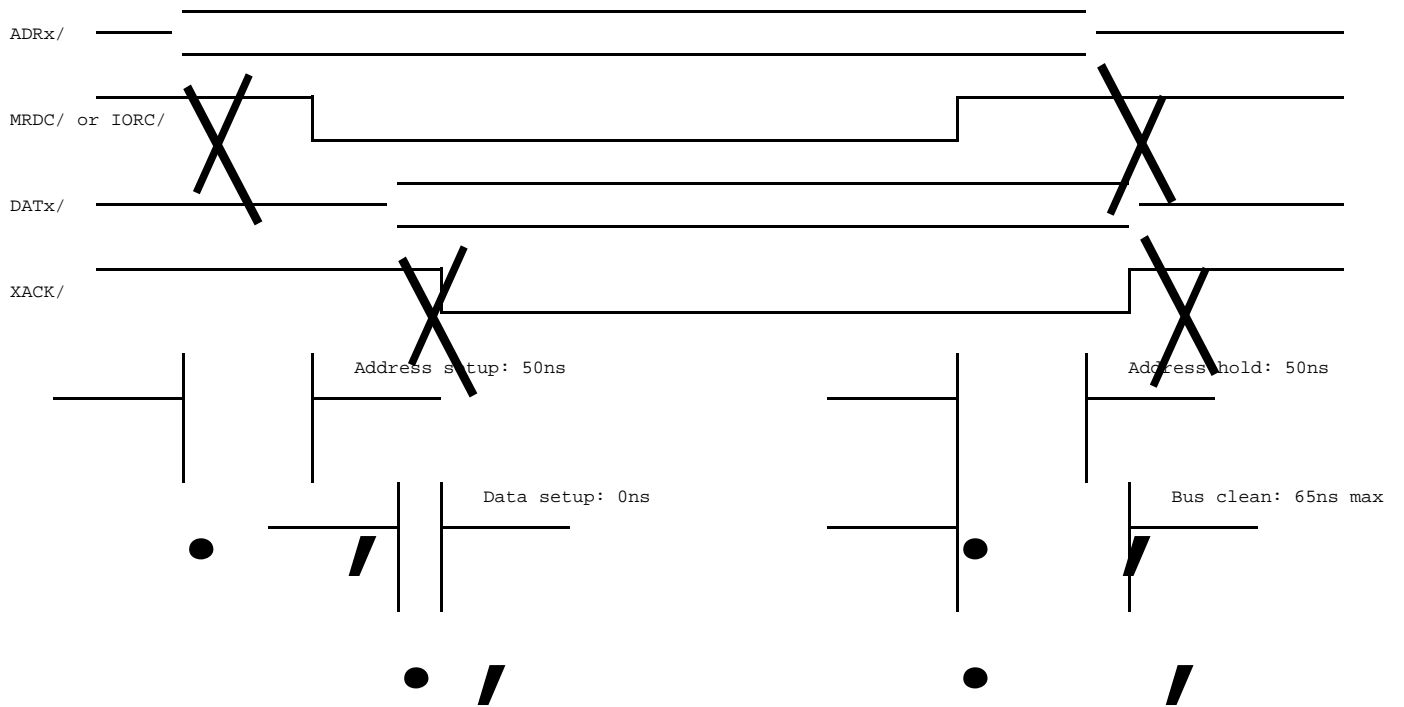
GND	175	GND	176
-12V	179	-12V	180
VCC	181	VCC	182
VCC	183	VCC	184
GND	185	GND	187

Component side P1 Solder side

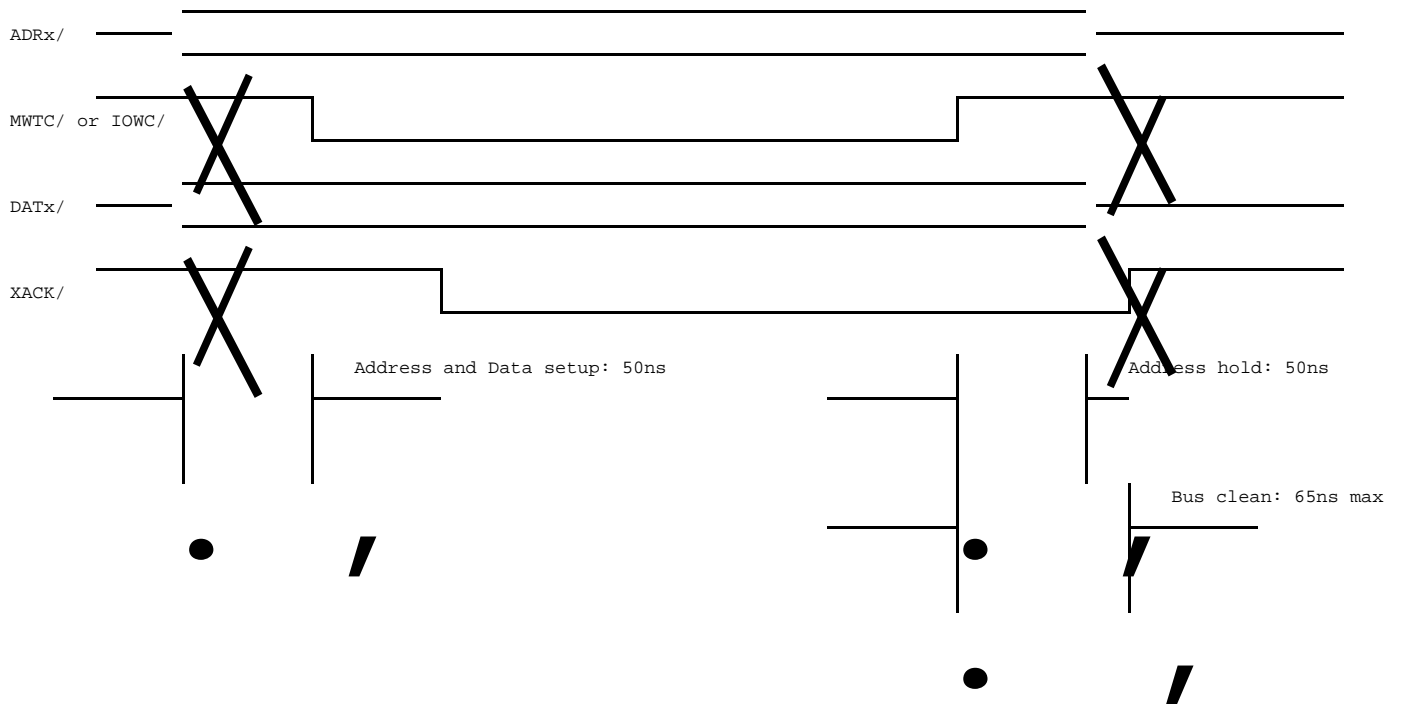
AD16/	255	AD17/	256
AD14/	257	AD15/	258

Component side P2 Solder side

### Read



### Write





\* These 4 bits are inverted when stored in CS RAM or ROM.

Field	Description
rA	2901 A reg addr, U addr [0-3]
rB	2901 B reg addr, RH addr
aS	2901 alu Source operand pair
aF	2901 alu Function
aD	2901 alu Destination/shift control
ep	Even Parity
Cin	2901 Carry In, Shift Ends, writesSU (if enSU=1)
enSU	enable SU reg file
mem	MAR_ (if c1), MDR_ (if c2), _MD (if c3)
fS	Function field Selector
fX	X Function
fY	Y Function
fZ	Z Function
INIA	Next Instruction Address

aS	R, S	aF	F	sh, aD	R[rB]	Q	Ybus
0	A, Q	0	R + S + Cin	0	no write	F	F
1	A, B	1	S - R - Cin'	1	no write	no write	F
2	0, Q	2	R - S - Cin'	2	F	no write	A
3	0, B	3	R or S	3	F	no write	F
4	0, A	4	R and S	4	F/2	Q/2	F
5	D, A	5	~R and S	5	F/2	no write	F
6	D, Q	6	R xor S	6	2F	2Q	F
7	D, 0	7	~R xor S	7	2F	no write	F

sh \_ (fX=shift) OR (fX=cycle) OR (fY=cycle)

fS[0-1]	fY=	fS[2-3]	fZ=	SU addr[0-7]
0	DispBr	0	fZNorm	0,,stackP
1	fyNorm	1	Nibble	0,,stackP
2	IOOut	2	Uaddr[4-7]	rA,,fZ   rA,,Y[12-15]* IF fZ=AltUaddr*
3	Byte	3	IOXin	rA,,fZ   rA,,Y[12-15]* IF fZ=AltUaddr*

\* as executed by previous u-instr

fX	fXNorm	fY	fYNorm	DispBr	IOOut	fZ	fZNorm	IOXIn
0	pCall/Ret0	0	*	NegBr	DebA_	0	*	ADR0
1	pCall/Ret1	1	*	ZeroBr	ExtCtrl_	1	IBPtr_1	
2	pCall/Ret2	2	ClrIntErr	NZeroBr		2	IBPtr_0	
3	pCall/Ret3	3	IBDisp	MesaIntBr		3	Cin_pc16	
4	pCall/Ret4	4	MesaIntRq	PgCarryBr		4		
5	pCall/Ret5	5	stackP_	CarryBr		5	pop	
6	pCall/Ret6	6	IB_	XRefBr		6	push	_ExtStat
7	pCall/Ret7	7	cycle	NibCarryBr		7	AltUaddr	IO_
8	Noop	8	Noop	XDisp		8	Noop	_DebB
9	RH_	9	Map_	YDisp		9		
A	shift	A	*	XC2npcDisp		A		_ErrnIBnStkp
B	cycle	B	push			B		_RH
C	Cin_pc16	C	IO_	XwdDisp	IO_, BHEN	C	LRot0	_ibNA
D	Map_	D	Bank_	XHDisp		D	LRot12	_ib
E	pop	E	RawRef	XLDisp	RawRef, BHEN	E	LRot8	_ibLow
F	push	F		PgCrOvDisp	BHEN	F	LRot4	_ibHigh

pCall when NIA[7]=0pRet when NIA[7]=1.

\* Left unused for Burdock DLion/Dicentra simplicity.