USB2.0 to Local CPU Bus controller

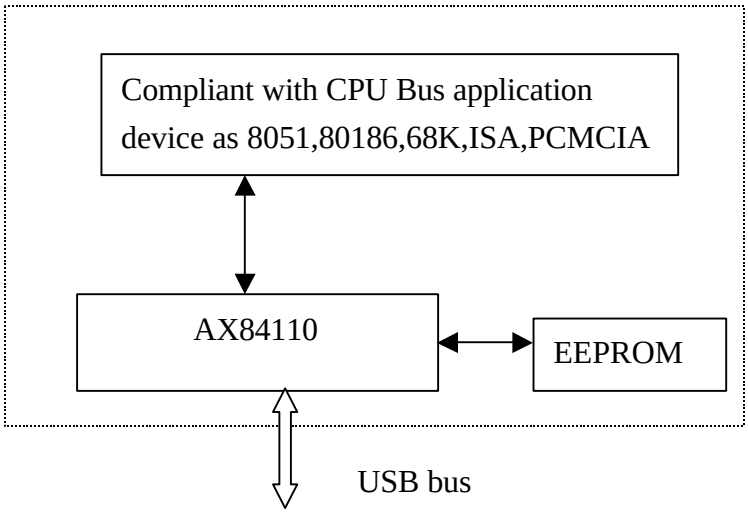
Document No.: AX84110 / V0.0 / Dec. 16 '02

Features:

- Compliant with USB specification 1.0, 1.1 and 2.0
- Support both 8 bit and 16 bit local CPU Interfaces include MCS-51 series, 80186 series, MC68K series CPU, ISA Bus and PCMCIA Bus
- Capability Full/High Speed USB Device with bus power
- Support 4 endpoints on USB and 4 General Purpose In/Out pins
- Embedded 2K*16 bit SRAM
- 128-pin LQFP low profile package
- Support suspended mode and remote wakeup
- Single 12MHz clock input, pure 3.3V operation with 5V I/O tolerance
- Support (93c56/93c66) 256/512 bytes serial EEPROM (used for saving USB Descriptors)
- Support 4 set of status change reply
- Support automatic loading of application device, from application device USB Descriptors and Adapter Configuration from EEPROM when power-on initialization

Product description

The AX84110 USB2.0 to Local CPU Bus controller is a high performance and with embedded 2K*16 SRAM and compliant with USB Standard V1.0, V1.1 and V2.0. The AX84110 supports Full/High Speed USB device with bus power capability and supports both 8 bit and 16 bit local CPU interfaces include MCS-51 series, 80186 series, MC68K series CPU and ISA Bus and PCMCIA Bus.



Always contact ASIX for possible updates before starting a design.

This data sheet contains new products information. ASIX ELECTRONICS reserves the rights to modify the product specification without notice.

No liability is assumed as a result of the use of this product. No rights under any patent accompany the sale of the product.

Contents

1.0Introduction

1.1General Description

The AX84110 USB2.0 to Local CPU Bus controller is a high performance and with embedded 2K*16 SRAM. The AX84110 supported Full/High Speed USB device with bus power capability and supports both 8 bit and 16bit local CPU interface include MCS-51 series,80186 series,MC68K series CPU and ISA Bus and PCMCIA Bus. The Chip also support up to 4 additional General Purpose In/Out pins and receive 4 set of status change from application device.

AX84110 use 128-pin LQFP low profile package, 12MHz operation for USB and CMOS process with pure 3.3V operation with 5V i/o tolerance.

1.2AX84110 Block Diagram

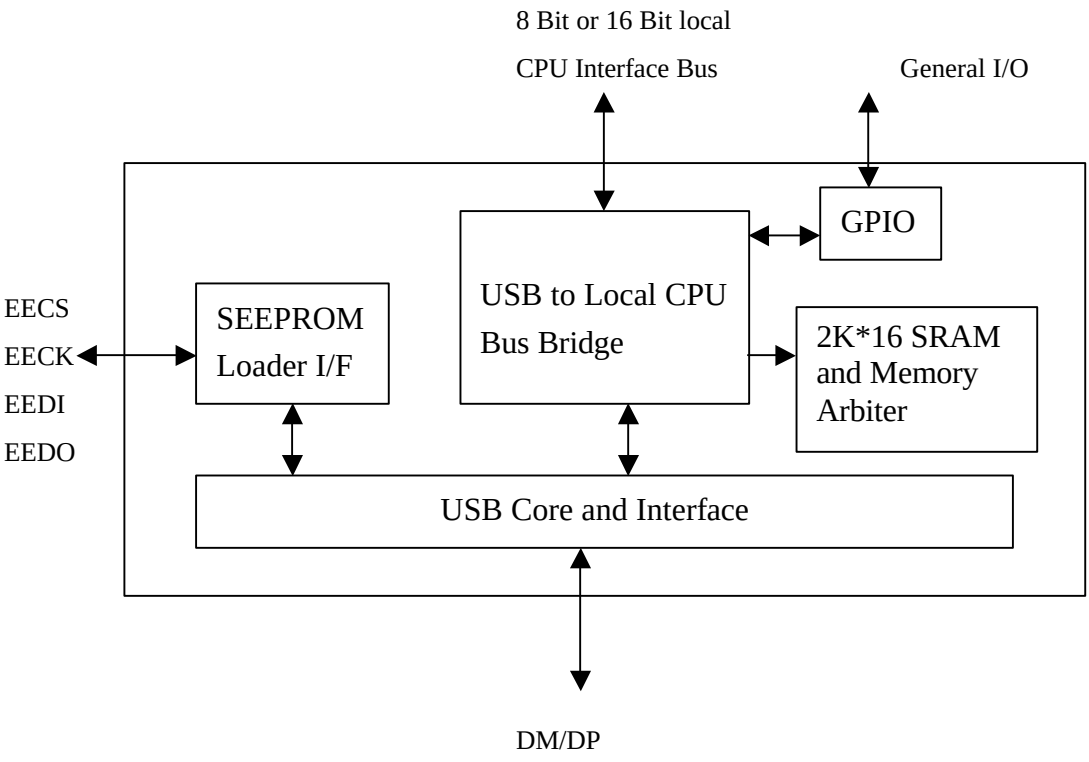
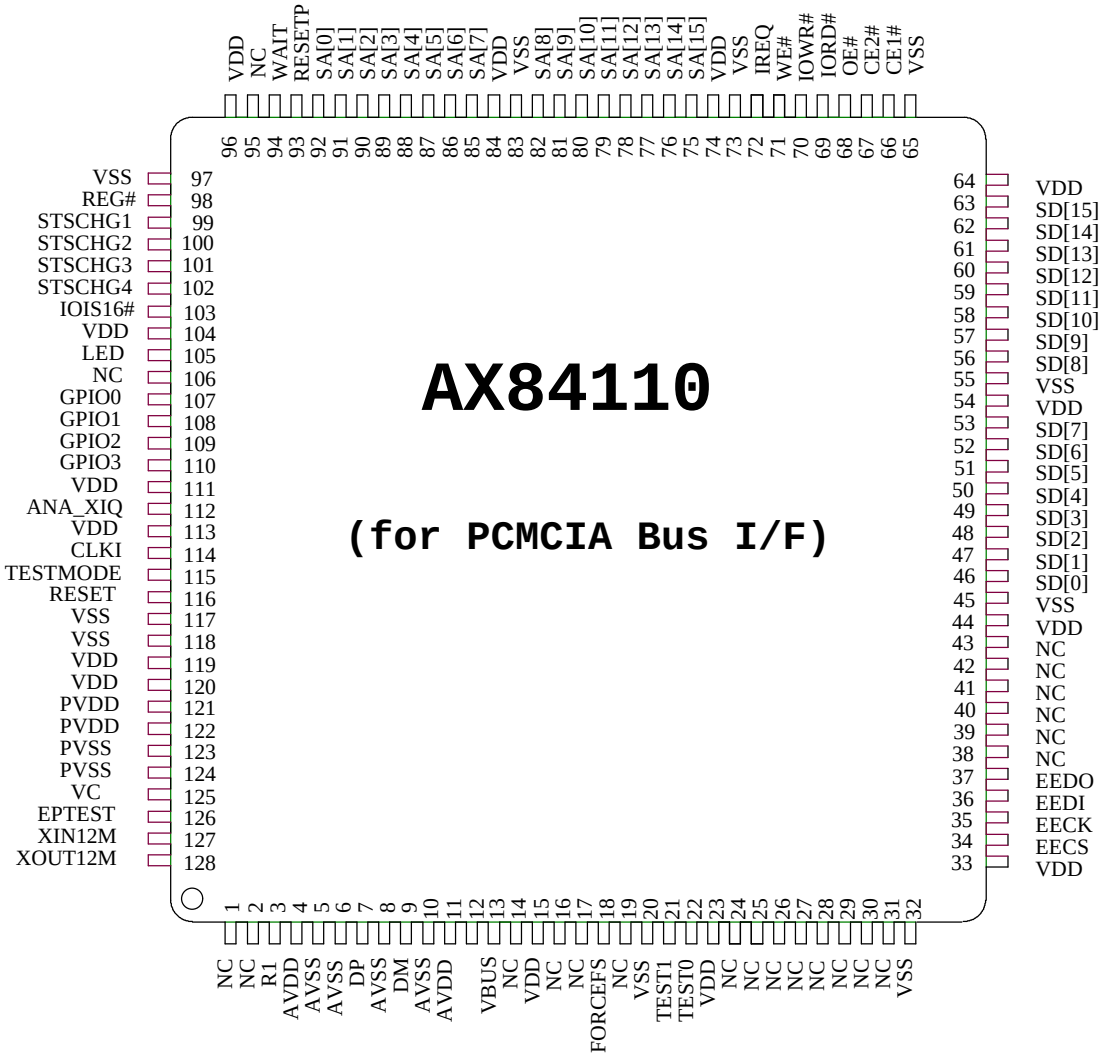


Fig-1 AX84110 Block Diagram

1.3AX84110 Pin Connection Diagram

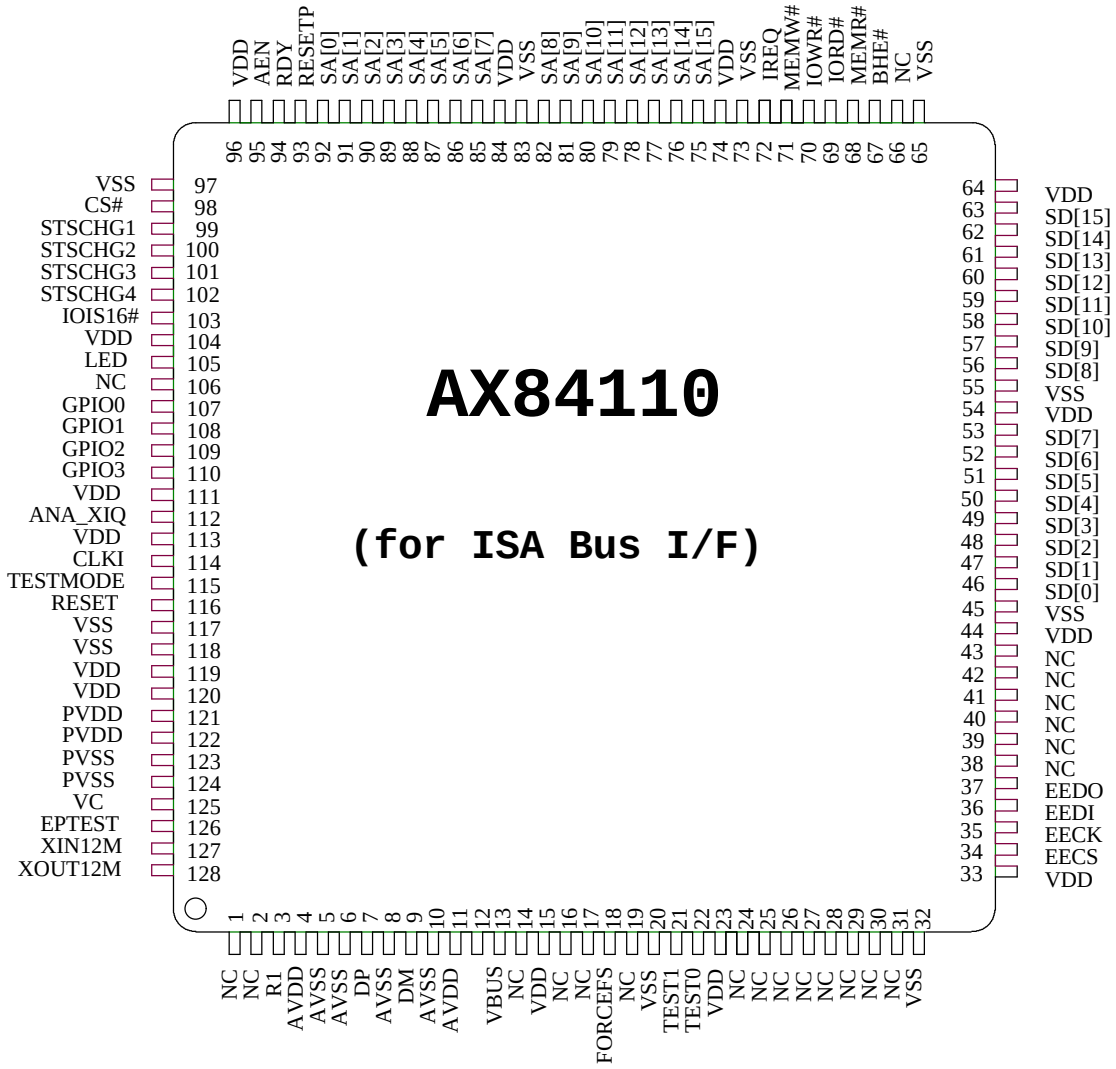
1.3.1 AX84110 Pin Connection Diagram for PCMCIA Mode



* Pin 12 is EXTWAKEUP

Fig-2 AX84110 Pin Connection Diagram for PCMCIA Bus Mode

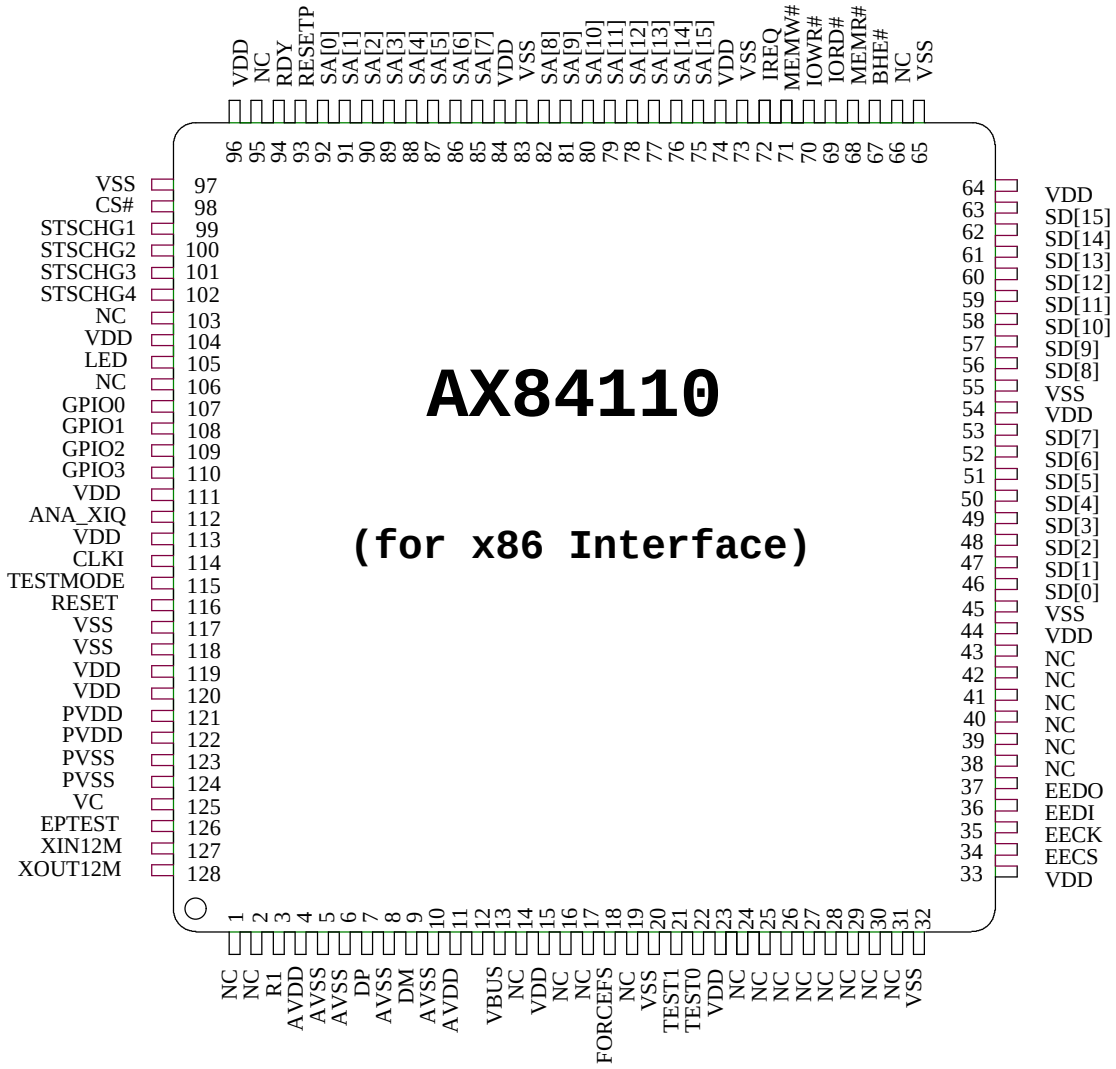
1.3.2 AX84110 Pin Connection Diagram for ISA Mode



* Pin 12 is EXTWAKEUP

Fig-3 AX84110 Pin Connection Diagram for ISA Bus Mode

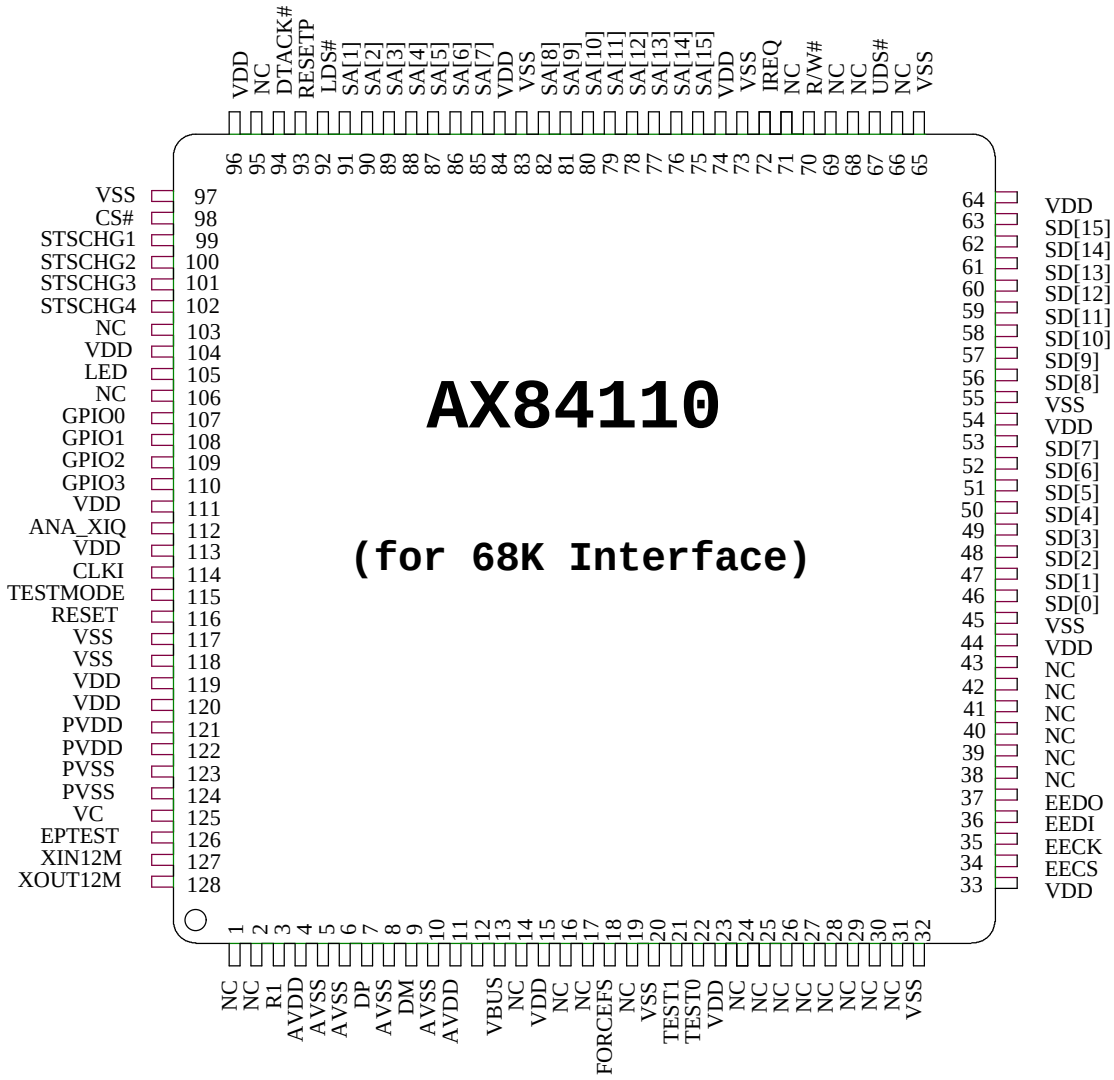
1.3.3 AX84110 Pin Connection Diagram for 80x86 Mode



* Pin 12 is EXTWAKEUP

Fig-4 AX84110 Pin Connection Diagram for 80x86 Mode

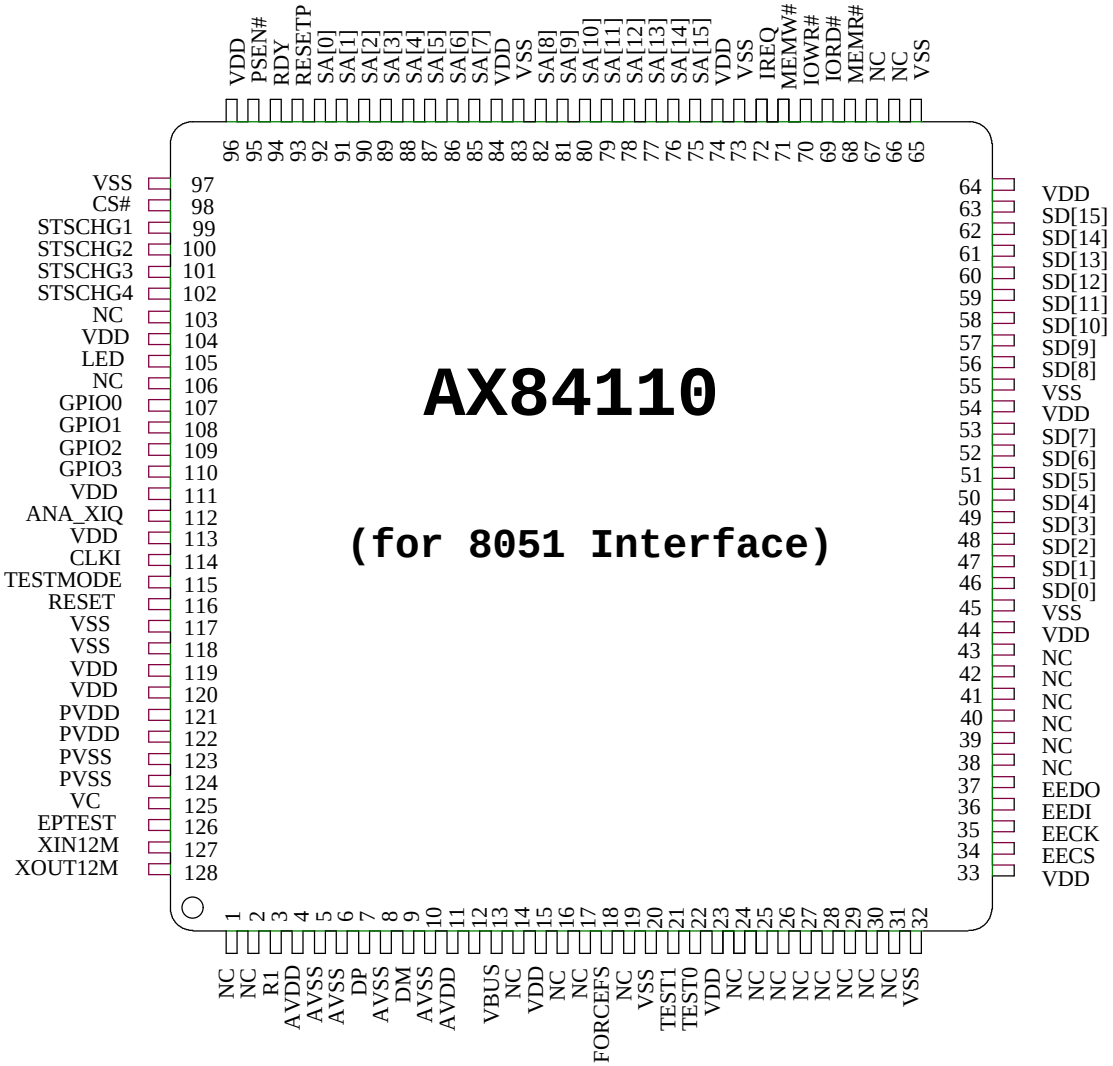
1.3.4 AX84110 Pin Connection Diagram for MC68K Mode



* Pin 12 is EXTWAKEUP

Fig-5 AX84110 Pin Connection Diagram for MC68K Mode

1.3.5 AX84110 Pin Connection Diagram for MCS-51 Mode



* Pin 12 is EXTWAKEUP

Fig-6 AX84110 Pin Connection Diagram for MCS-51 Mode

2.0 Signal Description

The following terms describe the AX84110 pin-out:

All pin names with the "#" suffix are asserted low.

The following abbreviations are used in following Tables.

I	Input	PU	Internal Pull Up (100K)
O	Output	PD	Internal Pull Down (100K)
I/O	Input/Output	P	Power Pin
OD	Open Drain	F	Failsafe

SIGNAL	TYPE	PIN NO.	DESCRIPTION
NC		1	No Connection
NC		2	No Connection
R1	I	3	Constant-voltage pin A 6.2K ¡ 1% resistors is connected to AVSS. Be sure to make the line between R1 and each resistor as short as possible.
AVDD	P	4	Power supply pin for analog circuits +3.3V DC
AVSS	P	5	Power supply pin for analog circuits Ground
AVSS	P	6	Power supply pin for analog circuits Ground
DP	B	7	USB data line Data+
AVSS	P	8	Power supply pin for analog circuits Ground
DM	B	9	USB data line Data-
AVSS	P	10	Power supply pin for analog circuits Ground
AVDD	P	11	Power supply pin for analog circuits +3.3V DC
EXTWAKEUP	I/PD	12	Remote-wakeup trigger from external pin. It active high and should be keep high over 2 clocks (12Mhz).
VBUS	I/PD	13	Usb Cable Power supply pin
NC	I/PD	14	For testing
VDD	P	15	Power supply pin for logic circuits +3.3V DC
NC	O	16	For testing
NC	O	17	For testing
FORCEFS	I/PD	18	When the pin is set, it must be force to work on full speed mode.
NC	I/PD	19	For testing
VSS	P	20	Power supply +0V DC or Ground Power
TEST1	I/PU	21	Test Pin: This pin for test purpose only Pull up the pin or keep no connection for normal operation
TEST0	I/PU	22	Test Pin: This pin for test purpose only Pull up the pin or keep no connection for normal operation
VDD	P	23	Power supply pin for logic circuits +3.3V DC
NC	O	24	For testing
NC	O	25	For testing
NC	O	26	For testing
NC	O	27	For testing
NC	O	28	For testing
NC	O	29	For testing
NC	O	30	For testing
NC	O	31	For testing
VSS	P	32	Power supply +0V DC or Ground Power
VDD	P	33	Power supply pin for logic circuits +3.3V DC
EECS	O	34	EEPROM Chip Select Signal

EECK	O	35	EEPROM Clock connected to EEPROM clock pin
EEDI	O	36	EEPROM Data In :Signal connected EEPROM data input pin
EEDO	I/PD	37	EEPROM Data Out :Signal connected EEPROM data output pin
NC	I/PD	38	For testing
NC	I/PD	39	For testing
NC	I/PD	40	For testing
NC	I/PD	41	For testing
NC	I/PD	42	For testing
NC	I/PD	43	For testing
VDD	P	44	Power supply pin for logic circuits +3.3V DC
VSS	P	45	Power supply +0V DC or Ground Power
SD[15:0]	I/O/F	63-56 53-46	System Data Bus: Signal SD[15:0] constitute the bi-directional data bus.
VDD	P	54	Power supply pin for logic circuits +3.3V DC
VSS	P	55	Power supply +0V DC or Ground Power
VDD	P	64	Power supply pin for logic circuits +3.3V DC
VSS	P	65	Power supply +0V DC or Ground Power
CE1#	O	66	Card Enable: The CE1# enables even numbered address bytes
CE2# or BHE# or UDS#	O	67	Card Enable: The CE2# enables odd numbered address bytes for pcmcia mode. Bus High Enable: Bus High Enable is active low signal in some 16-bit application mode. Upper Data Strobe: The signal also name Upper Data Strobe (/UDS) for 68K application mode.
OE# or MEMR#	O	68	Output Enable or Memory Read: The signal is low active for Memory Read.
IORD#	O	69	I/O Read: The signal is low active for I/O Read.
IOWR#	O	70	I/O Write: The signal is low active for I/O Write.
WE# or MEMW#	O	71	Write Enable or Memory Write: The signal is low active for Memory Write.
IREQ or IREQ#	I/PU/F	72	Interrupt Request: IREQ or IREQ# user can define high or low active. When the IREQ or IREQ# is asserted to indicate the host system that the application device requires host software service.
VSS	P	73	Power supply +0V DC or Ground Power
VDD	P	74	Power supply pin for logic circuits +3.3V DC
SA[15:1] ,SA[0] or LDS#	O	75-82 85-92	System Address: Signal SA[15:0] IO address decoding for up to 64 Kbyte. SA[0] also means Lower Data Strobe (LDS#) active low signal in 68K application mode.
VSS	P	83	Power supply +0V DC or Ground Power
VDD	P	84	Power supply pin for logic circuits +3.3V DC
RESETP or RESETP#	O	93	RESET: RESETP or RESETP# user can define high or low active.
WAIT or WAIT# or RDY or DTACK#	I/PU/F	94	Wait: WAIT or WAIT # user can define high or low active. This signal is set active, it insert wait state states during Remote DMA transfer. Ready: as the above same Dtack#: as above same for Motorola CPU, the pin is active inform that application device data is accepted.
AEN or PSEN#	O	95	Address Enable: The signal is asserted when the address bus is available for DMA cycle. PSEN: The signal is active low for 8051program access.
VDD	P	96	Power supply pin for logic circuits +3.3V DC
VSS	P	97	Power supply +0V DC or Ground Power
REG# or CS#	O	98	Attribute Memory and I/O Space Select: When the REG# signal is asserted, access is limited to Attribute Memory and to the I/O space (only for PCMCIA mode).

			Chip Select:When the CS# is asserted, the application is selected.
STSCHG1	I/PD	99	Status Change: reply application device status to Host
STSCHG2	I/PD	100	Status Change: reply application device status to Host
STSCHG3	I/PD	101	Status Change: reply application device status to Host
STSCHG4	I/PD	102	Status Change: reply application device status to Host
IOIS16#	I/PU/F	103	I/O is 16 Bit Port: The IOIS16# is asserted and I/O port addressed is capable of 16-bit access.
VDD	P	104	Power supply pin for logic circuits +3.3V DC
LED	O	105	LED indicator: When link FS, drives logic high always. When link HS, the pin drives logic low and it will drives high/low a period when line has activity (data transfer).
NC	I/PD	106	For testing
GPIO[3:0]	I/O/PU	110-107	General Purpose Input / Output Pins.
VDD	P	111	Power supply pin for logic circuits +3.3V DC
ANA_XIQ	I	112	Sets the IQ mode This pin is used during testing. It must be set to low in IQ measurement mode. 0: IQ mode 1: Normal operation mode
VDD	P	113	Power supply pin for logic circuits +3.3V DC
CLKI	I/PD	114	For testing
TESTMODE	I/PD	115	For testing (TESTMODE) 0: Normal operation mode 1: External clock Synchronization mode
RESET	I	116	When assert, place AX84110 into reset mode immediately. Reset complete loads the EEPROM data.
VSS	P	117	Power supply +0V DC or Ground Power
VSS	P	118	Power supply +0V DC or Ground Power
VDD	P	119	Power supply pin for logic circuits +3.3V DC
VDD	P	120	Power supply pin for logic circuits +3.3V DC
PVDD	P	121	Power supply pin for PLL and oscillator circuits +3.3V DC
PVDD	P	122	Power supply pin for PLL and oscillator circuits +3.3V DC
PVSS	P	123	Power supply pin for PLL and oscillator circuits +0V DC or Ground Power
PVSS	P	124	Power supply pin for PLL and oscillator circuits +0V DC or Ground Power
VC	I	125	Monitor pin for two PLL charge pumps Connect to GND on PCB when actually using
PTEST	I	126	Charge pump monitor ON/OFF: Connect to GND on PCB when actually using
XIN12M	I	127	12M crystal oscillator input
XOUT12M	O	128	12M crystal oscillator output

Tab-1 PIN signals

3.0 EEPROM Memory Mapping

EEPROM OFFSET	HIGH BYTE	LOW BYTE
00H	RESERVED	WORD COUNT FOR PRELOAD
01H	*FLAG	
02H	HIGH-SPEED LENGTH OF DEVICE DESCRIPTOR(BYTE)	HIGH-SPEED EEPROM OF DEVICE DESCRIPTOR
03H	HIGH-SPEED LENGTH OF CONFIGURATION DESCRIPTOR(BYTE)	HIGH-SPEED EEPROM OF CONFIGURATION DESCRIPTOR
04H	RESERVED	RESERVED
05H	RESERVED	RESERVED
06H	RESERVED	RESERVED
07H	LANGUAGE ID HIGH BYTE	LANGUAGE ID LOW BYTE
08H	LENGTH OF STRING INDEX 1	EEPROM OFFSET OF STRING INDEX 1
09H	LENGTH OF STRING INDEX 2	EEPROM OFFSET OF STRING INDEX 2
0AH	LENGTH OF STRING INDEX 3	EEPROM OFFSET OF STRING INDEX 3
0BH	LENGTH OF STRING INDEX 4	EEPROM OFFSET OF STRING INDEX 4
0CH	LENGTH OF STRING INDEX 5	EEPROM OFFSET OF STRING INDEX 5
0DH	LENGTH OF STRING INDEX 6	EEPROM OFFSET OF STRING INDEX 6
0EH	LENGTH OF STRING INDEX 7	EEPROM OFFSET OF STRING INDEX 7
0FH	RESERVED	RESERVED
10H	RESERVED	RESERVED
11H	RESERVED	RESERVED
12H	RESERVED	RESERVED
13H	FULL-SPEED LENGTH OF DEVICE DESCRIPTOR(BYTE)	FULL-SPEED EEPROM OF DEVICE DESCRIPTOR
14H	FULL-SPEED LENGTH OF CONFIGURATION DESCRIPTOR(BYTE)	FULL-SPEED EEPROM OF CONFIGURATION DESCRIPTOR
15H-1FH	RESERVED	RESERVED

Tab-2 EEPROM Memory Mapping

Note:
*FLAG:

- Bit0 ➔ Self Power (for USB GetStatus) 1:self power ; 0: bus power
- Bit1 ➔ Reserved
- Bit2 ➔ Remote WakeUP support
- Bit3 ➔ 1
- Bit4-6 ➔ Reserved
- Bit7 ➔ Force Full speed enable ; 1:force full speed 0:not enable
- Bit8 ➔ Set Reset active value ; 1:Reset active high ; 0: Reset active low
- Bit9-F ➔ Reserved

4.0 USB Commands

There are three command groups for endpoint0 in AX84110

- The USB standard commands
- USB Communication Class commands
- USB vendor commands

4.1 USB standard commands

- The Language ID is 0x0904 for English
- PPLL means buffer length
- CC means configuration number
- I I means Interface number

SETUP COMMAND	DATA IN/OUT	DESCRIPTION
80 06 00 01 00 00 LL PP	Data PPLL bytes	Get Device Descriptor
80 06 00 02 00 00 LL PP	Data PPLL bytes	Get Configuration Descriptor
80 06 00 03 00 00 LL PP	Data 2 bytes	Get Supported Language ID
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Manufacture String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Product String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Serial Number String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Configuration String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Interface 0 String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Interface 1/0 String
80 06 00 03 09 04 LL PP	Data PPLL bytes	Get Interface 1/1 String
80 08 00 00 00 00 01 00	Data 1 bytes	Get Configuration
00 09 CC 00 00 00 00 00	No Data	Set Configuration
81 0A 00 00 I I 00 01 00	Data 1 byte	Get Interface
01 0B AS 00 00 00 00 00	No Data	Set Interface

Tab-3 USB Standard commands

4.2 USB Vendor commands

SETUP COMMAND	DATA IN/OUT	DESCRIPTOR
40 80 0X 00 RR RR 00 00	No Data	Set Interface Register (1)
40 81 0X 00 RR RR 00 00	No Data	Set Interface Register (2)
40 82 0X 00 RR RR 00 00	No Data	Set Interface Register (3)
40 83 0X 00 RR RR 00 00	No Data	Set Interface Register (4)
40 84 IO I I 00 00 00 00	No Data	Set IO Data Port for OUT/IN
40 85 GP 00 00 00 00 00	No Data	Set Write GPIOs(***)
40 86 BC BC 00 00 00 00	No Data	Set Byte Count
40 87 TC 00 00 00 00 00	No Data	Set Data Access Timing Cycle
40 89 0Z 00 AA AA 00 00	No Data	Set IO or Memory address Function mode
C0 8A 00 00 00 00 01 00	Data 1 Bytes	Set Read GPIOs(***)
C0 8B DR 00 00 00 02 00	Data 2 Bytes	Read SROM
40 8C DR 00 MM SS 00 00	No Data	Write SROM
40 8D 00 00 00 00 00 00	No Data	Write SROM Enable
40 8E 00 00 00 00 00 00	No Data	Write SROM Disable
40 8F YY 00 00 00 00 00	No Data	Set Other Vaule

Tab-4 USB Vendor commands

X:

- Bit0 ➔ 1: Service Vaild ; 0: No Service
- Bit1 ➔ 1: Byte Valid ; 0: Word Valid
- Bit2 ➔ 1: Memory Area ; 0: I/O Area
- Bit3 ➔ Reserved

Y:

- Bit0 ➔ 1: Set EEPROM reload trigger
- Bit1 ➔ 1: Wait high active; 0: low active
- Bit2 ➔ 1: Ireq high active; 0: low active
- Bit3 ➔ 1: Set Reset (default Value com from EEPROM Autoload)
- Bit7 Bit6 Bit5 Bit4

x	0	0	0	➔	PCMCIA BUS mode
x	1	0	0	➔	ISA BUS mode
T	0	0	1	➔	80186 CPU mode
T	0	1	0	➔	68000 CPU mode
T	0	1	1	➔	8051CPU mode

T: 1: byte mode transmit/receive (only for CPU mode) x: don't care
0: word mode transmit/receive (only for CPU mode)

Z:

Bit3	Bit2	Bit1	Bit0	
x	F	0	0	for I/O device
x	M	0	1	Attribute memory
x	M	1	0	Common memory

F: 1 support fixed port 0: not support x: don't care
When work on Memory (only for PCMCIA or ISA Bus mode), user can define byte /word mode transmitting or receiving via set this bit. M:1 byte mode 0:word mode.
In CPU mode (as x86 or 68K or 51) must be set on xx01 (Attribute memory) as work on Memory mode and this M will be don't care because set T.

RR Int register offset address (user can set the address and it is absolute address)
I I Set IO Data Port In address (user can set I/O device data port by receiving)
IO Set IO Data Port Out address (user can set I/O device data port by transmitting)
BC Set Byte Count (user can set byte count of byte number for receiving)
TC Set Access Timing Cycle (min 3) (user can set read/write signal active timing of period)
AA Set IO Base or Memory (absolute address)

*** Read / Write GPIO

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Read	GPI3	X	GPI2	X	GPI1	X	GPI0	X
Write	GPO3	GPO3EN	GPO2	GPO2EN	GPO1	GPO1EN	GPO0	GPO0EN

4.3 USB Vendor and Normal commands format

SETUP COMMAND	DATA IN/OUT	DESCRIPTION
Application device internal register Normal Command & USB Vendor Command Register		
C0 AD1 DR 00 00 00 02 00	Data 2 Bytes	Usb Vendor Command Read ref Tab-4 Read SROM
40 AD1 DR 00 MM SS 00 00	No Data	Usb Vendor Command Write ref Tab-4 Write SROM
C0 AD1 AD2 AD3 AD4 AD5 wlength	No Data	Normal Command Read
40 AD1 DL1 DH1 00 00 wlength	Variable	Normal Command Write

Tab-5 vendor and normal commands format

Notes: wlength must be even numbers

- **40 : Write Command** (4,8,C,10…..) for Write Command
- **C0: Read Command** (0,2,4,6,8…..) for Read Command

AD1 Bit7 = 1 ➔ Vendor Command

0 ➔ Normal Command

AD1 or AD2 or AD3 or AD4 or AD5 or ADL

Bit6 = 1 ➔ D(data) only low byte valid

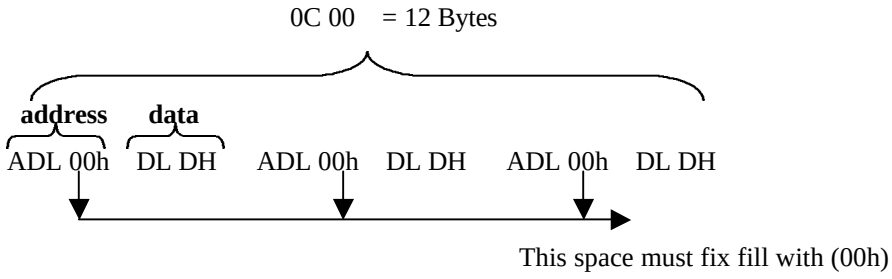
0 ➔ D(data) low byte & high byte valid

For Normal Command Write:

(low byte) (high byte)

Ex: wlength = 0C 00

Variable : Data OUT 12 Bytes



L: Low byte H: High Byte

Note: If wlength = 00 00 , it indicate no need other Data OUT .

For Normal Command Read:

If user want to read 3 set application device internal register, you can set this command as below:

C0 AD1AD2 AD3 AD4 AD5 wlength

➔ C0 02 04 06 x x 06 00 x : don't care

3x2 = 6 Bytes

5.0 USB Configuration Structure

5.1 USB Configuration

The AX84110 supports 1 Configuration only.

5.2 USB Interface

The AX84110 supports 2 interfaces, the interface 0 is Data Interface and interface 1 for Communication Interface.

5.3 USB Endpoints

The AX84110 supports 4 endpoints.

Endpoint0 ➔ Control endpoint, it is for configuration device.

Endpoint1 ➔ Interrupt endpoint , it is for reporting application device status.

Endpoint2 ➔ Bulk Out endpoint, it is for Transmitting application device Packet.
Buffer is 2k size and have 4 page. Each page have 512byte space.

Endpoint3 ➔ Bulk IN endpoint, it is for Receiving application device Packet.
Buffer is 1k size and have 2 page. Each page have 512 byte space.

6.0 Basic Operation

6.1 Process Flow

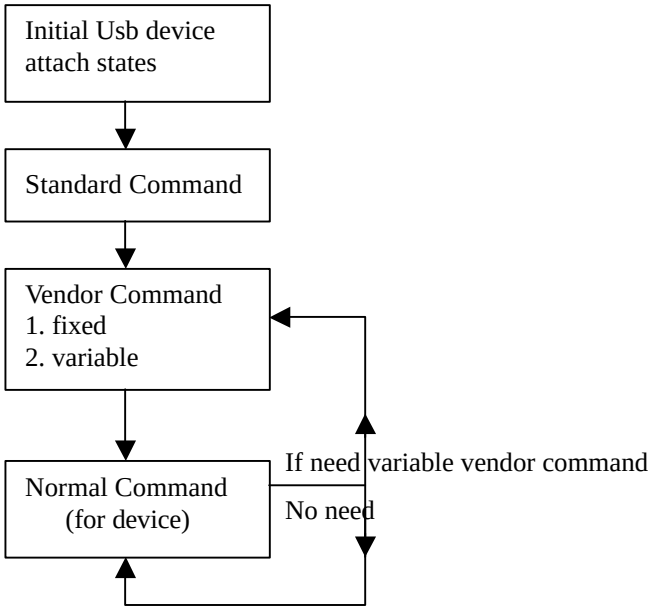


Fig-7 Basic Process Flow

1. Standard Command (Get USB Standard Command as Get Device Descriptor)

   Ex: (ref Tab-3)

   80 06 00 01 00 00 LL PP ➔ Get Device Descriptor

2. Vendor Command (Set AX84110 command)

- fixed of vendor command
- variable of vendor command

fixed of vendor command (for initial setup)

Ex: (ref Tab-4)

1.40 (80~83) 0X 00 RR RR 00 00 ➔ Set Interrupt Register4~1 (optional by application)

AX84110 provide 4 set of register about application device for interrupt status reply.

2.40 84 IO I I 00 00 00 00 ➔ Set I/O Data Port (IO:OUT ;I I :IN)

3.40 87 TC 00 00 00 00 00 ➔ Set Data Access Timing Cycle

Data Access Timing Cycle = Control signal read/write active of period

4. 40 8F YY 00 00 00 00 00 ➔ Set Other Value

AX84110 provide variable parameter for variable application device.

It can define some value high or low active and work on selectable mode.

variable of vendor command (depend on application device)

Ex: (ref Tab-4)

1.40 85 GP 00 00 00 00 00 ➔ Set Write GPIOs

2. C0 8A 00 00 00 00 01 00 ➔Set Read GPIOs

AX84110 provide optional 4 set General Purpose Pin (IN/OUT) for application device.

3. 40 87 TC 00 00 00 00 00 ➔ Set Byte Count

If want to receive data from application device, it can set this command. When the command is set , AX84110 will move byte count number from application device.

4. 40 89 0Z 00 AA AA 00 00 ➔ Set I/O or Memory address function mode

AX84110 provide selectable work mode on I/O or Memory (Attribute & Common only for Pcmcia Bus mode). You can set address by this command .

5. C0 8B DR 00 00 00 02 00 ➔ Set Read SROM

6. 40 8C DR 00 MM SS 00 00 ➔ Set Write SROM

7. 40 8D 00 00 00 00 00 00 ➔ Set SROM Enable

8. 40 8E 00 00 00 00 00 00 ➔ Set SROM Disable

above 4 command about SROM for needed. (ref Tab-2)

AX84110 provide additional space to store some special value. You can write or read DR (EEPROM offset of address) by used the command.

3. Normal Command (base on application device)

(ref Tab-5)

6.2 Packet Format Character

The AX84110 supports 4 endpoints.

- Control endpoint
- Int endpoint
- Bulk Out endpoint
- Bulk In endpoint

6.2.1 Control endpoint

It is for configuration device.

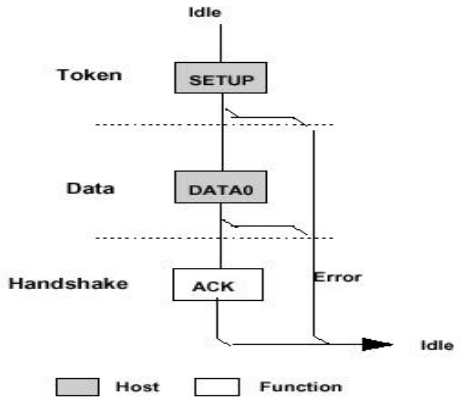


Fig-7 Control SEUP Transaction

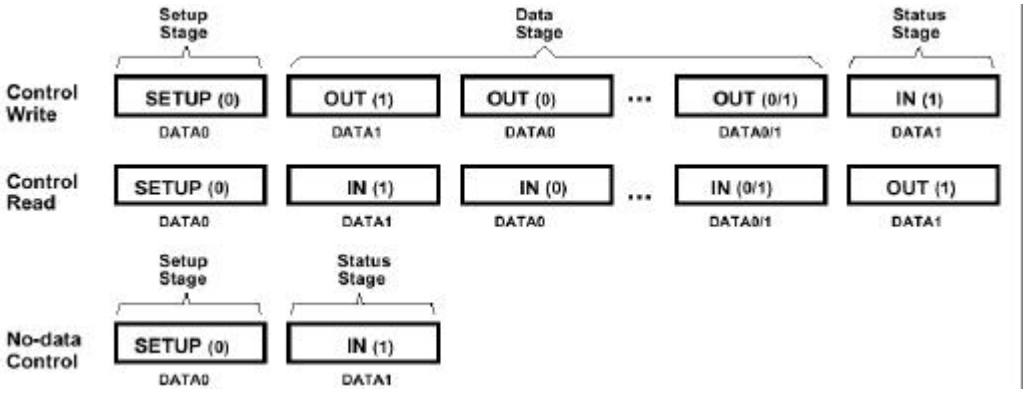
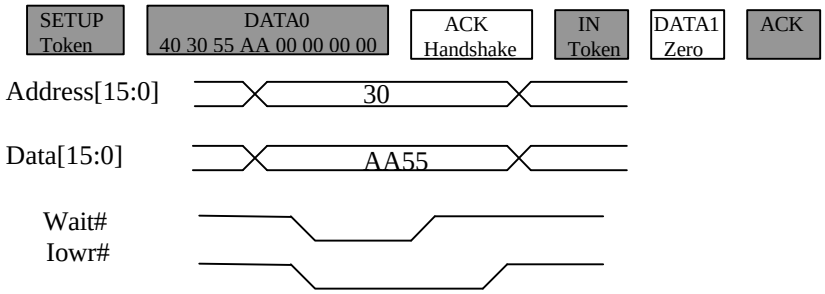


Fig-8 Control Read and Write Sequences

Example1: I/O single write (ref Tab-5)

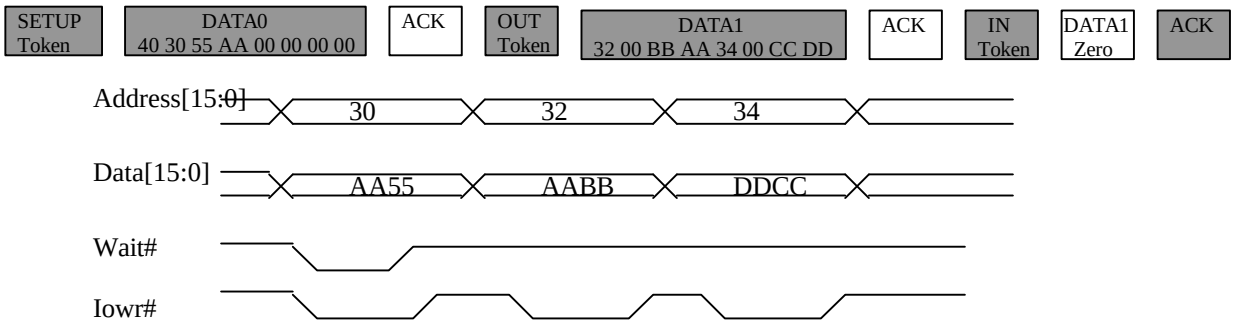
1. Host send SETUP token

2. Host send DATA frame (40 30 55 AA 00 00 00 00)
3.When the command decode by AX84110 and send appropriate packet to application device. If finish, AX84110 will reply a ACK to Host.
4. Host Send IN token.
5. AX84110 will send (zero length status stage) to Host.
6. Host reply a ACK. And No-data Control write is end.



Example2: I/O continue write (ref Tab-5)

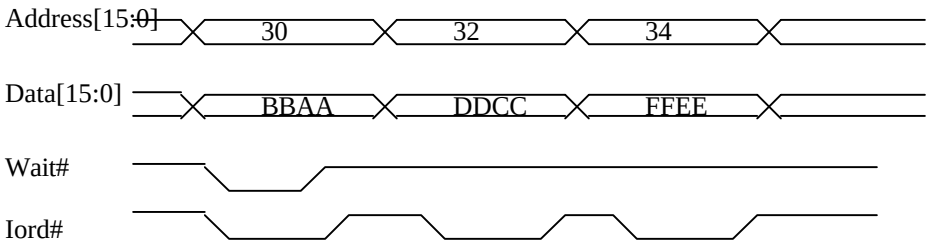
1. Host send SETUP token
2. Host send DATA frame (40 30 55 AA 00 00 08 00)
3.When the command decode by AX84110 and send appropriate packet to application device. If finish, AX84110 will reply a ACK to Host.
4. Host send OUT token
5. Host send DATA frame (32 00 BB AA 34 00 DD CC)
6. repeat 3 .
7. Host Send IN token
8. AX84110 will send (zero length status stage) to Host.
9. Host reply a ACK. And Control write is end.



Example2: I/O continue read (ref Tab-5)

1. Host send SETUP token
2. Host send DATA frame (C0 30 32 34 00 00 06 00)
3.When the command decode by AX84110 and send appropriate packet to application device. If finish, AX84110 will reply a ACK to Host.
4. Host send IN token
5. AX84110 receive DATA frame (AA BB CC DD EE FF) from application device and will send to Host.
6. Host receive Data frame and reply a ACK.
7. Host send OUT token
8. Host send zero length Data Frame.(status stage)
9. AX84110 reply a ACK. And Control read is end.

SETUP Token	DATA0 C0 30 32 34 00 00 06 00	ACK	IN	DATA1 AA BB CC DD EE FF	ACK	OUT	DATA1 Zero	ACK



6.2.2 INT endpoint

It is for reporting application device status

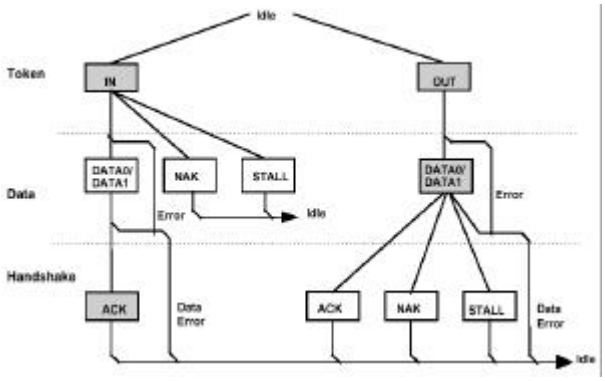
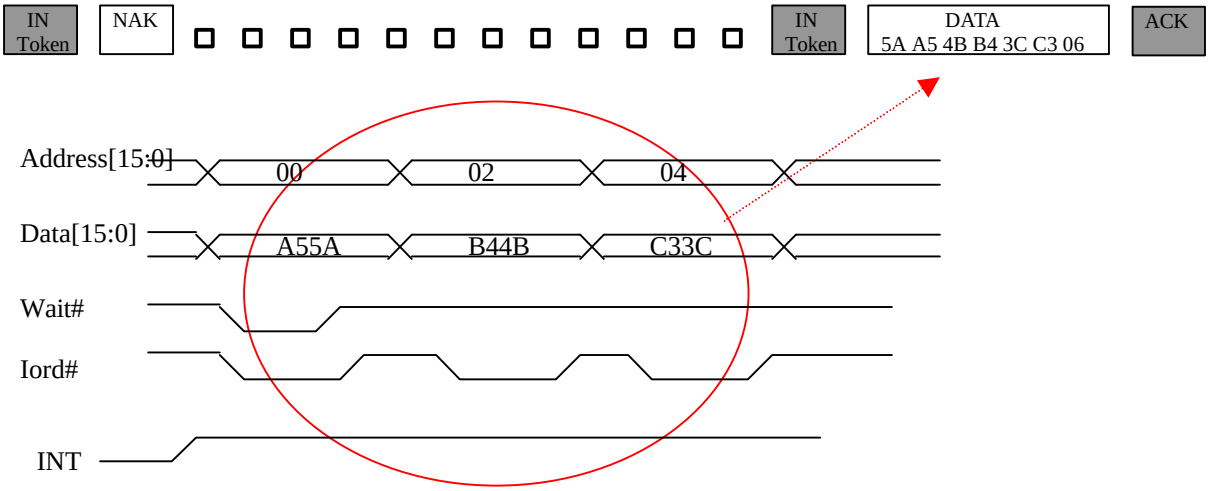


Fig-9 Interrupt Transaction Format

Example1: interrupt and status report

1. Host send Vendor Command (ref Tab-4) Set Interrupt Register4~1 (optional by application)
2. Host will send INT IN Token in a fixed time.
3. If no interrupt occur, AX84110 will reply a NAK , otherwise AX84110 will receive data base on Set Interrupt Register4~1 & status4~1(reply from application device) and wait Host send next IN token. If application reply status (0110 = 6)
4. When IN token occur and have int occur , AX84110 will send data frame to Host.
5. Host reply a ACK. And Int transaction is end.



6.2.3 Bulk Out endpoint

It is for Transmitting application device Packet.

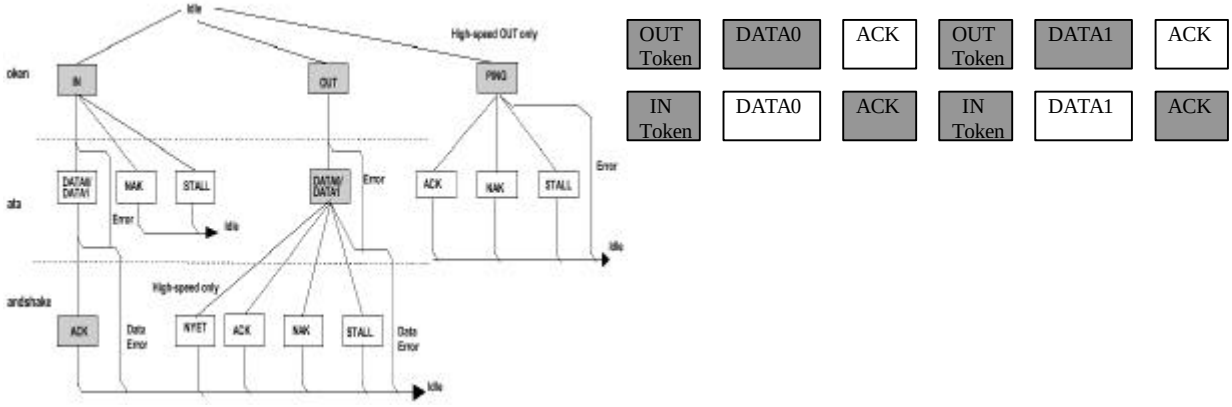
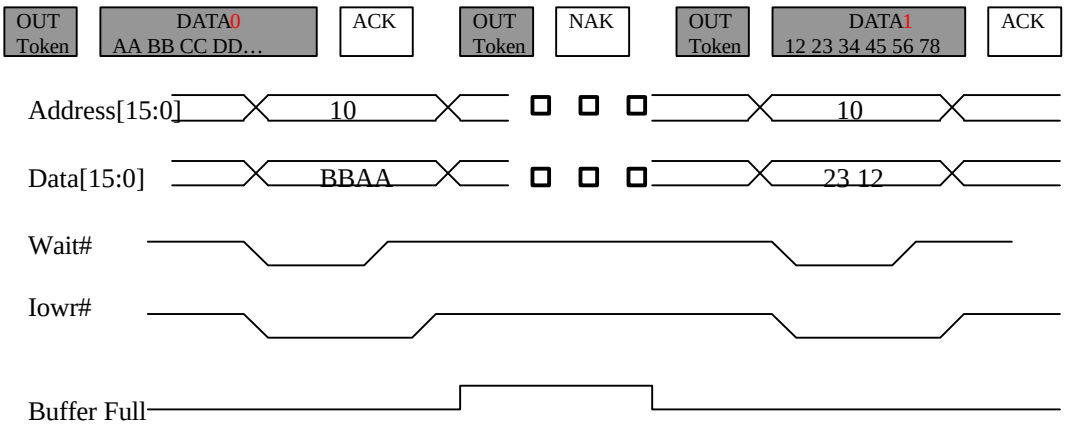


Fig-10 Bulk Transaction Format

Example1: I/O Bulk Out Transmit Packet

1. Host send Vendor Command (ref Tab-4) Set IO Data Port address.
2. Host send OUT token.(If AX84110 buffer is full, it will reply a NAK to Host)
3. Host send Data Frame(AA BB CC DD …….. PID (Data 0)
4. When the command decode by AX84110 and send appropriate packet (AA BB CC DD ….) to application device.
5. AX84110 reply a ACK. And Bulk Out transaction is end.
6. Host send OUT token. AX84110 buffer is full, it reply a NAK to Host.
7. Host send OUT token. (When buffer is not full)
8. Host send Data Frame (12 23 34 56 ……..). PID (Data 1)
9. When the command decode by AX84110 and send appropriate packet (12 23 34 56 ….) to application device.
10. AX84110 reply a ACK. And Bulk Out transaction is end.

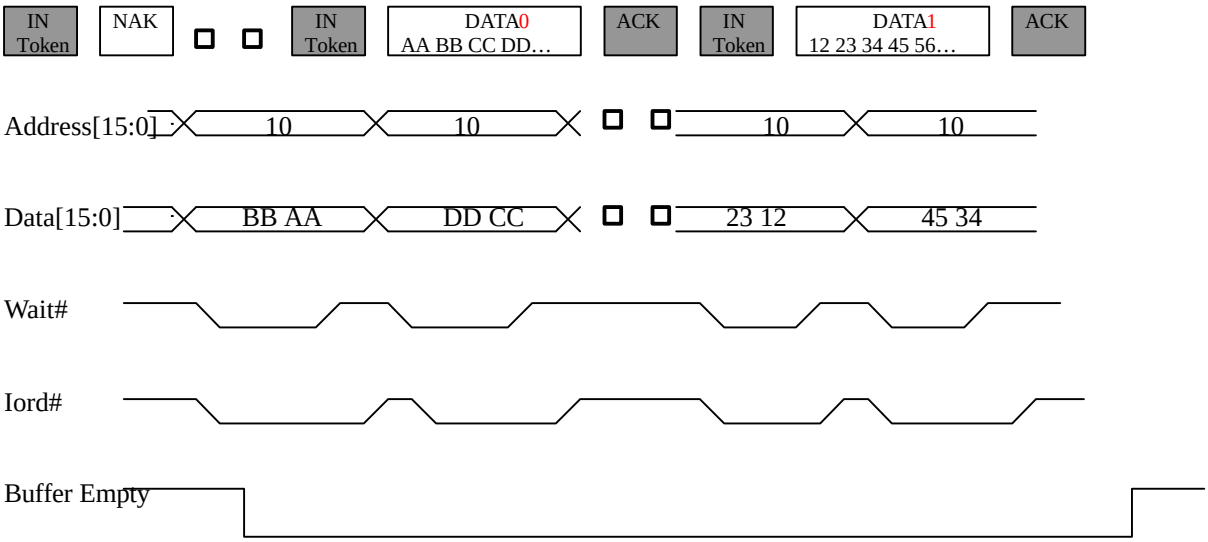


6.2.4 Bulk In endpoint

It is for Receiving application device Packet.

Example1: I/O Bulk IN Transmit Packet

1. Host send Vendor Command (ref Tab-4) Set IO Data Port address.
2. Host send Vendor Command (ref Tab-4) Set Byte Count
3. When the set byte count is set, AX84110 will move data from application device.
4. Host send IN token, but buffer is not ready (buffer is empty) and AX84110 will reply NAK.
5. When buffer is ready and IN token coming, AX84110 will send data to Host. It base on full/high speed (max 64 / 512 byte). If the set byte count more than max packet, it will divide a packet into 64 byte or 512 byte. PID (Data 0)
6. Host reply a ACK. And Bulk In transaction is end.
7. Host send IN token, but buffer is not ready (buffer is empty) and AX84110 will reply NAK.
8. When buffer is ready and IN token coming, AX84110 will send data to Host. It base on full/high speed (max 64 / 512 byte). If the set byte count more than max packet, it will divide a packet into 64 byte or 512 byte. PID (Data 1)
9. Host reply a ACK. And Bulk In transaction is end.



7.0 CPU & Device Access Functions

7.1 PCMCIA bus type access functions

7.1.1 Attribute Memory access function

Attribute Memory Read function

Function Mode	REG#	CE2#	CE1#	SA0	OE#	WE#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L	H	L	L	L	H	High-Z	Even-Byte
Word Access (16 bits)	L	L	L	X	L	H	Not Valid	Even-Byte

Attribute Memory Write function

Function Mode	REG#	CE2#	CE1#	SA0	OE#	WE#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	X	X
Byte Access (8 bits)	L	H	L	L	H	L	X	Even-Byte
Word Access (16 bits)	L	L	L	X	H	L	X	Even-Byte

7.1.2 Common Memory access function

Common Memory Read function

Function Mode	REG#	CE2#	CE1#	SA0	OE#	WE#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	H	H	L	L	L	H	High-Z	Even-Byte
	H	H	L	H	L	H	High-Z	Odd-Byte
Word Access (16 bits)	H	L	L	X	L	H	Odd-Byte	Even-Byte

Common Memory Write function

Function Mode	REG#	CE2#	CE1#	SA0	OE#	WE#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	X	X
Byte Access (8 bits)	H	H	L	L	H	L	X	Even-Byte
	H	H	L	H	H	L	X	Odd-Byte
Word Access (16 bits)	H	L	L	X	H	L	Odd-Byte	Even-Byte

7.1.3 I/O access function

I/O Read function

Function Mode	REG#	CE2#	CE1#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L	H	L	L	L	H	High-Z	Even-Byte
	L	H	L	H	L	H	High-Z	Odd-Byte
Word Access (16 bits)	L	L	L	L	L	H	Odd-Byte	Even-Byte

I/O Write function

Function Mode	REG#	CE2#	CE1#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	H	X	X	X	X	X
Byte Access (8 bits)	L	H	L	L	H	L	X	Even-Byte
	L	H	L	H	H	L	X	Odd-Byte
Word Access (16 bits)	L	L	L	L	H	L	Odd-Byte	Even-Byte

7.2 ISA bus type access functions

7.2.1 Memory access function

ISA bus Memory Read function

Function Mode	CS#	BHE#	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L L	H H	L H	L L	H H	Not Valid Not Valid	Even-Byte Odd-Byte
Word Access (16 bits)	L	L	L	L	H	Odd-Byte	Even-Byte

ISA bus Memory Write function

Function Mode	CS#	BHE#	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L L	H H	L H	H H	L L	X X	Even-Byte Odd-Byte
Word Access (16 bits)	L	L	L	H	L	Odd-Byte	Even-Byte

7.2.2 I/O access function

ISA bus I/O Read function

Function Mode	CS#	BHE#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L L	H H	L H	L L	H H	Not Valid Not Valid	Even-Byte Odd-Byte
Word Access (16 bits)	L	L	L	L	H	Odd-Byte	Even-Byte

ISA bus I/O Write function

Function Mode	CS#	BHE#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L L	H H	L H	H H	L L	X X	Even-Byte Odd-Byte
Word Access (16 bits)	L	L	L	H	L	Odd-Byte	Even-Byte

7.3 80186 CPU bus type access functions

7.3.1 Memory access function

80186 CPU bus Memory Read function

Function Mode	CS#	BHE#	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L L	H L	L H	L L	H H	Not Valid Odd-Byte	Even-Byte Not Valid
Word Access (16 bits)	L	L	L	L	H	Odd-Byte	Even-Byte

80186 CPU bus Memory Write function

Function Mode	CS#	BHE#	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L L	H L	L H	H H	L L	X Odd-Byte	Even-Byte X
Word Access (16 bits)	L	L	L	H	L	Odd-Byte	Even-Byte

7.3.2 I/O access function

80186 CPU bus I/O Read function

Function Mode	CS#	BHE#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L L	H L	L H	L L	H H	Not Valid Odd-Byte	Even-Byte Not Valid
Word Access (16 bits)	L	L	L	L	H	Odd-Byte	Even-Byte

80186 CPU bus I/O Write function

Function Mode	CS#	BHE#	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L L	H L	L H	H H	L L	X Odd-Byte	Even-Byte Odd-Byte
Word Access (16 bits)	L	L	L	H	L	Odd-Byte	Even-Byte

7.4 68000 CPU bus type access functions

7.4.1 *68000 CPU Bus access function*

68000 CPU bus Read function

Function Mode	CS#	UDS#	LDS#	R/W#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	High-Z	High-Z
Byte Access (8 bits)	L L	H L	L H	H H	Not Valid Even-Byte	Odd-Byte Not Valid
Word Access (16 bits)	L	L	L	H	Even-Byte	Odd-Byte

68000 CPU bus Write function

Function Mode	CS#	UDS#	LDS#	R/W#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X
Byte Access (8 bits)	L L	H L	L H	L L	X Even-Byte	Odd-Byte X
Word Access (16 bits)	L	L	L	L	Even-Byte	Odd-Byte

7.5 8051 CPU bus type access functions

7.5.1 Memory access function

8051 CPU bus Memory Read function

Function Mode	CS#	PSEN	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L	H	L	L	H	Not Valid	Even-Byte
	L	H	H	L	H	Not Valid	Odd-Byte

8051 CPU bus Memory Write function

Function Mode	CS#	PSEN	SA0	MEMR#	MEMW #	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L	H	L	H	L	X	Even-Byte
	L	H	H	H	L	X	Odd-Byte

7.3.2 I/O access function

8051 CPU bus I/O Read function

Function Mode	CS#	PSEN	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	High-Z	High-Z
Byte Access (8 bits)	L	H	L	L	H	Not Valid	Even-Byte
	L	H	H	L	H	Not Valid	Odd-Byte

8051 CPU bus I/O Write function

Function Mode	CS#	PSEN	SA0	IORD#	IOWR#	SD[15:8]	SD[7:0]
Standby Mode	X	H	X	X	X	X	X
Byte Access (8 bits)	L	H	L	H	L	X	Even-Byte
	L	H	H	H	L	X	Odd-Byte

8.0 Electrical Specification and Timing

8.1 Absolute Maximum Ratings

Description	SYM	Min	Max	Units
Operating Temperature	Ta	0	+85	¢ J
Storage Temperature	Ts	-65	+150	¢ J
Supply Voltage	Vdd	-0.3	+3.6	V
Input Voltage	Vin	-0.3	Vdd+0.3	V
Output Voltage	Vout	-0.3	Vdd+0.3	V
Lead Temperature (soldering 10 seconds maximum)	Tl	55	+240	¢ J

Note: Stress above those listed under Absolute Maximum Rating may cause permanent damage to the device. Exposure to Absolute Maximum Ratings condition for extended period, adversely affect device life and reliability.

8.2 General Operation Conditions

Description	SYM	Min	Tpy	Max	Units
Operating Temperature	Ta	0	25	+70	¢ J
Supply Voltage	Vdd	+3.0	+3.30	+3.6	V

8.3 D.C. Characteristics

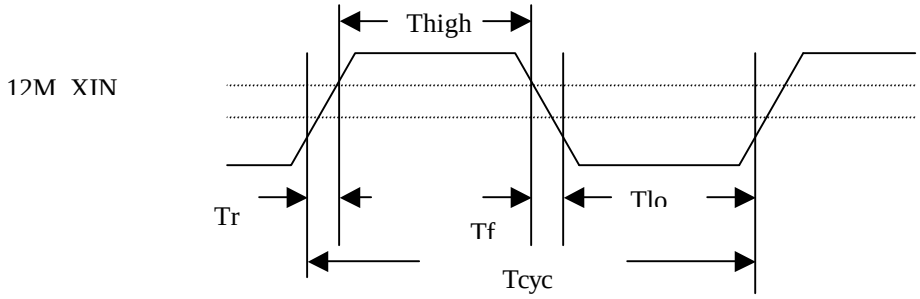
(Vdd = 3.0 to 3.6v , Vss = 0V , Ta = 0 ¢ J to 70 ¢ J

Description	SYM	Min	Tpy	Max	Units
Low Input Voltage	Vil	-		0.7*Vdd	V
High Input Voltage	Vih	0.7*Vdd		-	V
Low Output Voltage	Vol	-		0.4	V
High Output Voltage	Voh	2.4		-	V
Input Leakage Current	Iil	-1		+1	uA
Output Leakage Current	Iol	-10		+10	uA
Input Pull-up / down resistance	Ri		75		K ohm

Description	SYM	Min	Tpy	Max	Units
Power Consumption (3.3V)	SPt3v		150		mA

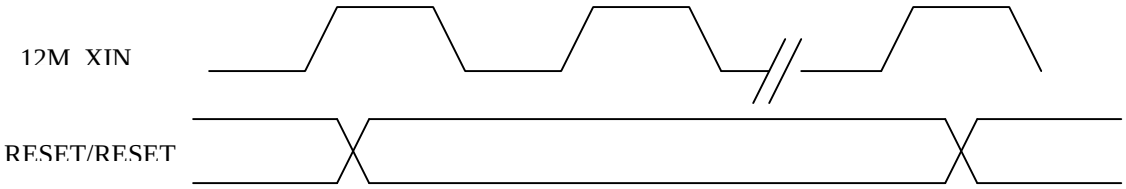
8.4 A.C. Timing Characteristics

8.4.1 12M_XIN



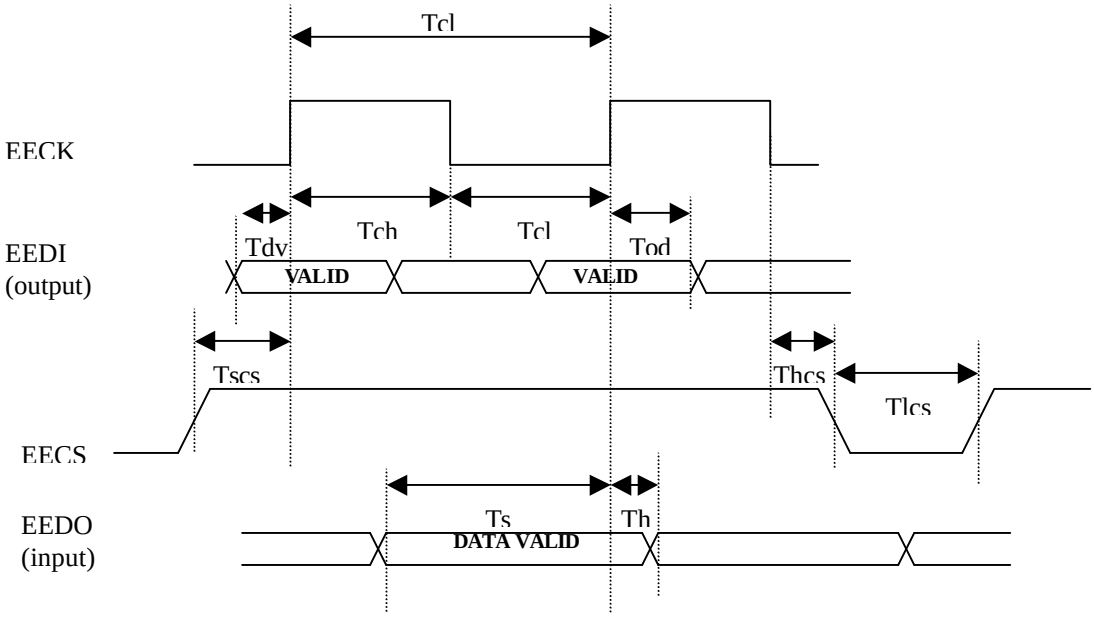
Symbol	Description	Min	Typ	Max	Units
Tcyc	CYCLE TIME		83.33		ns
Thigh	CLK HIGH TIME	34.71	41.66	49.99	ns
Tlow	CLK LOW TIME	34.71	41.66	49.99	ns
Tr / Tf	CLK SKEW RATE	1	-	4	ns

8.4.2 Reset Timing



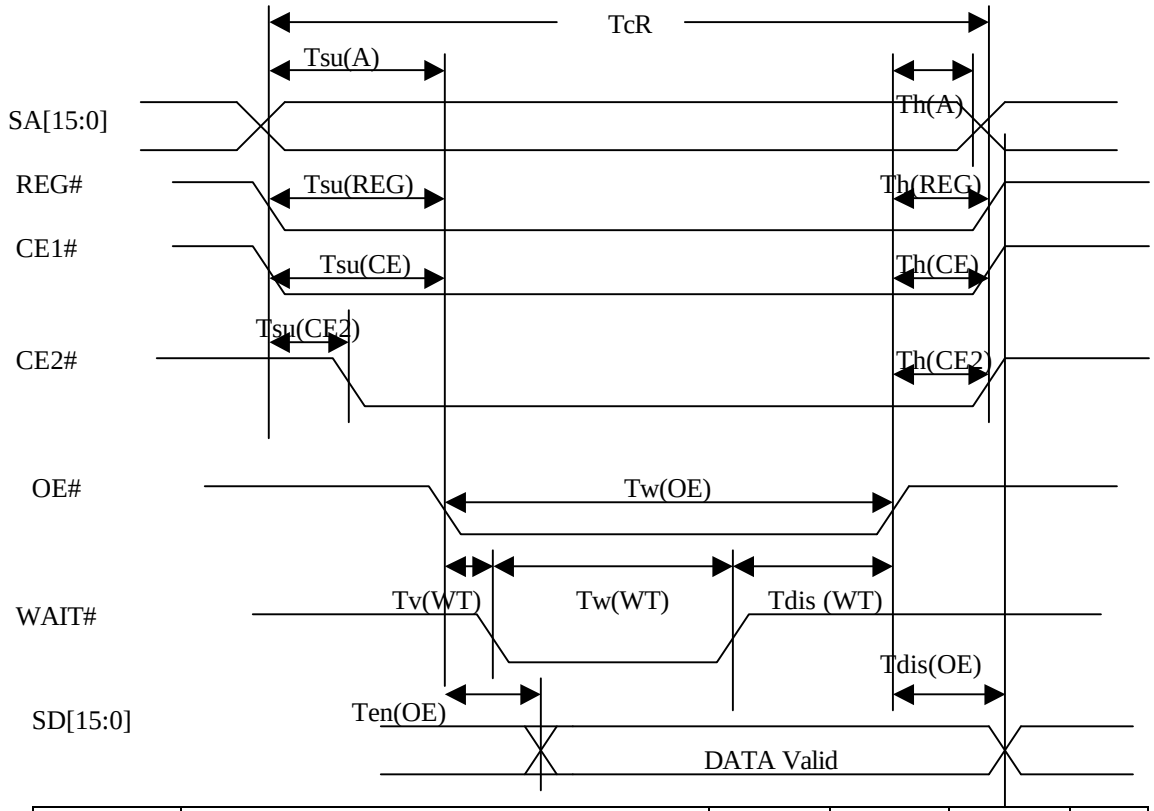
Symbol	Description	Min	Typ	Max	Units
Trst	Reset pulse width (6ms ~ 10 ms)	100	-	-	12M_XIN

8.4.3 Serial EEprom Timing



Symbol	Description	Min	Typ	Max	Units
Tclk	EECK Clock Cycle Time		187.5		KHZ
Tch	EECK Clock High Time		2666		ns
Tcl	EECK Clock Low Time		2666		ns
Tdv	EEDI Data Valid Output to EECK High Time	2666			ns
Tod	EECK High to EEDI Data Output Delay Time	2666			ns
Tscs	EECS Valid to EECK High Time	2666			ns
Thcs	EECK Low to EECS Invalid Time	0			ns
Tlcs	Minimum EECS Low Time	23904			ns
Ts	Data Input Setup Time	10			ns
Th	Data Input Hold Time	100			ns

8.4.4 Pcmcia Attribute Memory Read Timing



Symbol	Description	Min	Typ	Max	Units
TcR	READ CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Tsu(REG)	REGISTER SELECT SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Th(REG)	REGISTER SELECT HOLD TIME	30	-	-	ns
Tv(WT)	WAIT VALID FROM OE#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	OE# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(OE)	DATA ENABLE TIME FROM OE#	0	-	-	ns
Tdis(OE)	DATA DISABLE TIME FROM OE#	30	-	-	ns
Tw(OE)	OE# PULSE WIDTH	**	-	-	ns

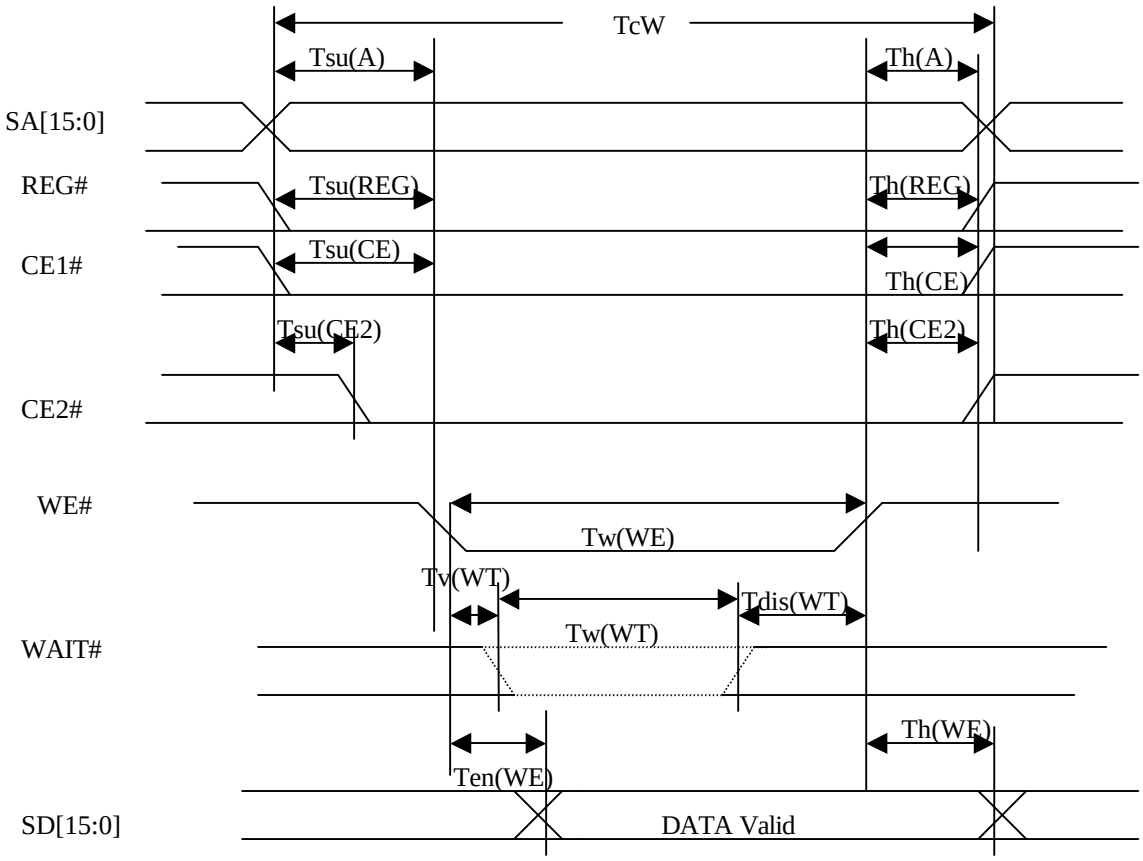
* TcR = Tsu(A) + Tw(OE) + Th(A)

** Tw(OE) = Base on Tab-4 Set Data Access Timing Cycle (TC) ➔ No wait condition (Tw(WT) = 0)
= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition (Tw(WT) != 0)

1. No wait condition
If you set TC = 06
Tw(OE) = TC * 30 = 6*30 = 180ns TcR = 270 ns

2. wait condition
If you set TC = 06
Tv(WT) = 20ns, Tw(WT) = 40ns, Tdis (WT) = 30ns
Tw(OE) = 90ns TcR = 180 ns

8.4.5 Pcmcia Attribute Memory Write Timing



Symbol	Description	Min	Typ	Max	Units
TcW	WRITE CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Tsu(REG)	REGISTER SELECT SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Th(REG)	REGISTER SELECT HOLD TIME	30	-	-	ns
Tv(WT)	WAIT VALID FROM WE#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	WE# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(WE)	DATA ENABLE TIME FROM WE#	30	-	-	ns
Th(WE)	DATA DISABLE TIME FROM WE#	30	-	-	ns
Tw(WE)	WE# PULSE WIDTH	**	-	-	ns

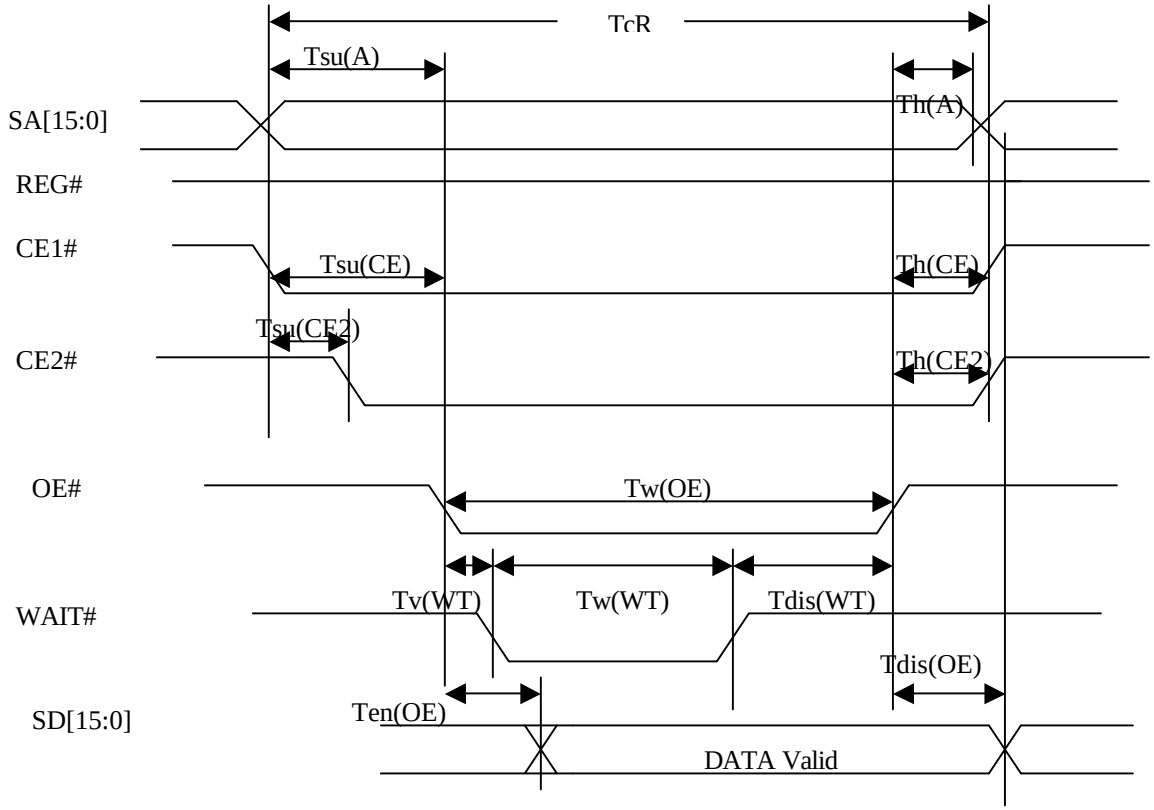
* TcW = Tsu(A) + Tw(WE) + Th(A)

** Tw(WE) = Base on Tab-4 Set Data Access Timing Cycle (TC) ➔ No wait condition (Tw(WT) = 0)
= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition (Tw(WT) != 0)

1. No wait condition
If you set TC = 06
Tw(WE) = TC * 30 = 6*30 = 180ns TcW = 270 ns

2. wait condition
If you set TC = 06
Tv(WT) = 20ns, Tw(WT) = 40ns, Tdis (WT) = 30ns
Tw(WE) = 90ns TcW = 180 ns

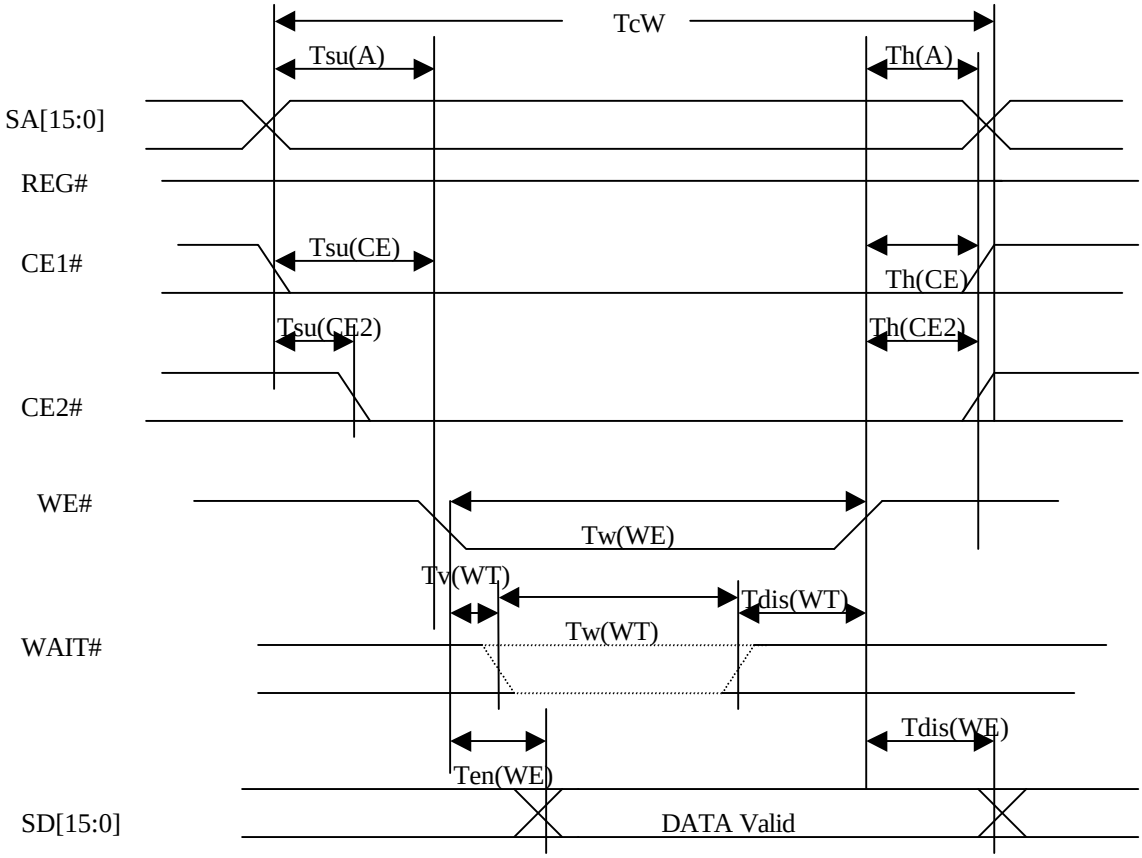
8.4.6 Pcmcia Common Memory Read Timing



Symbol	Description	Min	Typ	Max	Units
TcR	READ CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Tv(WT)	WAIT VALID FROM OE#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	OE# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(OE)	DATA ENABLE TIME FROM OE#	0	-	-	ns
Tdis(OE)	DATA DISABLE TIME FROM OE#	30	-	-	ns
Tw(OE)	OE# PULSE WIDTH	**	-	-	ns

* TcR = Tsu(A) + Tw(OE) + Th(A)
** Tw(OE) = Base on Tab-4 Set Data Access Timing Cycle (TC) ➔ No wait condition (Tw(WT) = 0)
= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition (Tw(WT) != 0)
1. No wait condition
If you set TC = 06
Tw(OE) = TC * 30 = 6*30 = 180ns TcR = 270 ns
2. wait condition
If you set TC = 06
Tv(WT) = 20ns, Tw(WT) = 40ns, Tdis (WT) = 30ns
Tw(OE) = 90ns TcR = 180 ns

8.4.7 Pcmcia Common Memory Write Timing

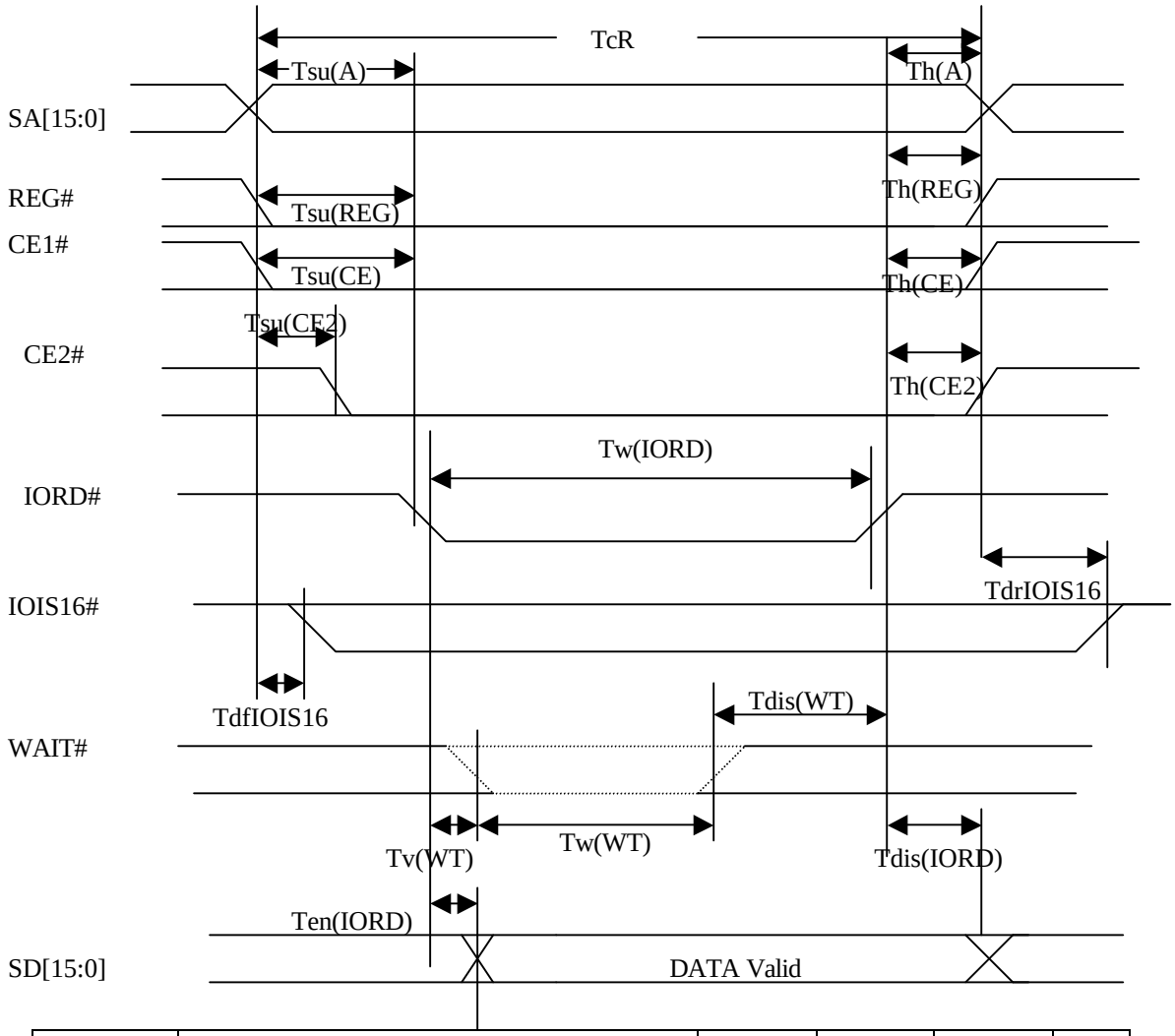


Symbol	Description	Min	Typ	Max	Units
TcW	WRITE CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Tv(WT)	WAIT VALID FROM WE#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	WE# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(WE)	DATA ENABLE TIME FROM WE#	30	-	-	ns
Th(WE)	DATA DISABLE TIME FROM WE#	30	-	-	ns
Tw(WE)	WE# PULSE WIDTH	**	-	-	ns

* TcW = Tsu(A) + Tw(WE) + Th(A)
** Tw(WE) = Base on Tab-4 Set Data Access Timing Cycle (TC) ➔ No wait condition (Tw(WT) = 0)
= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition (Tw(WT) != 0)
1. No wait condition
If you set TC = 06
Tw(WE) = TC * 30 = 6*30 = 180ns TcW = 270 ns
2. wait condition
If you set TC = 06
Tv(WT) = 20ns, Tw(WT) = 40ns, Tdis (WT) = 30ns

Tw(WE) = 90ns TcW = 180 ns

8.4.8 Pcmcia I/O Read Timing



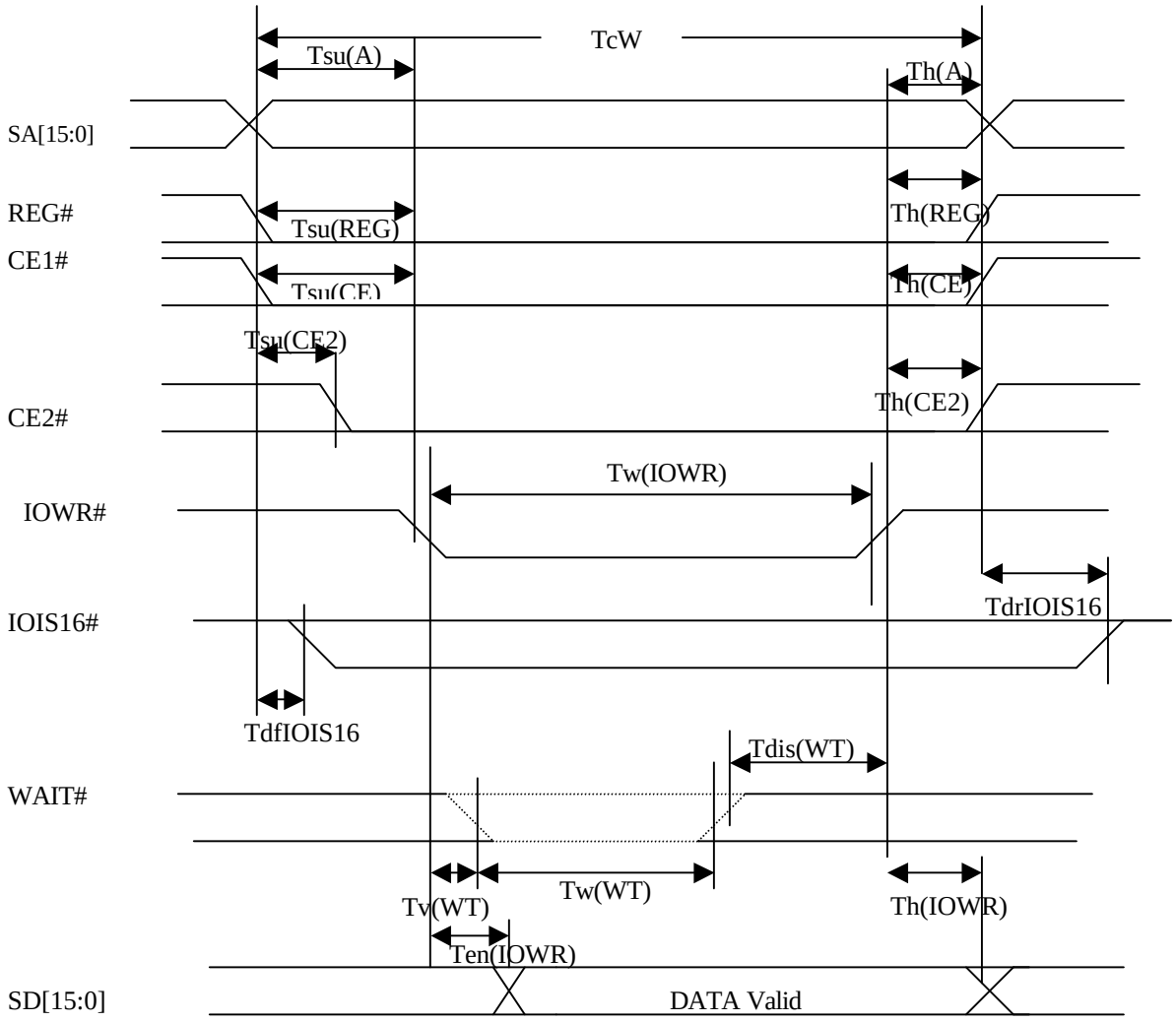
Symbol	Description	Min	Typ	Max	Units
TcR	READ CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Tsu(REG)	REGISTER SELECT SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Th(REG)	REGISTER SELECT HOLD TIME	30	-	-	ns
Tv(WT)	WAIT VALID FROM IORD#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	IORD# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(IORD)	DATA ENABLE TIME FROM IORD#	0	-	-	ns
Tdis(IORD)	DATA DISABLE TIME FROM IORD#	30	-	-	ns
Tw(IORD)	IORD# PULSE WIDTH	**	-	-	ns
TdfIOIS16	IOIS16# DELAY FALLING FROM ADDRESS	-	-	10	ns
TdrIOIS16	IOIS16# DELAY RISING FROM ADDRESS	0	-	-	ns

* TcR = Tsu(A) + Tw(IORD) + Th(A)

** Tw(IORD) = Base on Tab-4 Set Data Access Timing Cycle (TC) ➔ No wait condition

= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition

8.4.9 Pcmcia I/O Write Timing

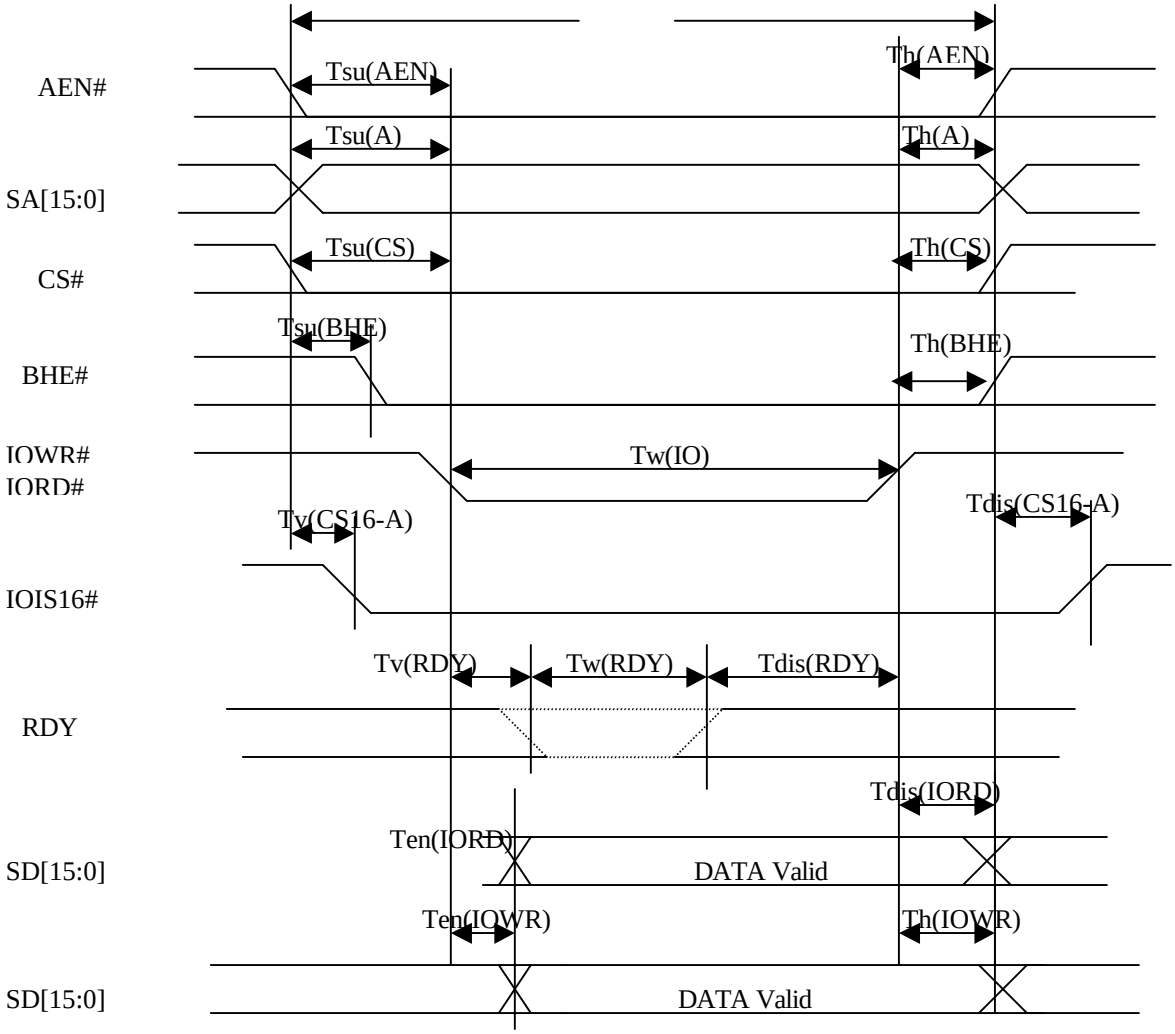


Symbol	Description	Min	Typ	Max	Units
TcW	WRITE CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CE)	CARD ENABLE SETUP TIME (CE1)	60	-	-	ns
Tsu(CE2)	CARD ENABLE SETUP TIME (CE2)	30	-	-	ns
Tsu(REG)	REGISTER SELECT SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CE)	CARD ENABLE HOLD TIME (CE1)	30	-	-	ns
Th(CE2)	CARD ENABLE HOLD TIME (CE2)	30	-	-	ns
Th(REG)	REGISTER SELECT HOLD TIME	30	-	-	ns
Tv(WT)	WAIT VALID FROM IOWR#	-	-	20	ns
Tw(WT)	WAIT# PULSE WIDTH	0	-	-	ns
Tdis(WT)	IOWR# HIGH FROM WAIT# DISABLE	30	-	-	ns
Ten(IOWR)	DATA ENABLE TIME FROM IOWR#	30	-	-	ns
Th(IOWR)	DATA DISABLE TIME FROM IOWR#	30	-	-	ns
Tw(IOWR)	IOWR# PULSE WIDTH	**	-	-	ns
TdfIOIS16	IOIS16# DELAY FALLING FROM ADDRESS	-	-	10	ns
TdrIOIS16	IOIS16# DELAY RISING FROM ADDRESS	0	-	-	ns

* TcW = Tsu(A) + Tw(IOWR) + Th(A)

** Tw(IOWR) = Base on Tab-4 Set Data Access Cycle (TC) ➔ No wait condition (Tw(WT) = 0)
= Tv(WT) + Tw(WT) + Tdis (WT) ➔ wait condition (Tw(WT) != 0)

8.4.9 ISA Bus Access I/O Timing

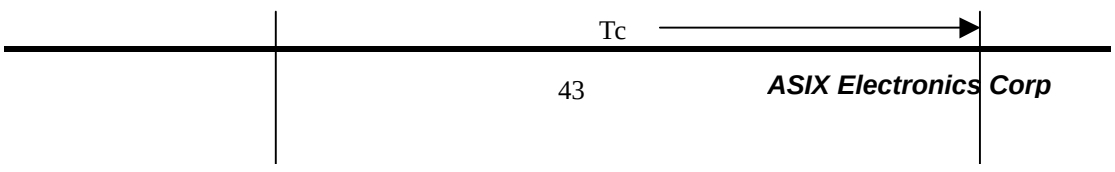


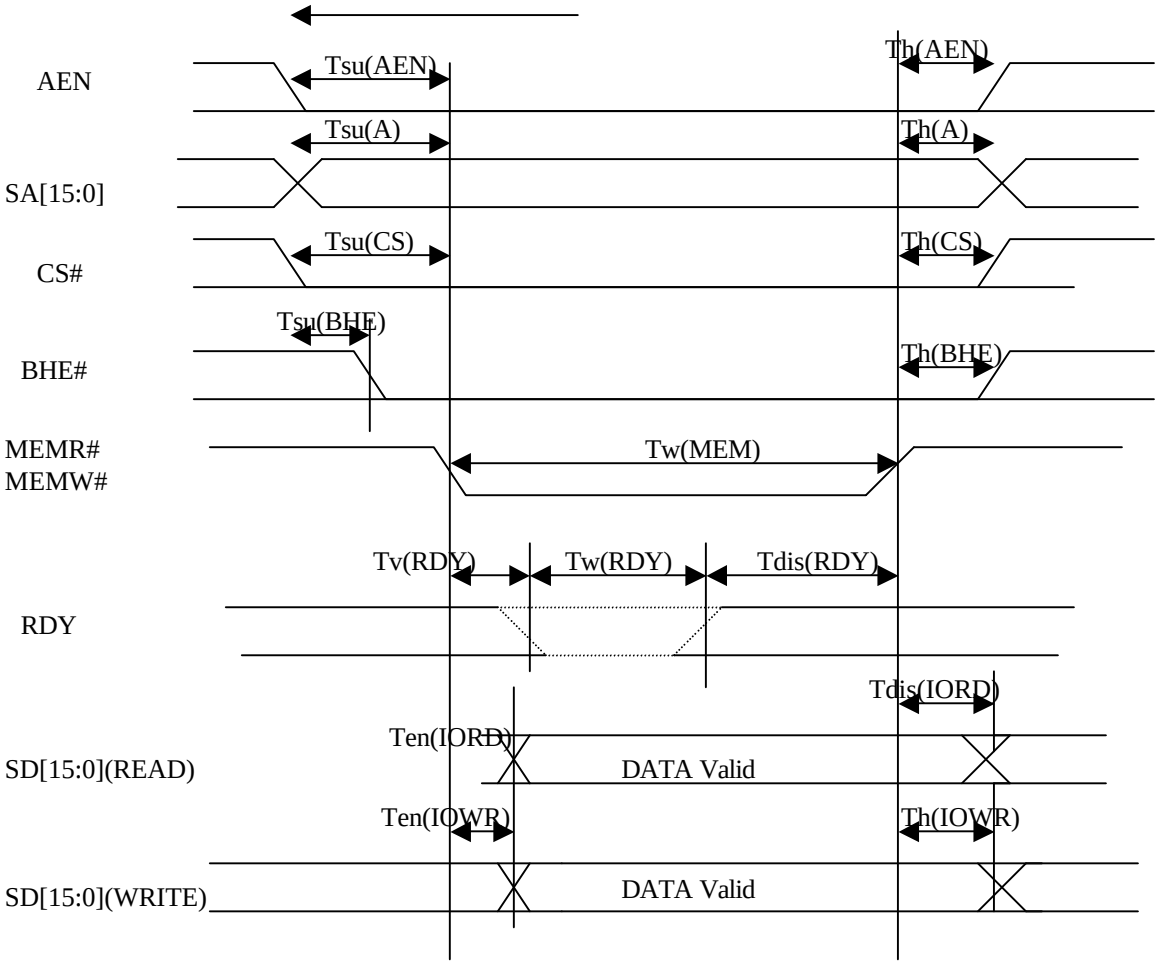
Symbol	Description	Min	Typ	Max	Units
Tc	CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CS)	CHIP SELECT SETUP TIME	60	-	-	ns
Tsu(BHE)	BYTE HIGH ENABLE SETUP TIME	30	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CS)	CHIP SELECT HOLD TIME	30	-	-	ns
Th(BHE)	BYTE HIGH ENABLE HOLD TIME	30	-	-	ns
Tv(RDY)	RDY VALID FROM IO#	-	-	20	ns
Tw(RDY)	RDY# PULSE WIDTH	0	-	-	ns
Tdis(RDY)	IO# HIGH FROM RDY# DISABLE	30	-	-	ns
Ten(IORD)	DATA ENABLE TIME FROM IORD#	0	-	-	ns
Tdis(IORD)	DATA DISABLE TIME FROM IORD#	30	-	-	ns
Ten(IOWR)	DATA ENABLE TIME FROM IOWR#	30	-	-	ns
Th(IOWR)	DATA DISABLE TIME FROM IOWR#	30	-	-	ns
Tw(IO)	IO# PULSE WIDTH	**	-	-	ns
Tv(CS16-A)	IOIS16# VALID FROM ADDRESS CHANGE	-	-	10	ns
Tdis(CS16-A)	IOIS16# DISABLE FROM ADDRESS CHANGE	0	-	-	ns

* Tc = Tsu(A) + Tw(IO) + Th(A) IO: include (IOWR & IORD)

** Tw(IO) = Base on Tab-4 Set Data Access Cycle (TC) ➔ No wait condition (Tw(RDY) = 0)
= Tv(RDY) + Tw(RDY) + Tdis (RDY) ➔ wait condition (Tw(RDY) != 0)

8.4.10 ISA Bus Access MEM Timing



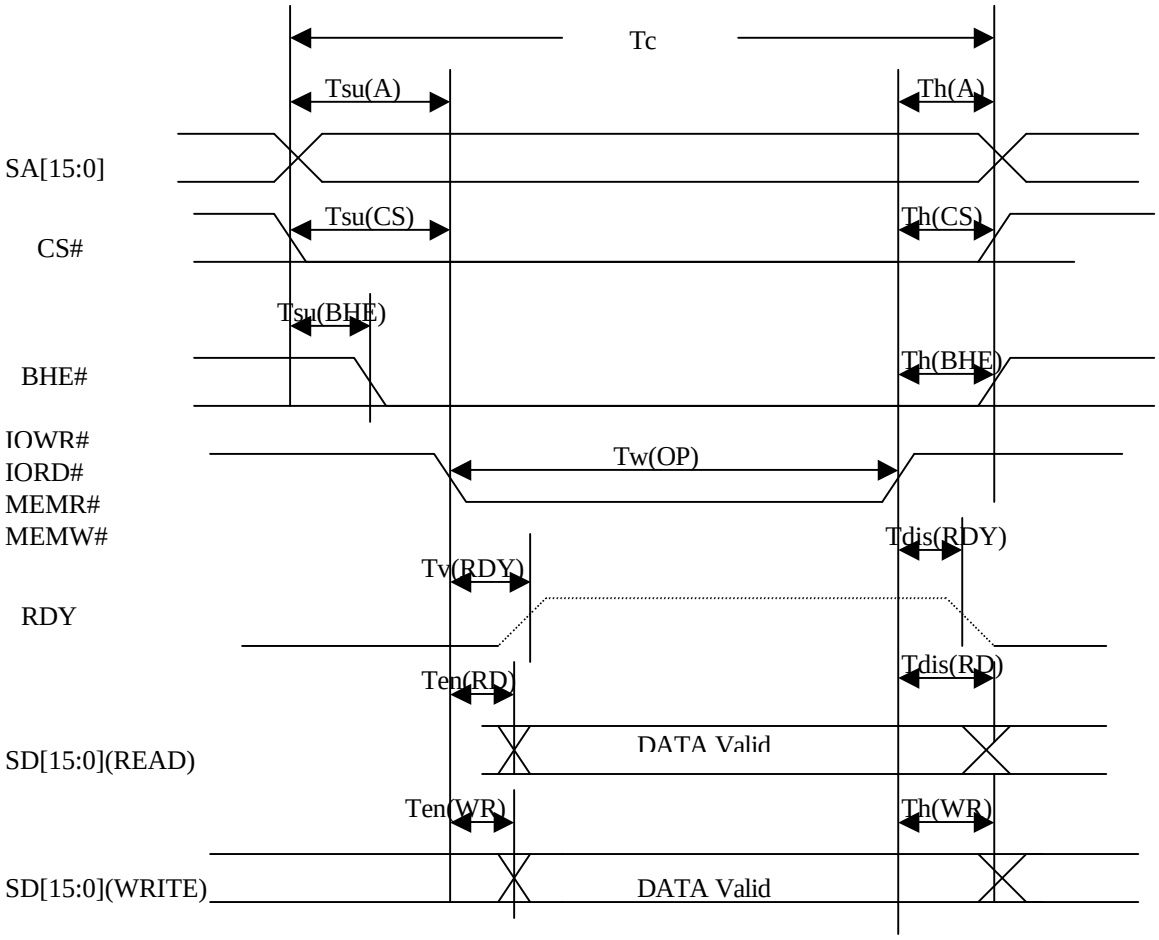


Symbol	Description	Min	Typ	Max	Units
Tc	CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CS)	CHIP SELECT SETUP TIME	60	-	-	ns
Tsu(BHE)	BYTE HIGH ENABLE SETUP TIME	30	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CS)	CHIP SELECT HOLD TIME	30	-	-	ns
Th(BHE)	BYTE HIGH ENABLE HOLD TIME	30	-	-	ns
Tv(RDY)	RDY VALID FROM MEM#	-	-	20	ns
Tw(RDY)	RDY# PULSE WIDTH	0	-	-	ns
Tdis(RDY)	MEM# HIGH FROM RDY# DISABLE	30	-	-	ns
Ten(MEMR)	DATA ENABLE TIME FROM MEMR#	0	-	-	ns
Tdis(MEMR)	DATA DISABLE TIME FROM MEMR#	30	-	-	ns
Ten(MEMW)	DATA ENABLE TIME FROM MEMW #	30	-	-	ns
Th(MEMW)	DATA DISABLE TIME FROM MEMW #	30	-	-	ns
Tw(MEM)	MEM# PULSE WIDTH	**	-	-	ns

* Tc = Tsu(A) + Tw(MEM) + Th(A) MEM: include (MEMR & MEMW)

** Tw(MEM) = Base on Tab-4 Set Data Access Cycle (TC) ➔ No wait condition (Tw(RDY) = 0)
= Tv(RDY) + Tw(RDY) + Tdis (RDY) ➔ wait condition (Tw(RDY) != 0)

8.4.11 80186 Type Access Timing

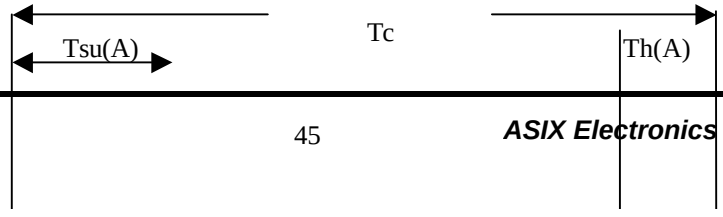


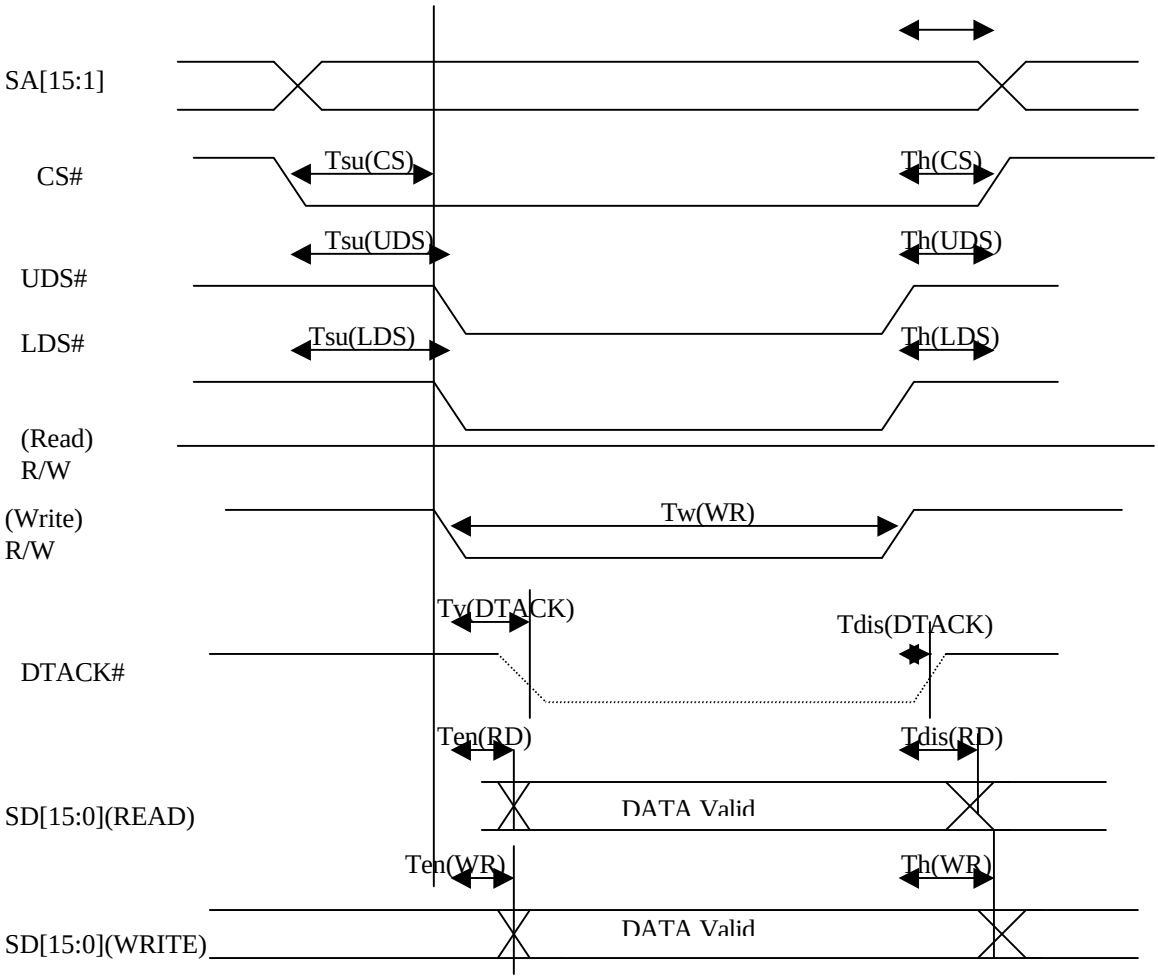
Symbol	Description	Min	Typ	Max	Units
Tc	CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CS)	CHIP SELECT SETUP TIME	60	-	-	ns
Tsu(BHE)	BYTE HIGH ENABLE SETUP TIME	30	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CS)	CHIP SELECT HOLD TIME	30	-	-	ns
Th(BHE)	BYTE HIGH ENABLE HOLD TIME	30	-	-	ns
Tv(RDY)	RDY VALID FROM MEM#	-	-	20	ns
Tdis(RDY)	OP# HIGH FROM RDY# DISABLE	0	-	-	ns
Ten(RD)	DATA ENABLE TIME FROM OP#	0	-	-	ns
Tdis(RD)	DATA DISABLE TIME FROM OP#	30	-	-	ns
Ten(WR)	DATA ENABLE TIME FROM OP#	30	-	-	ns
Th(WR)	DATA DISABLE TIME FROM OP#	30	-	-	ns
Tw(OP)	OP# PULSE WIDTH	**	-	-	ns

* Tc = Tsu(A) + Tw(OP) + Th(A) OP: include (MEMR & MEMW & IOWR & IORD)

** Tw(OP) = Base on Tab-4 Set Data Access Cycle (TC)

8.4.12 68K Type Access Timing





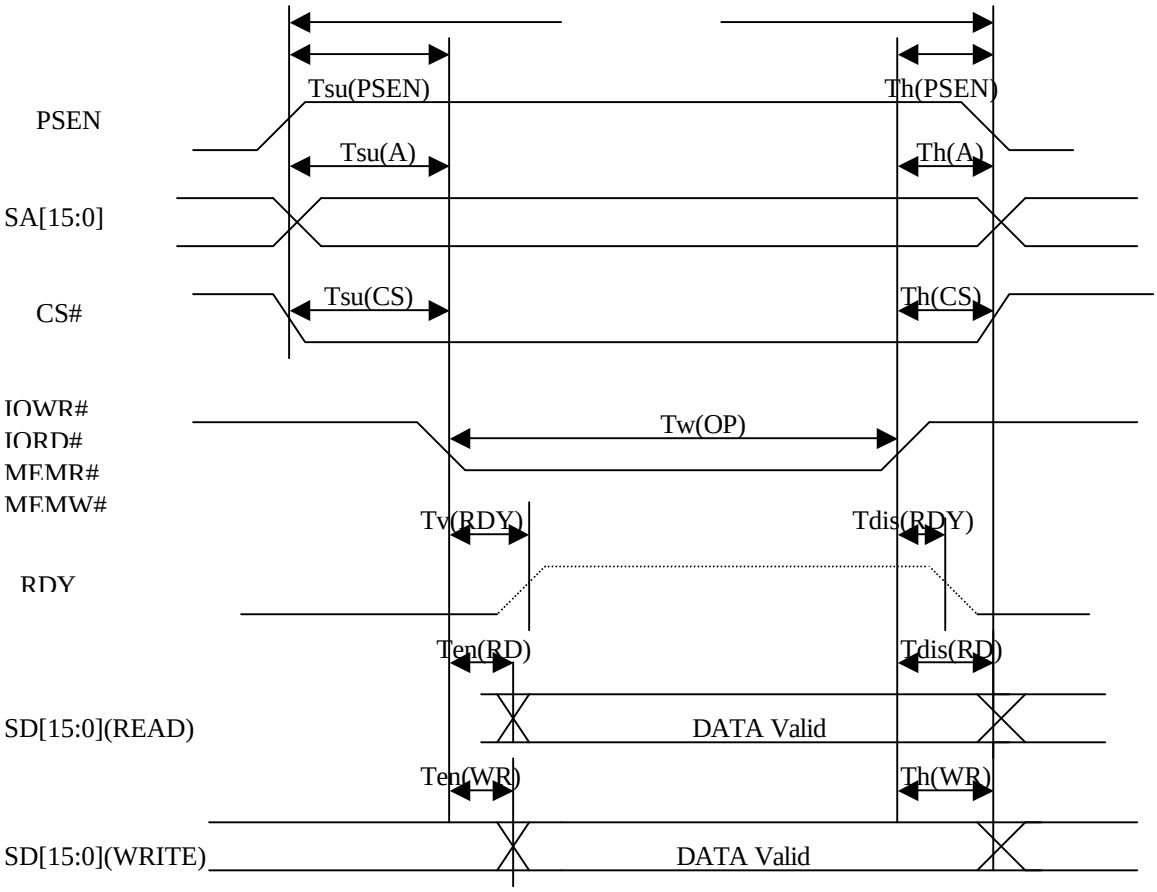
Symbol	Description	Min	Typ	Max	Units
Tc	CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CS)	CHIP SELECT SETUP TIME	60	-	-	ns
Tsu(UDS)	UPPER DATA STROBE SETUP TIME	60	-	-	ns
Tsu(LDS)	LOWER DATA STROBE SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CS)	CHIP SELECT HOLD TIME	30	-	-	ns
Th(UDS)	UPPER DATA STROBE HOLD TIME	30	-	-	ns
Th(LDS)	LOWER DATA STROBE HOLD TIME	30	-	-	ns
Tv(DTACK)	DTACK VALID FROM UDS# OR LDS#	-	-	20	ns
Tdis(DTACK)	DTACK DISABLE FROM UDS# OR LDS#	0	-	-	ns
Ten(RD)	DATA ENABLE TIME FROM UDS# OR LDS#	0	-	-	ns
Tdis(RD)	DATA DISABLE TIME FROM UDS# OR LDS#	30	-	-	ns
Ten(WR)	DATA ENABLE TIME FROM R/W (WRITE)	30	-	-	ns
Th(WR)	DATA DISABLE TIME FROM R/W (WRITE)	30	-	-	ns
Tw(WR)	WR# PULSE WIDTH	**	-	-	ns

* Tc = Tsu(A) + Tw(WR) + Th(A)

** Tw(WR) = Base on Tab-4 Set Data Access Cycle (TC)

8.4.12 8051 Bus Access Timing

Tc

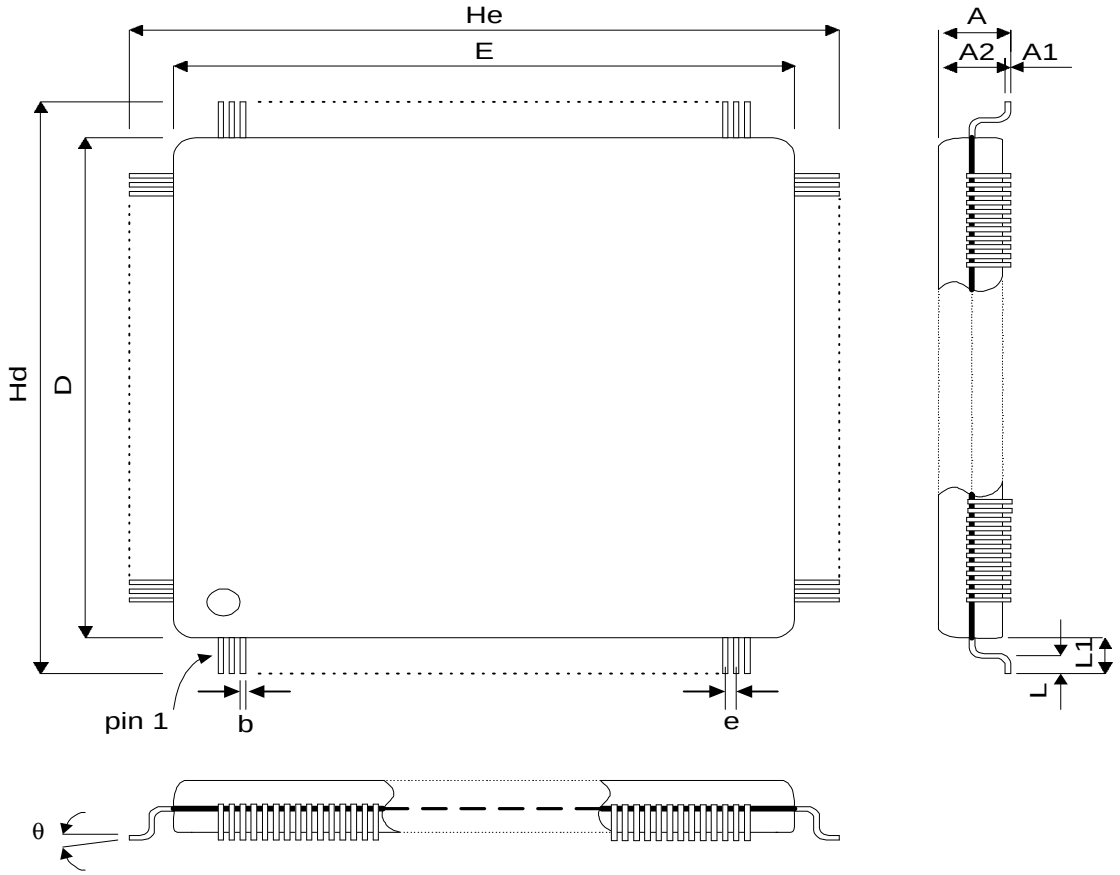


Symbol	Description	Min	Typ	Max	Units
Tc	CYCLE TIME	*	-	-	ns
Tsu(A)	ADDRESS SETUP TIME	60	-	-	ns
Tsu(CS)	CHIP SELECT SETUP TIME	60	-	-	ns
Tsu(PSEN)	PSEN SETUP TIME	60	-	-	ns
Th(A)	ADDRESS HOLD TIME	30	-	-	ns
Th(CS)	CHIP SELECT HOLD TIME	30	-	-	ns
Th(PSEN)	PSEN HOLD TIME	30	-	-	ns
Tv(RDY)	RDY VALID FROM OP#	-	-	20	ns
Tdis(RDY)	OP# HIGH FROM RDY# DISABLE	0	-	-	ns
Ten(RD)	DATA ENABLE TIME FROM OP#	0	-	-	ns
Tdis(RD)	DATA DISABLE TIME FROM OP#	30	-	-	ns
Ten(WR)	DATA ENABLE TIME FROM OP#	30	-	-	ns
Th(WR)	DATA DISABLE TIME FROM OP#	30	-	-	ns
Tw(OP)	OP# PULSE WIDTH	**	-	-	ns

* Tc = Tsu(A) + Tw(OP) + Th(A) OP: include (MEMR & MEMW & IOWR & IORD)

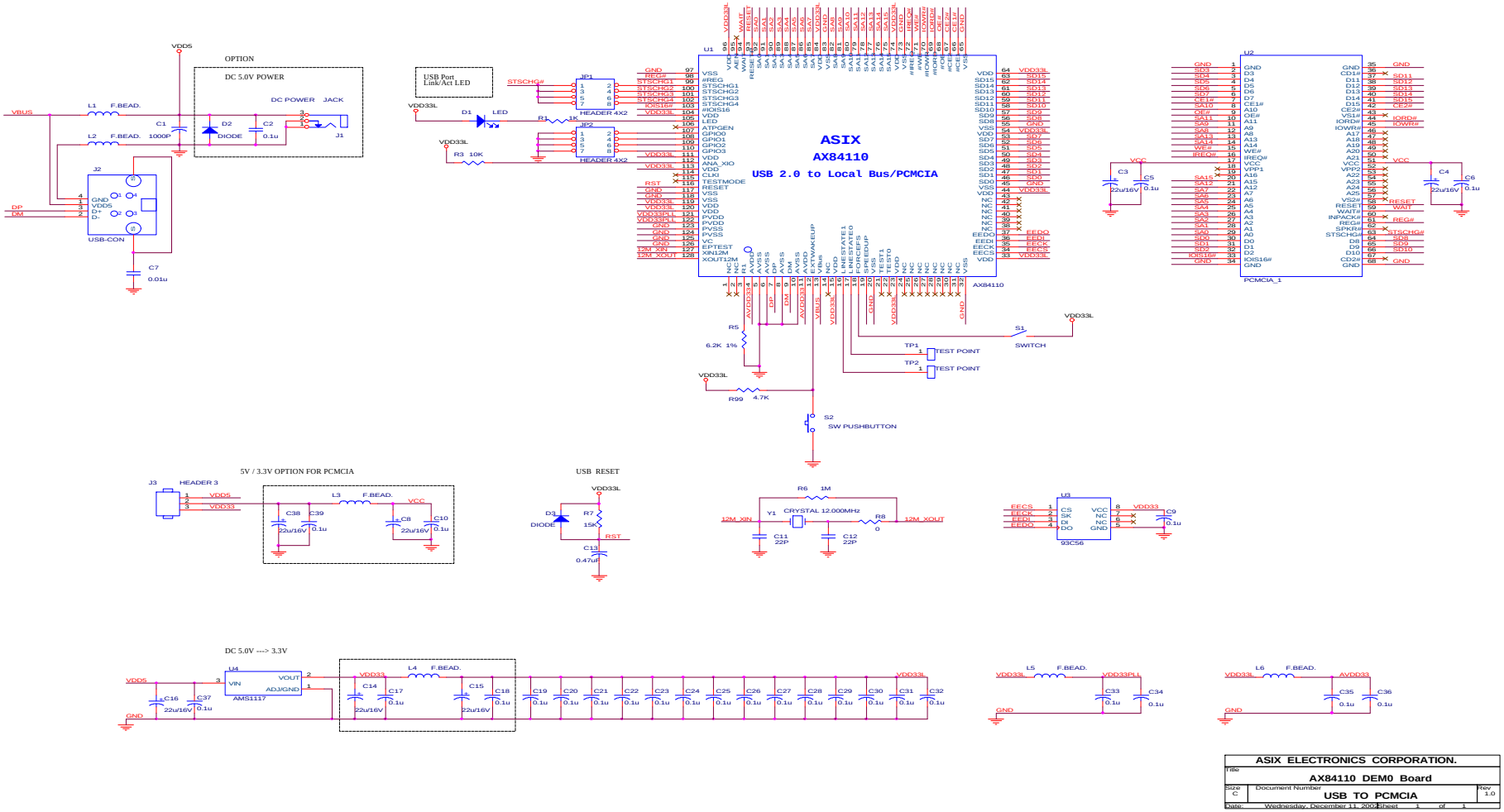
** Tw(OP) = Base on Tab-4 Set Data Access Cycle (TC)

9.0 Package information



SYMBOL	MILIMETER		
	MIN.	NOM	MAX
A1	0.05	0.1	0.15
A2	1.39	1.40	1.41
A			1.70
b	0.155	0.16	0.26
D	13.9	14.00	14.1
E	13.9	14.00	14.1
e		0.5	
Hd	15.6	16.00	16.4
He	15.6	16.00	16.4
L	0.3	0.50	0.7
L1		1.00	
θ	0°		10°

10.0 Reference design (PCMCIA to USB)



Revision	Date	Comment
V. 0.0	12/16/2002	Initial release.

ASIX Electronics Corporation.

4F, NO.8, HSIN ANN RD., SCIENCE-BASED

INDUSTRIAL PARK, HSINCHU, TAIWAN, R.O.C.

FAX: 886-3-5799558

Email: support@asix.com.tw

Web: http://www.asix.com.tw