

4. REGISTER DESCRIPTION

4.1. ADDRESSING

The ADDRESS[2:0] address lines of the processor interface provide access to six 8-bit registers – four read/write registers, one read-only register and one write-only register – as follows:

ADDRESS[2:0]	REGISTER	DESCRIPTION
0 0 0	ADDR	Slave address
1 0 0	XADDR	Extended slave address
0 0 1	DATA	Data byte
0 1 0	CNTR	Control register
0 1 1	STAT	Status register (read only)
0 1 1	CCR	Clock control register (write only)
1 1 1	SRST	Software reset (see Section 8)

On a hardware reset, ADDR, XADDR, DATA and CNTR are cleared to 00h; STAT is set to F8h and CCR is set to 00h. On a software reset, STAT is set to F8h and the STP, STA and IFLG bits of the CNTR register are set to '0'. (Reset is described in Section 8.)

4.2. ADDR REGISTER

Read/Write: Address 000b

POSN	7-BIT ADDRESSING		10-BIT ADDRESSING	
	BIT	DESCRIPTION	BIT	DESCRIPTION
D7	SLA6	Slave Address	1	
D6	SLA5	Slave Address	1	
D5	SLA4	Slave Address	1	
D4	SLA3	Slave Address	1	
D3	SLA2	Slave Address	0	
D2	SLA1	Slave Address	SLAX9	Extended Slave Address
D1	SLA0	Slave Address	SLAX8	Extended Slave Address
D0	GCE	General call address enable	GCE	General call address enable

For 7-bit addressing:

SLA6 – SLA0 is the 7-bit address of the MI2CV when in slave mode. When the MI2CV receives this address after a START condition, it will generate an interrupt and enter slave mode. (SLA6 corresponds to the first bit received from the I²C bus.) If GCE is set to ‘1’, the MI2CV will also recognize the general call address (00h).

For 10-bit addressing:

When the address received starts with 11110b, the MI2CV recognizes this as the first part of a 10-bit address and if the next two bits match ADDR[2:1] (i.e. SLAX9 and SLAX8 of the device’s extended address), it sends an ACK. (The device does not generate an interrupt at this point.) If the next byte of the address matches the XADDR register (SLAX7 – SLAX0), the MI2CV generates an interrupt and goes into slave mode.

4.3. XADDR REGISTER

Read/Write: Address 100b

POSN	BIT	DESCRIPTION
D7	SLAX7	Extended Slave Address
D6	SLAX6	Extended Slave Address
D5	SLAX5	Extended Slave Address
D4	SLAX4	Extended Slave Address
D3	SLAX3	Extended Slave Address
D2	SLAX2	Extended Slave Address
D1	SLAX1	Extended Slave Address
D0	SLAX0	Extended Slave Address

4.4. DATA REGISTER

Read/Write: Address 001b

This register contains the data byte/slave address to be transmitted or the data byte that has just been received. In transmit mode, the byte is sent MSB first; in receive mode, the first bit received will be placed in the MSB of the register.

After each byte is transmitted, the DATA register will contain the byte that was actually present on the bus so in the event of lost arbitration, it will contain the received byte.

4.5. CNTR REGISTER

Read/Write: Address 010b

POSN	BIT	DESCRIPTION
D7	IEN	Interrupt enable
D6	ENAB	Bus enable
D5	STA	Master mode start
D4	STP	Master mode stop
D3	IFLG	Interrupt flag
D2	AAK	Assert acknowledge
D1	–	Unused
D0	–	Unused

Bits 0 & 1 are read only and are read back as '0'.

4.5.1. IEN – INTERRUPT ENABLE

When IEN is set to '1', the interrupt line (INTR) will go high when the IFLG bit is set.

When IEN is '0', the interrupt line will always remain low.

4.5.2. ENAB – BUS ENABLE

When ENAB is '0', the I²C bus inputs ISDA/ISCL are ignored and the MI2CV will not respond to any address on the bus.

When ENAB set to '1', the MI2CV will respond to calls to its slave address – and to the general call address if the GCE bit in the ADDR register is set.

Note: The bus needs to be enabled (ENAB set to '1') for the MI2CV to operate in master mode.

4.5.3. STA – MASTER MODE START

When STA is set to '1', the MI2CV enters master mode and will transmit a START condition on the bus when the bus is free. If the STA bit is set to '1' when the MI2CV is already in master mode and one or more bytes have been transmitted, then a repeated START condition will be sent. If the STA bit is set to '1' when the MI2CV is being accessed in slave mode, the MI2CV will complete the data transfer in slave mode then enter master mode when the bus has been released.

The STA bit is cleared automatically after a START condition has been sent: writing a '0' to this bit has no effect.

4.5.4. STP – MASTER MODE STOP

If STP is set to ‘1’ in master mode, a STOP condition is transmitted on the I²C bus. If the STP bit is set to ‘1’ in slave mode, the MI2CV will behave as if a STOP condition has been received, but no STOP condition will be transmitted on the I²C bus. If both STA and STP bits are set, the MI2CV will first transmit the STOP condition (if in master mode) then transmit the START condition.

The STP bit is cleared automatically: writing a ‘0’ to this bit has no effect.

4.5.5. IFLG – INTERRUPT FLAG

IFLG is automatically set to ‘1’ when any of 28 (out of the possible 29) MI2CV states is entered (see ‘STAT Register’ below). The only state that does not set IFLG is state F8h. If the IEN bit is set, the interrupt line goes high when IFLG is set to ‘1’.

If the MI2CV is operating in slave mode, data transfer is suspended when IFLG is set and the low period of the I²C bus clock line (SCL) is stretched until ‘0’ is written to IFLG. The I²C clock line is then released and the interrupt line goes low.

4.5.6. AAK – ASSERT ACKNOWLEDGE

When AAK is set to ‘1’, an Acknowledge (low level on SDA) will be sent during the acknowledge clock pulse on the I²C bus if:

- i Either the whole of a matching 7-bit slave address or the first or the second byte of a matching 10-bit slave address has been received.
- ii The general call address has been received and the GCE bit in the ADDR register is set to ‘1’.
- iii A data byte has been received in master or slave mode.

When AAK is ‘0’, a Not Acknowledge (high level on SDA) will be sent when a data byte is received in master or slave mode.

If AAK is cleared to ‘0’ in slave transmitter mode, the byte in the DATA register is assumed to be the ‘last byte’. After this byte has been transmitted, the MI2CV will enter state C8h then return to the idle state (status code F8h) when IFLG is cleared.

The MI2CV will not respond as a slave unless AAK is set.

4.6. STAT REGISTER

Read-Only: Address 011b

This register contains a 5-bit status code in the five MSBs: the three LSBs are always ‘0’.

There are 32 possible codes, listed in the table overleaf, 29 of which are used as status codes and three of which are unused.

When STAT contains the status code F8h, no relevant status information is available, no interrupt is generated and the IFLG bit in the CNTR register is not set. All other status codes correspond to a defined state of the MI2CV. When any of these states is entered, the corresponding status code appears in this register and the IFLG bit in the CNTR register is set. When the IFLG bit is cleared, the status code returns to F8h.

CONFIDENTIAL



Code	Status
00h	Bus error
08h	START condition transmitted
10h	Repeated START condition transmitted
18h	Address + Write bit transmitted, ACK received
20h	Address + Write bit transmitted, ACK not received
28h	Data byte transmitted in master mode, ACK received
30h	Data byte transmitted in master mode, ACK not received
38h	Arbitration lost in address or data byte
40h	Address + Read bit transmitted, ACK received
48h	Address + Read bit transmitted, ACK not received
50h	Data byte received in master mode, ACK transmitted
58h	Data byte received in master mode, not ACK transmitted
60h	Slave address + Write bit received, ACK transmitted
68h	Arbitration lost in address as master, slave address + Write bit received, ACK transmitted
70h	General Call address received, ACK transmitted
78h	Arbitration lost in address as master, General Call address received, ACK transmitted
80h	Data byte received after slave address received, ACK transmitted
88h	Data byte received after slave address received, not ACK transmitted
90h	Data byte received after General Call received, ACK transmitted
98h	Data byte received after General Call received, not ACK transmitted
A0h	STOP or repeated START condition received in slave mode
A8h	Slave address + Read bit received, ACK transmitted
B0h	Arbitration lost in address as master, slave address + Read bit received, ACK transmitted
B8h	Data byte transmitted in slave mode, ACK received
C0h	Data byte transmitted in slave mode, ACK not received
C8h	Last byte transmitted in slave mode, ACK received
D0h	Second Address byte + Write bit transmitted, ACK received
D8h	Second Address byte + Write bit transmitted, ACK not received
E0h	Unused
E8h	Unused
F0h	Unused
F8h	No relevant status information, IFLG=0

If an illegal condition occurs on the I²C bus, the bus error state is entered (status code 00h). To recover from this state, the STP bit in the CNTR register must be set and the IFLG bit cleared. The MI2CV will then return to idle state (status code F8h): no STOP condition will be transmitted on the I²C bus. *Note:* To request resumption of transmission, set the STA bit to 1 at the same time as the STP bit is set. The MI2CV will then send a START on recovery from the bus error.

4.7. CCR REGISTER

Write-Only: Address 011b

This register is write only. The seven LSBs control the frequency at which the I²C bus is sampled and the frequency of the I²C clock line (SCL) when the MI2CV is in master mode.

POSN	BIT
D7	–
D6	M3
D5	M2
D4	M1
D3	M0
D2	N2
D1	N1
D0	N0

The input clock frequency (i.e. the frequency of CLOCK) is first divided by a factor of 2^N , where N is the value defined by bits 2–0 of CCR. The output of this clock divider is F_0 . F_0 is then divided by a further factor of M+1, where M is the value defined by bits 6–3 of CCR. The output of this clock divider is F_1 .

The I²C bus is sampled by the MI2CV at the frequency defined by F_0 :

$$F_{SAMP} = F_0 = F_{CLOCK} / 2^N$$

The MI2CV OSCL output frequency, in master mode, is $F_1 / 10$:

$$F_{OSCL} = F_1 / 10 = F_{CLOCK} / (2^N \cdot (M + 1) \cdot 10)$$

The use of two separately programmable dividers allows the master mode output frequency to be set independently of the frequency at which the I²C bus is sampled. This is particularly useful in multi-master systems because the frequency at which the I²C bus is sampled must be at least 10 times the frequency of the fastest master on the bus to ensure that START and STOP conditions are always detected. By using two programmable clock divider stages, a high sampling frequency can be ensured while allowing the master mode output to be set to a lower frequency.

5. BUS CLOCK SPEED

The I²C bus is defined for bus clock speeds up to 100kbits/s (400 kbits/s in 'fast-mode').

To ensure correct detection of START and STOP conditions on the bus, the MI2CV must sample the I²C bus at least ten times faster than the bus clock speed of the fastest master on the bus. The sampling frequency should therefore be at least 1 MHz (4 MHz in 'fast-mode') to guarantee correct operation with other bus masters.

The MI2CV sampling frequency is determined by the frequency of the input clock (CLOCK) and the value in CCR bits 2–0. The bus clock speed generated by the MI2CV in master mode is determined by the frequency of the input clock and the values in CCR bits 2–0 and CCR bits 6–3. (See 'CCR Register' above.)

6. CLOCK SYNCHRONIZATION

If another device on the I²C bus drives the clock line when the MI2CV is in master mode, the MI2CV will synchronize its clock to the I²C bus clock. The high period of the clock will be determined by the device that generates the shortest high clock period. The low period of the clock will be determined by the device that generates the longest low clock period.

When the MI2CV is in master mode and is communicating with a slow slave, the slave may stretch each bit period by holding the SCL line low until it is ready for the next bit. The MI2CV will automatically re-synchronize as described above.

When the MI2CV is in slave mode, it will hold the SCL line low after each byte has been transferred until IFLG has been cleared in the CNTR register.

7. BUS ARBITRATION

In master mode, the MI2CV will check that each transmitted logic 1 appears on the I²C bus as a logic 1. If another device on the bus over-rides and pulls the SDA line low, arbitration is lost. If arbitration is lost during the transmission of a data byte or a Not-Acknowledge bit, the MI2CV will return to idle state. If arbitration is lost during the transmission of an address, the MI2CV will switch to slave mode so that it can recognize its own slave address or the general call address.

8. RESET

The RESETN input provides an asynchronous power-up reset to the device. A software reset may also be applied by writing any value to address 111b.

When the MI2CV is reset using the RESETN pin, the ADDR, XADDR, DATA and CNTR registers are cleared to 00h; STAT is set to F8h and CCR is set to 00h.

A software reset sets the MI2CV back to idle (STAT set to F8h) and sets the STP, STA and IFLG bits of the CNTR (Control) register to '0'.

9. OPERATING MODES

Note: The following operations all require the ENAB bit of the CNTR register to be set to ‘1’.

9.1. MASTER TRANSMIT

In the master transmit mode, the MI2CV will transmit a number of bytes to a slave receiver.

The master transmit mode is entered by setting the STA in the CNTR register bit to ‘1’. The MI2CV will then test the I²C bus and will transmit a START condition when the bus is free. When a START condition has been transmitted, the IFLG bit will be set and the status code in the STAT register will be 08h. Before this interrupt is serviced, the DATA register must be loaded with either a 7-bit slave address or the first part of a 10-bit slave address, with the LSB cleared to ‘0’ (i.e. with an added Write bit) to specify transmit mode. The IFLG bit should now be cleared to ‘0’ to prompt the transfer to continue.

After the 7-bit slave address (or the first part of a 10-bit address) plus the Write bit have been transmitted, IFLG will be set again. A number of status codes are possible in the STAT register:

CODE	MI2CV STATE	MICROPROCESSOR RESPONSE	NEXT MI2CV ACTION
18h	Addr+W transmitted, ACK received	<i>For a 7-bit address:</i> Write byte to DATA, clear IFLG Or set STA, clear IFLG Or set STP, clear IFLG Or set STA & STP, clear IFLG <i>For a 10-bit address:</i> Write extended address byte to DATA, clear IFLG	Transmit data byte, receive ACK Transmit repeated START Transmit STOP Transmit STOP then START Transmit extended address byte
20h	Addr+W transmitted, ACK not received	As for code 18h	As for code 18h
38h	Arbitration lost	Clear IFLG Or set STA, clear IFLG	Return to idle Transmit START when bus free
68h	Arbitration lost, SLA+W received, ACK transmitted	Clear IFLG, AAK=0 Or clear IFLG, AAK=1	Receive data byte, transmit not ACK Receive data byte, transmit ACK
78h	Arbitration lost, general call addr received, ACK transmitted	As for code 68h	As for code 68h
B0h	Arbitration lost, SLA+R received, ACK transmitted	Write byte to DATA, clear IFLG, AAK=0 Or write byte to DATA, clear IFLG, AAK=1	Transmit last byte, receive ACK Transmit data byte, receive ACK

Note: ‘W’=Write bit i.e. LSB cleared to ‘0’; ‘R’ =Read bit i.e. LSB set to ‘1’.

CONFIDENTIAL



If 10-bit addressing is being used, then after the first part of a 10-bit address plus the Write bit have been successfully transmitted, the status code will be 18h or 20h.

After this interrupt has been serviced and the second part of the address transmitted, the STAT register will contain one of the following codes:

CODE	MI2CV STATE	MICROPROCESSOR RESPONSE	NEXT MI2CV ACTION
38h	Arbitration lost	Clear IFLG Or set STA, clear IFLG	Return to idle Transmit START when bus free
68h	Arbitration lost, SLAX+W received, ACK transmitted	Clear IFLG, AAK=0 Or clear IFLG, AAK=1	Receive data byte, transmit not ACK Receive data byte, transmit ACK
B0h	Arbitration lost, SLAX+R received, ACK transmitted	Write byte to DATA, clear IFLG, AAK=0 Or write byte to DATA, clear IFLG, AAK=1	Transmit last byte, receive ACK Transmit data byte, receive ACK
D0h	Second Address byte +W transmitted, ACK received	Write byte to DATA, clear IFLG Or set STA, clear IFLG Or set STP, clear IFLG Or set STA & STP, clear IFLG	Transmit data byte, receive ACK Transmit repeated START Transmit STOP Transmit STOP then START
D8h	Second Address byte +W transmitted, ACK not received	As for code D0h	As for code D0h

If a repeated START condition has been transmitted, the status code will be 10h instead of 08h.

After each data byte has been transmitted, IFLG will be set and one of three status codes will be in the STAT register:

CODE	MI2CV STATE	MICROPROCESSOR RESPONSE	NEXT MI2CV ACTION
28h	Data byte transmitted, ACK received	Write byte to DATA, clear IFLG Or set STA, clear IFLG Or set STP, clear IFLG Or set STA & STP, clear IFLG	Transmit data byte, receive ACK Transmit repeated START Transmit STOP Transmit START then STOP
30h	Data byte transmitted, ACK not received	As for code 28h	As for code 28h
38h	Arbitration lost	Clear IFLG Or set STA, clear IFLG	Return to idle Transmit START when bus free

When all bytes have been transmitted, the STP bit should be set by writing a 1 to this bit in the CNTR register. The MI2CV will then transmit a STOP condition, clear the STP bit and return to idle state (status code F8h).

9.2. MASTER RECEIVE

In the master receive mode, the MI2CV will receive a number of bytes from a slave transmitter.

After the START condition has been transmitted, the IFLG bit will be set and status code 08h will be in the STAT register. The DATA register should now be loaded with the slave address (or the first part of a 10-bit slave address), with the LSB set to '1' to signify a Read. The IFLG bit should now be cleared to '0' to prompt the transfer to continue.

When the 7-bit slave address (or the first part of a 10-bit address) and the Read bit have been transmitted, the IFLG bit will be set again. A number of status codes are possible in the STAT register:

CODE	MI2CV STATE	MICROPROCESSOR RESPONSE	NEXT MI2CV ACTION
40h	Addr +R transmitted, ACK received	<i>For a 7-bit/10-bit address:</i> Clear IFLG, AAK = 0 Or clear IFLG, AAK = 1	Receive data byte, transmit not ACK Receive data byte, transmit ACK
48h	Addr +R transmitted, ACK not received	<i>For a 7-bit/ 10-bit address:</i> Set STA, clear IFLG Or set STP, clear IFLG Or set STA & STP, clear IFLG	Transmit repeated START Transmit STOP Transmit STOP then START
38h	As for master transmit	As for master transmit	As for master transmit
68h	As for master transmit	As for master transmit	As for master transmit
78h	As for master transmit	As for master transmit	As for master transmit
B0h	As for master transmit	As for master transmit	As for master transmit

If 10-bit addressing is being used, the slave is first addressed using the full 10-bit address plus the Write bit. The master then issues a restart followed the first part of the 10-bit address again, but plus the Read bit – after which the status code will be 40h or 48h. It is the responsibility of the slave to remember that it had been selected prior to the restart.

If a repeated START condition has been transmitted, the status code will be 10h instead of 08h.

After each data byte has been received, IFLG will be set and one of three status codes will be in the STAT register:

CODE	MI2CV STATE	MICROPROCESSOR RESPONSE	NEXT MI2CV ACTION
50h	Data byte received, ACK transmitted	Read DATA, clear IFLG, AAK = 0	Receive data byte, transmit not ACK
		Or read DATA, clear IFLG, AAK = 1	Receive data byte, transmit ACK
58h	Data byte received, not ACK transmitted	Read DATA, set STA, clear IFLG	Transmit repeated START
		Or read DATA, set STP, clear IFLG	Transmit STOP
		Or read DATA, set STA & STP, clear IFLG	Transmit STOP then START
38h	Arbitration lost in not ACK bit	As for master transmit	As for master transmit

When all bytes have been received, a not ACK should be transmitted then the STP bit should be set by writing a ‘1’ to this bit in the CNTR register. The MI2CV will transmit a STOP condition, clear the STP bit and return to idle state (status code F8h).

9.3. SLAVE TRANSMIT

In the slave transmit mode, a number of bytes are transmitted to a master receiver. For the MI2CV to respond, the AAK bit in the CNTR register needs to be set.

The MI2CV will enter slave transmit mode when it receives its own slave address and a Read bit after a START condition. The MI2CV will then transmit an acknowledge bit and set the IFLG bit in the CNTR register. The STAT register will contain the status code A8h.

Note: Where the MI2CV has an extended slave address (signified by 11110b in ADDR[7:3]), it will first be selected, then there will be a restart followed by another address byte. If this address byte matches the value stored in ADDR, the MI2CV will transmit an acknowledge after this address byte is received. An interrupt will be generated, IFLG will be set and the status will be A8h. No second address byte will be sent by the master: it is up to the slave to remember that it had been selected prior to the restart.

Slave transmit mode can also be entered directly from a master mode if arbitration is lost in master mode during the transmission of an address and the slave address and Read bit are received. The status code in the STAT register will then be B0h.

The data byte to be transmitted should then be loaded into the DATA register and IFLG cleared. When the MI2CV has transmitted the byte and received an acknowledge, IFLG will be set and the STAT register will contain B8h. Once the last byte to be transmitted has been loaded into the DATA register, the AAK bit should be cleared when IFLG is cleared. After the last byte has been transmitted, IFLG will be set and the STAT register will contain C8h. The MI2CV will then return to idle state (status code F8h). The AAK bit must be set to ‘1’ before slave mode can be entered again.

If no acknowledge is received after transmitting a byte, IFLG will be set and the STAT register will contain C0h. The MI2CV will then return to idle state.

If the STOP condition is detected after an acknowledge bit, the MI2CV will return to idle state.

CONFIDENTIAL



9.4. SLAVE RECEIVE

In the slave receive mode, a number of data bytes are received from a master transmitter.

The MI2CV will enter slave receive mode when it receives its own slave address and a Write bit (LSB=0) after a START condition. The MI2CV will then transmit an acknowledge bit and set the IFLG bit in the CNTR register: the STAT register will then contain status code 60h. The MI2CV will also enter slave receive mode when it receives the general call address 00h (if the GCE bit in the ADDR register is set). The status code will then be 70h.

Note: Where the MI2CV has an extended slave address (signified by 11110b in ADDR[7:3]), it will transmit an acknowledge after the first address byte is received but no interrupt will be generated, IFLG will not be set and the status will not change. Only after the second address byte has been received will the MI2CV generate an interrupt, set the IFLG bit and the status code as described above.

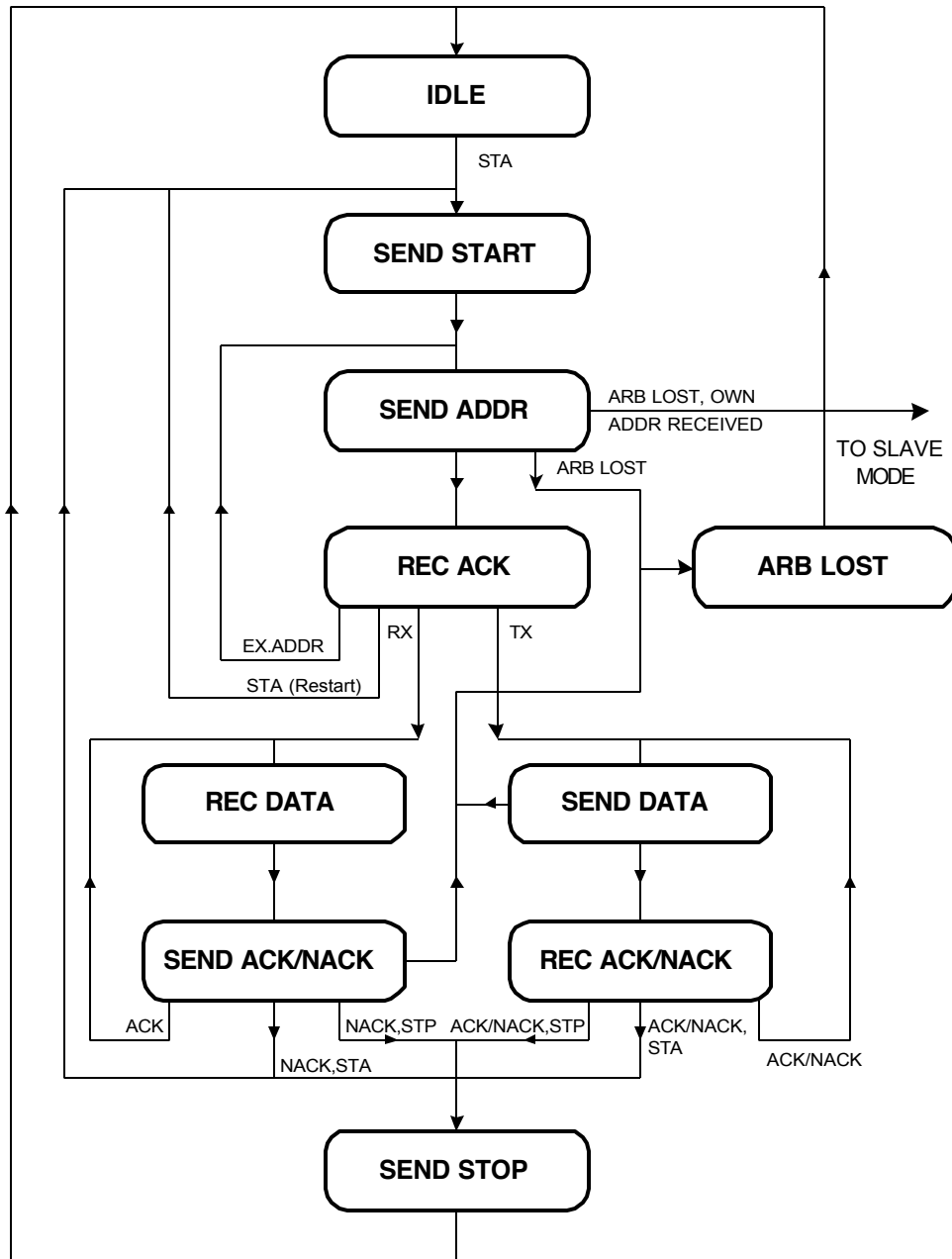
Slave receive mode can also be entered directly from a master mode if arbitration is lost in master mode during the transmission of an address and the slave address and Write bit (or the general call address if bit GCE in the ADDR register is set to '1') are received. The status code in the STAT register will then be 68h if the slave address was received or 78h if the general call address was received. The IFLG bit must be cleared to '0' to allow the data transfer to continue.

If the AAK bit in the CNTR register is set to '1', then after each byte is received, an acknowledge bit (low level on SDA) is transmitted and the IFLG bit is set: the STAT register will then contain status code 80h (or 90h if slave receive mode was entered with the general call address). The received data byte can be read from the DATA register and the IFLG bit must be cleared to allow the transfer to continue. When the STOP condition or a repeated START condition is detected after the acknowledge bit, then the IFLG bit is set and the STAT register will contain status code A0h.

If the AAK bit is cleared to '0' during a transfer, the MI2CV will transmit a not acknowledge bit (high level on SDA) after the next byte is received, and set the IFLG bit. The STAT register will contain status code 88h (or 98h if slave receive mode was entered with the general call address). When the IFLG bit has been cleared to '0', the MI2CV will return to idle state (status code F8h).

9.5. FLOW DIAGRAMS

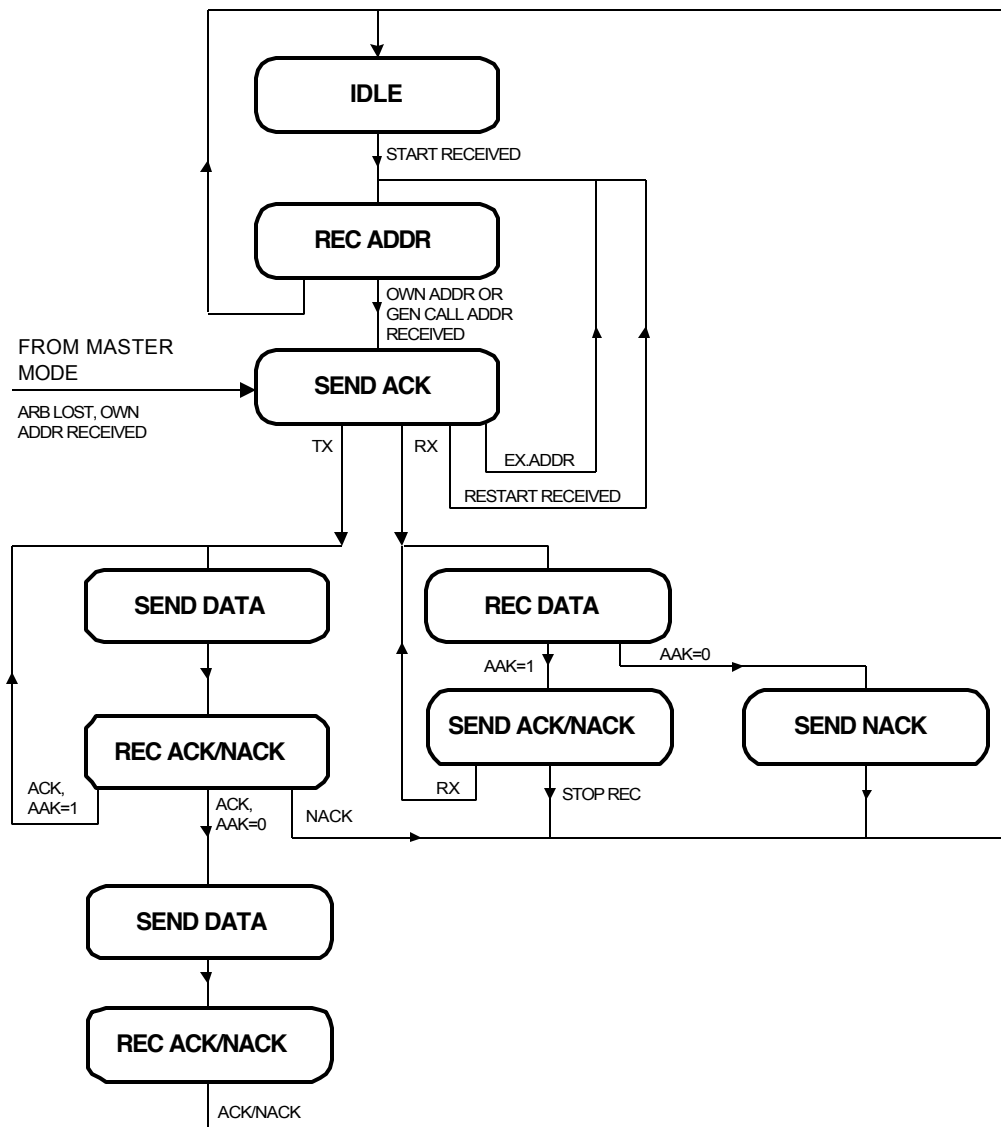
MASTER MODE



CONFIDENTIAL

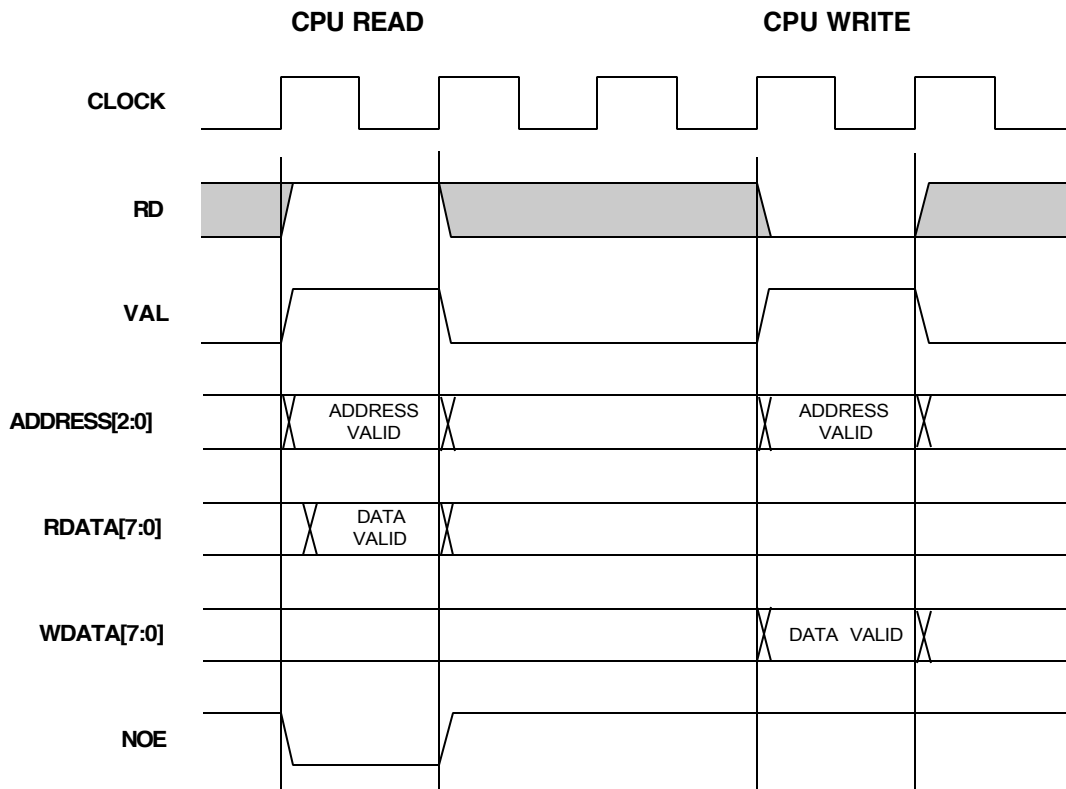


SLAVE MODE



10. TIMING INFORMATION

The following timing diagram shows when inputs to the MI2CV must be valid and when outputs are valid (e.g. which clock edges sample inputs and which clock edges cause outputs to change). The actual set-up and delay times will depend on the technology and layout used for the MI2CV.



Note: The ACK and EOP usually included in PPCI-compatible interfaces are omitted from the above diagram and from the core because the MI2CV implements the version of the PPCI interface with default acknowledge in which ACK and EOP are always logic '1'.