



AN<nnnnnnnnnn>

ATMEGA32 TWI/I2C

Rev 1.0

30-12-2009

Application Note

Revision History

Revision	Date	Description	Author
Rev 1.0	30-12-2009	Preliminary	Nguyen Trung Hieu

1. Giới thiệu khối TWI/I2C trên ATMEGA32

1.1. Tính năng tổng quát

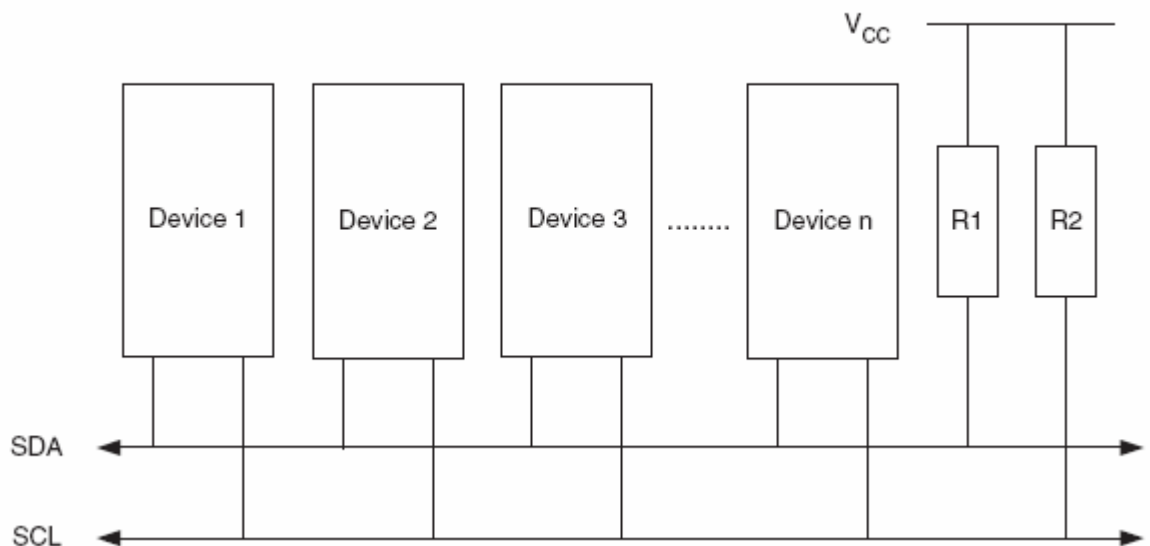
- I²C: hay IIC – Inter Integrated Circuit bus.
- TWI: Two Wire Interface.
- Kết nối đơn giản, bao gồm 2 dây SDA và SCL.
- Hỗ trợ hoạt động trong chế độ chủ (Master) hoặc tớ (Slave).
- Hoạt động 2 chiều truyền và nhận dữ liệu.
- Hỗ trợ Multi Master.
- 7bit địa chỉ cho phép kết nối 128 thiết bị với nhau.
- Tốc độ truyền dữ liệu tối đa: 400Khz.
- Địa chỉ Slave lập trình được.
- Tính năng đánh thức (wake-up) slave trong chế độ ngủ (sleep mode) khi nhận ra địa chỉ của mình.

(Từ lúc này, khái niệm được nhắc đến là I2C hay TWI xem như là một)

1.2. Định nghĩa TWI Bus – Two Wire Serial Interface Definition

Sơ đồ kết nối trên TWI bus – TWI Bus Interconnection

Figure 76. TWI Bus Interconnection



Thuật ngữ sử dụng - Terminology

- SCL là đường phát xung clock
- SDA là đường phát dữ liệu.
- Master: là thiết bị khởi động cho một quá trình truyền nhận, đồng thời là thiết bị phát xung clock trên SCL.
- Slave: là thiết bị được master gọi thông qua địa chỉ của nó
- Transmit: là quá trình thiết lập dữ liệu lên đường truyền

- Receive: là quá trình nhận dữ liệu trên đường truyền

Tính chất của các thiết bị trên TWI Bus

2 line (SDA và SCL) trên bus được kéo lên Vcc thông qua điện trở (Pull-Up resistor).

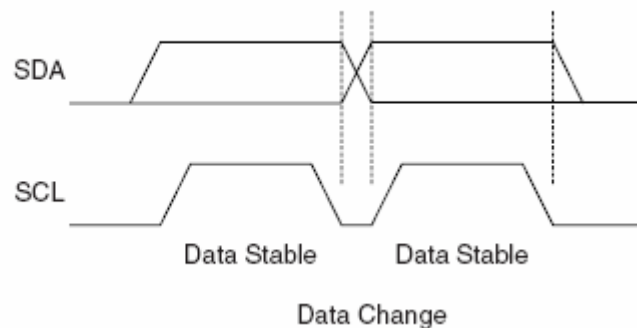
Ngõ ra nối với 2 line này trên TWI bus của tất cả các thiết bị tương thích chuẩn TWI đều là open-drain hay open collector (cực thu hở).

Điều này khiến cho hoạt động của bus TWI dựa trên tính chất AND, nghĩa là mức thấp trên bus line có thể do 1 hay nhiều thiết bị TWI thiết lập ngõ ra ở mức zero, còn mức cao trên bus line là khi tất cả các thiết bị TWI thiết lập ngõ ra của nó là tri-state (trạng thái nổi), cho phép điện trở kéo lên lúc này kéo bus line lên mức cao.

1.3. Truyền tải dữ liệu và định dạng khung truyền trên TWI Bus – Data Transfer and Frame Format

Data bits

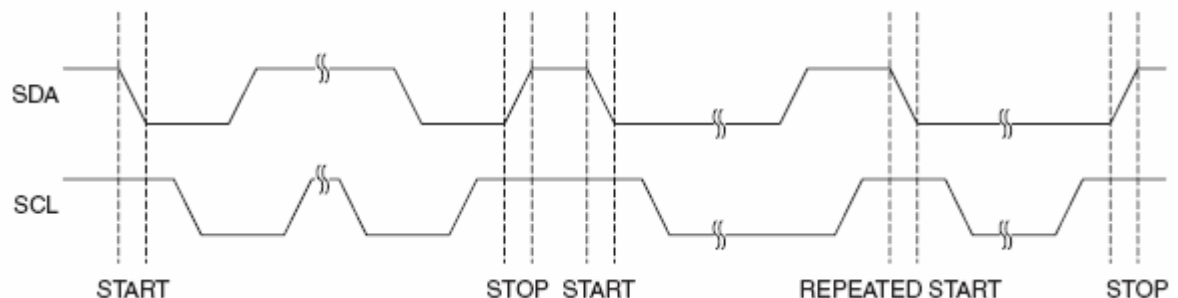
Figure 77. Data Validity



SDA được thiết lập/set (mức 0 hay 1) trước khi xung clock xuất hiện, nói cách khác, xung clock là căn cứ để đọc trạng thái của SDA trên đường truyền lúc đó (data validity).

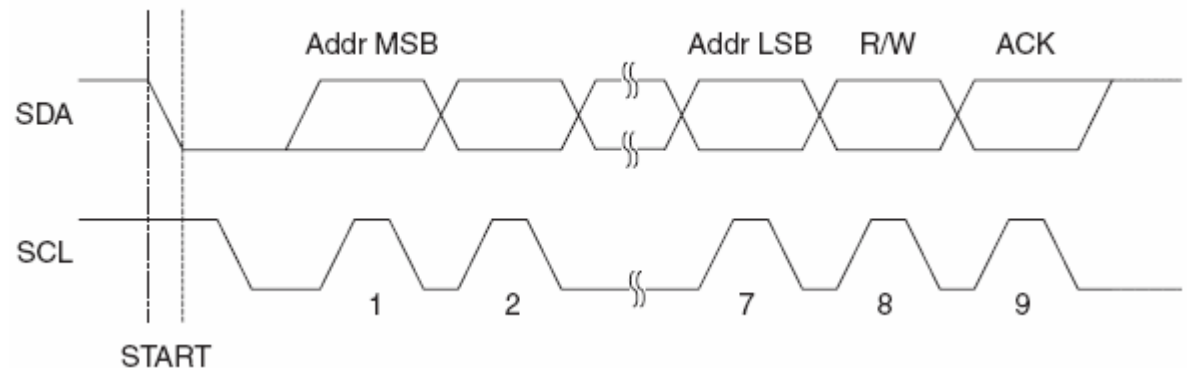
Điều kiện bắt đầu và kết thúc khung truyền – START and STOP condition

Figure 78. START, REPEATED START, and STOP Conditions



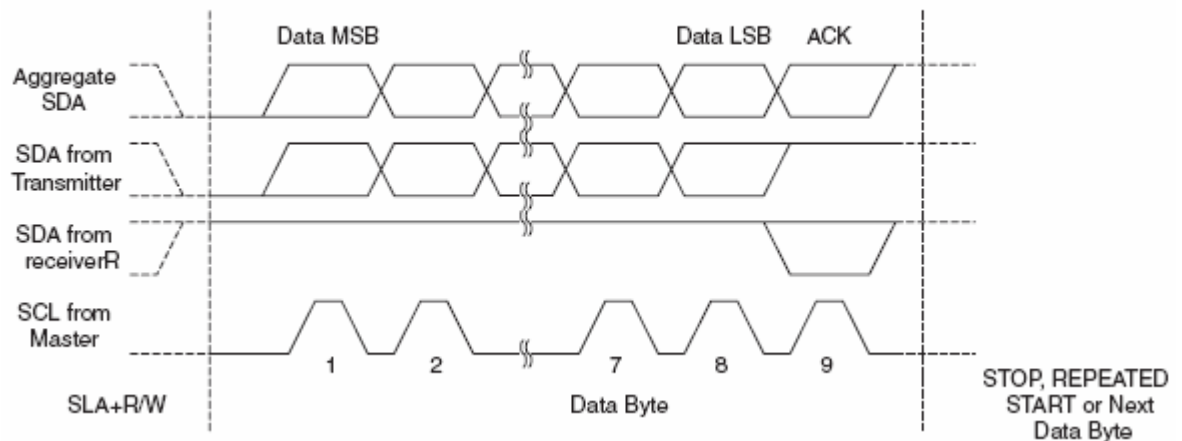
Điều kiện này xảy ra khi có sự thay đổi trạng thái SDA khi SCL đang ở mức cao. Cụ thể là:

- Start: sườn xuống SDA khi SCL mức cao
- Stop: sườn lên SDA khi SCL mức cao
- Repeat start: khi điều kiện start xảy ra tiếp theo sau một điều kiện start trước đó.

Định dạng gói địa chỉ - Address Packet Format**Figure 79. Address Packet Format**

Gói địa chỉ gửi đi sau khi xuất hiện điều kiện Start, việc này có tác dụng để các thiết bị slave nhận dạng đúng địa chỉ của mình để thực hiện những hành động tiếp theo.

- Bao gồm 9 bit
- 8 bit đầu bao gồm 7 bit địa chỉ và 1 bit $\overline{\text{READ/ WRITE}}$. Bit $\overline{\text{READ/ WRITE}}$ này sẽ báo cho thiết bị slave biết là nó chuẩn bị được “đọc” (tương ứng $\overline{\text{READ}}$, mức 1) hay “ghi” (tương ứng $\overline{\text{WRITE}}$, mức 0)
- Nếu 7bit address + $\overline{\text{READ}}$ thì kí hiệu là SLA+R, nếu là 7bit address + $\overline{\text{WRITE}}$ thì kí hiệu là SLA+W.
- Bit cuối cùng là ACK (Acknowledge), bit này sẽ do slave báo cho master biết bằng cách:
 - Master sẽ thả nổi SDA mức cao ở xung clock thứ 9 trên SCL.
 - Slave sẽ kéo SDA xuống thấp ở xung clock thứ 9 này trên SCL để báo cho master biết rằng nó đã nhận đúng địa chỉ của mình và sẵn sàng cho khung truyền tiếp theo.
 - Khi bit ACK không được slave nhận thì bit này trở thành NACK, lúc đó, SDA sẽ bị thả nổi mức cao trong xung clock thứ 9 trên SCL. Thiết bị master sẽ không thể biết được slave đã nhận được địa chỉ của mình hay chưa. Master sau đó sẽ khắc phục bằng cách phát đi điều kiện REPEAT START và gửi đi gói địa chỉ lần nữa...hoặc gửi tín hiệu STOP để kết thúc.
- Bit MSB sẽ gửi đi trước, địa chỉ thiết bị do người thiết kế quy định, nhưng địa chỉ 0000000 nên được để dành cho các lệnh gọi chung (General Call). Có nghĩa là nếu theo sau địa chỉ 0000000 là bit $\overline{\text{WRITE}}$ thì tất cả các slave trong hệ thống đều nhận dữ liệu từ master cùng lúc (nếu tất cả chúng đã kéo ACK xác nhận).
- Trường hợp còn lại là nếu theo sau địa chỉ 0000000 là bit $\overline{\text{READ}}$ sẽ gây ra sự tranh chấp dữ liệu.
- Các địa chỉ 1111xxx dành cho các mục đích khác.

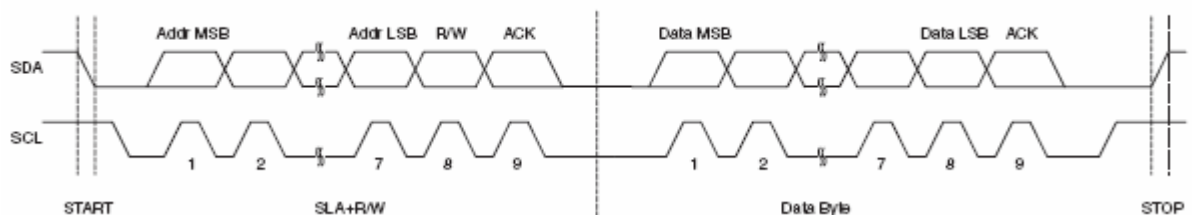
Định dạng gói dữ liệu – Data Packet Format**Figure 80. Data Packet Format**

Cũng bao gồm 9 bit, trong đó:

- 8bit dữ liệu
- 1 bit ACK
- Master vẫn tiếp tục sinh xung clock sau khi định hình dữ liệu trên SDA, đến xung clock thứ 9 thì thả nổi và chờ nhận bit ACK của slave xác nhận. Khi slave vì một lý do nào đó (như tràn dữ liệu, không nhận được thêm nữa), nó sẽ thả nổi bit ACK thành NACK.
- Bit MSB trong trường hợp này vẫn được chuyển đi trước.

Kết hợp gói địa chỉ và gói dữ liệu

Đóng gói khung truyền có thể là: Start → SLA+R/W → 1 hay nhiều data (theo sau là ACK) → Stop

Figure 81. Typical Data Transmission

Khi xảy ra điều kiện start và stop mà không có dữ liệu nào giữa hay điều kiện này, sẽ xảy ra lỗi.

Tính chất AND trên đường truyền bổ sung cho việc bắt tay giữa mater và slave. Slave có thể mở rộng (extend/stretch) xung SCL khi SCL đang ở chu kì mức thấp nếu clock sinh ra bởi master quá nhanh so với nó, hoặc để có thêm thời gian xử lý trong lúc truyền dữ liệu. Việc này không ảnh hưởng đến SCL khi SCL ở chu kì mức cao vì chu kì SCL mức cao do master sinh ra. Kết quả cuối cùng là slave có thể giảm tốc độ truyền dữ liệu trên TWI bus bằng cách nối dài chu kì xung SCL.

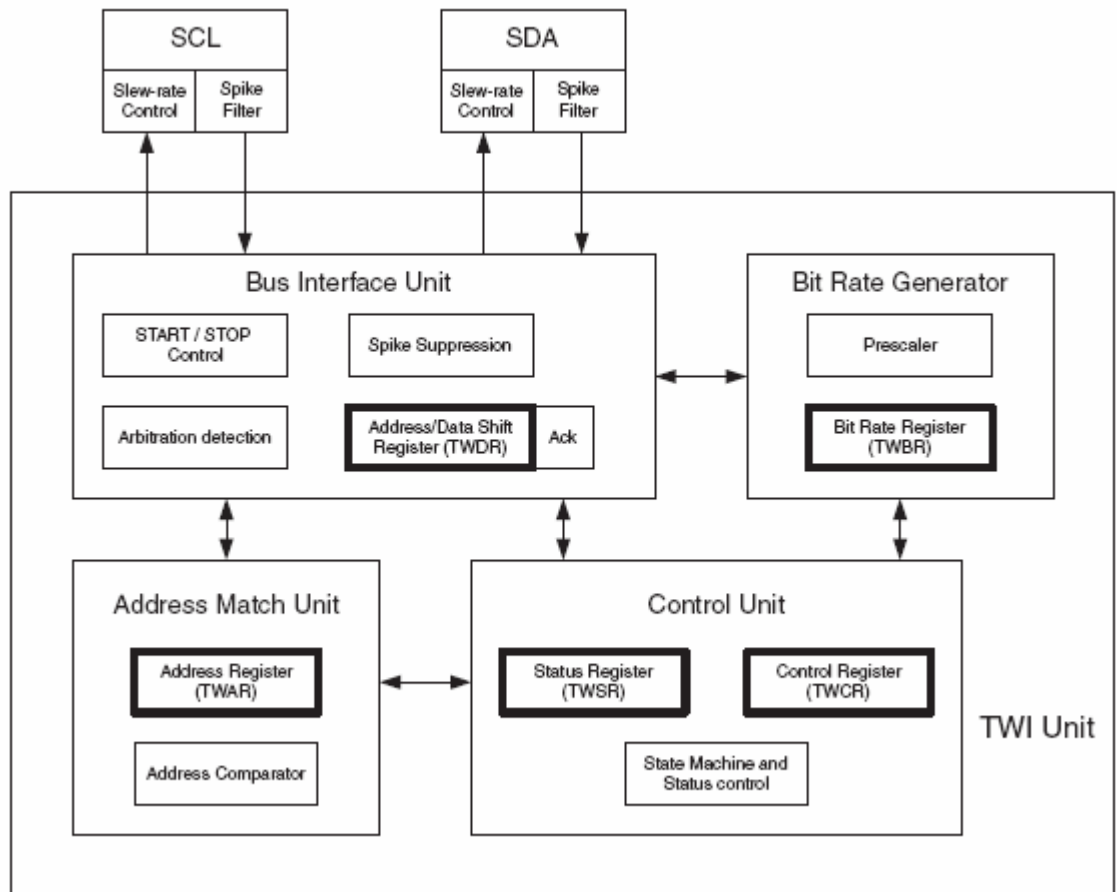
Số lượng byte truyền đi trong gói dữ liệu phụ thuộc vào phần mềm.

1.4. Hệ thống bus TWI đa chủ, tranh chấp và đồng bộ hóa – Multi Master Bus System, Arbitration and Synchronization

(* Tiếp tục cập nhật)

1.5. Tổng quan module TWI trong ATMEGA32

Figure 84. Overview of the TWI Module



Chân SDA và SCL

Là chân kết nối TWI trên ATMEGA32 với mạch điện bên ngoài, có hỗ trợ một số tính năng chống nhiễu.

Bộ phát xung – Bit rate Generator Unit

Phần tử quy định tốc độ phát xung, tính theo công thức:

$$\text{SCL frequency} = \frac{\text{CPU Clock frequency}}{16 + 2(\text{TWBR}) \cdot 4^{\text{TWPS}}}$$

Trong đó:

- TWBR là giá trị trong thanh ghi TWBR.
- TWPS là giá trị của 2 bit prescale trong thanh ghi trạng thái TWSR.

Bộ giao tiếp bus – Bus Interface Unit

(* Tiếp tục cập nhật)

Bộ so sánh địa chỉ - Address Match Unit

(* Tiếp tục cập nhật)

Bộ điều khiển

(* Tiếp tục cập nhật)

1.6. Mô tả thanh ghi TWI – TWI Register Definition

Thanh ghi điều khiển tốc độ xung SCL (TWBR – TWI bit rate register)

Bit	7	6	5	4	3	2	1	0	
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Thanh ghi điều khiển TWI (TWCR – TWI control register)

Bit	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit 7	TWINT – TWI interrupt	Bit này sẽ bị set khi TWI thực hiện xong một công việc nào đó, như phát xung điều kiện Start, Stop, truyền xong gói địa chỉ và nhận được ACK hoặc NACK, truyền xong gói dữ liệu và nhận được ACK hoặc NACK v.v...
Bit 6	TWEA – TWI enable acknowledge (ACK)	Bit cho phép phát xung ACK, sử dụng trong các trường hợp sau: Khi AVR (slave) nhận ra địa chỉ của mình. Khi AVR (slave) nhận được gói dữ liệu. Khi AVR (master) nhận xong gói dữ liệu từ slave
Bit 5	TWSTA – TWI start condition	Điều khiển việc phát đi điều kiện start
Bit 4	TWSTO – TWI stop condition	Điều khiển việc phát đi điều kiện stop.
Bit 3	TWWC – TWI write Collision flag	Cờ báo khi dữ liệu ghi vào TWDR khi cờ báo TWINT đang ở mức thấp
Bit 2	TWEN – TWI enable	Cho phép TWI hoạt động
Bit 1	Trống	
Bit 0	TWIE – TWI interrupt enable	Cho phép ngắt TWI

Thanh ghi trạng thái TWI (TWI Status Register – TWSR)

Bit	7	6	5	4	3	2	1	0	
	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	TWSR
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	1	1	1	1	1	0	0	0	

Bit 7	TWS7	5 bit này sẽ tạo thành code trạng thái trong quá trình diễn ra giao thức TWI Sẽ được nói rõ trong phần sau...
Bit 6	TWS6	
Bit 5	TWS5	

Bit 4	TWS4	
Bit 3	TWS3	
Bit 2	Trống	
Bit 1	TWPS1	2 bit prescale, ảnh hưởng đến việc thiết lập tốc độ (bitrate) TWI
Bit 0	TWPS2	

TWPS1	TWPS0	Giá trị prescale
0	0	1
0	1	4
1	0	16
1	1	64

Thanh ghi dữ liệu TWI (TWI Data Register – TWDR)

Bit	7	6	5	4	3	2	1	0	
	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0	TWDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	1	1	1	1	1	1	1	

Thanh ghi quy định địa chỉ slave (TWI (Slave) Address Register – TWAR)

Sử dụng khi AVR đóng vai trò là một slave trong bus TWI

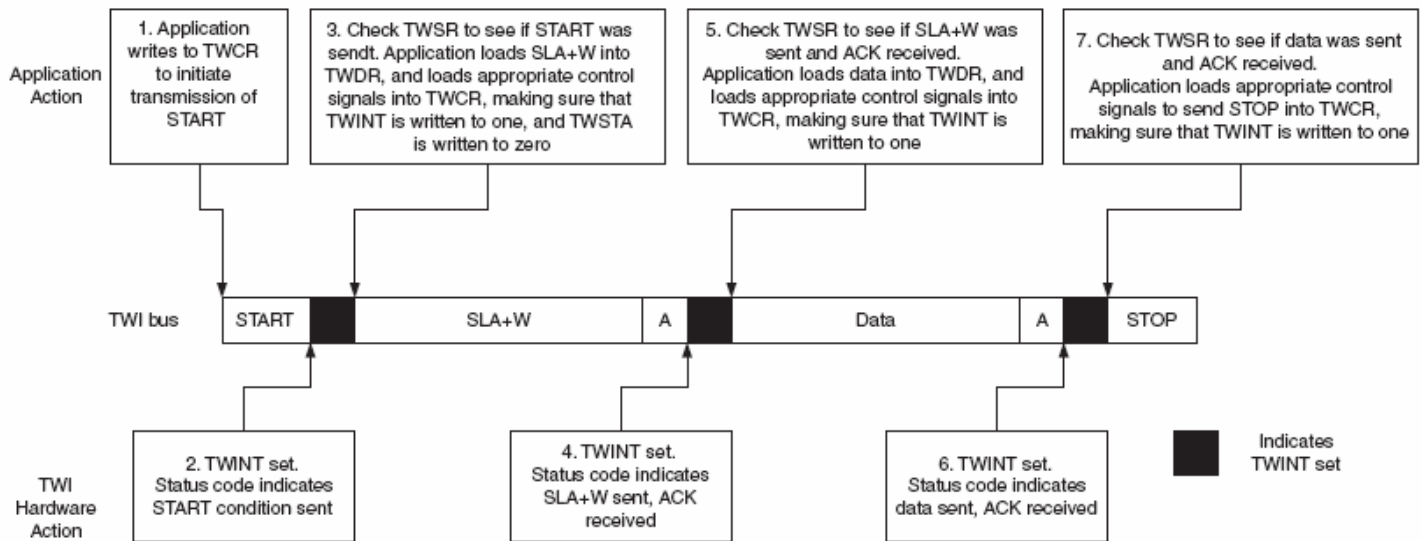
Bit	7	6	5	4	3	2	1	0	
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	TWAR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	1	1	1	1	1	1	0	

Bit 7 ..1	TWA – TWI (slave) address register	Quy định địa chỉ slave cho AVR
Bit 0	TWGCE - TWI General Call Recognition Enable	Cho phép nhận ra địa chỉ của cuộc gọi chung

1.7. Sử dụng TWI trong ATMEGA32

Lưu đồ sử dụng trong TWI đối với một quá trình truyền thông thường:

Figure 85. Interfacing the Application to the TWI in a Typical Transmission



Hoạt động TWI trên AVR dựa trên nền tảng định hướng byte (byte-oriented) và ngắt (interrupt).

Tất cả mọi sự kiện diễn ra trên bus đều gây ra ngắt, ví dụ như nhận xong 1 byte, hoặc phát hiện điều kiện START vừa diễn ra trên bus. Bởi vì dựa trên ngắt, nên trong quá trình truyền tải byte dữ liệu, MCU có thể rảnh rỗi để tiến hành hoạt động khác.

Cho phép ngắt TWI (dựa vào cờ ngắt – TWINT Flag) gây ra ngắt toàn cục bằng bit TWIE cùng với bit cho phép ngắt toàn cục (Global Interrupt Enable). Nếu sử dụng chế độ poll, cần thực hiện việc chờ cờ ngắt TWINT Flag này để phát hiện các sự kiện diễn ra trên bus.

Khi cờ ngắt TWINT được bật, nghĩa là TWI vừa hoàn tất một hoạt động nào đó và chờ sự phản hồi từ phần mềm. Trong trường hợp này, thanh ghi trạng thái TWSR chứa giá trị hiển thị trạng thái hiện tại trên TWI bus. Chương trình phần mềm sau đó quyết định hoạt động nào tiếp tục diễn ra cho chu kỳ TWI tiếp theo bằng cách thiết lập các giá trị phù hợp vào thanh ghi điều khiển TWCR và data TWDR.

TWI bus chia thành 2 hoạt động riêng biệt : phần mềm và phần cứng, ô tô đen biểu thị cho thời điểm cờ TWINT bị set.

- Đầu tiên, phần mềm sẽ ghi giá trị vào TWCR để bắt đầu phát điều kiện START.
- Phát xong điều kiện START, cờ TWINT bị set
- Phần mềm sau khi nhận thấy cờ TWINT bị set sẽ kiểm tra các thông số trong TWSR có phù hợp không, đảm bảo START đã được phát đi một cách hợp lệ. Sau đó, phần mềm sẽ load dữ liệu chứa tham số SLA+W (vì đây là một cuộc truyền dữ liệu) vào TWDR (xem như đệm dữ liệu trước khi phát đi) và load các thông số thích hợp vào TWCR, trong đó kể cả việc ghi giá trị “1” vào bit TWINT để cho phép dữ liệu truyền đi.
- Sau khi slave nhận dữ liệu xong và báo ACK, cờ TWINT tiếp tục bị set, cho thấy SLA+W đã truyền xong và ACK đã nhận được.
- Phần mềm lúc này sẽ kiểm tra thông số trong TWSR để kiểm tra chắc chắn rằng việc phát SLA+W đã được thực hiện và bit ACK nhận xong. Tiếp đó, phần mềm load dữ liệu vào thanh ghi TWDR, đây chính là dữ liệu cần truyền cho slave, tiếp tục load thông số thích hợp vào TWCR cho phép dữ liệu trong TWDR truyền đi.
- Sau khi truyền xong dữ liệu và nhận được bit ACK, cờ TWINT bị set tiếp.
- Phần mềm sẽ kiểm tra thanh ghi TWSR để đảm bảo quá trình vừa rồi thực hiện tốt. nó sẽ phát đi điều kiện STOP bằng cách ghi giá trị thích hợp vào thanh ghi TWCR.

Lưu ý: khi cờ TWINT bị set, để quá trình tiếp theo được thực hiện, phải ghi giá trị “1” vào bit TWINT này.

1.8. Các chế độ truyền dữ liệu của TWI trên ATMEGA32

TWI trên ATMEGA32 có thể hoạt động trên 1 trong 4 chế độ: Master Transmitter (MT), Master Receiver (MR), Slave Transmitter (ST) và Slave Receiver (SR).

Một số chú giải kí hiệu sử dụng cho các hình bên dưới như sau:

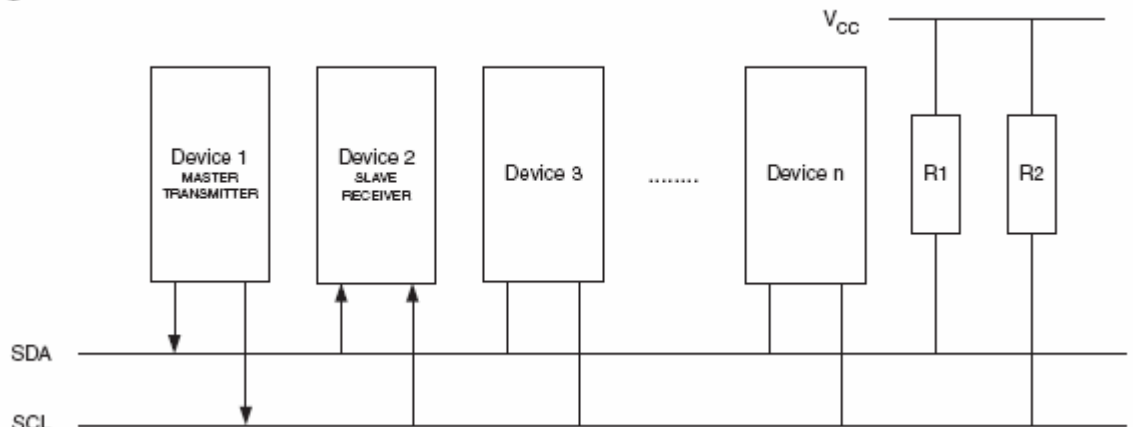
- S: START condition.
- Rs: REPEAT START condition.
- R: Read bit (high level at SDA).
- W: Write bit (low level at SDA).
- A: Acknowledge bit (low level at SDA).
- \bar{A} : Not Acknowledge bit (high level at SDA).
- Data: 8-bit data byte.
- P: STOP condition.
- SLA: Slave Address.

1.8.1. Chế độ MASTER TRANSMIT

Trong chế độ này, một số lượng data byte được truyền tải tới Slave. Để truy nhập vào chế độ Master, điều kiện START nhất thiết phải được truyền đi.

Định dạng gói địa chỉ gửi đi sẽ quy định Master trong chế độ nào (truyền/nhận). Nếu SLA+W được truyền đi, nghĩa là sẽ hoạt động bên chế độ MT.

Figure 86. Data Transfer in Master Transmitter Mode



Điều kiện START truyền đi khi ghi giá trị tương ứng sau vào thanh ghi TWCR:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
Value	1	X	1	0	X	1	0	X

Giải thích:

- TWEN phải được set để cho phép module TWI.
- TWSTA phải được set để phát đi điều kiện START.
- TWINT phải ghi vào “1” để xóa cờ ngắt này trước đó.
- Bộ TWI sẽ kiểm tra bus TWI và phát đi điều kiện START ngay khi bus free.
- Sau khi phát thành công điều kiện START, cờ TWINT sẽ set lên 1 (bởi phần cứng) và mã trạng thái (status code) trong TWSR sẽ là \$08 (xem thêm trong bảng 74).

Sau đó, vì là chế độ MASTER TRANSMITTER, để truyền gói địa chỉ đi (SLA+ \overline{W}) phải load giá trị địa chỉ này vào thanh ghi TWDR, (lưu ý là làm việc này trong khi TWINT đang ở mức cao, nếu không sẽ xảy ra tình trạng dữ liệu chưa được ghi hết vào bộ đệm đã truyền đi, điều này làm TWWC bị set), rồi ghi các giá trị sau vào thanh ghi TWCR:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	0	0	X	1	0	X

Sau khi truyền xong gói SLA+W và nhận được bit ACK, cờ TWINT set lần nữa và các mã trạng thái trả về có thể là \$18, \$20 và \$38.

Sau khi gói SLA+W đã truyền thành công, gói data phải được truyền tiếp theo. Giá trị DATA phải được load vào TWDR và ghi giá trị sau vào thanh ghi TWCR:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	0	0	X	1	0	X

Thao tác truyền gói DATA sẽ được lặp đi lặp lại cho đến byte cuối cùng được truyền xong. Việc truyền dữ liệu sẽ kết thúc bằng điều kiện STOP hoặc REPEAT START. Điều kiện STOP sẽ truyền đi khi các giá trị này được ghi vào TWCR như sau:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	0	1	X	1	0	X

Hoặc điều kiện REPEAT START truyền đi với các giá trị tương ứng như sau:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	1	0	X	1	0	X

Sau khi điều kiện REPEAT START được truyền xong (State \$10), TWI có thể truy xuất đến Slave hiện tại hay Slave khác mà không cần phải phát đi điều kiện STOP. REPEAT START cho phép Master chuyển đổi giữa các Slave, chuyển đổi giữa chế độ transmitter và receiver mà không mất quyền điều khiển trên bus.

Bảng mã thông số trạng thái trả về

Như đã nói như trên, bit thứ 7 → 3 trong thanh ghi TWSR hiển thị thông số liên quan đến việc truyền nhận trên TWI bus.

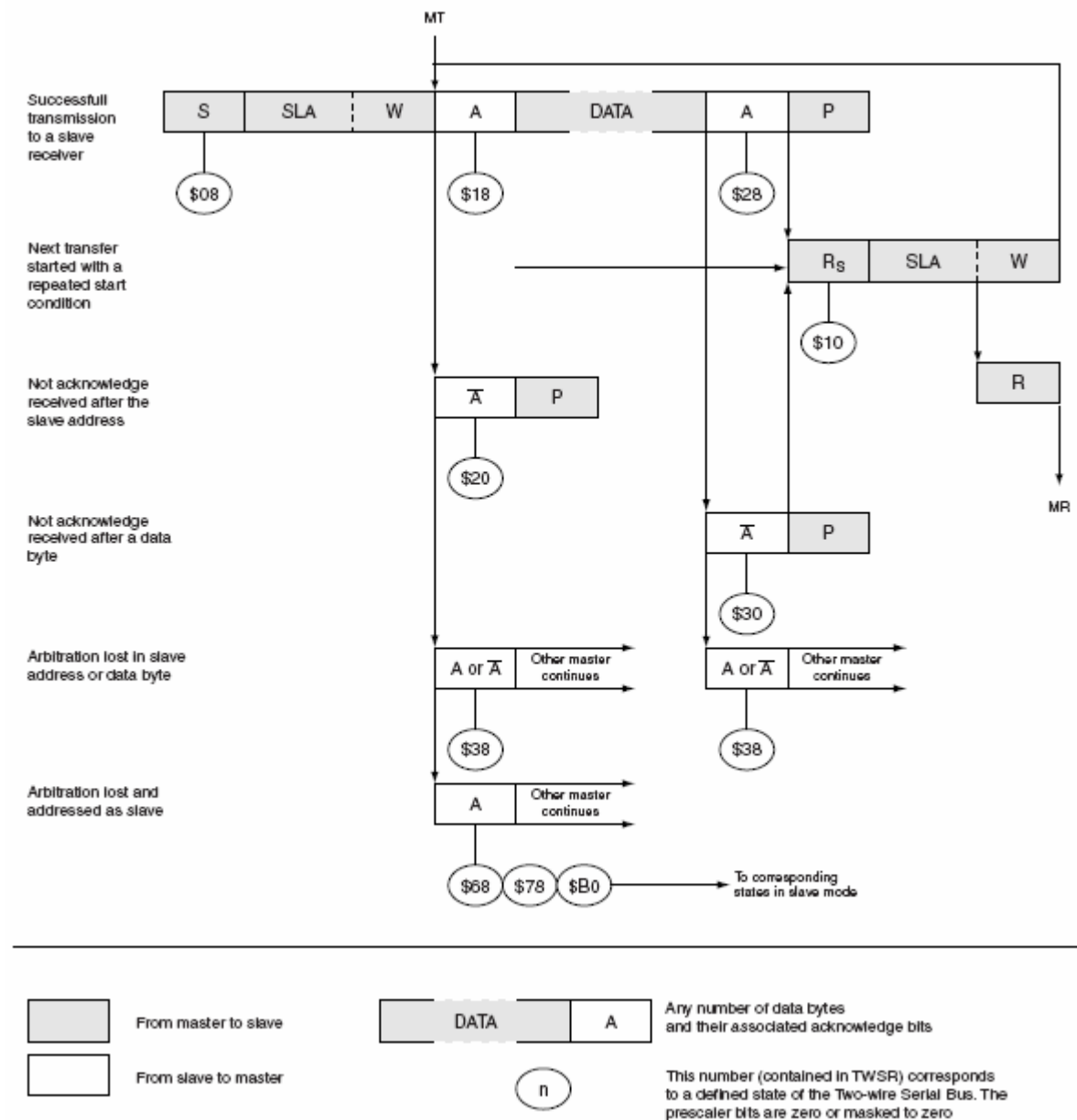
Bởi vì trong thanh ghi TWSR còn có 2 bit prescale, nên để đọc được trạng thái TWI trong thanh ghi này, phải che 2bit prescale này lại (mask) và các thông số trạng thái quy định trong bảng tương ứng với trường hợp 2 bit prescale = 0

Table 74. Status Codes for Master Transmitter Mode

Status Code (TWSR) Prescaler Bits are 0	Status of the Two-wire Serial Bus and Two-wire Serial Interface Hardware	Application Software Response					Next Action Taken by TWI Hardware
		To/from TWDR	To TWCR				
			STA	STO	TWINT	TWEA	
\$08	A START condition has been transmitted	Load SLA+W	0	0	1	X	SLA+W will be transmitted; ACK or NOT ACK will be received
\$10	A repeated START condition has been transmitted	Load SLA+W or	0	0	1	X	SLA+W will be transmitted; ACK or NOT ACK will be received SLA+R will be transmitted; Logic will switch to Master Receiver mode
		Load SLA+R	0	0	1	X	
\$18	SLA+W has been transmitted; ACK has been received	Load data byte or	0	0	1	X	Data byte will be transmitted and ACK or NOT ACK will be received Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be Reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be Reset
		No TWDR action or	1	0	1	X	
		No TWDR action or	0	1	1	X	
		No TWDR action	1	1	1	X	
\$20	SLA+W has been transmitted; NOT ACK has been received	Load data byte or	0	0	1	X	Data byte will be transmitted and ACK or NOT ACK will be received Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be reset
		No TWDR action or	1	0	1	X	
		No TWDR action or	0	1	1	X	
		No TWDR action	1	1	1	X	
\$28	Data byte has been transmitted; ACK has been received	Load data byte or	0	0	1	X	Data byte will be transmitted and ACK or NOT ACK will be received Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be reset
		No TWDR action or	1	0	1	X	
		No TWDR action or	0	1	1	X	
		No TWDR action	1	1	1	X	
\$30	Data byte has been transmitted; NOT ACK has been received	Load data byte or	0	0	1	X	Data byte will be transmitted and ACK or NOT ACK will be received Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be reset
		No TWDR action or	1	0	1	X	
		No TWDR action or	0	1	1	X	
		No TWDR action	1	1	1	X	
\$38	Arbitration lost in SLA+W or data bytes	No TWDR action or	0	0	1	X	Two-wire Serial Bus will be released and not addressed slave mode entered A START condition will be transmitted when the bus becomes free
		No TWDR action	1	0	1	X	

Định dạng và trạng thái TWI có thể trong chế độ Master Transmitter

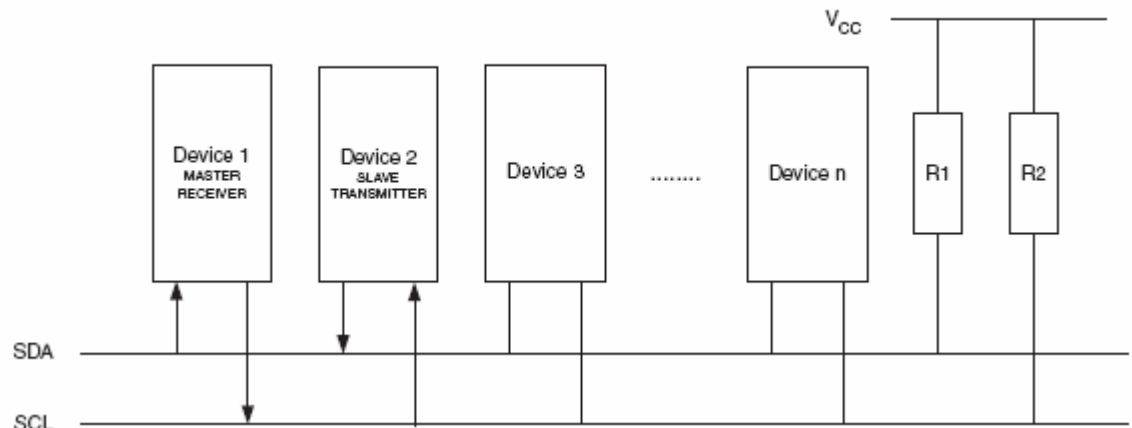
Figure 87. Formats and States in the Master Transmitter Mode



1.8.2. Chế độ MASTER RECEIVE

Trong chế độ này, một số lượng data byte có thể nhận từ Slave. Để truy nhập vào chế độ Master, điều kiện START nhất thiết phải được truyền đi.

Định dạng gói địa chỉ gửi đi sẽ quy định Master trong chế độ nào (truyền/nhận). Nếu SLA+R được truyền đi, nghĩa là sẽ hoạt động bên chế độ MR.

Figure 88. Data Transfer in Master Receiver Mode

Điều kiện START truyền đi khi ghi các giá trị tương ứng bên dưới vào TWCR:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	1	0	X	1	0	X

Giải thích:

- TWEN phải được set để cho phép module TWI.
- TWSTA phải được set để phát đi điều kiện START.
- TWINT phải ghi vào “1” để xóa cờ ngắt này trước đó.
- Bộ TWI sẽ kiểm tra bus TWI và phát đi điều kiện START ngay khi bus free.

Sau khi phát thành công điều kiện START, cờ TWINT sẽ set lên 1 (bởi phần cứng) và mã trạng thái (status code) trong TWSR sẽ là \$08 (xem thêm trong bảng 74).

Vì là chế độ MASTER RECEIVE, nên phải load gói địa chỉ SLA+R vào thanh ghi TWDR, sau đó cho phép truyền đi bằng cách ghi các giá trị sau vào TWCR:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	0	0	X	1	0	X

Sau khi SLA+R truyền xong và bit ACK nhận được, cờ TWINT sẽ set lên 1 lần nữa và mã trạng thái trả về trong TWSR có thể là \$38, \$40 hoặc \$48.

Để bắt đầu nhận data từ Slave và xác nhận - gửi ACK cho Slave – sau khi nhận được byte đó, cờ TWINT phải được xóa bằng cách ghi “1” và bit TWEA phải được set lên 1. Cờ TWINT sẽ được set lên 1 (bằng phần cứng) sau khi nhận xong 1 byte từ Slave, giá trị nhận được từ Slave được đọc ra từ thanh ghi TWDR. Quá trình đọc dữ liệu từ Slave sẽ được lặp lại như trên cho đến khi byte cuối cùng nhận xong, Master khi đó phải thông báo cho Slave bằng cách gửi bit NACK (xóa bit TWEA trước khi nhận byte tiếp theo).

Quá trình nhận sẽ kết thúc bằng cách gửi đi điều kiện STOP hoặc REPEAT START. Điều kiện STOP sẽ truyền đi khi các giá trị này được ghi vào TWCR như sau:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	0	1	X	1	0	X

Hoặc điều kiện REPEAT START truyền đi với các giá trị tương ứng như sau:

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	1	X	1	0	X	1	0	X

Sau khi điều kiện REPEAT START được truyền xong (State \$10), TWI có thể truy xuất đến Slave hiện tại hay Slave khác mà không cần phải phát đi điều kiện STOP. REPEAT START cho phép Master chuyển đổi giữa các Slave, chuyển đổi giữa chế độ transmitter và receiver mà không mất quyền điều khiển trên bus.

Bảng mã thông số trạng thái trả về

Như đã nói như trên, bit thứ 7 → 3 trong thanh ghi TWSR hiển thị thông số liên quan đến việc truyền nhận trên TWI bus.

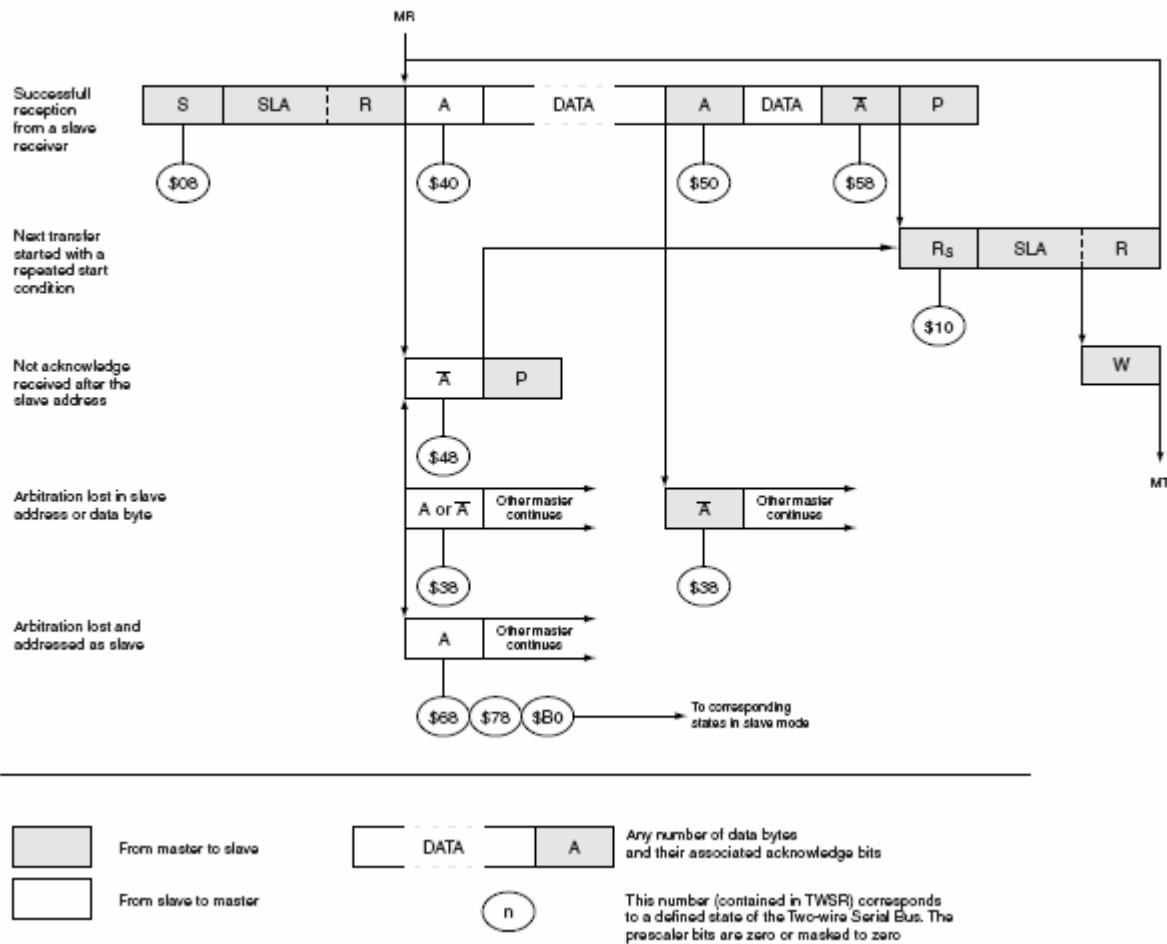
Bởi vì trong thanh ghi TWSR còn có 2 bit prescale, nên để đọc được trạng thái TWI trong thanh ghi này, phải che 2bit prescale này lại (mask) và các thông số trạng thái quy định trong bảng tương ứng với trường hợp 2 bit prescale = 0

Table 75. Status Codes for Master Receiver Mode

Status Code (TWSR) Prescaler Bits are 0	Status of the Two-wire Serial Bus and Two-wire Serial Inter- face Hardware	Application Software Response					Next Action Taken by TWI Hardware
		To/from TWDR	To TWCR				
			STA	STO	TWINT	TWEA	
\$08	A START condition has been transmitted	Load SLA+R	0	0	1	X	SLA+R will be transmitted ACK or NOT ACK will be received
\$10	A repeated START condition has been transmitted	Load SLA+R or	0	0	1	X	SLA+R will be transmitted ACK or NOT ACK will be received SLA+W will be transmitted Logic will switch to masTer Transmitter mode
		Load SLA+W	0	0	1	X	
\$38	Arbitration lost in SLA+R or NOT ACK bit	No TWDR action or	0	0	1	X	Two-wire Serial Bus will be released and not addressed slave mode will be entered A START condition will be transmitted when the bus becomes free
		No TWDR action	1	0	1	X	
\$40	SLA+R has been transmitted; ACK has been received	No TWDR action or	0	0	1	0	Data byte will be received and NOT ACK will be returned Data byte will be received and ACK will be returned
		No TWDR action	0	0	1	1	
\$48	SLA+R has been transmitted; NOT ACK has been received	No TWDR action or	1	0	1	X	Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be reset
		No TWDR action or	0	1	1	X	
		No TWDR action	1	1	1	X	
\$50	Data byte has been received; ACK has been returned	Read data byte or	0	0	1	0	Data byte will be received and NOT ACK will be returned Data byte will be received and ACK will be returned
		Read data byte	0	0	1	1	
\$58	Data byte has been received; NOT ACK has been returned	Read data byte or	1	0	1	X	Repeated START will be transmitted STOP condition will be transmitted and TWSTO Flag will be reset STOP condition followed by a START condition will be transmitted and TWSTO Flag will be reset
		Read data byte or	0	1	1	X	
		Read data byte	1	1	1	X	

Định dạng và trạng thái TWI có thể trong chế độ Master Receiver

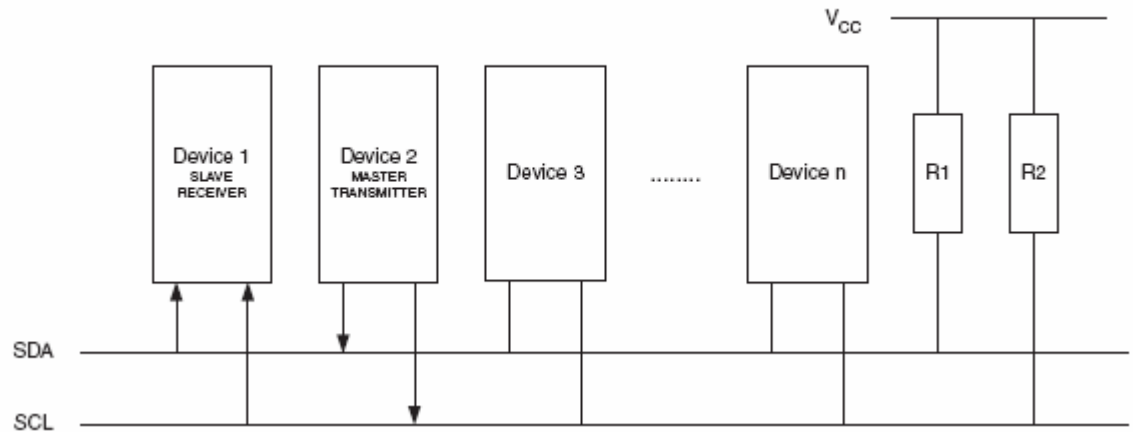
Figure 89. Formats and States in the Master Receiver Mode



1.8.3. Chế độ SLAVE RECEIVER

Trong chế độ này, một số lượng data bytes được nhận từ master khác truyền tới trên TWI bus.

Figure 90. Data Transfer in Slave Receiver Mode



Để khởi tạo chế độ Slave cho ATMEGA, phải ghi các giá trị tương ứng sau vào thanh ghi TWAR và TWCR.

TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Value	Device's Own Slave Address							

Địa chỉ Slave được định nghĩa bằng 7bit cao trong TWAR, bit TWGCE để cho phép/cấm (1/0) tính năng nhận cuộc gọi chung từ master (General Call Enable – địa chỉ 0x00).

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	0	1	0	0	0	1	0	X

Bit TWEN cho phép module TWI hoạt động, bit TWEA cho phép xác nhận (ACK) khi địa chỉ thiết bị master đang truyền đến trùng khớp với địa chỉ slave đã định nghĩa trong thanh ghi TWAR.

Sau khi thanh ghi TWCR và TWAR đã khởi tạo xong, module TWI sẽ chờ đến khi khung truyền nào trên TWI bus có địa chỉ slave thiết bị trùng khớp với địa chỉ slave của nó thì cờ trạng thái TWINT sẽ set lên 1 và mã trạng thái trả về khi đó lưu trong thanh ghi TWSR. Nếu theo sau địa chỉ thiết bị là bit W thì module TWI tiếp tục hoạt động trong chế độ Slave Receive (SR), ngược lại, theo sau địa chỉ thiết bị là bit R thì module TWI tiếp tục hoạt động trong chế độ Slave Transmit (ST).

Module TWI cũng có thể chuyển sang chế độ SR khi module TWI đang trong chế độ master bị mất khả năng tranh chấp (Lost Arbitration).

Trong quá trình truyền nhận, bit TWEA nếu bị reset về 0, module TWI sẽ báo NACK sau lần nhận byte tiếp theo. Tính năng này có thể sử dụng để thông báo với master rằng slave không thể tiếp tục nhận thêm data nữa. Tuy nhiên, module TWI vẫn giám sát mọi hoạt động diễn ra trên bus và khả năng xác nhận (ACK) vẫn có thể tiếp tục bất kì khi nào bit TWEA được set trở lại. Như vậy, ngầm định rằng bit TWEA sử dụng để cách ly tạm thời module TWI với TWI bus.

Trong chế độ ngủ (Sleep mode), nguồn clock nội cấp cho module TWI bị ngắt. Tuy nhiên, khi bit TWEA được set lên 1, module TWI vẫn có khả năng giám sát TWI bus và vẫn có khả năng xác nhận (ACK) khi khung truyền có địa chỉ thiết bị trùng với địa chỉ slave của module TWI. Module TWI khi đó sẽ kéo đường SCL xuống thấp (stretch), đánh thức MCU và việc module TWI kéo dài chu kỳ mức thấp của đường SCL sẽ tiếp tục cho đến khi nào cờ TWINT bị xóa bằng phần mềm (ghi 1). Tính năng này có tác dụng tạm thời trì hoãn các phiên truyền nhận data tiếp theo từ master do thời gian đánh thức MCU có thể diễn ra lâu.

Lưu ý rằng thanh ghi data TWDR sẽ không chứa nội dung của byte vừa nhận trước đó sau khi bị đánh thức từ chế độ ngủ.

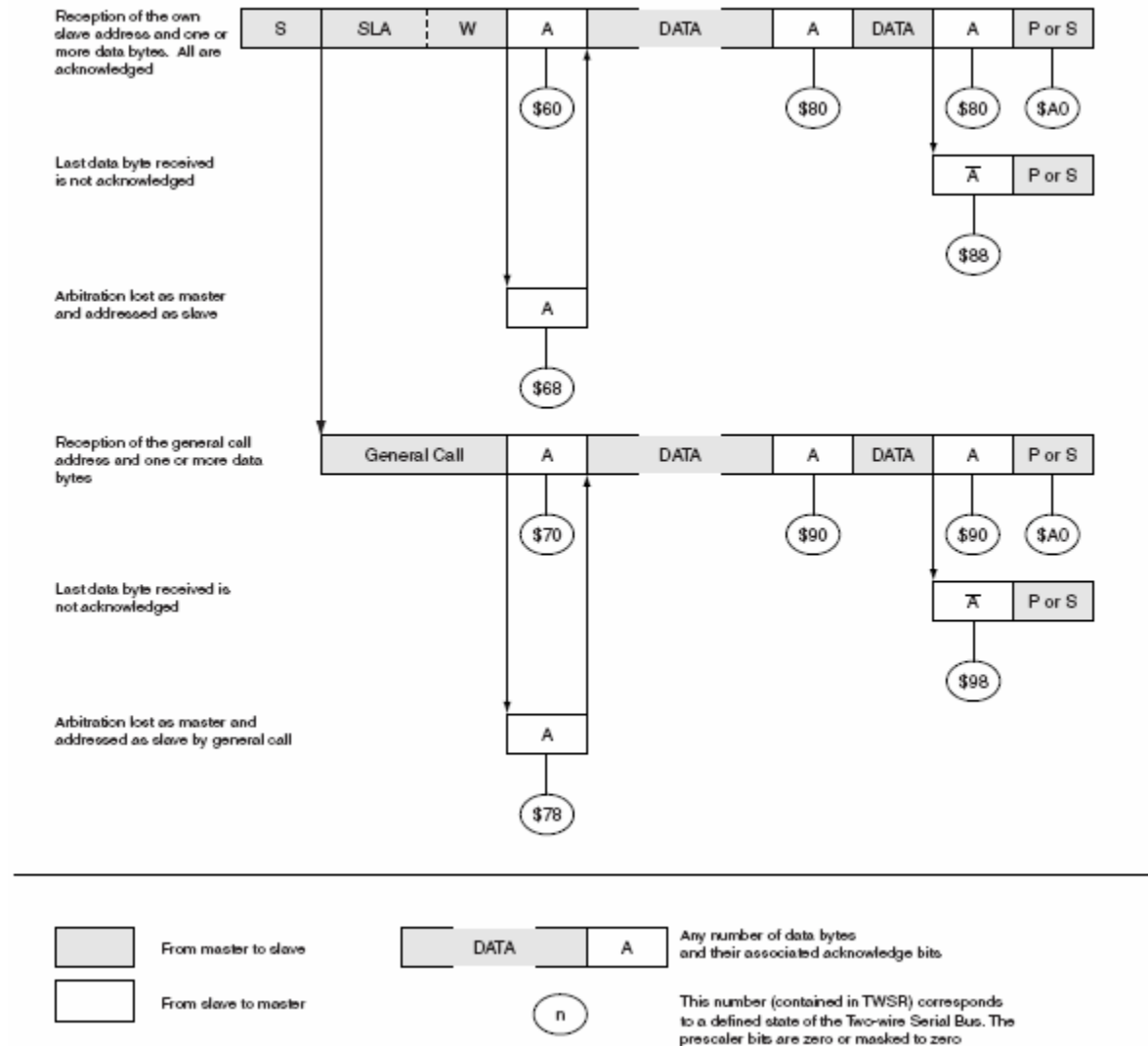
Bảng mã trạng thái trong chế độ SR

Table 76. Status Codes for Slave Receiver Mode

Status Code (TWSR) Prescaler Bits are 0	Status of the Two-wire Serial Bus and Two-wire Serial Interface Hardware	Application Software Response					Next Action Taken by TWI Hardware
		To/from TWDR	To TWCR				
			STA	STO	TWINT	TWEA	
\$60	Own SLA+W has been received; ACK has been returned	No TWDR action or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		No TWDR action	X	0	1	1	Data byte will be received and ACK will be returned
\$68	Arbitration lost in SLA+R/W as master; own SLA+W has been received; ACK has been returned	No TWDR action or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		No TWDR action	X	0	1	1	Data byte will be received and ACK will be returned
\$70	General call address has been received; ACK has been returned	No TWDR action or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		No TWDR action	X	0	1	1	Data byte will be received and ACK will be returned
\$78	Arbitration lost in SLA+R/W as master; General call address has been received; ACK has been returned	No TWDR action or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		No TWDR action	X	0	1	1	Data byte will be received and ACK will be returned
\$80	Previously addressed with own SLA+W; data has been received; ACK has been returned	Read data byte or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		Read data byte	X	0	1	1	Data byte will be received and ACK will be returned
\$88	Previously addressed with own SLA+W; data has been received; NOT ACK has been returned	Read data byte or	0	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA
		Read data byte or	0	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"
		Read data byte or	1	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA; a START condition will be transmitted when the bus becomes free
		Read data byte	1	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"; a START condition will be transmitted when the bus becomes free
\$90	Previously addressed with general call; data has been received; ACK has been returned	Read data byte or	X	0	1	0	Data byte will be received and NOT ACK will be returned
		Read data byte	X	0	1	1	Data byte will be received and ACK will be returned
\$98	Previously addressed with general call; data has been received; NOT ACK has been returned	Read data byte or	0	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA
		Read data byte or	0	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"
		Read data byte or	1	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA; a START condition will be transmitted when the bus becomes free
		Read data byte	1	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"; a START condition will be transmitted when the bus becomes free
\$A0	A STOP condition or repeated START condition has been received while still addressed as slave	No action	0	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA
			0	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"
			1	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA; a START condition will be transmitted when the bus becomes free
			1	0	1	1	Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"; a START condition will be transmitted when the bus becomes free

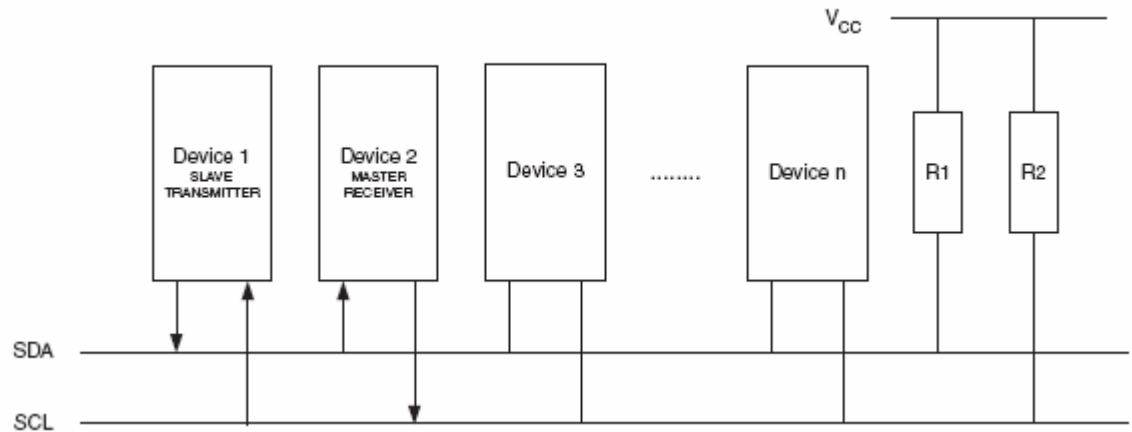
Định dạng và các trạng thái có thể có trong chế độ SR

Figure 91. Formats and States in the Slave Receiver Mode



1.8.4. Chế độ SLAVE TRANSMITTER

Trong chế độ này, một số lượng data bytes sẽ được truyền tới master.

Figure 92. Data Transfer in Slave Transmitter Mode

Để khởi tạo chế độ Slave cho ATMEGA, phải ghi các giá trị tương ứng sau vào thanh ghi TWAR và TWCR (* *Tương tự chế độ SR*)

TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Value	Device's Own Slave Address							

Địa chỉ Slave được định nghĩa bằng 7bit cao trong TWAR, bit TWGCE để cho phép/cấm (1/0) tính năng nhận cuộc gọi chung từ master (General Call Enable – địa chỉ 0x00).

TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
Value	0	1	0	0	0	1	0	X

Bit TWEN cho phép module TWI hoạt động, bit TWEA cho phép xác nhận (ACK) khi địa chỉ thiết bị master đang truyền đến trùng khớp với địa chỉ slave đã định nghĩa trong thanh ghi TWAR (* *Tương tự chế độ SR*).

Sau khi thanh ghi TWCR và TWAR đã khởi tạo xong, module TWI sẽ chờ đến khi khung truyền nào trên TWI bus có địa chỉ slave thiết bị trùng khớp với địa chỉ slave của nó thì cờ trạng thái TWINT sẽ set lên 1 và mã trạng thái trả về khi đó lưu trong thanh ghi TWSR. Nếu theo sau địa chỉ thiết bị là bit W thì module TWI tiếp tục hoạt động trong chế độ Slave Receive (SR), ngược lại, theo sau địa chỉ thiết bị là bit R thì module TWI tiếp tục hoạt động trong chế độ Slave Transmit (ST) (* *Tương tự chế độ SR*).

Module TWI cũng có thể chuyển sang chế độ ST khi module TWI đang trong chế độ master bị mất khả năng tranh chấp (Lost Arbitration) (* *Tương tự chế độ SR*).

Nếu bit TWEA bị xóa trong quá trình truyền nhận data, module TWI sẽ truyền đi byte cuối cùng, việc đó tùy thuộc vào khi nào master xác nhận (ACK) hay không (NACK) sau byte cuối cùng đó. Khi đó, module TWI sẽ chuyển sang chế độ slave không được định địa chỉ (not addressed slave mode), và sẽ bỏ qua tất cả việc truyền data byte tiếp theo nếu master tiếp tục. Dĩ nhiên là master khi đó chỉ nhận được data có toàn bộ các bit đều là “1”. Trạng thái \$C8 (xem Định dạng và các mã trạng thái có thể có trong chế độ ST) sẽ chuyển sang khi Slave đã truyền xong byte cuối cùng mà master vẫn yêu cầu thêm data (bằng cách xác nhận ACK).

Khi bit TWEA bằng 0, module TWI sẽ không phản hồi ACK khi nhận được địa chỉ slave của chính nó. Tuy nhiên, module TWI vẫn giám sát mọi hoạt động diễn ra trên bus và khả năng xác nhận (ACK) vẫn có thể tiếp tục bất kì khi nào bit TWEA được set trở lại. Như vậy, ngầm định rằng bit TWEA sử dụng để cách ly tạm thời module TWI với TWI bus (* *Tương tự chế độ SR*).

Trong chế độ ngủ (Sleep mode), nguồn clock nội cấp cho module TWI bị ngắt. Tuy nhiên, khi bit TWEA được set lên 1, module TWI vẫn có khả năng giám sát TWI bus và vẫn có khả năng xác nhận (ACK) khi khung truyền có địa chỉ thiết bị trùng với địa chỉ slave của module TWI. Module TWI khi đó sẽ kéo đường SCL xuống thấp (stretch), đánh thức MCU và việc module TWI kéo dài chu kỳ mức

thấp của đường SCL sẽ tiếp tục cho đến khi nào cờ TWINT bị xóa bằng phần mềm (ghi 1). Tính năng này có tác dụng tạm thời hoãn các phiên truyền nhận data tiếp theo từ master do thời gian đánh thức MCU có thể diễn ra lâu (* *Tương tự chế độ SR*).

Lưu ý rằng thanh ghi data TWDR sẽ không chứa nội dung của byte vừa nhận trước đó sau khi bị đánh thức từ chế độ ngủ (* *Tương tự chế độ SR*).

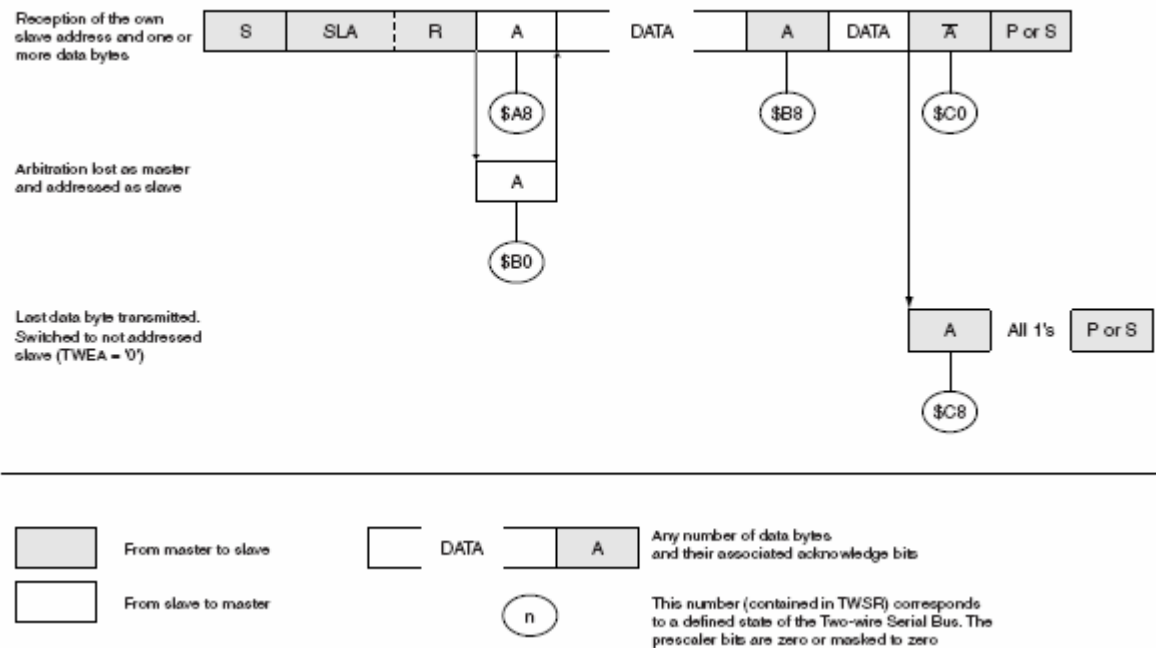
Mã trạng thái trong chế độ ST

Table 77. Status Codes for Slave Transmitter Mode

Status Code (TWSR) Prescaler Bits are 0	Status of the Two-wire Serial Bus and Two-wire Serial Interface Hardware	Application Software Response					Next Action Taken by TWI Hardware
		To/from TWDR	To TWCR				
			STA	STO	TWINT	TWEA	
\$A8	Own SLA+R has been received; ACK has been returned	Load data byte or	X	0	1	0	Last data byte will be transmitted and NOT ACK should be received Data byte will be transmitted and ACK should be received
		Load data byte	X	0	1	1	
\$B0	Arbitration lost in SLA+R/W as master; own SLA+R has been received; ACK has been returned	Load data byte or	X	0	1	0	Last data byte will be transmitted and NOT ACK should be received Data byte will be transmitted and ACK should be received
		Load data byte	X	0	1	1	
\$B8	Data byte in TWDR has been transmitted; ACK has been received	Load data byte or	X	0	1	0	Last data byte will be transmitted and NOT ACK should be received Data byte will be transmitted and ACK should be received
		Load data byte	X	0	1	1	
\$C0	Data byte in TWDR has been transmitted; NOT ACK has been received	No TWDR action or	0	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1" Switched to the not addressed Slave mode; no recognition of own SLA or GCA; a START condition will be transmitted when the bus becomes free Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"; a START condition will be transmitted when the bus becomes free
		No TWDR action or	0	0	1	1	
		No TWDR action or	1	0	1	0	
		No TWDR action	1	0	1	1	
\$C8	Last data byte in TWDR has been transmitted (TWEA = "0"); ACK has been received	No TWDR action or	0	0	1	0	Switched to the not addressed Slave mode; no recognition of own SLA or GCA Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1" Switched to the not addressed Slave mode; no recognition of own SLA or GCA; a START condition will be transmitted when the bus becomes free Switched to the not addressed Slave mode; own SLA will be recognized; GCA will be recognized if TWGCE = "1"; a START condition will be transmitted when the bus becomes free
		No TWDR action or	0	0	1	1	
		No TWDR action or	1	0	1	0	
		No TWDR action	1	0	1	1	

Định dạng và các mã trạng thái có thể có trong chế độ ST

Figure 93. Formats and States in the Slave Transmitter Mode



1.8.5. Các trạng thái hỗn tạp – Miscellaneous State

Có 2 trạng thái không được đề cập trong các trạng thái đã cho ở 4 chế độ trên.

Trạng thái \$F8 trong khi cờ TWINT chưa được set (bởi hardware) và là trạng thái trung gian chuyển tiếp từ trạng thái này sang trạng thái kia.

Trạng thái \$00 cho thấy có lỗi trên bus xảy ra khi gửi đi START hoặc STOP. Khi có lỗi này xảy ra, cờ TWINT được set lên 1 (bằng hardware). Để phục hồi lại khi xảy ra lỗi này, bit TWSTO phải set lên 1 và cờ TWINT phải được xóa đi (bằng cách ghi 1). Việc này có tác dụng chuyển module TWI sang chế độ slave không được định địa chỉ và xóa cờ TWSTO đi (những bit còn lại trong TWCR không bị tác động). Dây SDA và SCL khi đó sẽ được thả nổi và không có điều kiện STOP nào được phát đi.

Table 78. Miscellaneous States

Status Code (TWSR) Prescaler Bits are 0	Status of the Two-wire Serial Bus and Two-wire Serial Inter- face Hardware	Application Software Response					Next Action Taken by TWI Hardware
		To/from TWDR	To TWCR				
			STA	STO	TWINT	TWEA	
\$F8	No relevant state information available; TWINT = "0"	No TWDR action	No TWCR action				Wait or proceed current transfer
\$00	Bus error due to an illegal START or STOP condition	No TWDR action	0	1	1	X	Only the internal hardware is affected, no STOP con- dition is sent on the bus. In all cases, the bus is released and TWSTO is cleared.

1.8.6. Kết hợp các chế độ TWI

Trong một số trường hợp cần kết hợp vài chế độ TWI lại với nhau để hoàn tất một hoạt động truyền nhận data mong muốn.

Ví dụ trong trường hợp này là bộ nhớ EEPROM.

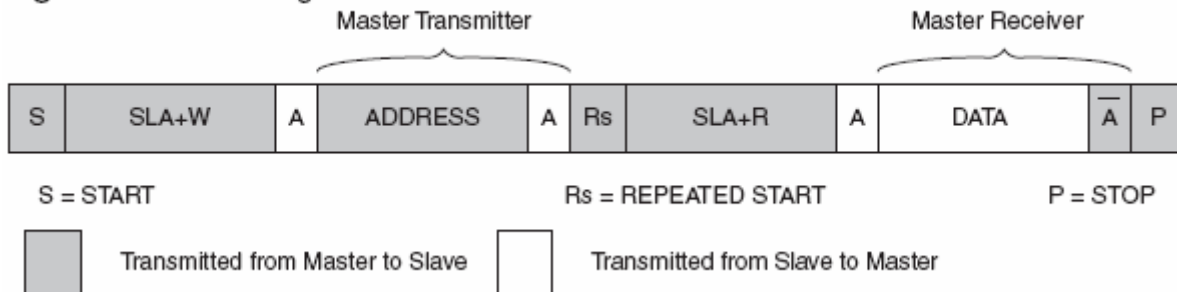
Quá trình đọc data tại địa chỉ ô nhớ bất kỳ trong EEPROM cần phải theo các bước sau:

- Khởi tạo module TWI.
- Gửi START (MT mode)

- Gửi SLA+W (MT mode)
- Gửi địa chỉ cần đọc (MT mode)
- Gửi REPEAT START (MR mode)
- Gửi SLA+R (MR mode)
- Bắt đầu đọc data (MR mode)
- Gửi STOP (MR mode)

Lưu ý rằng sau khi gửi REPEAT START xong thì master vẫn chiếm quyền điều khiển TWI bus.

Figure 94. Combining Several TWI Modes to Access a Serial EEPROM



1.8.7. Bus đa chủ và tranh chấp

(* Tiếp tục cập nhật)

2. Thiết bị sử dụng chuẩn giao tiếp I2C/TWI

2.1. EEPROM AT24Cxx

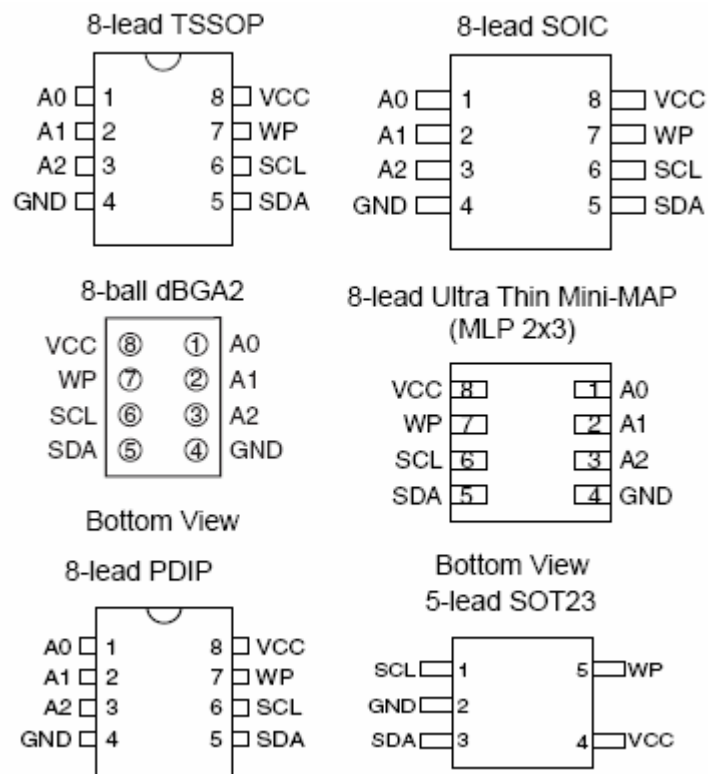
2.1.1. Giới thiệu

- Loại bộ nhớ EEPROM do ATMEL sản xuất.
- Hoạt động ở 2 chế độ điện áp 2.7 (Vcc = 2.7 to 5.5V) và 1.8 (Vcc = 1.8 to 5.5V)
- Dung lượng bộ nhớ tùy thuộc theo từng dòng sản phẩm (1Kbit, 2 Kbit, 4Kbit, 8 Kbit hay 16 Kbit)
- Giao tiếp qua giao thức I2C/TWI.
- Tốc độ có khả năng lên đến 100KHz (ở 1.8V) và 400KHz (ở 2.7V, 5V).
- Chế độ ghi liên tục từng page (trang): page 8byte (với loại 1K, 2K) và page 16byte (loại 4K, 8K, 16K).
- Khả năng ghi/xóa 1 triệu lần, thời gian lưu trữ lên đến 100năm.

Kí hiệu chức năng chân như hình dưới:

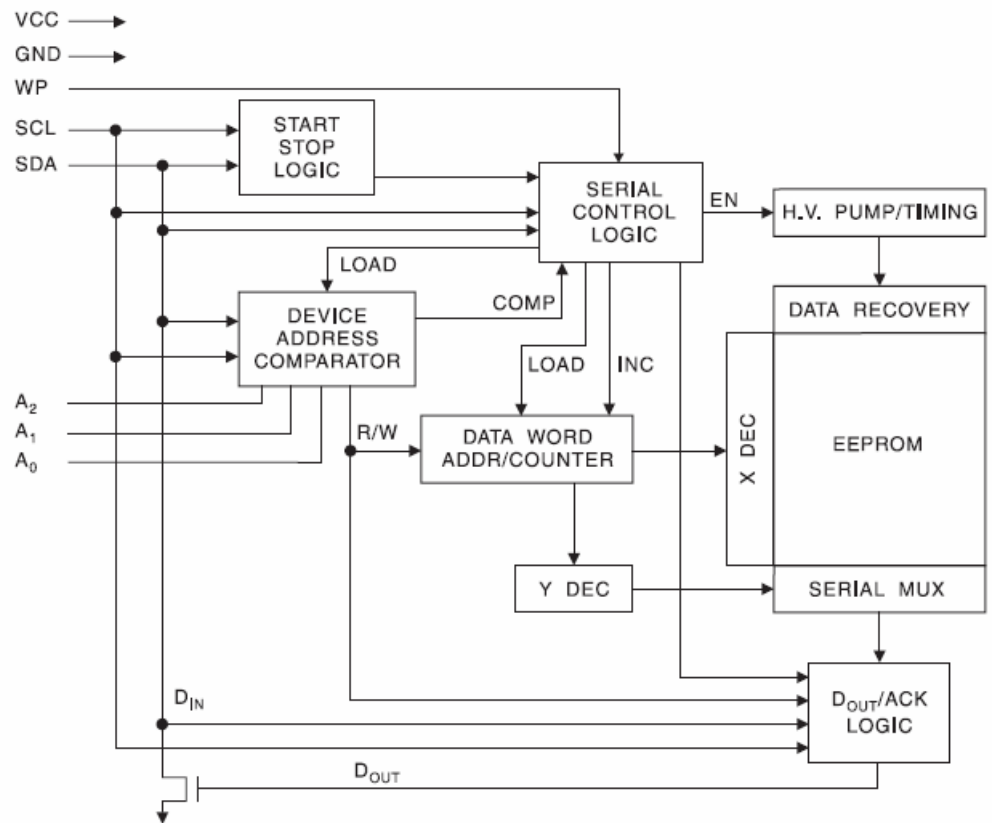
Table 1. Pin Configuration

Pin Name	Function
A0 - A2	Address Inputs
SDA	Serial Data
SCL	Serial Clock Input
WP	Write Protect
NC	No Connect
GND	Ground
VCC	Power Supply



2.1.2. Sơ đồ khối tổng quát

Figure 1. Block Diagram



2.1.3. Mô tả chức năng chân

- Serial Clock (SCL): chân clock TWI/I2C.
- Serial Data (SDA): chân data TWI/I2C.
- Device/Page Addresses (A₂, A₁, A₀): Chân lựa quy định địa chỉ/trang của thiết bị.
- Write Protect (WP): chân chống ghi.

2.1.4. Tổ chức bộ nhớ từng dòng sản phẩm trong AT24Cxx

AT24C01A, 1K SERIAL EEPROM: Internally organized with 16 pages of 8 bytes each, the 1K requires a 7-bit data word address for random word addressing.

AT24C02, 2K SERIAL EEPROM: Internally organized with 32 pages of 8 bytes each, the 2K requires an 8-bit data word address for random word addressing.

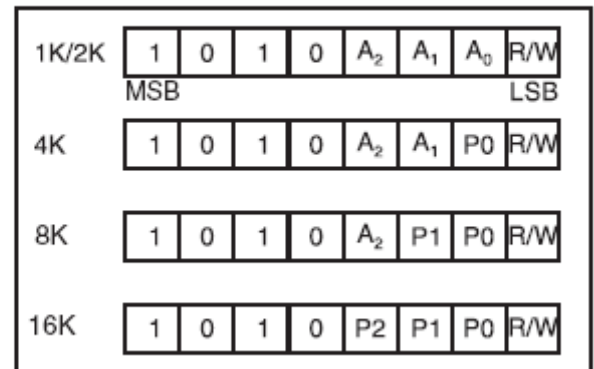
AT24C04, 4K SERIAL EEPROM: Internally organized with 32 pages of 16 bytes each, the 4K requires a 9-bit data word address for random word addressing.

AT24C08A, 8K SERIAL EEPROM: Internally organized with 64 pages of 16 bytes each, the 8K requires a 10-bit data word address for random word addressing.

AT24C16A, 16K SERIAL EEPROM: Internally organized with 128 pages of 16 bytes each, the 16K requires an 11-bit data word address for random word addressing.

2.1.5. Định địa chỉ thiết bị

Figure 7. Device Address



Địa chỉ thiết bị (slave address) được thiết lập bằng phần cứng (chân A2, A1, A0), khác nhau trên từng dòng sản phẩm và nó được kết hợp để quy định địa chỉ trang (page) trong EEPROM.

2.1.6. Hoạt động ghi dữ liệu

Có 2 kiểu ghi dữ liệu: ghi từng byte hoặc ghi từng “trang” byte.

Figure 8. Byte Write

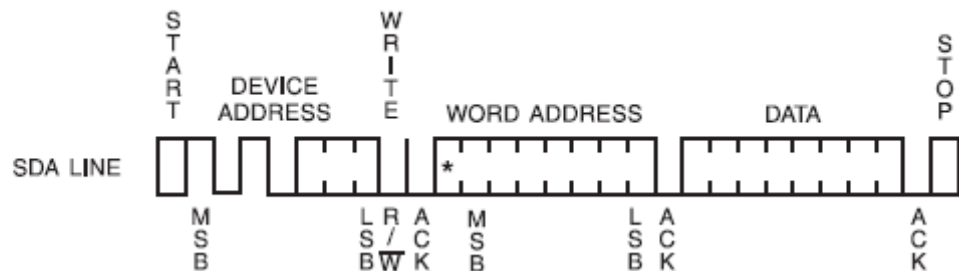
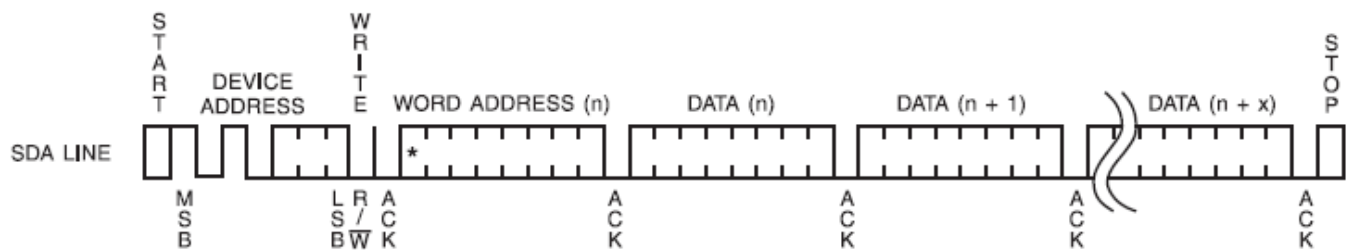


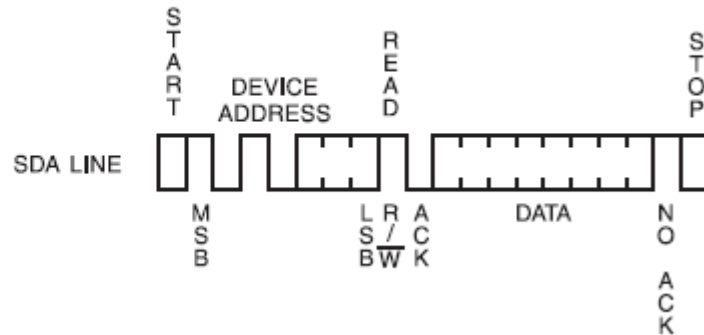
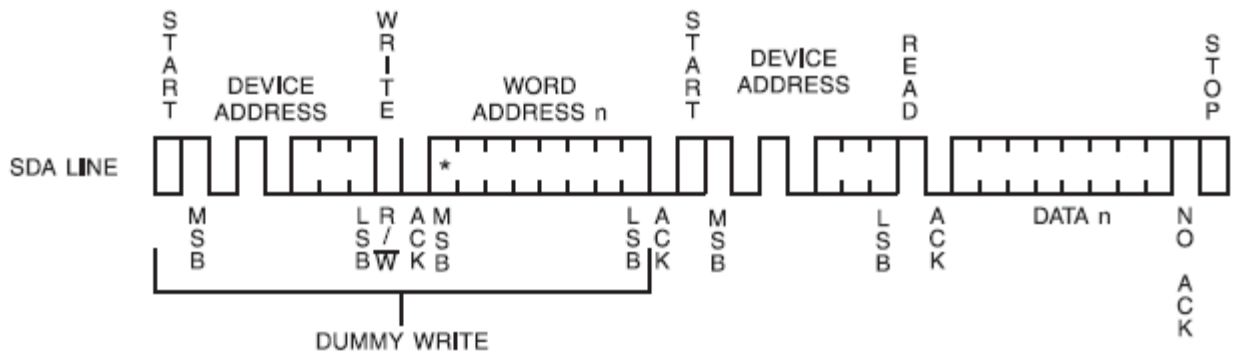
Figure 9. Page Write



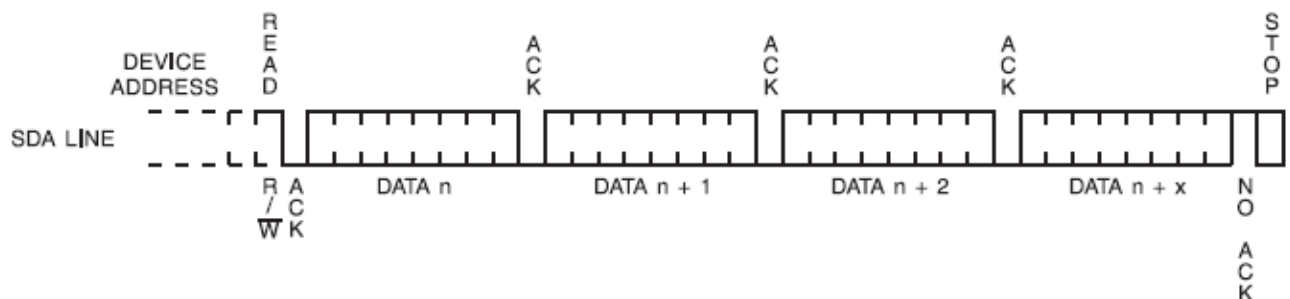
(* = DON'T CARE bit for 1K)

2.1.7. Hoạt động đọc dữ liệu

Có các kiểu: đọc từng byte hay từng trang, đọc ở địa chỉ hiện tại hay địa chỉ bất kì

Figure 10. Current Address Read**Figure 11.** Random Read

(* = DON'T CARE bit for 1K)

Figure 12. Sequential Read

Đối với kiểu đọc dữ liệu ở địa chỉ hiện tại, tức là con trỏ dữ liệu trong EEPROM đang đứng ở đâu thì nó sẽ bắt đầu truy xuất dữ liệu bắt đầu từ địa chỉ này. Còn khi đọc dữ liệu ở địa chỉ bất kì, đầu tiên phải ghi địa chỉ cho con trỏ nhảy tới, sau đó mới cho phép truy xuất dữ liệu tại ô nhớ vừa nhảy.

2.2. I2C Realtime Clock DS1307

2.2.1. Giới thiệu

Là đồng hồ thời gian thực với khả năng đếm thời gian như giây, phút, giờ và ngày tháng như ngày trong tháng, tháng, ngày trong tuần, năm bình thường và năm nhuận.

Sử dụng giao thức I2C/TWI chuẩn.

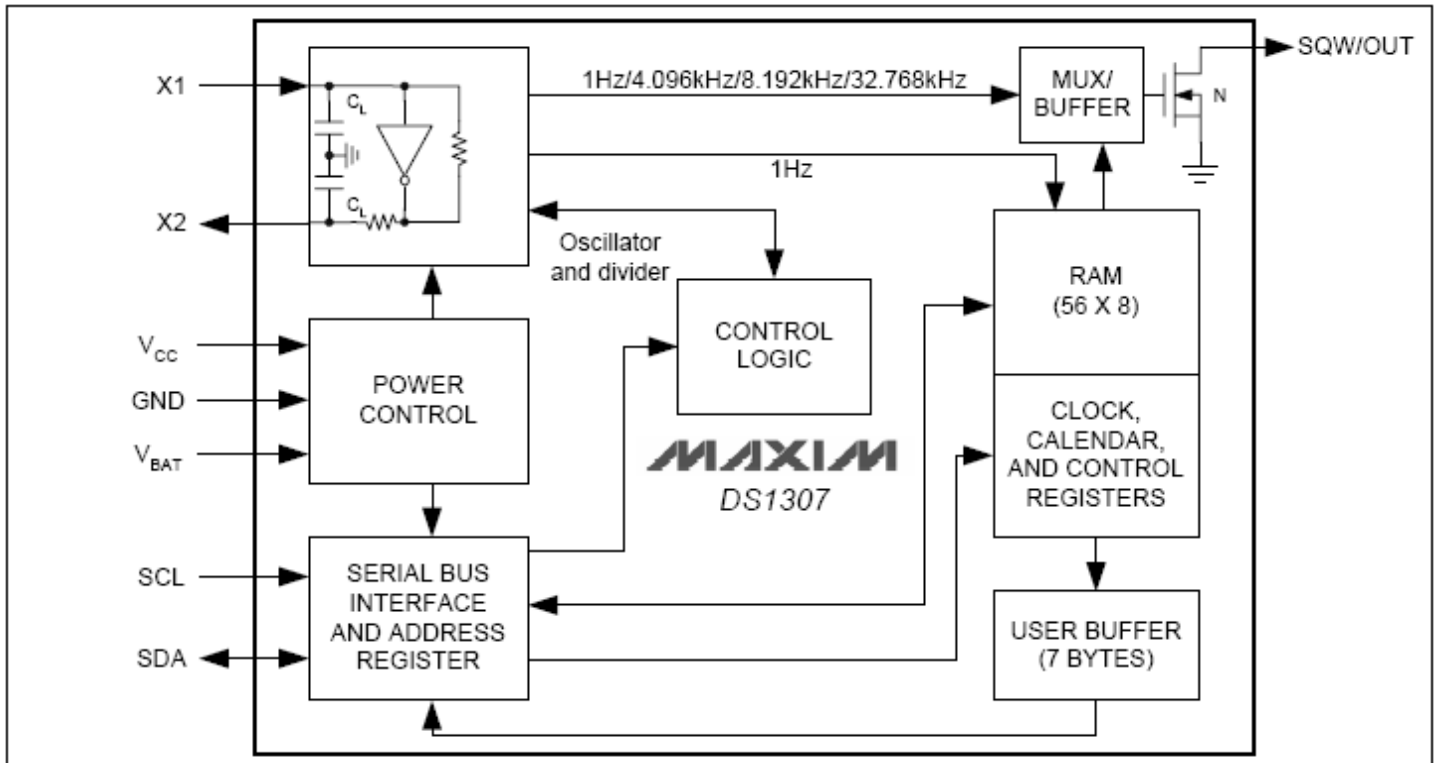
Có khả năng đếm đến năm 2100.

Vùng RAM đa mục đích với dung lượng 56 bytes và khả năng ghi không giới hạn.

Khả năng phát xung vuông lập trình được.

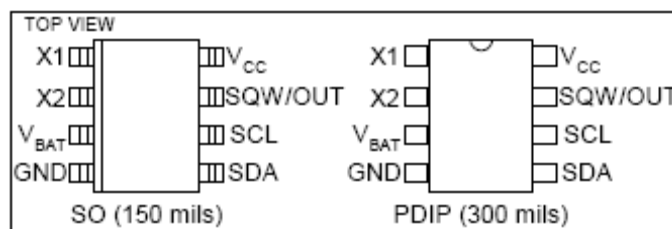
Tự động phát hiện nguồn giảm và chuyển mạch bên trong, tiêu thụ dưới 500nA khi sử dụng nguồn từ pin backup.

2.2.2. Sơ đồ khối



2.2.3. Mô tả chức năng chân

Đóng gói dưới dạng DIP hoặc SO



STT chân	Tên chân	Mô tả chức năng
1 & 2	X1 & X2	Kết nối với thạch anh chuẩn 32.768KHz
3	VBAT	Kết nối với pin backup
4	GND	Chân đất GND
5	SDA	Chân SDA nối với I2C/TWI bus
6	SCL	Chân SCL nối với I2C/TWI bus
7	SQW/OUT	Chân ngõ ra của bộ phát xung vuông

8	VCC	Chân nối nguồn VCC
---	-----	--------------------

2.2.4. Thanh ghi và RAM trong DS1307

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/ AM							
03h	0	0	0	0	0	DAY			Day	01–07
04h	0	0	10 Date		Date				Date	01–31
05h	0	0	0	10 Month	Month				Month	01–12
06h	10 Year				Year				Year	00–99
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

0 = Always reads back as 0.

Thời gian và ngày tháng được xác định bằng cách đọc giá trị trong các thanh ghi tương ứng cho trong bảng trên.

Thời gian và ngày tháng cũng có thể được khởi tạo bằng cách ghi giá trị mong muốn vào các thanh ghi trên.

Nội dung của thời gian và ngày tháng được định dạng theo mã BCD.

Bit 7 trong thanh ghi tại thanh ghi đầu tiên – Clock Halt (CH) bit – dùng để cho phép/cấm bộ dao động hoạt động. Nếu bit này set lên 1, bộ dao động bị cấm và ngược lại.

DS1307 có thể hoạt động trong chế độ 12 giờ hoặc 24 giờ bằng cách thiết lập giá trị thích hợp cho bit 6 của thanh ghi giờ. Khi set bit này lên 1, chế độ 12h được chọn và ngược lại. Trong chế độ 12h, bit 5 (AM/PM) sẽ set lên 1 khi đang ở PM. Trong chế độ 24h, bit 5 lúc này được xem như bit 10-hour thứ 2 (từ 20-23h). Giá trị giờ phải thiết đặt lại khi bit chọn chế độ 12/24h bị thay đổi.

2.2.5. Thanh ghi điều khiển

Thanh ghi này chủ yếu để điều khiển các chế độ hoạt động cho ngõ ra của bộ phát xung vuông

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

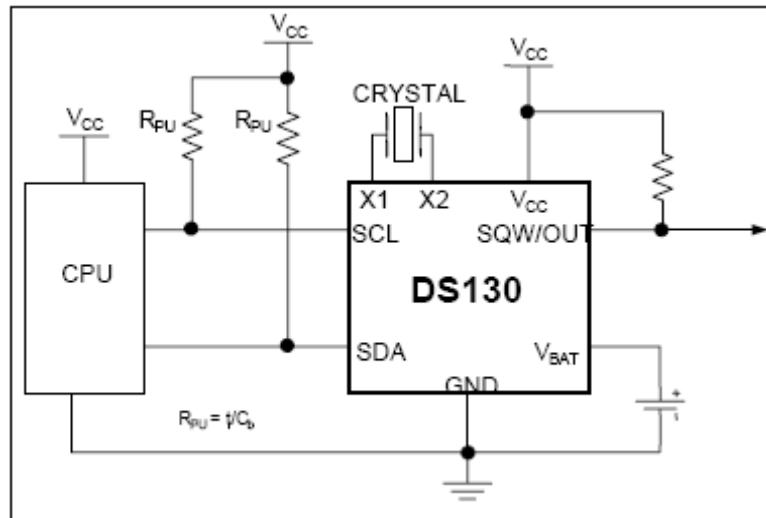
Bit 7 – Output Control - OUT: Điều khiển mức logic trên chân OUT khi chức năng phát xung bị cấm.

Bit 4 – Square Wave Enable - SQWE: Cho phép bộ phát xung hoạt động.

Bit 1 và 0 – RS10 – Rate Select: Bit chọn tốc độ phát xung, như trong bảng bên dưới.

RS1	RS0	SQW/OUT OUTPUT	SQWE	OUT
0	0	1Hz	1	X
0	1	4.096kHz	1	X
1	0	8.192kHz	1	X
1	1	32.768kHz	1	X
X	X	0	0	0
X	X	1	0	1

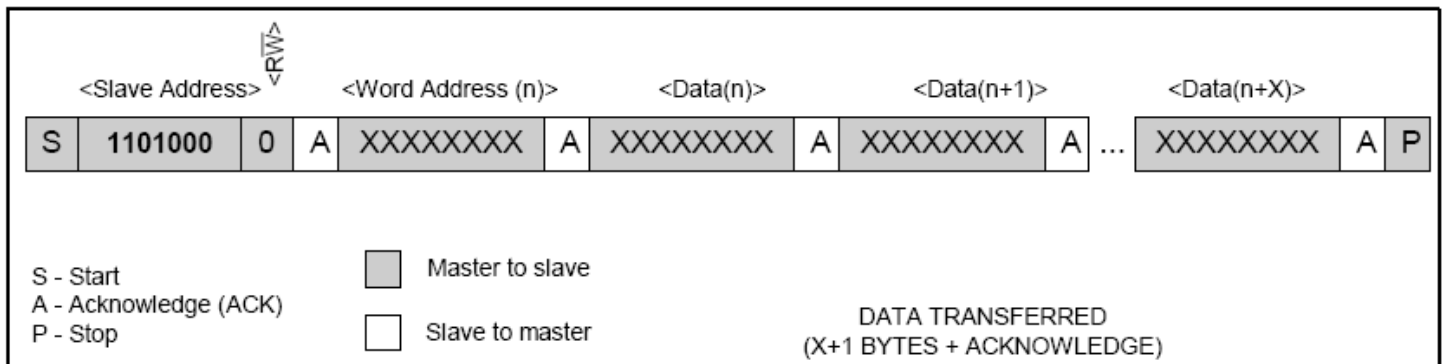
2.2.6. Sơ đồ mạch ứng dụng thông thường



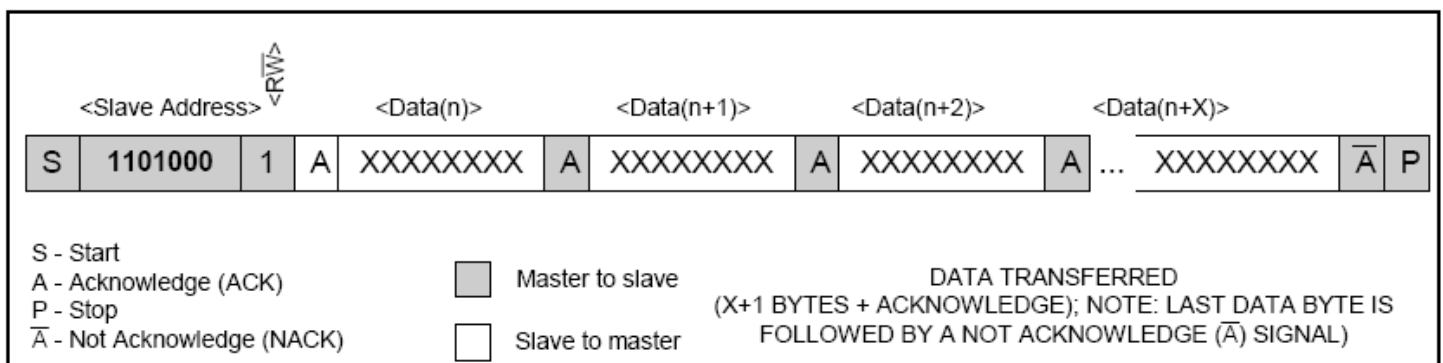
2.2.7. Mô tả hoạt động

Lưu ý: Địa chỉ I2C của DS1307 được gán cố định và có giá trị bằng 0b11010000 (8-bit) hoặc 0b1101000 (7-bit).

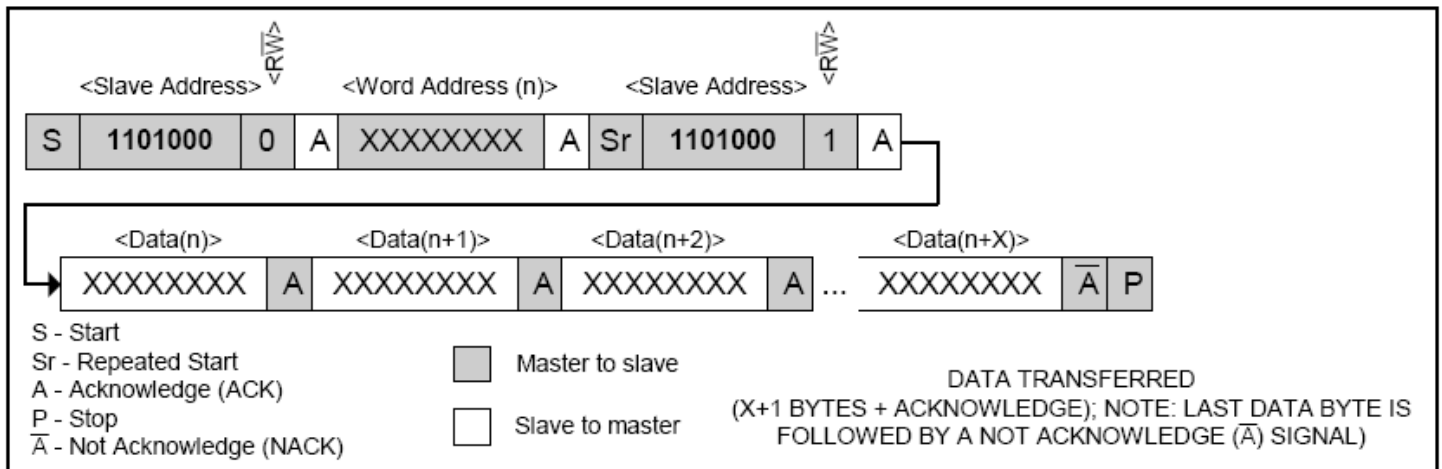
Hoạt động ghi dữ liệu



Hoạt động đọc dữ liệu



Ghi và đọc trong cùng 1 frame truyền



3. Nội dung thực hành

3.1. Lập trình ứng dụng với EEPROM AT24C04 (Master mode)

Nội dung

Viết chương trình truy xuất đọc/ghi dữ liệu với EEPROM AT24C04 qua I2C bus.

Mô phỏng trên Proteus, các thao tác nhập xuất (thiết đặt giá trị và hiển thị nội dung) thực hiện thông qua màn hình terminal.

Trong quá trình viết, sử dụng công cụ I2C Debugger trong Proteus để debug chương trình.

Sau khi mô phỏng thành công, hiện thực hóa trên KIT TN AT32.

Lưu ý:

- EEPROM sử dụng trong Proteus là loại AT24C1024, dung lượng 1024Kbit = 131,072byte = 128Kbyte.
- EEPROM sử dụng trên module rời của KIT AT32 là loại AT24C04, dung lượng 4Kbit = 512byte.
- Địa chỉ thiết bị của EEPROM với 2 loại trên được quy định khác nhau. Tham khảo trong datasheet để biết chi tiết về việc quy định địa chỉ EEPROM cho AT24C1024, còn đối với AT24C04 việc quy định địa chỉ đã được đề cập trong phần 2 của tài liệu này.

Các bước thực hiện

(HD của GV)

3.2. Lập trình ứng dụng với Realtime clock DS1307 (Master mode)

Nội dung

Viết chương trình truy xuất đọc/ghi dữ liệu với RTC DS1307 qua I2C bus với yêu cầu cụ thể như sau:

- Khởi tạo giá trị giờ phút giây, ngày tháng năm thông qua quá trình ghi dữ liệu.
- Cập nhật giá trị và in ra màn hình terminal hoặc TEXT LCD thông qua quá trình đọc dữ liệu.
- Truy xuất nội dung các thanh ghi trong vùng RAM (General Purpose RAM) của DS1307.

Các bước thực hiện

(HD của GV)

3.3. Lập trình giao tiếp 2 thiết bị TWI/I2C master-slave (Master và Slave mode)

Nội dung

Sử dụng 2 MCU ATMEGA32, một làm master và một làm slave giao tiếp qua ngõ TWI.

Trên mỗi thiết bị (master và slave) đều sử dụng PORTA.0 làm ngõ lái LED (tích cực mức cao) và PINB.0 làm ngõ vào nút nhấn (tích cực mức cao).

Viết chương trình ứng dụng, sao cho khi nhấn nút trên master thì LED trên slave sáng, còn khi nhấn nút trên slave thì LED trên master sáng.

Sử dụng công cụ I2C debugger hỗ trợ debug chương trình.

Các bước thực hiện

(HD của GV)

Table Of Content

1. Giới thiệu khối TWI/I2C trên ATMEGA32	3
1.1. Tính năng tổng quát	3
1.2. Định nghĩa TWI Bus – Two Wire Serial Interface Definition	3
1.3. Truyền tải dữ liệu và định dạng khung truyền trên TWI Bus – Data Transfer and Frame Format	4
1.4. Hệ thống bus TWI đa chủ, tranh chấp và đồng bộ hóa – Multi Master Bus System, Arbitration and Synchronization	6
1.5. Tổng quan module TWI trong ATMEGA32	7
1.6. Mô tả thanh ghi TWI – TWI Register Definition	8
1.7. Sử dụng TWI trong ATMEGA32	9
1.8. Các chế độ truyền dữ liệu của TWI trên ATMEGA32	11
1.8.1. Chế độ MASTER TRANSMIT	11
1.8.2. Chế độ MASTER RECEIVE	14
1.8.3. Chế độ SLAVE RECEIVER	17
1.8.4. Chế độ SLAVE TRANSMITTER	20
1.8.5. Các trạng thái hỗn tạp – Miscellaneous State	23
1.8.6. Kết hợp các chế độ TWI	23
1.8.7. Bus đa chủ và tranh chấp	24
2. Thiết bị sử dụng chuẩn giao tiếp I2C/TWI	24
2.1. EEPROM AT24Cxx	24
2.1.1. Giới thiệu	24
2.1.2. Sơ đồ khối tổng quát	26
2.1.3. Mô tả chức năng chân	26
2.1.4. Tổ chức bộ nhớ từng dòng sản phẩm trong AT24Cxx	26
2.1.5. Định địa chỉ thiết bị	27
2.1.6. Hoạt động ghi dữ liệu	27
2.1.7. Hoạt động đọc dữ liệu	27
2.2. I2C Realtime Clock DS1307	28
2.2.1. Giới thiệu	28
2.2.2. Sơ đồ khối	29
2.2.3. Mô tả chức năng chân	29
2.2.4. Thanh ghi và RAM trong DS1307	30
2.2.5. Thanh ghi điều khiển	30
2.2.6. Sơ đồ mạch ứng dụng thông thường	31
2.2.7. Mô tả hoạt động	31
3. Nội dung thực hành	32
3.1. Lập trình ứng dụng với EEPROM AT24C04 (Master mode)	32
3.2. Lập trình ứng dụng với Realtime clock DS1307 (Master mode)	32
3.3. Lập trình giao tiếp 2 thiết bị TWI/I2C master-slave (Master và Slave mode)	33

Appendix