Tutorial for interfacing DS1307 (RTC) with AT89C5131A using I^2C

Author - Yogesh Patil, 123079052@iitb.ac.in

I^2C (Inter IC)-

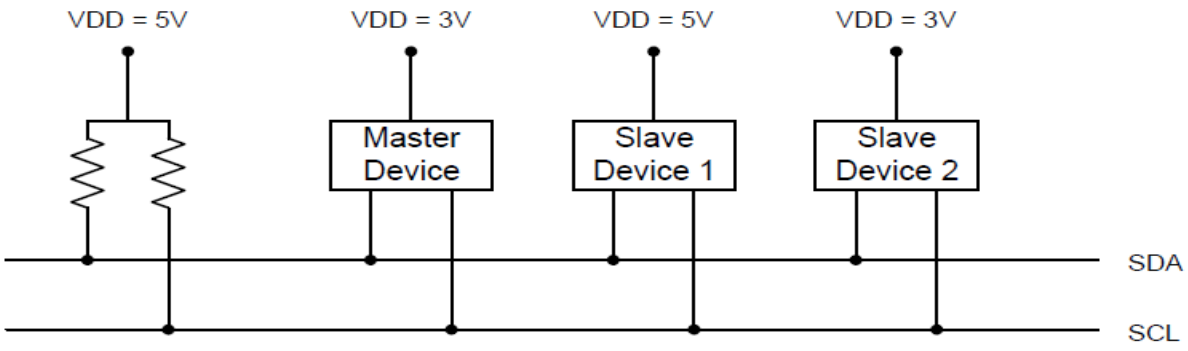

Figure 1 - Typical I^2C bus (From C8051F380 datasheet by Silicon Labs)

Important features of I^2C configuration:

1. Only two lines are required SDA (serial data line) and SCL (serial clock line) for communication.
2. Each device is identified by its unique 7-bit address.
3. Any device with capability of generating clock, START and STOP conditions can act as MASTER.
4. MASTER controls the SCL line and initiates the data transfer.
5. The 8-bit bidirectional serial data transfer occurs at 100Kbits/s in standard mode and 400 Kbits/s in fast mode. (DS1307 only supports standard mode.)
6. The no. of devices that can be connected is limited by capacitance on the bus.
7. Each device has to have open-drain/open-collector output configuration. The lines are pulled up as shown in figure 1.

I^2C bus states

START : HIGH to LOW transition on SDA line when SCL is HIGH defines the START condition.

STOP : LOW TO HIGH transition on SDA line when SCL is HIGH defines the STOP condition.

ACK : The receiver has to hold the SDA line low during 9^{th} cycle after 8 bit transfer from transmitter to receiver.

NACK : The receiver has to release the SDA line during 9^{th} cycle after 8 bit transfer from transmitter to receiver.

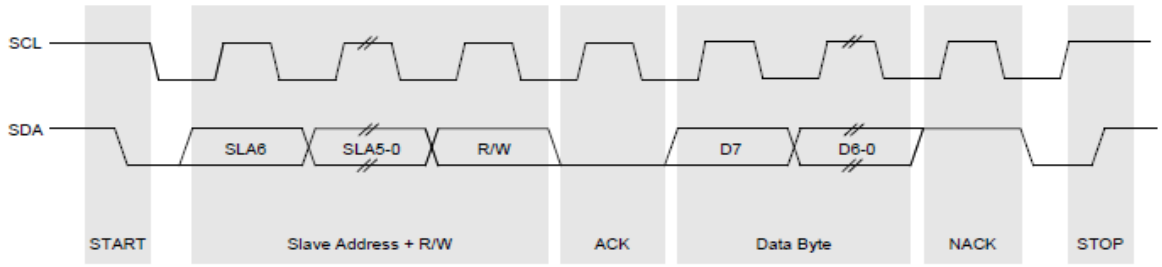

Figure 2- I²C bus states (From C8051F380 datasheet by Silicon Labs)

There can be 2 possible way of data transfer.

1. MASTER TRANSMITTER - SLAVE RECEIVER
2. MASTER RECEIVER - SLAVE TRANSMITTER

AT89C5131A has SSLC i.e. synchronous serial link controller (SSLC) for two wire interface i.e. TWI. For communication using *I²C* protocol, TWI can be used.

The registers to be used are

- SSCON – Control register for configuring TWI.
- SSCS – Status register for TWI.
- SSDAT – Data register. Stores data to be transmitted /received.

Serial (TWI) interrupt

The serial interrupt is generated when

1. START condition generated on SDA line.
2. Address/Data byte has been transmitted and acknowledgement is received.
3. Address/Data byte has been received and acknowledgement as per AA (SSCON.2) is sent. (Set AA=1 for sending positive ACK i.e. low on SDA line)
4. Arbitration lost.

When the interrupt is generated, SI (SSCON.3) flag is set to 1. The clock is disabled for that time. SI has to be cleared in software before leaving the ISR routine. The interrupt number is 8.

Serial clock generation

The SSLC has to generate clock on SCL line. It is configured by setting appropriate values in CR2, CR1 and CR0 bit of SSCON. (Refer page No. 104 from datasheet. Clock frequency is same as bit frequency).

(Refer datasheet from http://www.atmel.in/Images/doc4136.pdf)

TWI operation

WRITE:

Just after the START condition is generated by MASTER, it transmits the SLAVE address + R/W bit. For this operation R/W bit will be 0. After address byte + R/W is acknowledged by slave, in the next byte MASTER starts sending the data in 8-bit format. After a byte is transferred, transmitter will wait for the ACK from the receiver. Note that MASTER is always a transmitter in this function.

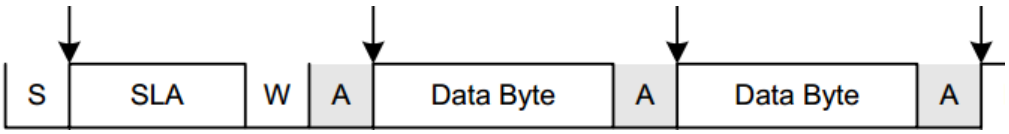

Figure 3 - Write operation (From C8051F380 datasheet by Silicon Labs)

All the arrows show the instances at which the serial interrupt is generated.

READ:

Just after the START condition is generated by MASTER, in next byte it transmits the SLAVE address + R/W bit. For this operation R/W bit will be 1. After address byte is acknowledged by SLAVE, in the next byte MASTER waits for the data from receiver in 8-bit format. After byte is transferred, transmitter will wait for the ACK from receiver. Note that MASTER is transmitter during address byte and receiver during data bytes.

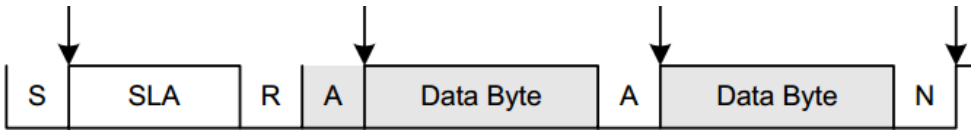

Figure 4 - Read operation (From C8051F380 datasheet by Silicon Labs)

Steps to configure TWI for write:

1. Enable the global and TWI interrupt (IEN0.7 and IEN1.1).
2. Generate START condition (SSCON.5).
3. Handle the interrupt generated by taking appropriate action depending on the status in SSCS register. (Look at the table 80 in datasheet on page no. 104)

DS 1307

It's a real time clock. The slave address of DS1307 is 1011000. So for WRITE, after START condition 1011000 + 0 = 0xD0 has to be transmitted. For read, 0xD1 has to be transmitted. DS1307 can only act as slave. DS1307 has the memory map as follows.

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

Figure 5 - Memory map of DS1307 (From DS1307 datasheet by Maxim Integrated)

When DS1307 is powered on, the data and time registers are reset to 01/01/00 01 00:00:00 (MM/DD/YY DAY HH:MM:SS) and the CH bit is seconds register is set to 1. CH=1 disables the clock of RTC. ***So, CH has to be cleared before beginning any operation using DS1307***. The data is stored in BCD format. When bit 6 of hour register is set to 1, 12 hour format is selected. Otherwise 24 hour format is selected. In 12 hour format bit 5 defines AM/PM. Logic high indicates PM.

DS1307 maintains an address pointer. The address pointed by address pointer is used for read operation. When a WRITE operation is performed at particular address the pointer is set to that location. After which if READ operation is performed then data at that particular location is transmitted. Once a read/write operation is performed address pointer automatically gets incremented. So, in next cycle value at next location is transmitted. It becomes clear in the following figures.

The **WRITE** cycle is,

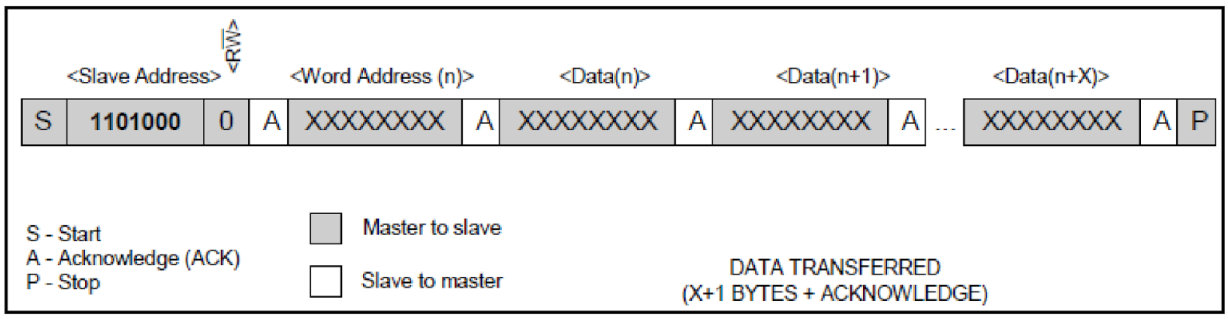

Figure 6 - Write operation on DS1307(From DS1307 datasheet by Maxim Integrated)

Here ‘n’ denotes the location at which address pointer is pointing. Data (n) is the data to WRITE at location ‘n’. Note that it is expecting word address after the slave address is received.

The **READ** cycle is,

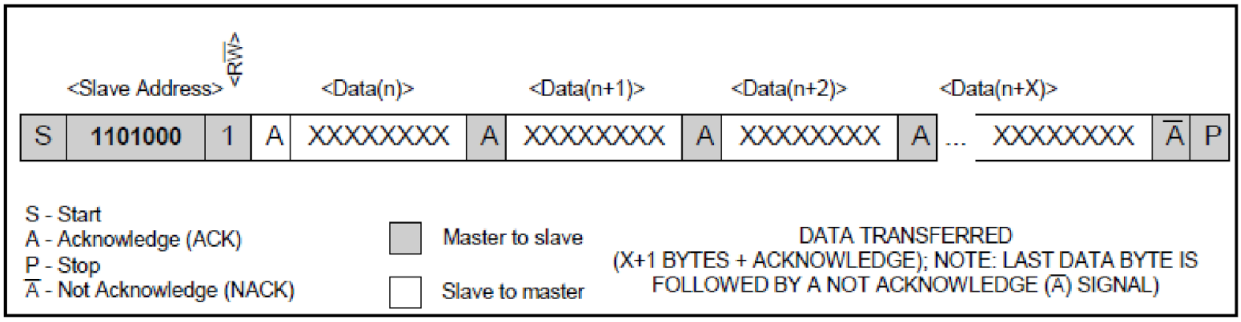

Figure 7 - Read operation from DS1307(From DS1307 datasheet by Maxim Integrated)

Here ‘n’ denotes the location at which address pointer is pointing. Data(n) is the data at location ‘n’ which is transmitted.

Dummy write - For reading from a particular location first we need to set the address pointer to that location. This can be done by dummy write.

The steps to be followed are,

1. After start condition, send slave address+0 i.e., write mode.
2. After ACK, send the address to be read.
3. After ACK, generate start condition and send slave address+1 i.e., read mode.
4. Receive data from slave.

Step 2 makes sure that address pointer gets pointed to the location to be read. In step 4 it sends the data at that location.