SC2107/CE2107 Microprocessor System Design and Development

Tutorial 6 (with Solutions)

Serial Bus and Display

1. Discuss the common and different characteristics of SPI and I2C?

Solution:

Common characteristics:

- Serial (single data line)
    - Synchronous
    - Data is clocked along with a clock signal
    - Master-Slave protocol
- Master device controls the clock (SCLK/SCL)
- Usually one master at a time

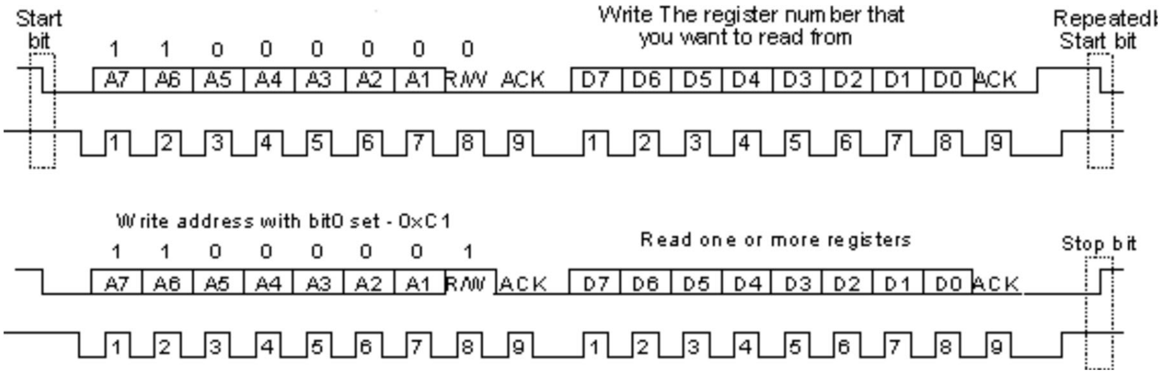
Different characteristics:

- SPI requires 3 or 4 wires, while I2C requires 2 wires (power and ground not considered).
- SPI has higher throughput due to full duplex. I2C is slower due to half duplex.
- In SPI, slave is selected using CS# (increase wiring). In I2C, slave is selected using 7-bit address.

2. In the I2C protocol, in order to read from the slave device, you must first tell it which of its internal addresses you want to read from. So a read of the slave actually starts off by writing to it. Let's look at the sequence of communicating with a peripheral module.
   - Send the Start Sequence
   - Send 0xC0 (I2C address of the peripheral module with R/W set to low)
   - Send 0x01 (Internal address of the register that we want to read from)
   - Send a Start Sequence again
   - Send 0xC1 (I2C address of the peripheral module with R/W set to high)
   - Read a byte from the peripheral device
   - Send the Stop Sequence

   Illustrate the SCL and SDA lines of the I2C bus to reflect the above sequence.

   Solution:


Start bit
1 1 0 0 0 0 0 0
A7 A6 A5 A4 A3 A2 A1 R/W ACK
Write The register number that you want to read from
D7 D6 D5 D4 D3 D2 D1 D0 ACK
Repeated Start bit
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
Write address with bit0 set - 0xC1
1 1 0 0 0 0 0 1
A7 A6 A5 A4 A3 A2 A1 R/W ACK
Read one or more registers
D7 D6 D5 D4 D3 D2 D1 D0 ACK
Stop bit
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9

3. In the I2C protocol, there could be a situation where the slave is not ready to send data.

   Why is this a problem and describe the solution in the I2C protocol to overcome this problem.

Solution

Clock Stretching. The slave is allowed to hold the SCL line low. When the slave gets the read command from the master it holds the clock line low. The slave then gets the requested data, places it in the transmission register and releases the clock line allowing the pull-up resistor to finally pull it high. From the masters point of view, it will issue the first clock pulse of the read by making SCL high and then check to see if it really has gone high. If it's still low then the slave that holding it low and the master should wait until it goes high before continuing.

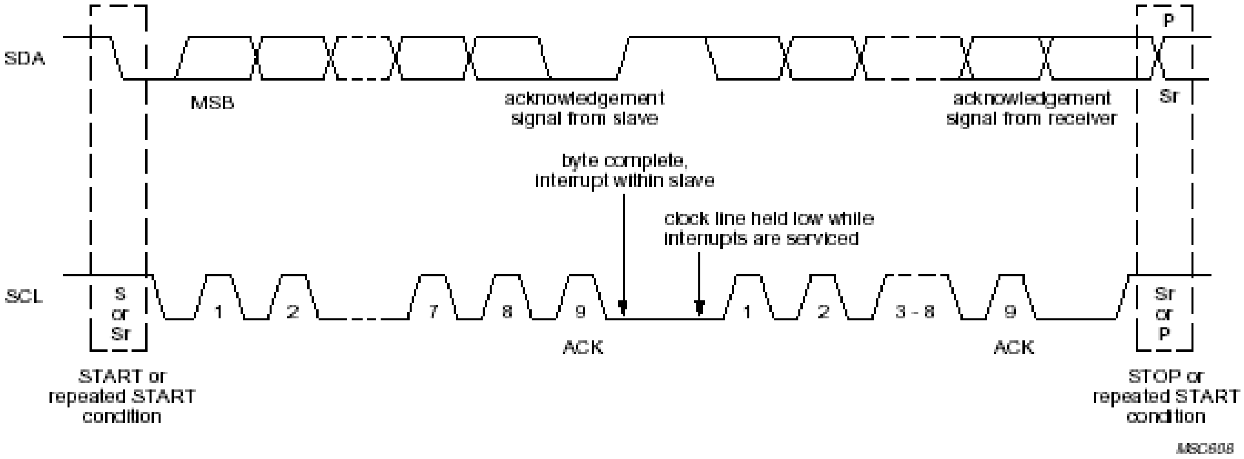
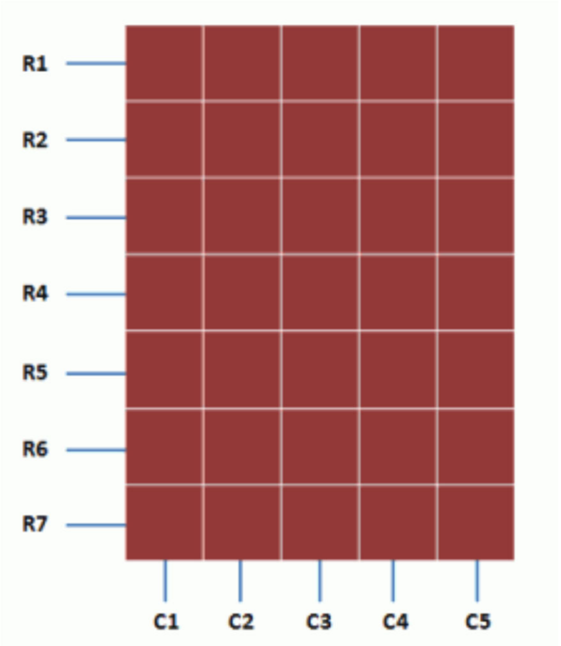

SDA
MSB
acknowledgement signal from slave
acknowledgement signal from receiver
P
Sr
byte complete, interrupt within slave
clock line held low while interrupts are serviced
SCL
S or Sr
1
2
7
8
9
ACK
1
2
3 - 8
9
ACK
Sr or P
START or repeated START condition
STOP or repeated START condition
MSC608

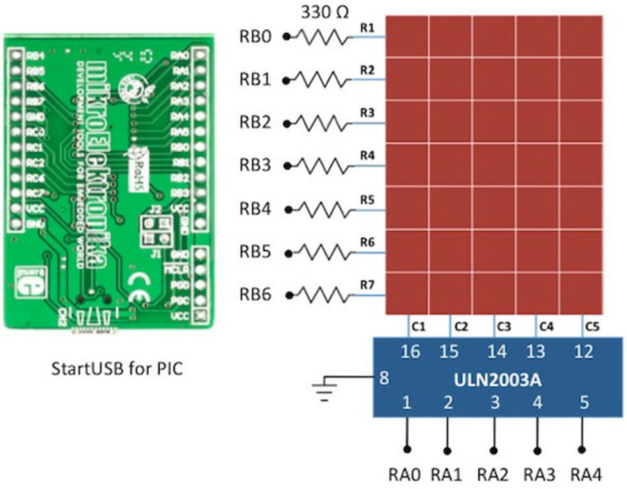
Fig.6 Data transfer on the I^{2}C-bus.

4. A matrix LED as shown in the figure below is to be interfaced to a microprocessor. Explain how the interfacing will be done and the factors that should be considered.



Solution:

All the rows and columns need to be interfaced with GPIO lines of the microprocessor.



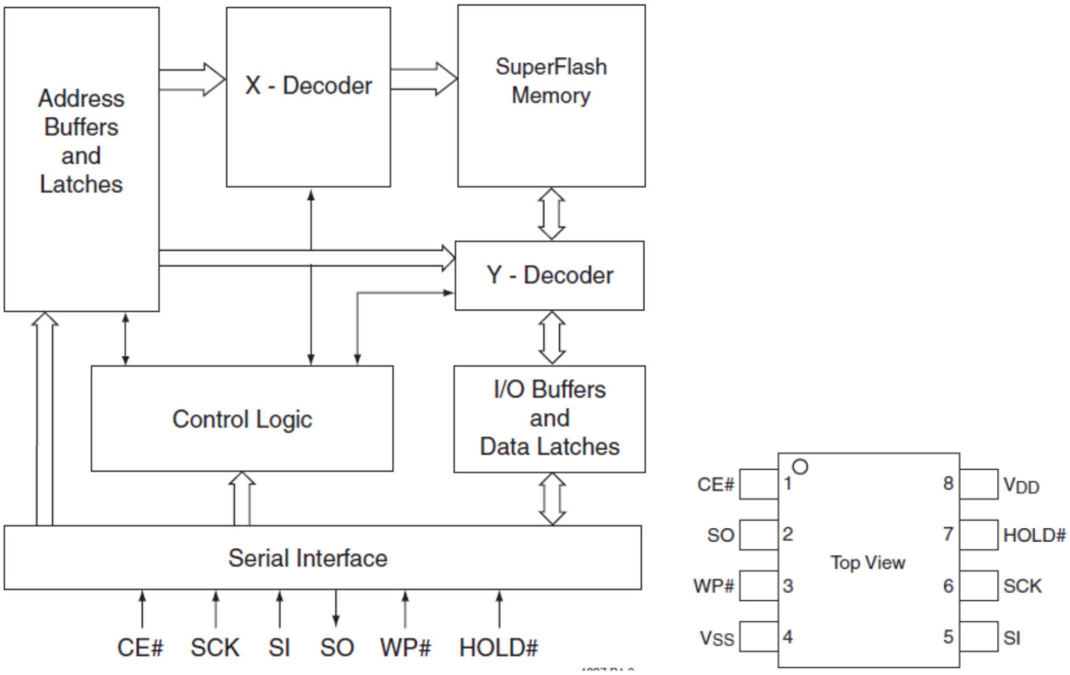
For the Rows, they can be interfaced through a current-limiting resistor. It is important to know the current-handling capability of the LEDs so as to ensure that the appropriate resistor value is chosen. This will also help in selecting the appropriate power rating for the resistors.

For the Columns, they need to be interfaced through a current-sinking transistor array. This is important because when multiple rows are biased through a single column, the current flowing through the column pin will be very high. This can potentially damage the microprocessor pin.

Optional

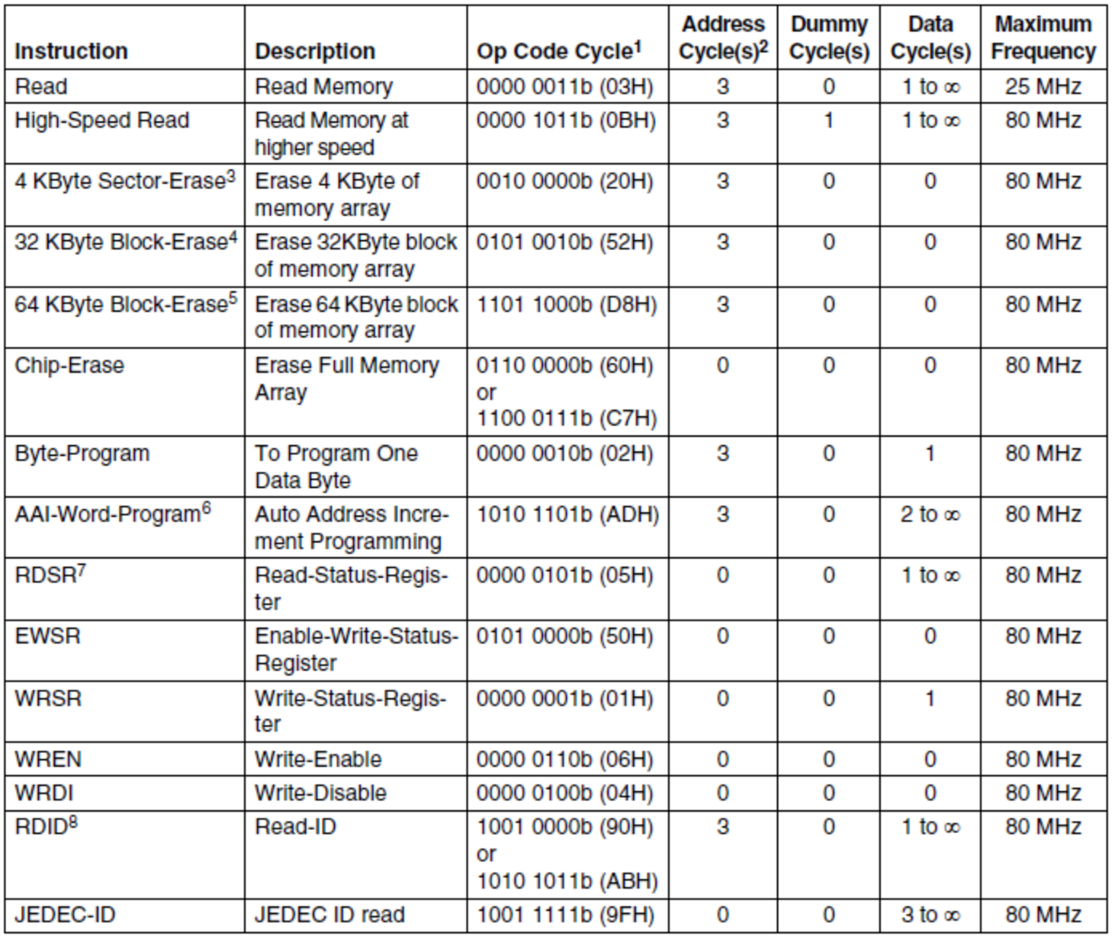
5. The SPI is a common interface for a microprocessor to communicate with peripheral chips. In order to communicate effectively, the programmer must have a good understanding of the protocols associated with that particular chip. These protocols may be unique to that particular chip or may be generalized due to an industry standard.

   We will look at some of the protocols of the 32Mbit SPI Flash chip (SST25VF032B) to have a better understanding.



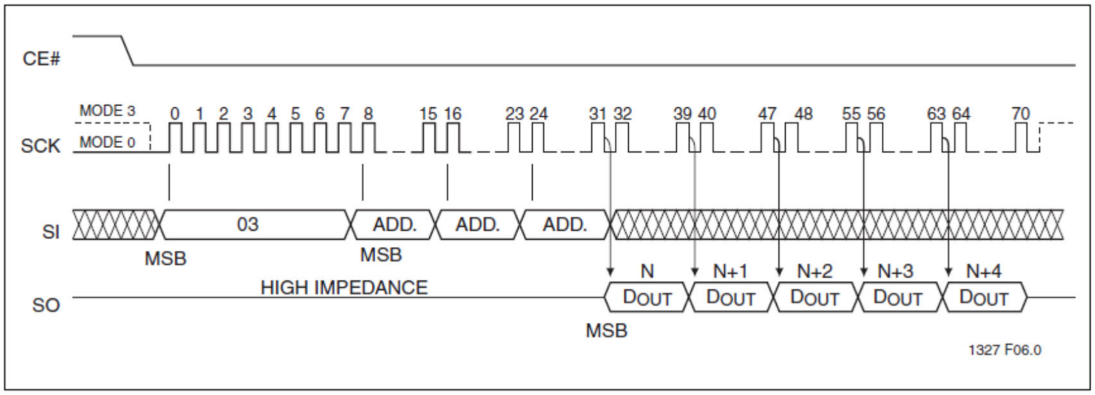
Block Diagram and Pin-Out of the SPI Flash Chip

The Device Operation Protocol is as follows:



Instruction	Description	Op Code Cycle[1]	Address Cycle(s)[2]	Dummy Cycle(s)	Data Cycle(s)	Maximum Frequency
Read	Read Memory	0000 0011b (03H)	3	0	1 to ∞	25 MHz
High-Speed Read	Read Memory at higher speed	0000 1011b (0BH)	3	1	1 to ∞	80 MHz
4 KByte Sector-Erase[3]	Erase 4 KByte of memory array	0010 0000b (20H)	3	0	0	80 MHz
32 KByte Block-Erase[4]	Erase 32KByte block of memory array	0101 0010b (52H)	3	0	0	80 MHz
64 KByte Block-Erase[5]	Erase 64 KByte block of memory array	1101 1000b (D8H)	3	0	0	80 MHz
Chip-Erase	Erase Full Memory Array	0110 0000b (60H) or 1100 0111b (C7H)	0	0	0	80 MHz
Byte-Program	To Program One Data Byte	0000 0010b (02H)	3	0	1	80 MHz
AAI-Word-Program[6]	Auto Address Increment Programming	1010 1101b (ADH)	3	0	2 to ∞	80 MHz
RDSR[7]	Read-Status-Register	0000 0101b (05H)	0	0	1 to ∞	80 MHz
EWSR	Enable-Write-Status-Register	0101 0000b (50H)	0	0	0	80 MHz
WRSR	Write-Status-Register	0000 0001b (01H)	0	0	1	80 MHz
WREN	Write-Enable	0000 0110b (06H)	0	0	0	80 MHz
WRDI	Write-Disable	0000 0100b (04H)	0	0	0	80 MHz
RDID[8]	Read-ID	1001 0000b (90H) or 1010 1011b (ABH)	3	0	1 to ∞	80 MHz
JEDEC-ID	JEDEC ID read	1001 1111b (9FH)	0	0	3 to ∞	80 MHz

The following figure shows the waveform for the 'Read Sequence' command.

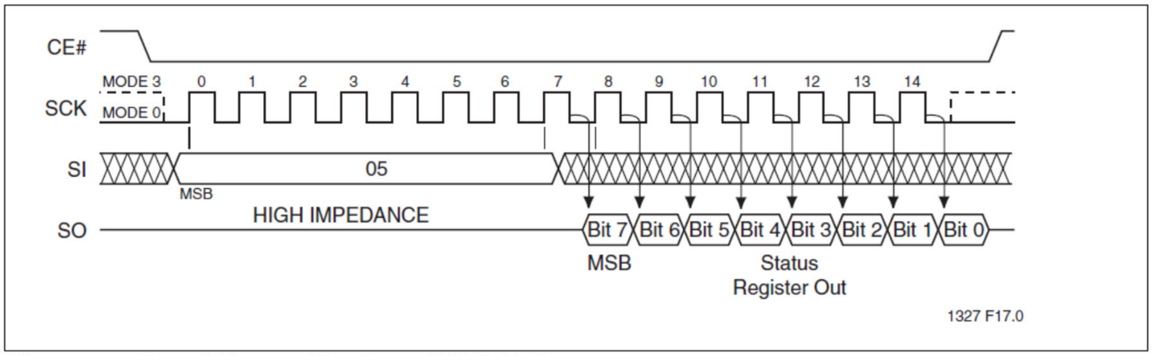


a. Decode the protocol sequence from the microprocessor's perspective.

Solution:

- The microprocessor must first select the device by driving its CE# low.
- The READ command (0x3h) is written to SI, followed by a 3-byte address (A23 – A0) that can be at any location in the device.
- The data at that address is available at the SO line.
- The device automatically increments to the next higher address after each byte of data is output.
- The data output can continue indefinitely.
- The READ command is terminated by driving CE# high at any time

The following figure shows another SPI transaction with the device.



b. Decode the protocol command.

Solution:

The command is encoded as 0x05 and according to the protocol table, it refers to Read-Status-Register (RDSR).

c. Decode the protocol sequence from the microprocessor's perspective.

Solution:

- The microprocessor must first select the device by driving its CE# low.
- The RDSR command (0x5h) is written to SI.
- The status register information is clocked out as the next byte.