EMBEDDED SYSTEM DESIGN LAB ASSIGNMENT -5

Implementation of SPI on STM8

SUBMITTED BY- SANCHIT AGRAWAL (2016H140106P)
ABHINAV GARG (2016H140103P)

Objective – To learn the implementation of SPI for STM8S105C6 microcontroller.

Project Workspace: -

- Used IAR embedded workbench with embedded C coding and used ST Visual Programmer for programming the STM8S micro controller

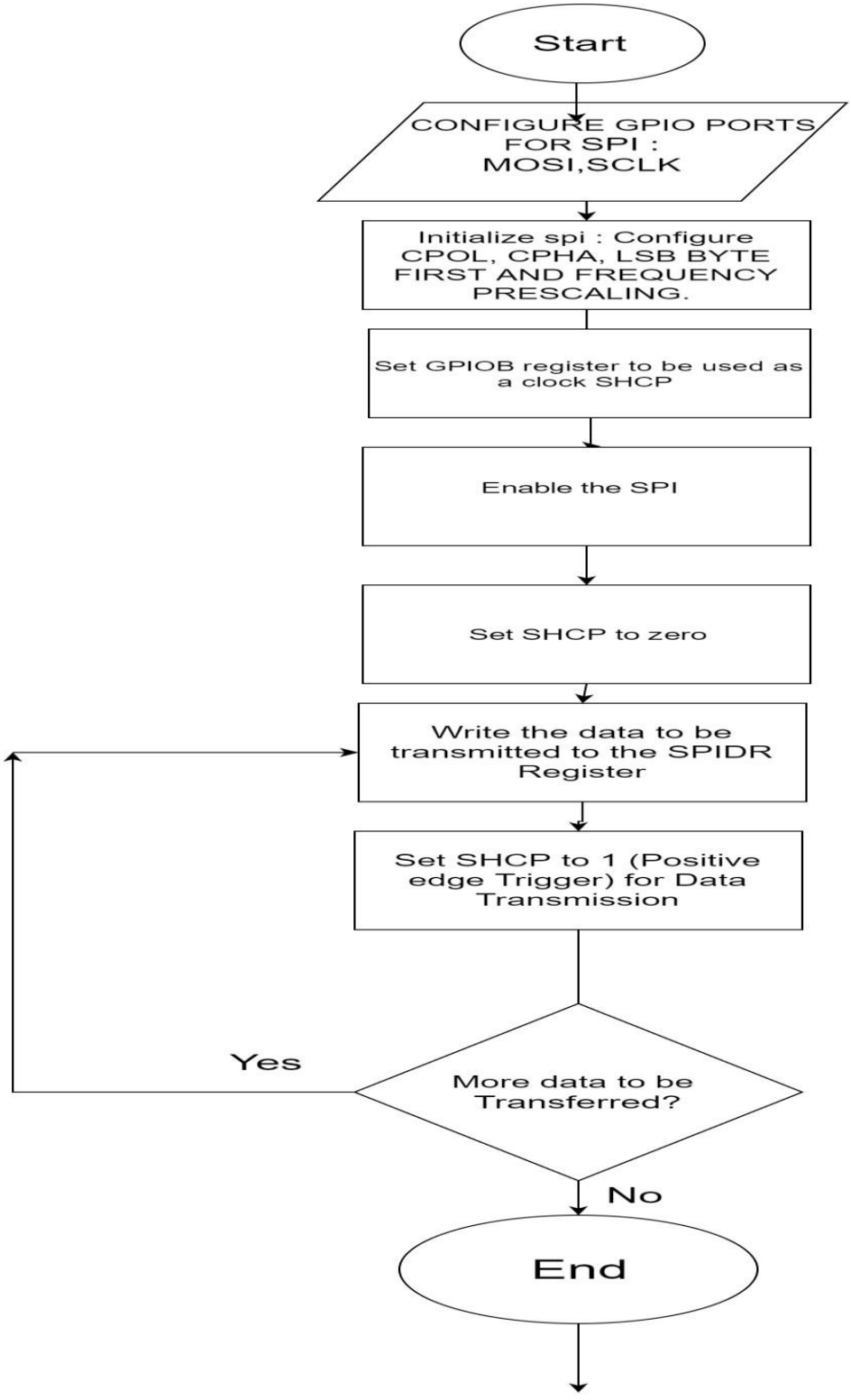
Critical Issues: -

- SHCP should be triggered from 'low' to 'high' on every data transmission to 74HC595,
- Synchronization of data transmission with the clock.

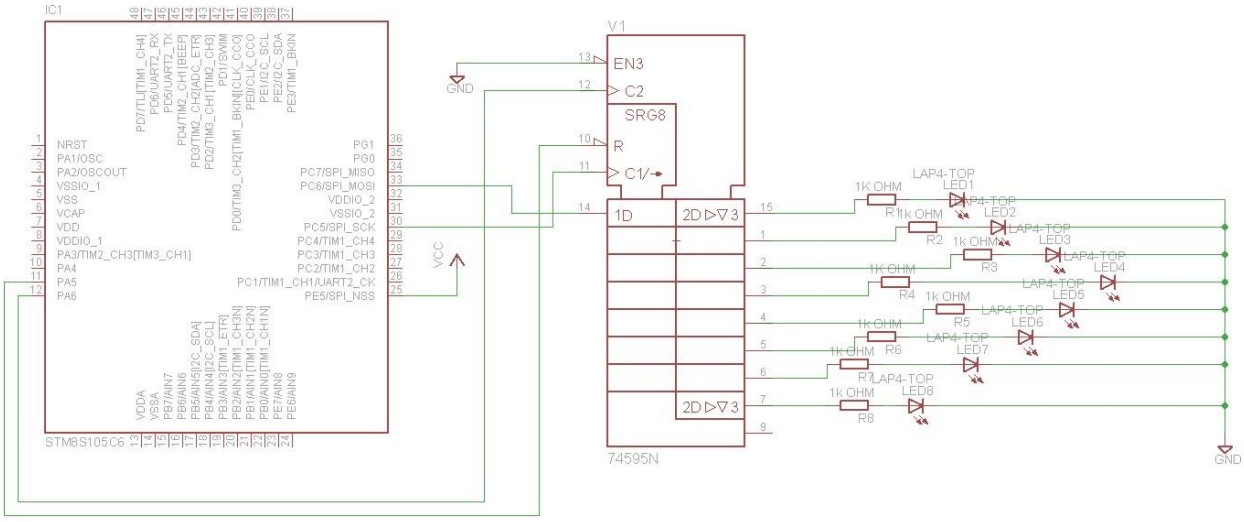
Methodology – SPI mode of microcontroller is enabled and write the data in the Shift register. Then transmit data to the IC serially and latch that data to the output pins.

Display the data on the led.

FLOW CHART


Start
CONFIGURE GPIO PORTS FOR SPI : MOSI,SCLK
Initialize spi : Configure CPOL, CPHA, LSB BYTE FIRST AND FREQUENCY PRESCALING.
Set GPIOB register to be used as a clock SHCP
Enable the SPI
Set SHCP to zero
Write the data to be transmitted to the SPIDR Register
Set SHCP to 1 (Positive edge Trigger) for Data Transmission
More data to be Transferred?
Yes
No
End

HARDWARE DESIGN –



Schematic diagram of STM8S105C6 showing interfacing with 74HC595

- Connect LEDs to the output pins of IC 74HC595 through resistors.

- Connect MOSI pin (PC6) of microcontroller to the DS pin of IC 74HC595 (pin 14) .
- PA6 is connected to pin 12 (ST_CP) in 74HC595 IC.
- PC5 (SPI_CLK) is connected to pin 11 (SH_CP) of IC 74HC595 .
- OE_BAR is connected to logic.
- SPI_NSS is connected to LOGIC HIGH.
  MR_BAR connected to PA5.
- In the schematic supply line pins are not connected because they are connected internally in the PCB design.

SOFTWARE DESIGN

SPI is configured as given below,

1. STM8S105C6 is configured as master (MOSI).
2. MSB is sent first, by resetting the LSBFRST bit of SP_CR1 register
3. fclk is divided by 256 by setting BRR of SPI_CR1 register to use it in slow mode to reduce chances of error.
4. MSTR bit of SPI_CR1 register is set to use it in master mode
5. CPOL and CPHL bits are set to 0 & 0 respectively so that data is captured in rising edge and transmitted in the falling edge of CPHL.
6. SPI is enabled by writing 1 on SPE of SPI_CR1.
7. Write the data to be transmitted is put on SPI_DR register, and then wait until data is transmitted from TX_buffer.
8. Disable SPI by resetting the SPE of SPI_CR1.
9. PA5 is configured as output port to latch STCP

OBSERVATION-

Data is successfully transmitted through SPI. The LEDs were glown according to the data sent.

RESULT-

SPI was successfully used to transmit serial data to the port and it was perfectly latched to glow the LEDs.

References: -

- RM0016 – STM8S105C6 user manual
- STM8S Datasheet for connections.