

4-pin SPI initialization & Bitmap Indexing

Mentor: Mr. Gaurav Tyagi

Aim: To initialize 4-pin SPI communication for LCD & create bitmap indexing for graphics

Walkthrough:

For reducing cost of LCD panel, which is currently being used in Solar Inverter, a new Graphical LCD panel was to be used in place of usual LCD. For sending signals from the controller Serial Peripheral Interface.

Serial Peripheral Interface (SPI)

SPI is a synchronous serial communication interface used for short distance communication. It communicates in full duplex mode using master-slave architecture. In our case there would be a single slave, the LCD panel. The data is transferred is through a single wire, where the bits are sent one-by-one. There are other three pins, which would be for sending clock signals, slave select and control select pin.

We would use R5F100FC microcontroller, and define pins for the communication.

```
#define RESET P14_bit.no6      // lcd res
#define A0 P14_bit.no7        // lcd a0
#define ENABLE P2_bit.no7     // slave select
```

PIN10 and PIN12 would be used for clock signals and data signals respectively.

For setting up connections, we first need to create clock, ports, serial communication and watchdog timer by calling IDE defined functions.

```
R_CGC_Get_ResetSource();
R_CGC_Create();
R_PORT_Create();
R_SAU0_Create();
R_WDT_Create();
```

Start serial communication and reset and enable the LCD.

```
R_CSI00_Start();
RESET = 0;
Delay(100);
RESET = 1;    // lcd reset
ENABLE = 1;
Delay(100);
ENABLE = 0;   // slave select enabled
LCDinit();
```

LCDinit() function will be sending commands to initialize the screen which are stored in an instruction set insSet2[].

```
unsigned char insSet2[]={0xa4,0xa6,0xa0,0xc8,0x40,0x25,0x81,0x1b,0xa2,0x2c,0x2e,0x2f,0xaf};
```

```

void LCDinit(void){
    Delay(50);
    ENABLE = 1;
    Delay(100);
    ENABLE = 0;    // slave select enabled
    Delay(50);
    RESET = 0;
    Delay(100);
    RESET = 1;    // lcd reset
    Delay(50);
    R_CSI00_Send(&insSet2[0], 8);
    R_CSI00_Send(&insSet2[1], 8);
    R_CSI00_Send(&insSet2[2], 8);
    R_CSI00_Send(&insSet2[3], 8);
    R_CSI00_Send(&insSet2[4], 8);
    R_CSI00_Send(&insSet2[5], 8);
    R_CSI00_Send(&insSet2[6], 8);
    R_CSI00_Send(&insSet2[7], 8);
    R_CSI00_Send(&insSet2[8], 8);
    R_CSI00_Send(&insSet2[9], 8);
    R_CSI00_Send(&insSet2[10], 8);
    R_CSI00_Send(&insSet2[11], 8);
    R_CSI00_Send(&insSet2[12], 8);
}

```

These instructions are explained in the datasheet of the LCD. In such a way LCD is initialized.

Bitmap Indexing

Bitmap Indexing is an indexing technique which uses single key, to determine whether a function should be called or not. In our case, the pixel can be either be glowing or not glowing. So, for different graphics we will create arrays having 0, 1 as elements. If 1 is found at a point, pixel needs to be glowed.

But there is a problem, for large arrays the space in the RAM, that is 2KB would be less. So, we would break the arrays and call them one after other from the ROM, so RAM does not have overflow.

Now, for printing



Figure can be divided into smaller parts and smaller instances can be called to create the figure

```

void page1_inst1(){
    int r= 15;
    int c1=18, c2 = 23;
    int k=0;
    int key[]={0,1,1,1,1,1,1};
    for(int c = c1; c<=c2;c++){
        if(key[k]==1) LED[r][c]='#';
        k++;
    }
}

void page1_inst2(){
    int r= 15;
    int c1=105, c2 = 120;
    int k=0;
    int key[]={1,1,1,1,1,0,0,1,1,1,0,1,1,0,1,1};
    for(int c = c1; c<=c2;c++){
        if(key[k]==1) LED[r][c]='#';
        k++;
    }
}

void page2_inst1(){
    int r1=16, r2 = 17 ;
    int c1=18, c2 = 23;
    int key[]={0,1,1,1,1,1,1,1,1,1,1};
    int k =0;
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

void page2_inst2(){
    int r1=16, r2 = 17 ;
    int c1=105, c2 = 120;
    int key[]={1,1,1,1,1,0,0,0,1,0,0,1,0,1,0,1,1,1,1,1,0,0,0,1,0,0,1,0,0,0,1};
    int k =0;

```

```

for(int r=r1; r<=r2;r++){
    for(int c =c1; c<=c2;c++){
        if(key[k]==1){
            LED[r][c]='#';
        }
        k++;
    }
}

}

void page2_inst3(){
    int r1=18, r2 = 19 ;
    int c1=17, c2 = 22;
    int key[]={0,1,1,1,1,1,0,1,1,1,1,1,1,0};
    int k =0;
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

void page2_inst4(){
    int r1=18, r2 = 19 ;
    int c1=105, c2 = 109;
    int key[]={1,1,1,1,1,1,1,1,1,1};
    int k =0;
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

void page2_inst5(){
    int r1=20, r2 = 23 ;
    int c1=17, c2 = 109;

```

```

int k =0;
int key[]={ 1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,0,1,1,1,1,1,0,0,1,1,
            1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,
            1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,
            1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,
            1,1,1,1,0,1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,1,1,1,1,1,
            1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1,1,1,
            1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1,1,1,
            1,1,1,1,1,0,0,1,1,1,1,1,1,1,0,0,1,1,1,1,1,1,1,
            1,1,1,1,1,1,1,1,1,0,0,0,0,0,1,1,1,1,1,0,1,1,1,1,0,0,
            1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,
            1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,
            1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,
            0,0,1,1,1,1,1,1,0,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,
            1,1,0,1,1,1,1,1,1,0,1,1,1,1,1,1,0,0,1,1,1,1,0,1,1
            ,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,1,1,1,1,
            1,0,0,1,1,1,1,1,0

};

for(int r=r1; r<=r2;r++){
    for(int c =c1; c<=c2;c++){
        if(key[k]==1){
            LED[r][c]='#';
        }
        k++;
    }
}

}

void page3_inst1(){
    int r1=24, r2 = 27 ;
    int c1=16, c2 = 108;
    int k =0;

    int key[] = { 1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,0,1,1,1,1,1,0,0,0,1,1,
                  1,1,0,1,1,1,1,1,0,0,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,
                  1,1,1,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,0,1,
                  1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,
                  0,0,1,1,1,1,1,1,1,0,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,
                  1,0,1,1,1,1,1,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,
                  1,0,0,1,1,1,1,1,0,0,0,1,1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,
                  1,1,1,0,0,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,0,0,

```

```

        1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,
        0,0,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,0,0,1,1,1,1,
        1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,0,0,0,
        1,1,1,1,0,0,1,1,1,1,0,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,
        1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,0,0,1,1,1,1,1,0,1,
        1,1,1,1,1,0,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1};

for(int r=r1; r<=r2;r++){
    for(int c =c1; c<=c2;c++){
        if(key[k]==1){
            LED[r][c]='#';
        }
        k++;
    }
}

void page3_inst2(){

int key[]={ 0,1,1,1,1,1,0,0,0,1,1,1,1,1,1,0,0,0,1,1,1,0,0,1,1,1,
            1,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,0,0,1,1,
            1,1,1,0,1,1,1,1,1,0,0,0,1,1,1,1,0,1,1,1,1,1,0,0,0,0,
            1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,1,1,1,1,
            1,0,0,0,1,1,1,0,1,1,1,1,1,0,0,1,1,1,1,1,1,0,1,1,1,1,1,
            1,0,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,1,1,1,0,0,1,1,1,
            1,1,0,1,1,1,1,1,0,0,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,1,1,
            1,1,1,1,1,1,0,1,1,1,1,1,1,0,0,0,1,1,1,0,1,1,1,1,0,0,0,
            1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,
            0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,0,0,0,1,1,1,
            1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,0,0,
            0,1,1,0,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,1,
            1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,
            1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1};

int r1=28, r2 = 31 ;
int c1=15, c2 = 107;
int k =0;

for(int r=r1; r<=r2;r++){
    for(int c =c1; c<=c2;c++){
        if(key[k]==1){
            LED[r][c]='#';
        }
        k++;
    }
}

```

```

    }
}
}

void page4_inst1(){
    int r1=32, r2 = 32 ;
    int c1=15, c2 = 107;
    int k =0;
    int key[] ={1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,0,0,0,1,1,
                0,1,1,1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,
                0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,
                1,1,0,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,
                1,1,1,1,1,1,1,1,1,1};
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

void page4_inst2(){
    int r1=33, r2 = 37 ;
    int c1=46, c2 = 56;
    int k =0;
    int key[] ={0,0,0,0,0,0,0,1,1,1,1,1,
                1,1,1,1,1,1,1,1,1,1,1,1,
                1,1,1,1,1,1,1,1,1,1,0,
                1,1,1,1,1,1,1,1,1,1,0,
                1,1,1,1,1,1,1,1,0,0,0};
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}
}

```

[illegible]


```

        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

```

```

void page6_inst1(){
    int r1=48, r2 = 48 ;
    int c1=60, c2 = 61;
    int k =0;
    int key[] ={1,1};
    for(int r=r1; r<=r2;r++){
        for(int c =c1; c<=c2;c++){
            if(key[k]==1){
                LED[r][c]='#';
            }
            k++;
        }
    }
}

```

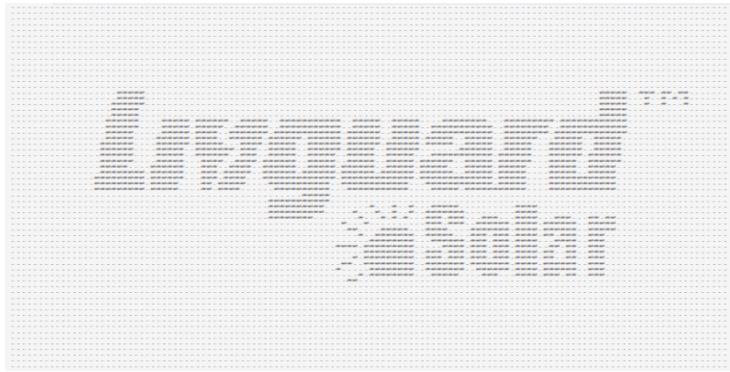
```

void PRINT_LIVGUARD(){
    page1_inst1();
    page1_inst2();
    page2_inst1();
    page2_inst2();
    page2_inst3();
    page2_inst4();
    page2_inst5();
    page3_inst1();
    page3_inst2();
    page4_inst1();
    page4_inst2();
    page4_inst3();
    page5_inst1();
    page6_inst1();

}

```

This code will have following output:



Similarly, we will create graphics for all other icons were created.

Result: 4-pin SPI was initialized and bitmap indexing was completed for graphics.