Q1)

A main code and two subroutines, GPIO initialization & Signal Send, are provided in the following codes:

GPIO Initialization

```
GPIO_PORTB_DATA     EQU 0x400053FC
GPIO_PORTB_DIR              EQU 0x40005400
GPIO_PORTB_AFSEL    EQU 0x40005420
GPIO_PORTB_DEN              EQU 0x4000551C
GPIO_PORTB_PUR              EQU 0x40005510
GPIO_PORTB_PDR              EQU 0x40005514
IOB                         EQU 0x0F
SYSCTL_RCGCGPIO     EQU 0x400FE608
;**************************************************************
; Program section
;**************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
;LABEL DIRECTIVE VALUE COMMENT
                        AREA rutins , CODE, READONLY
                        THUMB
                        EXPORT InitGPIO ;
InitGPIO        PROC
                        LDR R1 , =SYSCTL_RCGCGPIO
                        LDR R0 , [ R1 ]
                        ORR R0 , R0 , #0x2;Port B clock enabled
                        STR R0 , [ R1 ]
                        NOP             ;Wait for clock to stabilize
                        NOP
                        NOP
                        LDR R1 , =GPIO_PORTB_DIR ; Config of Port B starts
                        LDR R0 , [ R1 ]
                        BIC R0 , #0xFF
                        ORR R0 , #IOB;00001111 1->output
                        STR R0 , [ R1 ]
                        LDR R1 , =GPIO_PORTB_AFSEL
                        LDR R0 , [ R1 ]
                        BIC R0 , #0xFF
                        STR R0 , [ R1 ]
                        LDR R1 , =GPIO_PORTB_DEN
                        LDR R0 , [ R1 ]
                        ORR R0 , #0xFF
                        STR R0 , [ R1 ]
                        LDR R1 , =GPIO_PORTB_PUR
                        LDR R0 , [ R1 ]
                        ORR R0 , #0xF0
                        STR R0 , [ R1 ]
```

```
                    BX LR; end

;************************************************************
; End of the program  section
;************************************************************
;LABEL     DIRECTIVE     VALUE                COMMENT
                    ALIGN
                    ENDP
                    END
```

Signal Send

```
GPIO_PORTB_DATA     EQU 0x400053FC
;************************************************************
; Program section
;************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
;LABEL DIRECTIVE VALUE COMMENT
                    AREA rutins , CODE, READONLY
                    THUMB
                    EXPORT SignalSend ;
SignalSend      PROC
                    ORR             R0, R0, #0xF0 ; Data is set
                    LDR             R1,=GPIO_PORTB_DATA; Data address in R1
                    STR             R0,[R1];          Corresponding Outputs set high
                    BX LR; end

;************************************************************
; End of the program  section
;************************************************************
;LABEL     DIRECTIVE     VALUE                COMMENT
                    ALIGN
                    ENDP
                    END
```

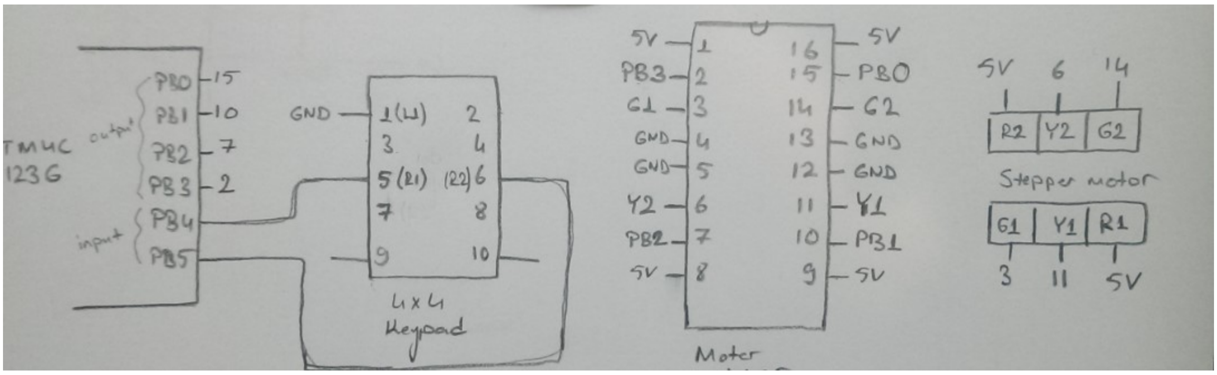
Main Code

```
;************************************************************
; Program_Directives.s
; Copies the table from one location
; to another memory location.
; Directives and Addressing modes are
; explained with this program.
;************************************************************
;************************************************************
; EQU Directives
; These directives do not allocate memory
;************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
OFFSET          EQU     0x10
FIRST           EQU             0x20000480
```

```
STORE           EQU                     0x20000410
;*************************************************************
; Directives - This Data Section is part of the code
; It is in the read only section  so values cannot be changed.
;*************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
        AREA      sdata, DATA, READONLY
        THUMB
CTR1    DCB     0x10
MSG     DCB     "Copying table..."
                        DCB                     0x0D
                        DCB                     0x04
;*************************************************************
; Program section
;*************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
                        AREA    main, READONLY, CODE
                        THUMB
                        EXTERN          OutStr  ; Reference external subroutine
                        EXTERN          InChar; Serial input Added
                        EXTERN          SignalSend;      GPIO signal send
                        EXTERN          InitGPIO; GPIO initialize
                        EXPORT          __main ; Make available

__main
                        BL              InitGPIO; GPIO initialized
Begin           MOV     R0, #0x0001; Set for Full Step config
                        BL              SignalSend;
                        B               Begin;
;*************************************************************
; End of the program  section
;*************************************************************
;LABEL     DIRECTIVE      VALUE                 COMMENT
                        ALIGN
                        END
```

Q2)



Q3)

A main code with an additional subroutines, InputCheck, are provided.

InputCheck

```
GPIO_PORTB_DATA       EQU 0x400053FC
;************************************************************
; Program section
;************************************************************
;LABEL            DIRECTIVE       VALUE           COMMENT
;LABEL DIRECTIVE VALUE COMMENT
                        AREA rutins , CODE, READONLY
                        THUMB
                        EXPORT CheckInput ;
                        EXTERN delay;
CheckInput        PROC
                        LDR               R1,=GPIO_PORTB_DATA; Data address in R1
                        MOV               R2,#0; R2 holds the information cw or ccw
Check             LDR   R0,[R1];Checks for any input
                        LSR               R0,#4;
                        LSRS      R0,#1;
                        BCC               Delay100
                        LSRS      R0,#1;
                        BCC               Delay100
                        ;LSRS     R0,#1;
                        ;BCC              Delay100
                        ;LSRS     R0,#1;
                        ;BCC              Delay100
                        B                 Check
Delay100          MOV32 R0,#1600000;If any input is detected
                        PUSH{LR}
                        BL                delay
                        POP{LR}
                        LDR       R0,[R1];          Check Again
                        LSR               R0,#4;
                        LSRS      R0,#1;
                        MOVCCR2,#1; 1 is the cw direction PB4 pressed
                        BCC               Released ; If input is detected again wait for relase
                        LSRS      R0,#1;
                        MOVCCR2,#2; 2 is the ccw direction PB5 pressed
                        BCC               Released
                        BX LR; if no signal


Released          LDR   R0,[R1];          It checks for if the switch is open again
                        LSR               R0,#4;
                        LSRS      R0,#1;
                        BCC               Released;         If it is not open
                        LSRS      R0,#1;
                        BCC               Released;
                        BX LR; end

;************************************************************
; End of the program  section
;************************************************************
;LABEL     DIRECTIVE      VALUE                 COMMENT
                        ALIGN
                        ENDP
                        END
```

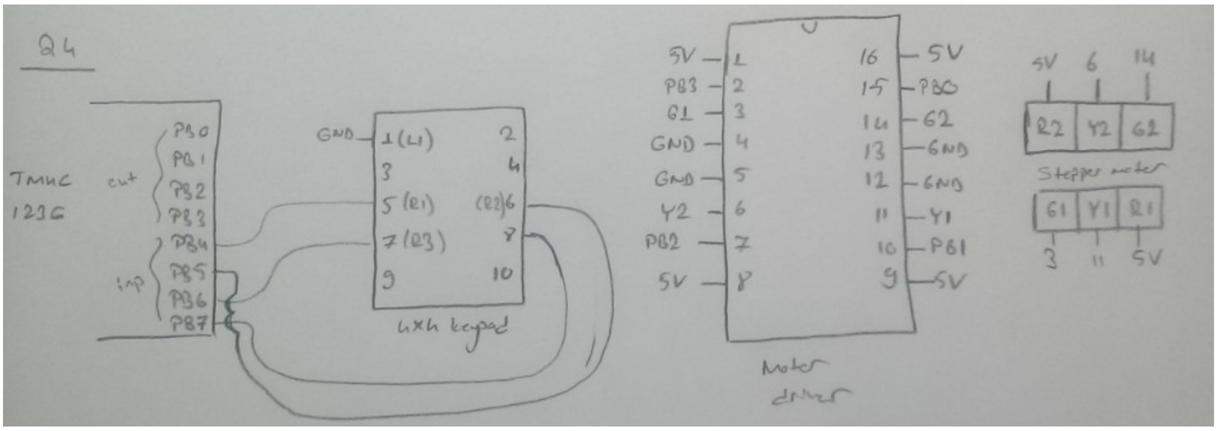
Main Code

```
;************************************************************
; Program_Directives.s
; Copies the table from one location
; to another memory location.
; Directives and Addressing modes are
; explained with this program.
;************************************************************
;************************************************************
; EQU Directives
; These directives do not allocate memory
;************************************************************
;LABEL           DIRECTIVE       VALUE           COMMENT
OFFSET           EQU     0x10
FIRST            EQU             0x20000480
STORE            EQU                     0x20000410
;************************************************************
; Directives - This Data Section is part of the code
; It is in the read only section  so values cannot be changed.
;************************************************************
;LABEL           DIRECTIVE       VALUE           COMMENT
        AREA      sdata, DATA, READONLY
        THUMB
CTR1    DCB     0x10
MSG     DCB     "Copying table..."
                        DCB                     0x0D
                        DCB                     0x04
;************************************************************
; Program section
;************************************************************
;LABEL           DIRECTIVE       VALUE           COMMENT
                        AREA    main, READONLY, CODE
                        THUMB
                        EXTERN          OutStr  ; Reference external subroutine
                        EXTERN          InChar; Serial input Added
                        EXTERN          SignalSend;      GPIO signal send
                        EXTERN          InitGPIO; GPIO initialize
                        EXTERN          delay; Delay is available
                        EXTERN          CheckInput ; Input Check is available
```

```
            EXPORT  __main ; Make available

__main
            BL      InitGPIO; GPIO initialized
            MOV     R3,#0x00000001;
            MOV     R6,#0;
            BL      SignalSend;
Begin   BL      CheckInput;
            CMP R2,#0
            BEQ     Begin;
            CMP     R2,#1
            BEQ     CW
            CMP     R2,#2
            BEQ     CCW
            B       Begin
CW          AND     R3,R3,#15;
            CMP     R3,#1
            MOVEQ   R3,#8;
            LSRNE R3,R3,#1;
            MOV     R6,R3;
            BL      SignalSend;
            B       Begin;
CCW         AND     R3,R3,#15;
            CMP     R3,#8
            MOVEQ   R3,#1;
            LSLNE R3,R3,#1;
            MOV     R6,R3;
            BL      SignalSend;
            B       Begin;
;***********************************************************
; End of the program  section
;***********************************************************
;LABEL     DIRECTIVE      VALUE          COMMENT
            ALIGN
            END
```

Q4)



Q5)

Main code with interrupt changes are provided and modified InputCheck subroutine is provided also.

ISR Code

```
NVIC_ST_RELOAD          EQU 0xE000E014
;***************************************************************
; Program section
;***************************************************************
;LABEL          DIRECTIVE       VALUE           COMMENT
;LABEL DIRECTIVE VALUE COMMENT
                        AREA rutins , CODE, READONLY
                        ;THUMB
                        EXTERN SignalSend;
                        EXPORT DriveMotor ;
DriveMotor      PROC
                        MOV             R4,#2;
                        CMP             R2,#0;
                        BX              LR;
                        CMP             R2,#1
                        BEQ             CW
                        CMP             R2,#2
                        BEQ             CCW
                        CMP             R2,#3
                        BEQ             SpeedUp
                        CMP             R2,#4
                        BEQ             SpeedDown
CW                      AND             R3,R3,#15;
                        CMP             R3,#1
                        MOVEQ           R3,#8;
                        LSRNE   R3,R3,#1;
                        MOV             R6,R3;
                        PUSH    {LR}
                        BL              SignalSend;
                        POP             {LR}
                        BX              LR; end
CCW                     AND             R3,R3,#15;
```

```
                        CMP             R3,#8
                        MOVEQ           R3,#1;
                        LSLNE   R3,R3,#1;
                        MOV             R6,R3;
                        PUSH    {LR}
                        BL              SignalSend;
                        POP             {LR}
                        BX              LR; end
SpeedUp                 LDR     R1,=NVIC_ST_RELOAD
                        LDR             R0,[R1]
                  UDIV R0,R0,R4 ;Interrupt period is decreased
                        STR             R0,[R1]
                        MOV             R2,R7;
                        BX              LR; end
SpeedDown       LDR     R1,=NVIC_ST_RELOAD
                        LDR             R0,[R1]
                        MUL             R0,R0,R4 ;Interrupt period is increased
                        STR             R0,[R1]
                        MOV             R2,R7;
                        BX              LR; end

;***************************************************************
; End of the program  section
;***************************************************************
;LABEL      DIRECTIVE       VALUE                   COMMENT

                        ENDP
                        END
```

StartUp.s Change

```
                                IMPORT          DriveMotor
SysTick_Handler PROC
         EXPORT  SysTick_Handler          [WEAK]
                                PUSH {LR}
                                BL DriveMotor
                                POP {LR}
                                BX        LR
         ENDP
```

InputCheck

```
GPIO_PORTB_DATA       EQU 0x400053FC
;***************************************************************
; Program section
;***************************************************************
;LABEL             DIRECTIVE        VALUE             COMMENT
;LABEL DIRECTIVE VALUE COMMENT
                        AREA rutins , CODE, READONLY
                        THUMB
                        EXPORT CheckInput ;
                        EXTERN delay;
```

```
CheckInput      PROC
                        LDR             R1,=GPIO_PORTB_DATA; Data address in R1
                        MOV             R2,#0; R2 holds the information cw or ccw
Check           LDR     R0,[R1];Checks for any input
                        LSR             R0,#4;
                        LSRS    R0,#1;
                        BCC             Delay100
                        LSRS    R0,#1;
                        BCC             Delay100
                        LSRS    R0,#1;
                        BCC             Delay100
                        LSRS    R0,#1;
                        BCC             Delay100
                        BX              LR; End
Delay100        MOV32   R0,#1600000;If any input is detected
                        PUSH{LR}
                        BL              delay
                        POP{LR}
                        LDR     R0,[R1];        Check Again
                        LSR             R0,#4;
                        LSRS    R0,#1;
                        MOVCCR2,#1; 1 is the cw direction PB4 pressed
                        MOVCC R7,#1;
                        BCC             Released ; If input is detected again wait for relase
                        LSRS    R0,#1;
                        MOVCC R2,#2; 2 is the ccw direction PB5 pressed
                        MOVCC R2,#2;
                        BCC             Released
                        LSRS    R0,#1;
                        MOVCCR2,#3; 1 is speed up, PB6 pressed
                        BCC             Released ; If input is detected again wait for relase
                        LSRS    R0,#1;
                        MOVCCR2,#4; 2 is speed down, PB7 pressed
                        BCC             Released
                        BX LR; if no signal


Released        LDR     R0,[R1];        It checks for if the switch is open again
                        LSR             R0,#4;
                        LSRS    R0,#1;
                        BCC             Released;       If it is not open
                        LSRS    R0,#1;
                        BCC             Released;
                        LSRS    R0,#1;
                        BCC             Released;       If it is not open
                        LSRS    R0,#1;
                        BCC             Released;
                        BX LR; end
```

```
;************************************************************
; End of the program  section
;************************************************************
;LABEL      DIRECTIVE       VALUE                   COMMENT
                            ALIGN
                            ENDP
                            END
```

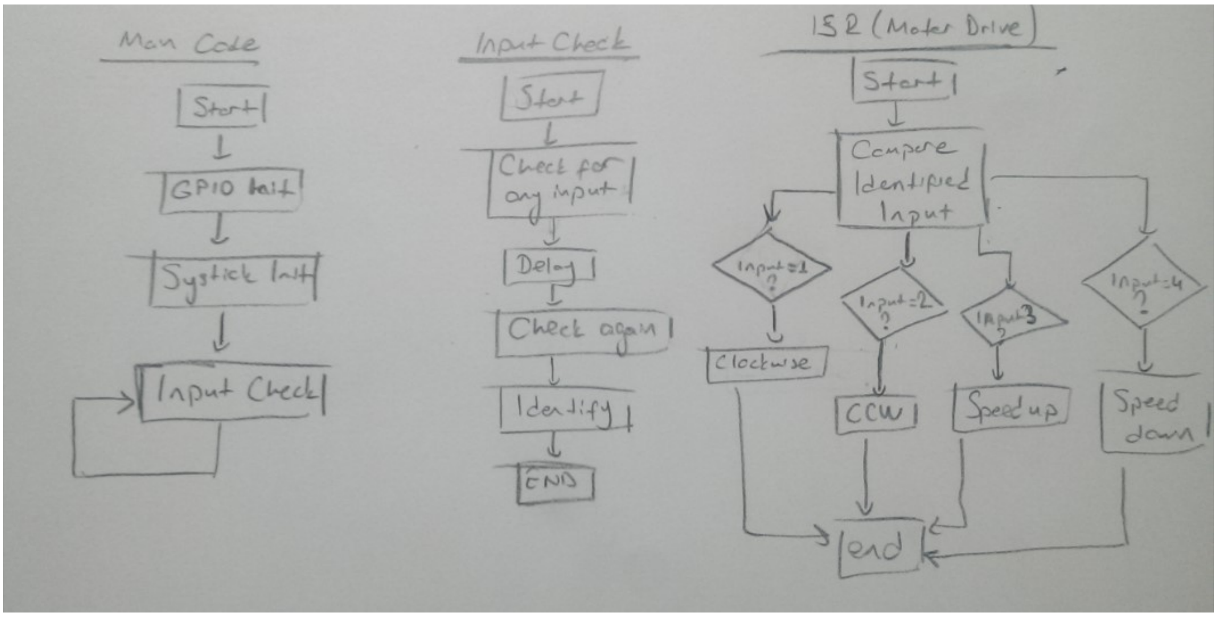
Main Code

```
;************************************************************
; Program_Directives.s
; Copies the table from one location
; to another memory location.
; Directives and Addressing modes are
; explained with this program.
;************************************************************
;************************************************************
; EQU Directives
; These directives do not allocate memory
;************************************************************
;LABEL            DIRECTIVE       VALUE           COMMENT
OFFSET            EQU     0x10
FIRST             EQU             0x20000480
STORE             EQU                     0x20000410
;************************************************************
; Directives - This Data Section is part of the code
; It is in the read only section  so values cannot be changed.
;************************************************************
;LABEL            DIRECTIVE       VALUE           COMMENT
          AREA     sdata, DATA, READONLY
          THUMB
CTR1    DCB     0x10
MSG     DCB     "Copying table..."
                            DCB                     0x0D
                            DCB                     0x04
;************************************************************
; Program section
;************************************************************
;LABEL            DIRECTIVE       VALUE           COMMENT
                            AREA    main, READONLY, CODE
                            THUMB
                            EXTERN          OutStr  ; Reference external subroutine
                            EXTERN          InChar; Serial input Added
                            EXTERN          SignalSend;     GPIO signal send
                            EXTERN          InitGPIO; GPIO initialize
                            EXTERN          delay; Delay is available
                            EXTERN          CheckInput ; Input Check is available
                            EXTERN          InitSysTick;
```

```
                    EXPORT      __main ; Make available

__main
                    BL          InitGPIO; GPIO initialized
                    MOV         R3,#0x00000001;
                    MOV         R6,#0;
                    MOV         R7,#0;
                    BL          InitSysTick;
                    CPSIE   I;

Begin           BL          CheckInput;
                    B           Begin
;*************************************************************
; End of the program  section
;*************************************************************
;LABEL      DIRECTIVE       VALUE               COMMENT
                    ALIGN
                    END
```



How I approached is that, my input check subroutine can identify the input type, CW, CCW, SpeedUp, SpeedDown, R2 register gets 1,2,3,4 w.r.t. And this subroutine is called in main subroutine continuously. When an interrupt occurs, it looks the what is last input and it processes.