```
GPIO_PORTB_DATAIN   EQU     0x4000503C  ;data address to all pins
GPIO_PORTB_DATAOUT  EQU     0x400053C0
GPIO_PORTB_DIR      EQU     0x40005400
GPIO_PORTB_AFSEL    EQU     0x40005420
GPIO_PORTB_AMSEL    EQU     0x40005428
GPIO_PORTB_DEN      EQU     0x4000551C
GPIO_PORTB_PUR      EQU     0x40005510
IOB                 EQU     0xF0
PUB                 EQU     0x0F

SYSCTL_RCGCGPIO     EQU     0x400FE608


            AREA        main, READONLY, CODE
            THUMB
            EXTERN      delay
            EXPORT      __main  ; Make available

__main      PROC
            LDR         R1, =SYSCTL_RCGCGPIO
            LDR         R0,[R1]
            ORR         R0,#0x02
            STR         R0,[R1]

            NOP
            NOP
            NOP                                 ;Stabilize clock

            LDR         R1,=GPIO_PORTB_DIR
            LDR         R0,[R1]
            BIC         R0,#0xFF                                    ;INPUTS         OUTPUTS
            ORR         R0,#IOB                                 ;s1 pb3         d1  pb7
            STR         R0,[R1]                                 ;s2 pb2         d2  pb6
                                                                ;s3 pb1         d3  pb5
            LDR         R1,=GPIO_PORTB_AFSEL                    ;s4 pb0         d4  pb4
            LDR         R0,[R1]
            BIC         R0,#0xFF
            STR         R0,[R1]

            LDR         R1,=GPIO_PORTB_DEN
            LDR         R0,[R1]
            MOV         R0,#0xFF
            STR         R0,[R1]

            LDR         R1,=GPIO_PORTB_AMSEL                ;PORTB initilization part
            LDR         R0,[R1]
            BIC         R0,#0xFF
            STR         R0,[R1]

            LDR         R1,=GPIO_PORTB_PUR
            MOV         R0,#PUB
            STR         R0,[R1]


            LDR         R1,=GPIO_PORTB_DATAOUT
            LDR         R0,[R1]
            ORR         R0,#0xF0
            STR         R0,[R1]

checkrows   LDR         R1,=GPIO_PORTB_DATAIN                   ;Debounce algorithm for pressing
            LDR         R10,[R1]
            CMP         R10,#0x0F                           ;wait a delay between two data
            BEQ         checkrows                           ;samples and if they are the same
            BLNE        delay
            LDR         R1,=GPIO_PORTB_DATAIN                   ;it continues to check columns
            LDR         R9,[R1]
            CMP         R9,R10                              ;it loads the data onto R9 reg.
            BEQ         pressed
            B           checkrows

pressed     LDR         R1,=GPIO_PORTB_DATAIN
            LDR         R10,[R1]
            CMP         R9,R10
            BEQ         pressed
            B           led

```

```
led         LDR         R1,=GPIO_PORTB_DATAOUT
            CMP         R9,#0x07
            BEQ         first
            CMP         R9,#0x0B
            BEQ         second
            CMP         R9,#0x0D
            BEQ         third
            CMP         R9,#0x0E
            BEQ         fourth

first       LDR         R0,[R1]
            ANDS        R5,R0,#0x80
            ORREQ       R0,#0x80
            STREQ       R0,[R1]
            BEQ         checkrows
            BIC         R0,#0x80
            STR         R0,[R1]
            B           checkrows


second      LDR         R0,[R1]
            ANDS        R5,R0,#0x40
            ORREQ       R0,#0x40
            STREQ       R0,[R1]
            BEQ         checkrows
            BIC         R0,#0x40
            STR         R0,[R1]
            B           checkrows

third       LDR         R0,[R1]
            ANDS        R5,R0,#0x20
            ORREQ       R0,#0x20
            STREQ       R0,[R1]
            BEQ         checkrows
            BIC         R0,#0x20
            STR         R0,[R1]
            B           checkrows

fourth      LDR         R0,[R1]
            ANDS        R5,R0,#0x10
            ORREQ       R0,#0x10
            STREQ       R0,[R1]
            BEQ         checkrows
            BIC         R0,#0x10
            STR         R0,[R1]
            B           checkrows

                                                            ;if everything goes fine code prints the
                                                            ;key's character since it already holds
            NOP                                             ;it as ASCII value
            NOP
            NOP


            B           checkrows                               ;code starts over

            ENDP
            ALIGN
            END

```