

```

1      AREA                main, READONLY, CODE
2      THUMB
3      EXTERN              DELAY150
4      EXTERN              PB_INIT
5      EXTERN              OutChar
6      EXPORT              __main
7  WRITE EQU 0x20000400
8  GPIO_PORTB_DATA EQU 0x400053FC ; data address to all pins
9  GPIO_PORTB_DIR EQU 0x40005400
10 GPIO_PORTB_AFSEL EQU 0x40005420
11 GPIO_PORTB_DEN EQU 0x4000551C
12 GPIO_PORTB_PDR EQU 0x40005514
13 GPIO_PORTB_PUR EQU 0x40005510
14 SYSCTL_RCGCGPIO EQU 0x400FE608 ; these are written in Week-6
15
16 __main PROC
17 BL PB_INIT ;Port B init
18 start LDR R2,=GPIO_PORTB_DATA
19 MOV R3,#0xF0
20 MOV R0,#48 ;Define R0 as 0
21 MOV R7,#0
22 STR R3,[R2]
23 LDR R3,[R2] ;Debouncing Effect
24 BL DELAY150
25 LDR R4,[R2]
26 CMP R3,R4 ;If there is no debouncing , continue
27 BNE start
28 LSR R5,R3,#4 ;in this part of the code we search row and
column in only one loop
29 CMP R5,#0xD ;to determine the number of button we use column
number
30 ADDEQ R0,#1 ;if pressed button is in R1 we add to R2 1
31 CMP R5,#0xB ;if pressed button is in R1 we add to R3 2
32 ADDEQ R0,#2
33 CMP R5,#0x7 ;if pressed button is in R1 we add to R4 3
34 ADDEQ R0,#3
35 CMP R5,#0xF ;Determine button is being pressed
36 BNE ROW_finder ;If there is button pressed continue
37 B start
38
39
40 ROW_finder MOV R6,#0x7 ;ROW1 Which means L4
41 STR R6,[R2]
42 NOP
43 NOP
44 NOP
45 LDR R7,[R2]
46 LSR R7,R7,#4 ;Output is taken
47 CMP R7,R5 ;If output is same with R5 we can assure it is
true
48 ADDEQ R0,#12 ;Since we are on L4, we should add 12 to the R0
49 MOV R6,#0xB; ;Same process continues
50 STR R6,[R2]
51 NOP
52 NOP
53 NOP
54 LDR R7,[R2]
55 LSR R7,R7,#4
56 CMP R7,R5
57 ADDEQ R0,#8
58 MOV R6,#0xD
59 STR R6,[R2]
60 NOP
61 NOP
62 NOP
63 LDR R7,[R2]
64 LSR R7,R7,#4
65 CMP R7,R5
66 ADDEQ R0,#4
67 MOV R6,#0xE
68 STR R6,[R2]
69 NOP
70 NOP
71 NOP
72 LDR R7,[R2]
73 LSR R7,R7,#4
74 CMP R7,R5

```

```
75          ADDEQ      R0,#0
76          LDR        R8,=0xF0
77          STR        R8,[R2]
78          NOP
79          NOP
80          NOP
81  out      LDR        R9,[R2]          ;This function determines the if button is keep
being pressed or not
82          NOP
83          NOP
84          NOP
85          CMP        R9,R8
86          BNE        out
87          CMP        R0,#58          ;We have done lots of addition but we did not
take care of letters. If R0 is bigger than 9+48, we should add 7 to get letters
88          BCC        noletter
89          ADD        R0,#7
90  noletter BL        OutChar
91          B          start
92          ALIGN
93          ENDP
94
```