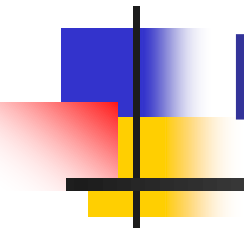
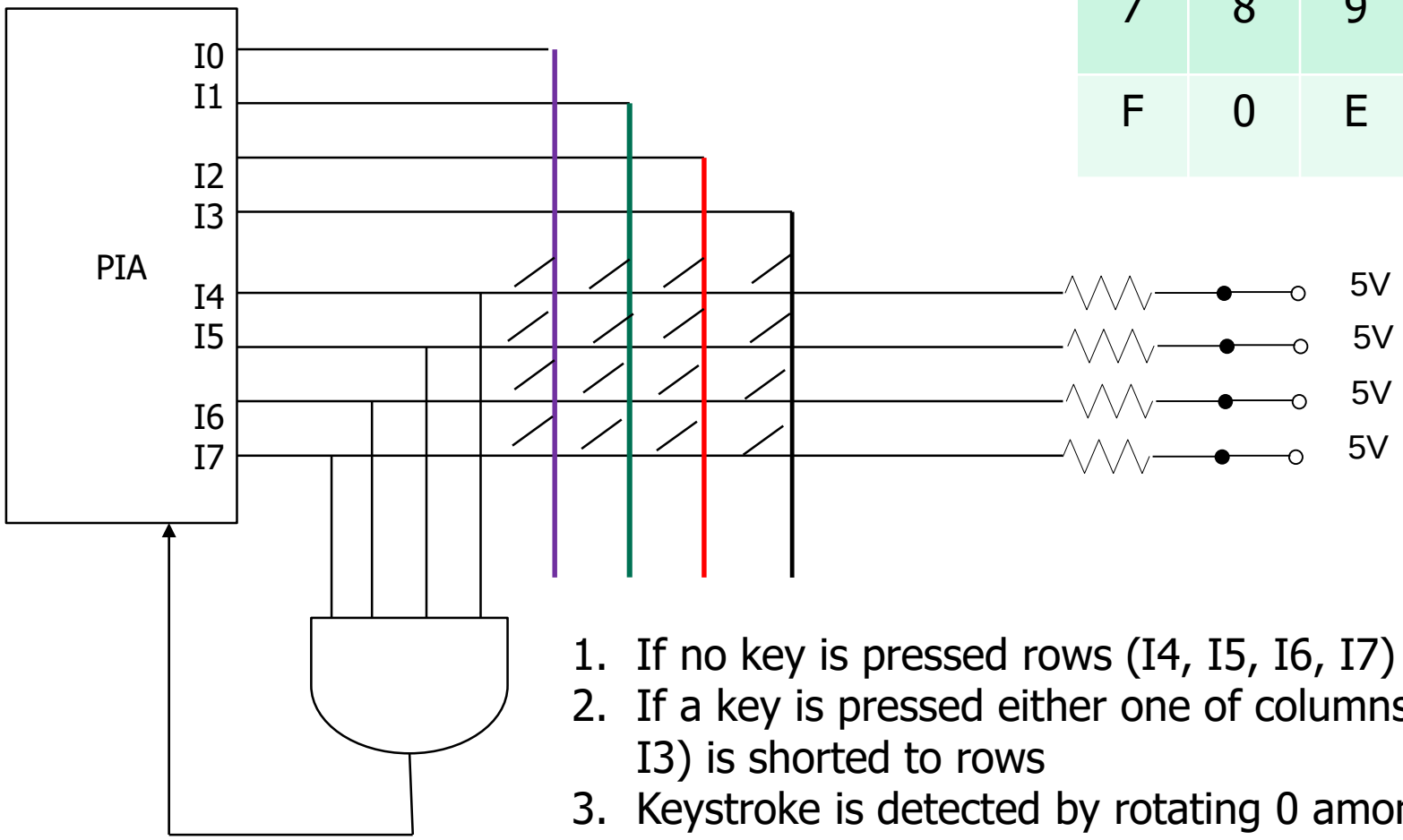


Microprocessor Systems

Dr. Gökhan İnce

Keyboard Design

1	2	3	A
4	5	6	B
7	8	9	C
F	0	E	D



1. If no key is pressed rows (I4, I5, I6, I7) reads high
2. If a key is pressed either one of columns (I0, I1, I2, I3) is shorted to rows
3. Keystroke is detected by rotating 0 among columns

Keyboard Design

Array Index	Row, Column	Symbol	Data in Memory
1	1,1	1	0000 0001
2	1,2	2	0000 0010
3	1,3	3	0000 0011
4	1,4	A	0000 1010
5	2,1	4	0000 0100
6	2,2	5	0000 0101
:			
16	4,4	D	1101 0000

1	2	3	A
4	5	6	B
7	8	9	C
F	0	E	D

Array_Index= (Row-1)*NumberOfColumns+Column

Ex: Array Index of 5:
Row=2; Column=2
Array_Index= 1*4+2=6

Keyboard Design

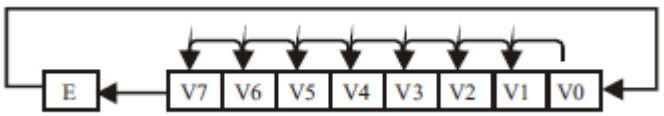
```
COND    LDA A, $0F
        STA A, <DIRECT>
        LDA A, $02
        STA A, <STAT/COND>
        RTS

SHIFT   LDA A, $FE
REW1    STA A, <PORT>
        BSR DELAY
        ROL A
        CMP A, $EF
        BEQ SHIFT
        BR REW1
        RTS
```

```
DELAY   LDA IX, $AAAA
DECR    DEC IX
        BNEQ DECR
        RTS

START   LDA SP, $A000
        BSR COND
REW2    BSR SHIFT
        BR REW2
```

ROL(Rotate Left)



1 1 1 1 1 1 1 0 E=1

1 1 1 1 1 1 0 1 E=1

1 1 1 1 1 0 1 1 E=1

1 1 1 1 0 1 1 1 E=1

Defined variables:
TABLE,
INDEX,
ROW,
COLUMN,
KEY

«Read Key» and «Find Key» subroutines

```
RDKEY   SET E
        LDA B, $FE
REW3    STA B, <PORT>
        LDA A, <PORT>
        AND A, $F0
        CMP A, $F0
        BEQ SCAN
        STA A, ROW
        STA B, COLUMN
        BSR KEY
        RTS
SCAN    ROL B
        CMP B, $EF
        BNEQ REW3
        RTS
```

1.Column: 1111 1110
2.Column: 1111 1101
3.Column: 1111 1011
4.Column: 1111 0111

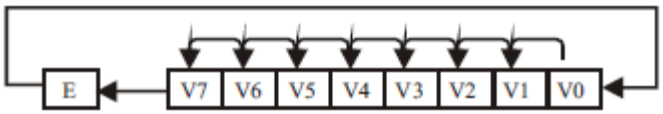
1	2	3	A
4	5	6	B
7	8	9	C
F	0	E	D

B: 1111 1101
A: 1101 1111
B->Column=2
A->Row=2

1111 1110 E=1
1111 1101 E=1
1111 1011 E=1
1111 0111 E=1

```
KEY     CLR E
        CLR C
        COM B
COLNR   INC C
        SHR B
        BNC COLNR
        STA C, COLUMN
        SHR A
        SHR A
        SHR A
        SHR A
        COM A
        CLR E
        CLR C
ROWNR   INC C
        SHR A
        BNC ROWNR
        STA C, ROW
        RTS
```

ROL (Rotate Left)



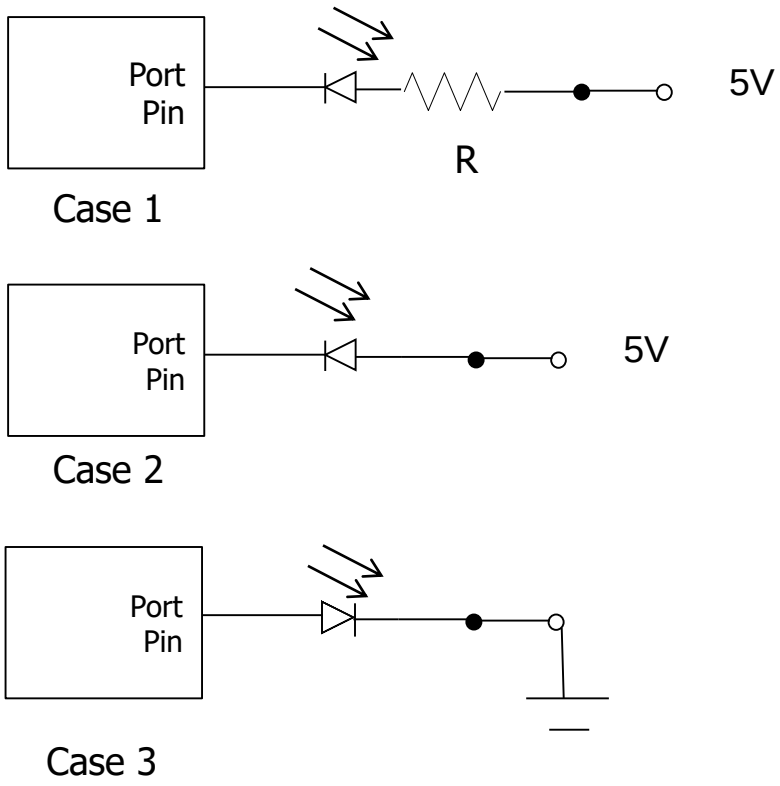
«Compute Key Index» and «Interrupt» subroutines

```
KEYIX   LDA A, <ROW>
        DEC A
        MUL A, $04
        LDA A, <COLUMN>
        ADD A, B
        STA A, INDEX
        RTS
```

```
INTRPT  BSR RDKEY
        BSR KEYIX
        LDA IX, <TABLE>
        CLR C
        LDA D, <INDEX>
        LDA A, <IX+CD+00>
        STA A, KEY
        RTI
```

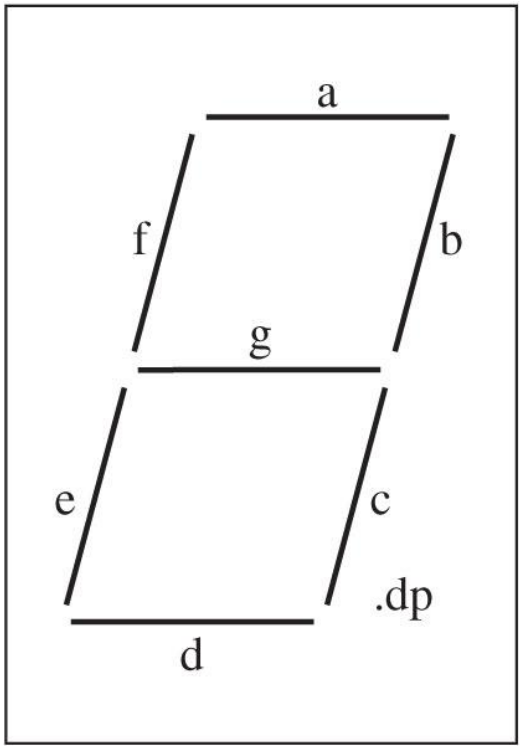
Display Design

- Simple output Device: LED
  - Case-1
    - LED is ON for an output of zero
    - Most LEDs drop 1.7 to 2.5 volts and need about 10ma
    - Current is (5-2)/R
  - Case-2
    - Too much current
    - Failure of Port or LED
  - Case-3
    - Not enough drive (1ma)
    - LED too dim

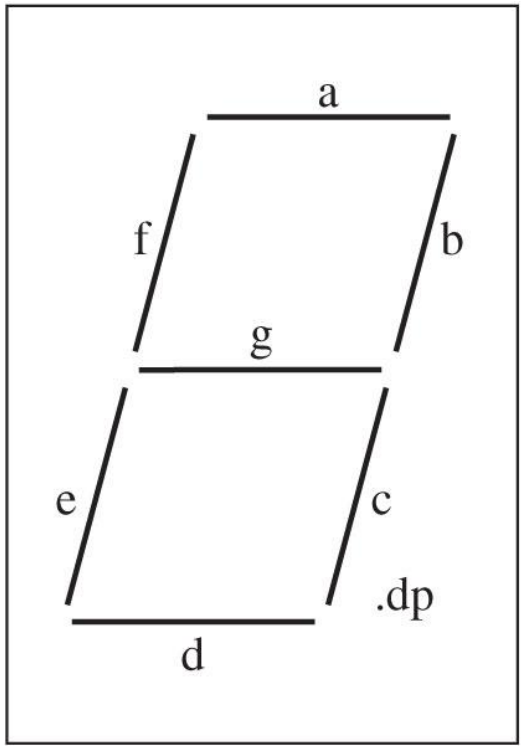


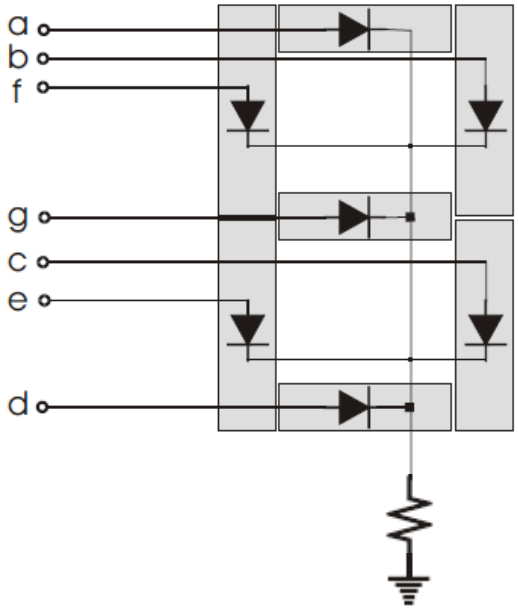
Display Design

- Seven-segment LEDs
  - Often used to display BCD numbers (1 through 9) and a few letters
  - A group of seven LEDs physically mounted in the shape of the number eight
  - Each LED is called a segment and labeled as ‘a’ through ‘g’.



Display Design

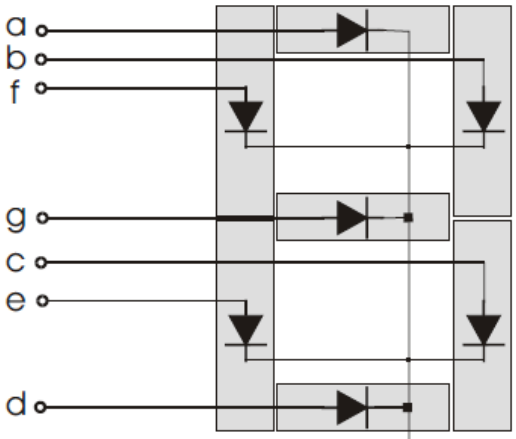




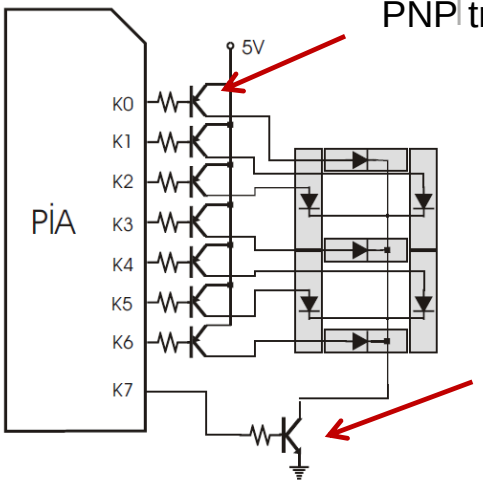
Common Cathode
(Common Ground) Segments need
Logic High to display

- Two types of seven-segment LEDs
  - Common anode
  - Common cathode

Display Design



PNP transistor – Needs 0 to conduct



NPN transistor – Needs 1 to conduct

	K7	K6/d	K5/e	K4/c	K3/g	K2/f	K1/b	K0/a
0	1	0	0	0	1	0	0	0
1	1	1	1	0	1	1	0	1
2	1	0	0	1	0	1	0	0
3	1	0	1	0	0	1	0	0
4	1	1	1	0	0	0	0	1
5	1	0	1	0	0	0	1	0
6	1	0	0	0	0	0	1	0
7	1	1	1	0	1	1	0	0
8	1	0	0	0	0	0	0	0
9	1	1	1	0	0	0	0	0
A	1	1	0	0	0	0	0	0
C	1	0	0	1	0	1	0	0
E	1	0	0	1	0	0	1	0