# BINARY to 7-SEGMENT CONVERTER

Zafar Ullah
Scenix Semiconductor Inc.

BINARY to 7-SEG CONVERTER

RB0
RB1
RB2
RB3

SX28

RC0
RC1
RC2
RC3
RC4
RC5
RC6

COMMON ANODE
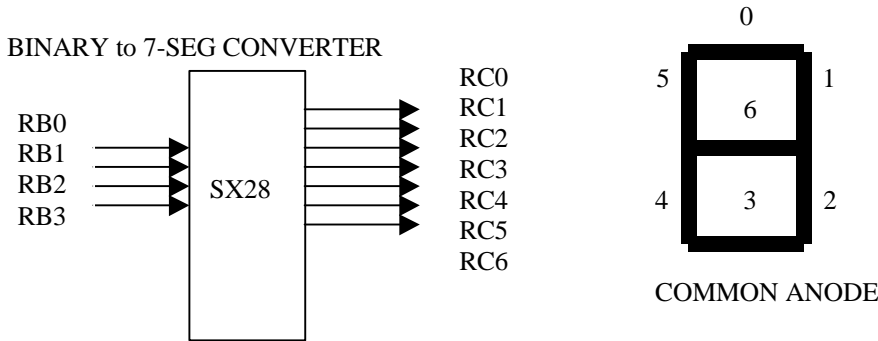
Truth Table:

| Hex | RB3 | RB2 | RB1 | RB0 | RC6 | RC5 | RC4 | RC3 | RC2 | RC1 | RC0 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| A | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| B | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| C | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| D | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| E | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| F | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |

A simple table lookup method is implemented for this binary to 7-seg converter. The code reads the binary input through the RB ports, reads the table and output the value through the RC port to turn on/off LEDs. This design used hp's HDSP 7301 LED, which is a common anode type. The truth table above shows the input-output relationship to implement this converter with this LED.

```
;*********************************************************************
;       Binary to Seven-Segment Display Routine
;       Author:Zafar Ullah
;       Written: 98/7/30
;
;*********************************************************************
;
;****** Assembler directives
;
; uses: SX28AC, 2 pages of program memory, 8 banks of RAM, high speed
;osc.
;    operating in turbo mode, with 8-level stack & extended option reg.
;
            DEVICE  pins28,pages2,banks8,oschs
            DEVICE  turbo,stackx,optionx
            ID      'Seven-Se'                 ;program ID label
            RESET   reset_entry             ;set reset/boot address

            FREQ  50000000
;
;
;
;
            ORG     100h
;
; Program execution begins here on power-up or after a reset
;
reset_entry

        MOV  M, #$0F            ;set the mode registar.
        MOV  !RB, #$FF          ; Set Port B as Input
        MOV  !RC, #$00          ; Set the port C as Output

:LOOP
        MOV  W, RB              ; Read Port B
        AND  W, #$0F            ; Masking significant bits
        CALL  CHECK_TABLE       ; Go to look up table
        MOV  RC, W              ; Output the value to port C
        NOP
        NOP
        JMP :LOOP

        ORG   10h

CHECK_TABLE    JMP PC+W  ; Look up table

RETW $40,$F9,$24,$30,$19,$12,$02,$78,$00,$18,$08,$03,$46,$21,$06,$0E
```