Decimal Adjust Routines

APPLICATION OPERATION

When implementing packed BCD logic with PICmicros™ it may be necessary to perform additions or increments. In this case, it is necessary to execute a decimal adjustment to ensure a valid result.

I've included two routines.

Routine #1

This routine emulates the 'decimal adjust after addition' which is part of the instruction set of some microprocessor families. Of course, it works only after additions, but remember that a packed BCD number can be decremented by adding '99'.

ROM words: 16
Additional RAM: 1 byte (1 bit used)
Execution time: 15 to 17 instruction cycles

Routine #2

This routine is to be used when only an increment is necessary (i.e., up counters, clocks, etc.). It increments the 'number' register and performs the decimal adjust. In fact, this routine will work with additions of up to '6' in any or both of the packed BCD digits.

ROM words: 12
Additional RAM: none
Execution time: 10 to 13 instruction cycles

Both routines return W = 0, if the result is '99' or less and W = 1, if the result is '100' or more. This allows for multiple precision packed BCD numbers manipulation.

MICROCHIP TOOLS USED

Assembler/Compiler version
MPASM V01.50, MPLAB V3.22.02

ROM words: 12
Additional RAM: none
Execution time: 10 to 13 instruction cycles
FIGURE 1: ROUTINE #1

Clear flag0

Is Carry set?
  yes
  Set flag0
  no

Is Decimal Carry set?
  yes
  Add 06 to number
  no

Add 06 to number

Is Decimal Carry set?
  yes
  Subtract 06 from number
  no

Subtract 06 from number

Add 06 to number

Is Carry set?
  no
  Is flag0 set?
    yes
    Return 0 (number < 100)
    no
  Return 1 (Number >= 100)
  yes
FIGURE 2: ROUTINE #2:

Add 07 to number

yes

Iss Decimal Carry

no

Subtract 06 from number

Add 60 to number

Is Carry set?

yes

Return 0 (Number< 100)

Return 1 (Number>= 100)

no

Subtract 60 from number
APPENDIX A: SOURCE CODE

A.1 Routine #1: Decimal Adjust After Addition

Note: ‘daa’ must be called immediately after the addition if performed, which result must be in ‘number’ register.

flag  equ 07h ; Register for flag storing (only bit 0 is used)
number equ 08h ; Register for number storing

; daa
BCF flag, 0
BTFSC STATUS, C
BSF flag, 0
MOVLW 06h
BTFSC STATUS, DC
ADDF W number, F

; ADDWF number, F
BTFSS STATUS, DC
SUBWF number, F
MOVLW 60h
ADDF W number, F
BTFSS STATUS, C
BTFSC flag, 0
RETLW 01h
SUBWF number, F
RETLW 00h

A.2 Routine #2: Increment and Perform Decimal Adjust to the ‘Number’ Register

number  equ 07h ; Register for number storing

; incdaa
MOVLW 07h
ADDF W number, F
BTFSC STATUS, DC
GOTO test_c
MOVLW 06h
SUBWF number, F

test_c
MOVLW 60h
ADDF W number, F
BTFSC STATUS, C
RETLW 01h
SUBWF number, F
RETLW 00h