

Universidad nacional Autónoma de México

Facultad de Ingeniería

Microcomputadoras

- Alumno: Alfonso Murrieta Villegas
- Profesor Rubén Anaya García

Tarea 2:

Realizar un programa que convierta un número de 8 bits en hexadecimal a su correspondiente decimal; la conversión deberá estar ubicada en dos registros.

Desarrollo del algoritmo

A continuación el planteamiento de la futura solución en pseudocódigo, con el objetivo de abordar un algoritmo destinado a realizar la conversión de bases numéricas.

```
Proceso HexaToDec
  clearRegistros;
  obtener hexa;
  nDigitos<-longitud(hexa); //va a estar limitado por el tamaño del registro
  nDecimal<-0;

  para i<-1 hasta nDigitos con paso 1 //nos desplazamos en cada posición del
número
    compararHextoDec() // validar del 9 en adelante
    divFunc()
    saveDato
  FinPara
  escribir(suma(saveDato.pos()))
FinProceso
```

Ejemplo:

Hexadecimal = AA

1. Se guarda en el registro
2. Obtenemos el valor de la decena
 1. Hacemos la conversión considerando la posición
 1. Factor de conversión = 16^1
 2. Valor del dígito = A = 10
 2. Factor de conversión * Valor de dígito = 160
3. Obtenemos el valor de la unidad
 1. Hacemos la conversión considerando la posición
 1. Factor de conversión = 16^0
 2. Valor del dígito = A = 10

2. Factor de conversión * Valor de dígito = 10
4. Sumamos los valores obtenidos de cada posición tras ser convertidos
5. Guardamos el valor final en nuevos registros

Código

A continuación el código final de la presente tarea:

```
processor 16f877
include <p16f877.inc>

in_He    equ h'20'    ; Input hexadecimal
outD_C    equ h'21'    ; Output Decimal - Centenas
outD_DU    equ h'22'    ; Output Decimal - Decenas /Unidades

aux       equ h'30'
des       equ h'0A'
cen       equ h'64'

        ORG 0
        GOTO INICIO
        ORG 5

INICIO: CLRf outD_C
        CLRf outD_DU
        CLRf aux
        MOVLW cen

        CALL divPart;Llamada para decena
        ADDWF in_He,F
        MOVF aux,W
        MOVWF outD_C
        CLRf aux
        MOVLW des

        CALL divPart ; Llamada para unidad
        ADDWF in_He,F
        SWAPF aux,W
        MOVWF outD_DU
        MOVF in_He,W
        ADDWF outD_DU
        GOTO$

divPart: SUBWF in_He,F ;División y recorrido
        BTFSS STATUS,C
        RETURN
        INCF aux
        GOTO $-4

        END
```

NOTA: En la siguiente rutina realizamos parte de la división para la obtención respecto a la posición ya sea en la centena o decena:

```
divPart: SUBWF in_He, F
        BTFSS STATUS, C
        RETURN
        INCF aux
        GOTO $-4
```

Capturas de Pantalla (Evidencias de funcionamiento)

A continuación una tabla con el código en el MPLAB antes y después de compilar y ensamblar en código con los respectivos valores a probar.

NOTA: Hacer zoom sobre las capturas de pantalla para más detalles.

[illegible][illegible]

A1 =
161

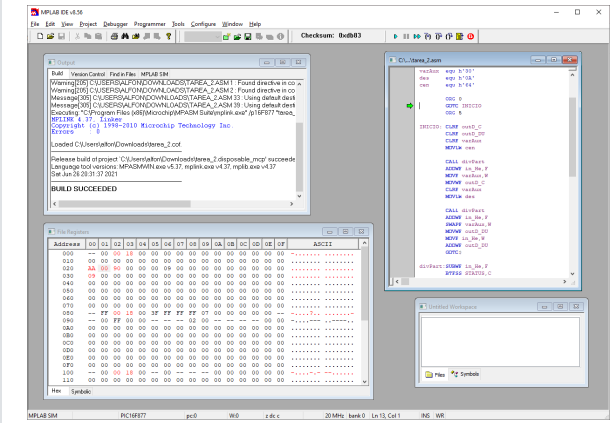
[illegible]

A Test Region										C2 C3		C4 C5		C6 C7		C8 C9		C10 C11		C12 C13		C14 C15		C16 C17		C18 C19		C20 C21		C22 C23		C24 C25		C26 C27		C28 C29		C30 C31		C32 C33		C34 C35		C36 C37		C38 C39		C40 C41		C42 C43		C44 C45		C46 C47		C48 C49		C50 C51		C52 C53		C54 C55		C56 C57		C58 C59		C60 C61		C62 C63		C64 C65		C66 C67		C68 C69		C70 C71		C72 C73		C74 C75		C76 C77		C78 C79		C80 C81		C82 C83		C84 C85		C86 C87		C88 C89		C90 C91		C92 C93		C94 C95		C96 C97		C98 C99		C100 C101		C102 C103		C104 C105		C106 C107		C108 C109		C110 C111		C112 C113		C114 C115		C116 C117		C118 C119		C120 C121		C122 C123		C124 C125		C126 C127		C128 C129		C130 C131		C132 C133		C134 C135		C136 C137		C138 C139		C140 C141		C142 C143		C144 C145		C146 C147		C148 C149		C150 C151		C152 C153		C154 C155		C156 C157		C158 C159		C160 C161		C162 C163		C164 C165		C166 C167		C168 C169		C170 C171		C172 C173		C174 C175		C176 C177		C178 C179		C180 C181		C182 C183		C184 C185		C186 C187		C188 C189		C190 C191		C192 C193		C194 C195		C196 C197		C198 C199		C200 C201		C202 C203		C204 C205		C206 C207		C208 C209		C210 C211		C212 C213		C214 C215		C216 C217		C218 C219		C220 C221		C222 C223		C224 C225		C226 C227		C228 C229		C230 C231		C232 C233		C234 C235		C236 C237		C238 C239		C240 C241		C242 C243		C244 C245		C246 C247		C248 C249		C250 C251		C252 C253		C254 C255		C256 C257		C258 C259		C260 C261		C262 C263		C264 C
---------------	--	--	--	--	--	--	--	--	--	-------	--	-------	--	-------	--	-------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	---------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	-----------	--	--------

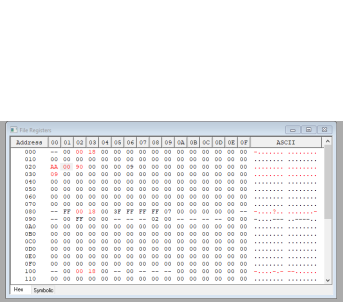
BB =
187

[illegible]
$$5A = 90$$

Antes de compilación

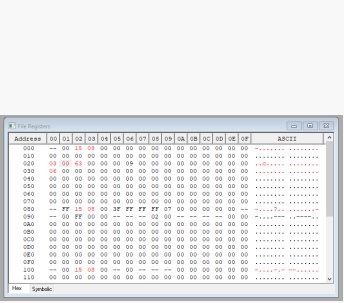
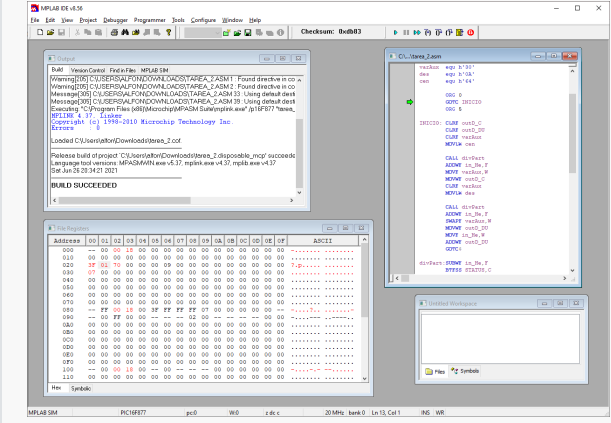


Después de compilación



Valores

AA = 170



3F = 63