



INSTITUTO POLITÉCNICO NACIONAL  
“ESCOM”

INGENIERÍA EN SISTEMAS COMPUTACIONALES  
REPORTE: SUMADOR Y MULTIPLICADOR DE 8 BITS

ESQUIVEL PÉREZ JONATHAN ALFREDO  
HERNÁNDEZ LÓPEZ ÁNGEL ZAIT  
SALGADO GALLEGOS JESUS  
SANCHEZ PIZANO IRVING DANIEL

INTRODUCCIÓN A LOS MICROCONTROLADORES  
PEREZ PEREZ JOSE JUAN

3CM8

19/JUNIO/2020

## OBJETIVO. –

Implementar un programa para lograr sumar y multiplicar dos datos en el Microcontrolador Atmega8535, mediante la escritura de código y construcción del circuito.

## INTRODUCCIÓN. –

### **SUMADOR**

El sumador binario es un circuito capaz de hacer la suma algebraica entre dos números binarios con un bit cada uno, que hace que la suma de dos bits a la vez, es decir, dos dígitos binarios a la vez. Primero vemos un semisumador, es decir, un circuito que recibe como entrada dos dígitos binarios y da en salida a su suma, más el acarreo.

Aunque los sumadores se pueden construir para muchas representaciones numéricas, tales como decimal codificado en binario o exceso-3, los sumadores más comunes funcionan en números binarios. En los casos en que se utiliza el complemento a dos o el complemento a uno para representar números negativos, es trivial modificar un sumador para convertirlo en un sumador-restador. Otras representaciones de números con signo requieren más lógica alrededor del sumador básico.

Un sumador completo suma números binarios junto con las cantidades de acarreo. Un sumador completo de un bit añade tres bits, a menudo escritos como A, B y  $C_{in}$ ; siendo A y B los sumandos y  $C_{in}$  el acarreo que proviene de la anterior etapa menos significativa. El sumador completo suele ser un componente de una cascada de sumadores, que suman 8, 16, 32, etc. números binarios de bits. El circuito produce una salida de dos bits, al igual que el semisumador, denominadas acarreo de salida ( $C_{out}$ ) y suma S.

### **MULTIPLICADOR**

Un **multiplicador binario** es un circuito electrónico utilizado en la electrónica digital, tales como un ordenador, para multiplicar dos números binarios. Está construido usando sumadores binarios.

Existen varios métodos básicos para el cálculo de la multiplicación de dos números (A, B) de N bits:

- Almacenamiento de los  $2 \cdot 2^N$  resultados posibles en una memoria ROM y utilizar los  $2^N$  bits para el direccionamiento.
- Calcular los  $2 \cdot N$  funciones lógicas y realizar la suma correspondiente.
- Con base en la codificación anterior optimizar teniendo en cuenta una relación de dependencia entre los números A y B y el resultado M.

La multiplicación consiste en una serie de operaciones AND entre los distintos bits y una serie de sumas.

- Se requieren de  $2 \cdot N$  compuertas AND.
- Se requiere de N sumadores de N bits
- Problema: Extensión del signo
- Problema: Tratamiento del signo del operando B.

## MATERIAL Y EQUIPO UTILIZADO

- Protoboard
- Dos dip switch de 8 entradas
- Resistencias 100Ω
- display 7 segmentos
- microcontrolador ATmega 8535
- software proteus
- Alambres para conexión
- leds

## QUE SE REALIZARÁ EN LA PRÁCTICA

### **SUMADOR**

Se realizará un sumador el cual dependerá de 2 dip-switch, los cuales serán los encargados obtener la entrada de los números que vamos a sumar. El resultado del sumador debe ser mostrado en leds.

### **MULTIPLICADOR**

Para el multiplicador se ocuparan los mismos elementos sólo que esta vez en vez de sumar, se multiplicará..Es decir se introduce un número binario en cada dip-switch y el resultado se mostrará en la tira de leds.

## DESARROLLO DE LA PRÁCTICA INCLUYENDO DIAGRAMAS DE FLUJO, CODIFICACIÓN COMENTADA

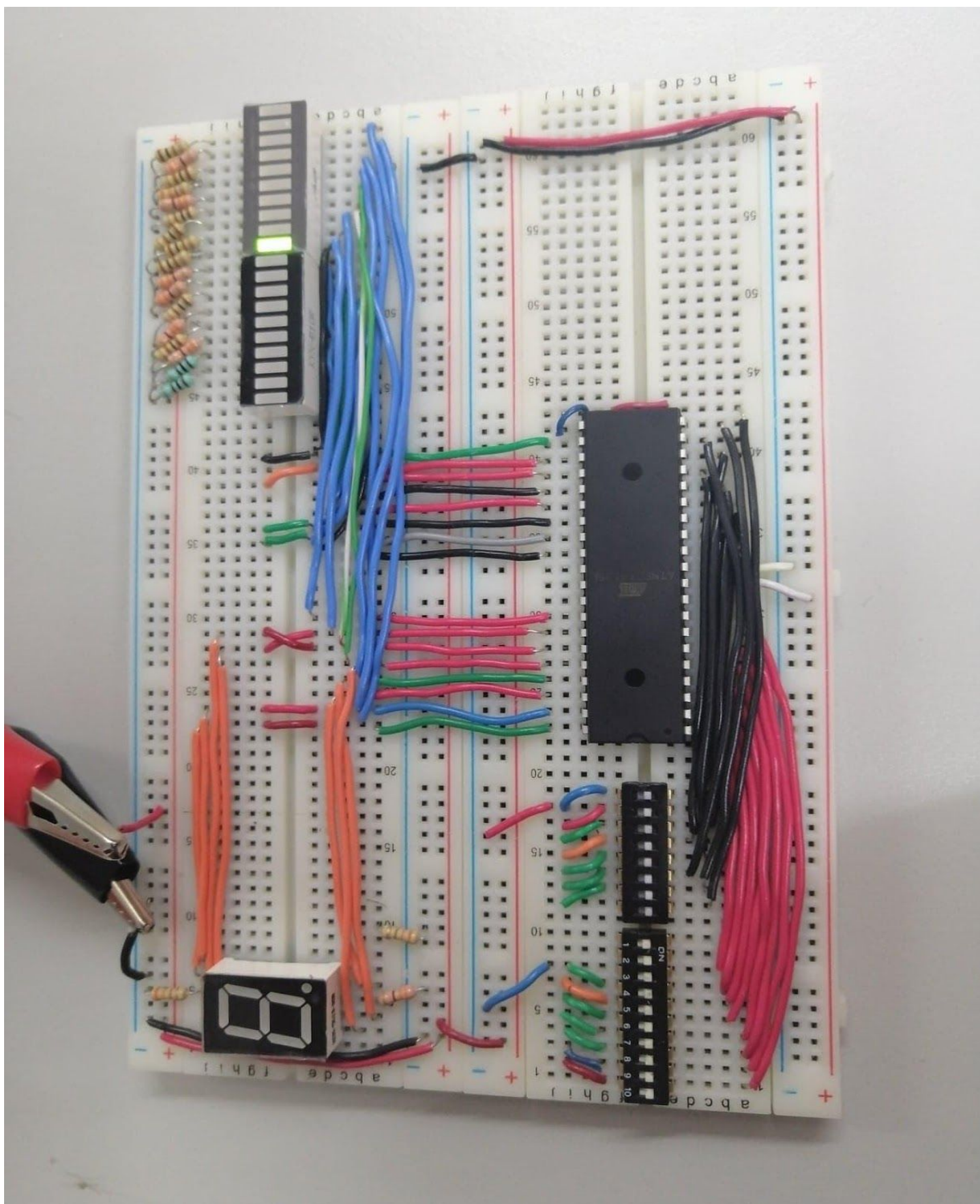
### **CODIFICACIÓN SUMADOR:**

```
1. ser R16 ;Se asigna al R16 $ff (R16 <- $ff)
2. out DDRA,R16 ;Se habilita como salida El puerto A y se asigna a R16
3. out DDRC,R16 ;Se habilita como salida El puerto C y se asigna a R16
4. out PORTB,R16 ;Se habilita pull up El puerto B y se asigna a R16
5. out PORTD,R16 ;Se habilita pull up El puerto D y se asigna a R16
6. uno: ;Etiqueta que indica el retorno para ejecutar un ciclo
7. in R16,PINB ;Se utiliza como entrada el PIN B indicando que será usado en el R16
8. in R17,PIND ;Se utiliza como entrada el PIN D indicando que será usado en el R17
9. add R16,R17 ;Se Realiza la operación suma R16 <- R16 + R17
10. out PORTA,R16 ;Se Utiliza como salida el Puerto A usando el R16
11. in R16,SREG ;Se usa el R16 para mostrar el contenido de SREG
12. out PORTC,R16 ;Se usa como salida el puerto C mostrando el contenido de R16
13. rjmp uno ;Instrucción que salta a la etiqueta llamada uno
```

## **CODIFICACIÓN MULTIPLICADOR:**

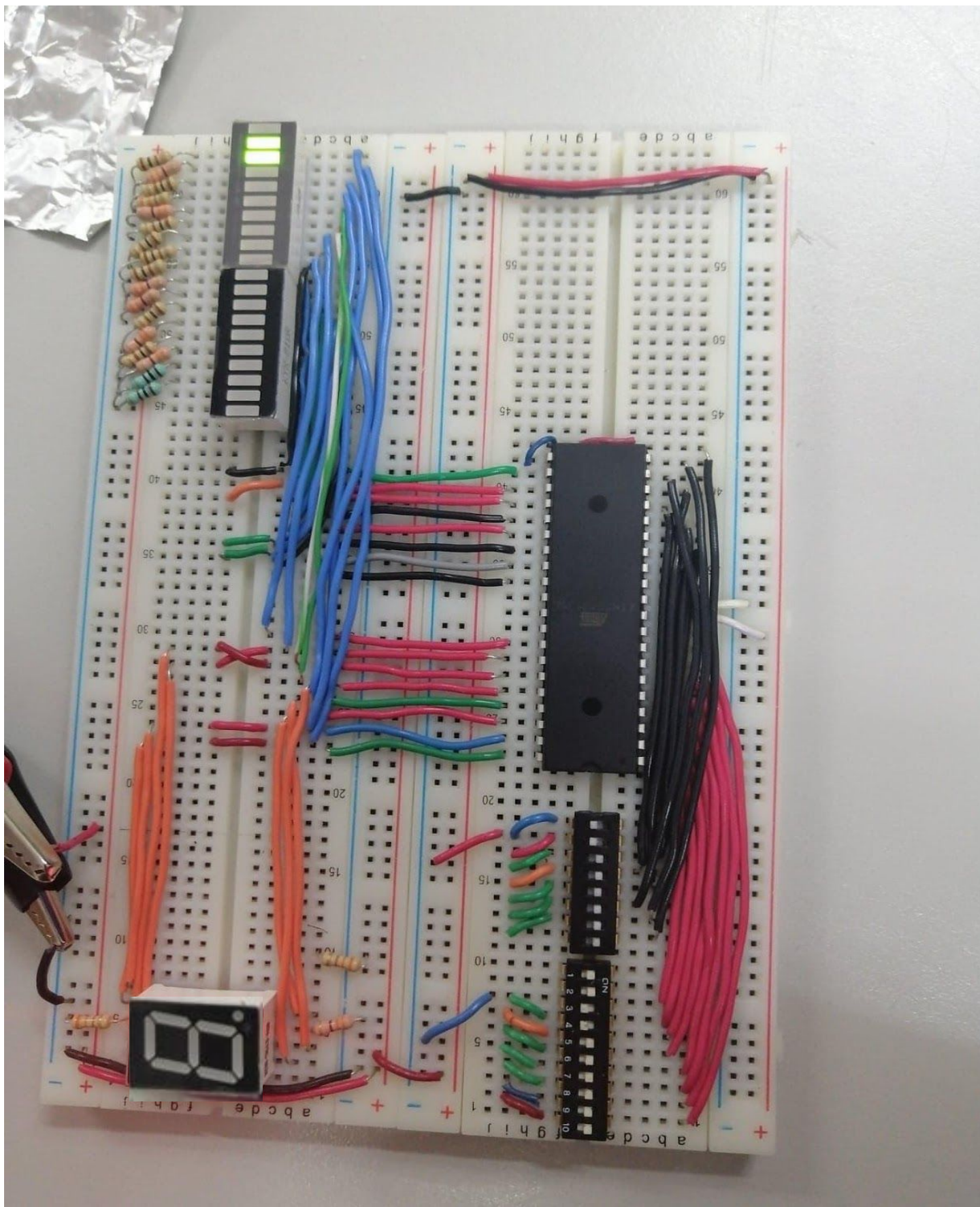
```
1. .include "m8535def.inc"
2. reset:
3.     ser R16 ;Se asigna al R16 $ff (R16 <- $ff)
4.     out DDRA,R16 ;Se habilita como salida El puerto A y se asigna a R16
5.     out DDRC,R16 ;Se habilita como salida El puerto C y se asigna a R16
6.     out PORTB,R16 ;Se habilita pull up El puerto B y se asigna a R16
7.     out PORTD,R16 ;Se habilita pull up El puerto D y se asigna a R16
8.
9. uno: ;Etiqueta que indica el retorno para ejecutar un ciclo
10.     in R16,PINB ;Se utiliza como entrada el PIN B indicando que será usado en el
    R16
11.     in R17,PIND ;Se utiliza como entrada el PIN D indicando que será usado en el
    R17
12.     mul R16,R17 ;Se Realiza la operación multiplicacion R16 <- R16 * R17
13.     out PORTA,R16 ;Se Utiliza como salida el Puerto A usando el R16
14.     in R16,SREG ;Se usa el R16 para mostrar el contenido de SREG
15.     out PORTC,R16 ;Se usa como salida el puerto C mostrando el contenido de R16
16.     rjmp uno ;Instrucción que salta a la etiqueta llamada uno
```

## **SIMULACIÓN SUMADOR:**



**SIMULACIÓN MULTIPLICADOR:**





## CONCLUSIONES Y OBSERVACIONES INDIVIDUALES

- ❖ Esquivel Pérez Jonathan Alfredo: En la presente práctica aplicamos los conocimientos de las banderas para aumentar la escritura más allá de los registros de propósito general. Una vez que se tuvieron en cuenta estos ajustes recuperamos el dato binario aplicando los conocimientos de la práctica anterior.
- ❖ Hernández López Ángel Zait  
Para esta práctica se vio más a detalle el cómo podemos ingresar datos al microcontrolador de forma externa, al igual de ver distintos tipos de comandos para poder hacer esto posible, habilitando y poniendo instrucciones en específico para que se pudiera leer de forma binaria. Además se vieron involucrados banderas de los registros, los cual teníamos que apreciar para entender el cómo hace la suma y los acarreos que conlleva.
- ❖ Salgado Gallegos Jesús  
Gracias al uso de las banderas del registro de estados, es posible usar diferentes instrucciones dependiendo del valor que tengan. Como es en este caso, el de los brincos condicionales, al realizar una operación entre dos números; como la suma, se modifican los valores de los bits de banderas y de acuerdo a operaciones de compuertas lógicas y/o algunas comparaciones básicas entre ellos.
- ❖ Sanchez Pizano Irving Daniel  
En esta práctica pude aprender sobre el funcionamiento de los registros y el saber cómo implementarlos en los microcontroladores, si nos surgieron varios problemas mas que nada porque aún no estábamos tan familiarizados con el lenguaje, pero revisando los apuntes que nos brindó pudimos solucionar esos problemas.