

Reporte de práctica 1 - Calculadora analógica”

1288433 - Pedro I. López [dreilopz@gmail.com]

4 de mayo de 2012

Maestro: Dr. Luis M. Torres Treviño

Materia: 483 PERCEPCIÓN (Laboratorio)

Dependencia: FIME UANL

1. Introducción

El ejercicio práctico consiste en implementar una calculadora analógica. Los operandos son 2 señales de voltaje con rango 0-5 V, cuyo valor se controla con un potenciómetro rotacional. Las operaciones aritméticas que se pueden realizar con los operandos son suma, resta, multiplicación y división; la operación se selecciona con entradas digitales al sistema. Una sola salida analógica con rango 0-5 V muestra el resultado de la operación.

2. Descripción

Los operandos serán voltajes de 0-5 V variados con potenciómetros rotacionales. La operación se selecciona con el estado de los dos primeros switches del DIP switch azul. El resultado de la operación aritmética se reporta con una señal de voltaje de 0-5 V, medida con un multímetro. Ya que los operandos serán números computables de 0 a 5, los resultados pueden ser menores a 0 o mayores a 5, pero físicamente el resultado de la operación aritmética se acota al rango original 0-5.

La NI USB-6008 (USB6008) es una interfaz de adquisición y salida de datos para PC con conexión USB. Los operandos se envían a los canales analógicos de entrada USB6008.AI0 y USB6008.AI1, y la salida se detecta en USB6008.A00. Los estados de los switches que seleccionan la operación aritmética se leen con USB6008.P0.0 y USB6008.P0.1.

El software se desarrolló en MATLAB, utilizando el *Data Acquisition Toolbox* para obtener acceso a la API de la USB6008: *NIDAQ-mx*. La interfaz del programa es una línea de comandos interactiva.

Componentes utilizados:

- 2 potenciómetros de 10K Ω
- 1 DIP switch

- 1 NI USB6008
- Alambre
- Conectores
- 1 multímetro

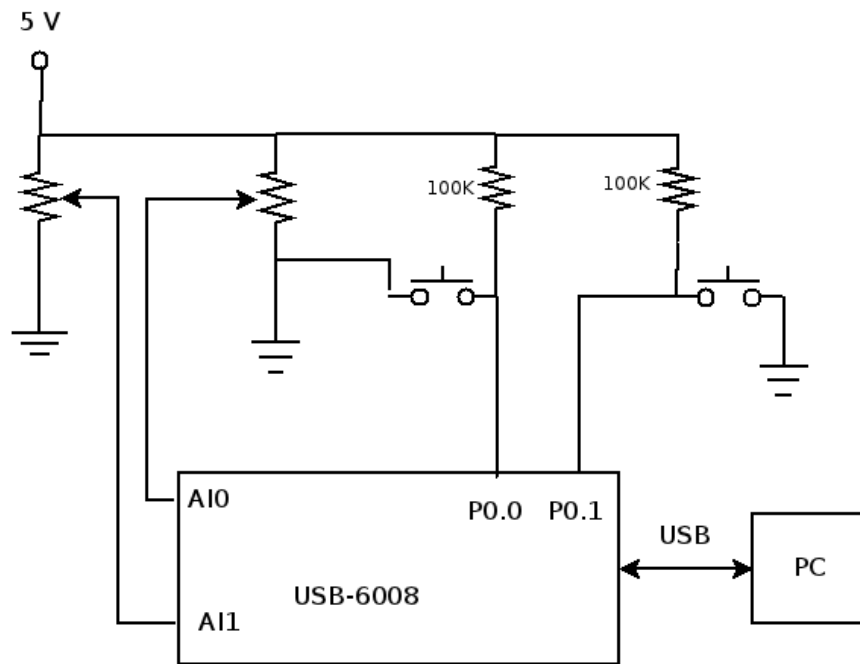


Figura 1: Diagrama de circuito.

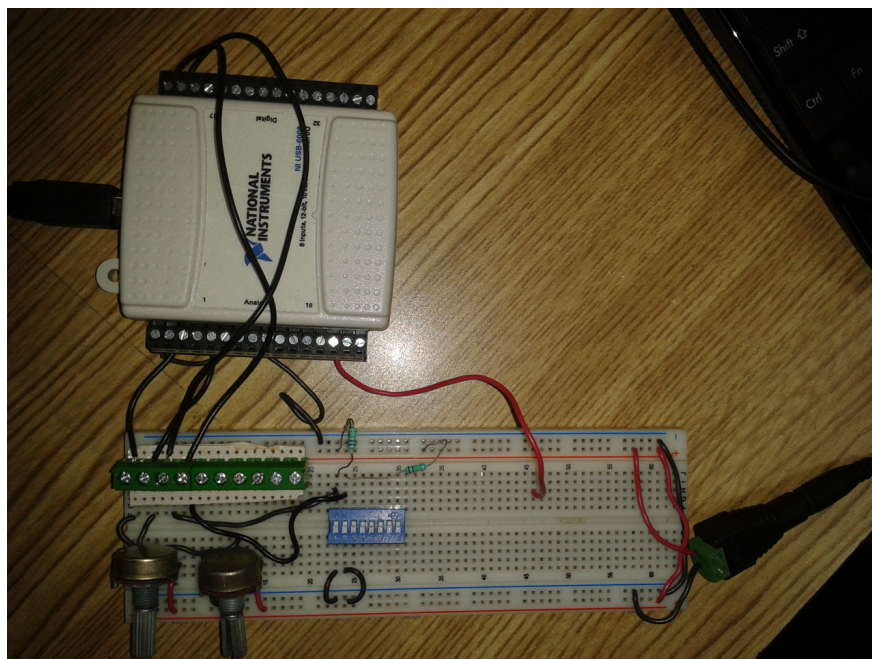


Figura 2: Implementación de circuito.

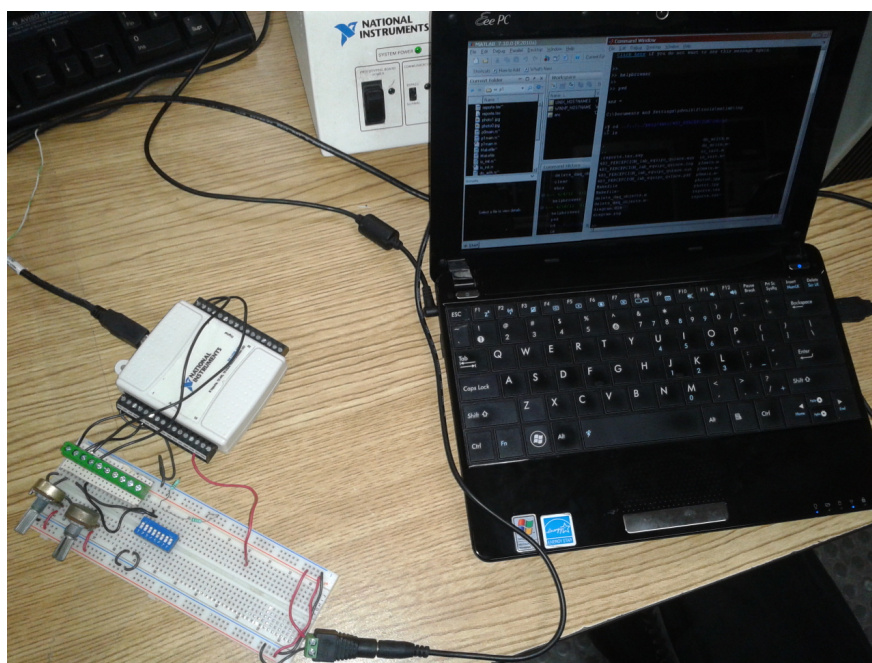


Figura 3: Sistema.

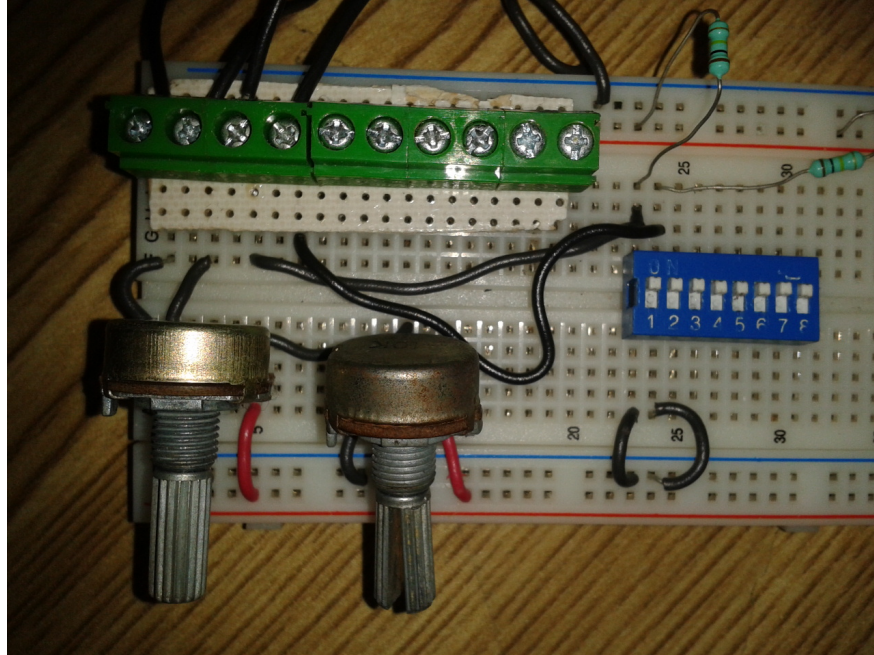


Figura 4: Entradas de sistema.

3. Resultados

Las operaciones se ejecutan correctamente. Como ya se mencionó el resultado físico está acotado al rango 0-5, pero la salida del programa reporta el resultado correcto sin acotarse, incluyendo números negativos.

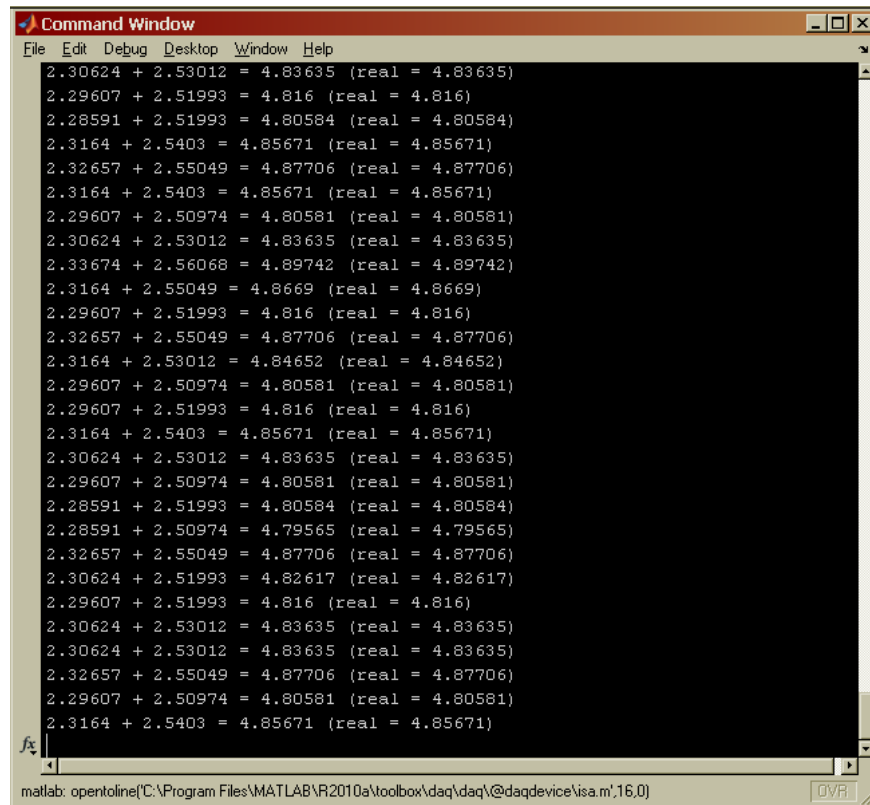


Figura 5: Programa de la calculadora ejecutando una suma.

4. Pruebas

Las pruebas se realizaron en el salón de laboratorio con la presencia y aprobación del instructor.

5. Código

Listing 1: p1main: Script principal de la calculadora.

```

1 delete_daq_objects;
2 io_init;
3 while true
4     do_arith;
5 end

```

Listing 2: `delete_daq_objects`: Script que elimina objetos de dispositivos de adquisición de datos

```
1 try
2     delete(operands);
3     delete(result);
4     delete(op_select);
5     clear('operands' , 'result' , 'op_select');
6 end
```

Listing 3: `io_init`: Inicializando el hardware.

```
1 DAQ_DEVICE_NAME = 'usb6008';
2 ADAPTOR = daqhwinfo('nidaq');
3
4 delete_daq_objects
5
6 operands = analoginput(ADAPTOR.AdaptorName, DAQ_DEVICE_NAME);
7 operands.SampleRate = 1000;
8 operands.InputType = 'SingleEnded';
9 operands.TriggerType = 'Immediate';
10 operands.SamplesPerTrigger = 2;
11 addchannel(operands, 0:1);
12 result = analogoutput(ADAPTOR.AdaptorName, DAQ_DEVICE_NAME);
13 addchannel(result, 0);
14 op_select = digitalio(ADAPTOR.AdaptorName, DAQ_DEVICE_NAME);
15 addline(op_select, 0:1, 'in');
```

Referencias

- [1] *MATLAB R2010a documentation*. The MathWorks, Inc.
- [2] *NI USB-6008/6009 - User guide and specifications*. 2004-2008 National Instruments Corporation.

Listing 4: do_arith: Este código ejecuta las operaciones aritméticas.

```
1 start(operands);
2 pause(0.15);
3 data = getdata(operands);
4 data = data(1,:);
5
6 operation_selector = getvalue(op_select);
7
8 if isequal([0 0], operation_selector)
9     arithmetic_result = data(1) + data(2);
10    operation = '+';
11 elseif isequal([0 1], operation_selector)
12    arithmetic_result = data(1) - data(2);
13    operation = '-';
14 elseif isequal([1 0], operation_selector)
15    arithmetic_result = data(1) * data(2);
16    operation = '*';
17 elseif isequal([1 1], operation_selector)
18    arithmetic_result = data(1) / data(2);
19    operation = '/';
20 end
21
22 arithmetic_result_real = arithmetic_result;
23 if arithmetic_result > 5.0
24     arithmetic_result = 5.0;
25 end
26 if arithmetic_result < 0.0
27     arithmetic_result = 0.0;
28 end
29
30 putsample(result, arithmetic_result);
31
32 report_str = sprintf('%g %s %g = %g (real = %g)', data(1), operation, ...
33 data(2), arithmetic_result, arithmetic_result_real);
34 disp(report_str);
```