

## DISEÑO Y MONTAJE DE UN SISTEMA PARA INGRESO Y VISUALIZACIÓN DE DATOS NUMÉRICOS [SIVDN] BASADO EN EL PIC18F6520

Stalin Jiménez Miranda  
Universidad Técnica Particular de Loja  
Electrónica y Telecomunicaciones  
[sdjimenez@utpl.edu.ec](mailto:sdjimenez@utpl.edu.ec)

Las palabras diseño y montaje dentro del campo de la ingeniería, siempre surgen como parte importante en el proceso de dar solución a un problema, sea cual sea su índole, sin embargo, en el avance de estas líneas aquellas mismas palabras no serán tomadas como etapas necesarias en la solución de una problemática, pero si como puntos medulares para divertirnos con el conocimiento, pues es así como se ha ido edificando todo un cúmulo de ciencia que hoy por hoy nos rodea a cada paso.

Entrando de lleno en lo que respecta al artículo, de ante mano es preciso mencionar que con este *Sistema para Ingreso y Visualización de Datos Numéricos [SIVDN]*, no pretendo solucionar absolutamente nada, únicamente mi intención es brindarle unas algunas ideas para “juguetear”, con componentes que los electrónicos [permítaseme considerarme uno de ellos] tenemos a mano. Siendo más preciso, al hablar de un sistema estamos haciendo referencia a un conjunto de partes que ordenadamente relacionadas entre sí, contribuyen a un fin determinado, interpretando el título del artículo podemos extraer tres de características sobre el sistema: a) debe contar con una etapa destinada a la recepción de datos, b) es indispensable disponer de otra fase para visualización de datos, c) se hace necesario que tanto el ingreso como la visualización interactúen entre sí, por lo tanto, existe algún bloque de acoplamiento. Una vez aclarado el panorama, podemos empezar a cincelar las ideas, lo más parecido que he visto al **SIVDN**, es una calculadora de bolsillo, la misma que cuenta con un teclado, una pantalla y un CI de acoplamiento, en efecto la “calcu” y nuestro **SIVDN** están estructurados de las mismas

partes, consecuentemente el paso siguiente es *analizar y construir cada componente* por separado. La pregunta de rigor en esta línea de texto tiene que ver con los componentes, ¿qué es más conveniente: comprarlos o construirlos? Al respecto yo prefiero construirlos [en la medida de lo posible], pues generalmente ahorro un 50% o más de valor que pago por comprarlos ya hechos, a ello se suma la satisfacción de saber que he fabricado algo con mis propias manos.

Hablando técnicamente, un teclado es una configuración adecuada de conductores a manera de array, los mismos que cortocircuitados brindan una determinada posición, pudiendo relacionarla a esta ubicación con un número, una letra o cualquier otro tipo de dato, en la Fig. 1 se resume cómo está dispuesto un teclado.

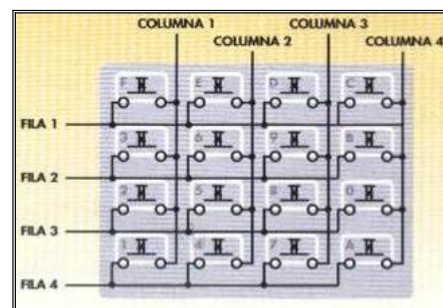


Fig. 1. Circuito de un teclado matricial.

Un matemático es a una ecuación, como un electrónico es a un circuito, es decir ambos saben cómo leer en sus respectivos idiomas, sin embargo para aquellas personas que aún no lo entienden, la Fig. 1 nos está explicando todo y si pudiese hablar diría lo siguiente: “Hola, soy el teclado, constrúyeme con 8 conductores y 16 pulsadores para ingresar 16 datos”, simplemente acatando las indicaciones que el “señor teclado” nos

marca, podemos llegar a un producto como el de la Fig. 2, al cual le he quitado una columna, personalmente considero que para ser un teclado casero es de buena calidad [puede corroborarlo haciendo un zoom a la imagen], por tres razones: a) el teclado es estéticamente aceptable, b) es económico debido al empleo de componentes tales como pulsadores, baquelita, etc., y c) funciona correctamente.

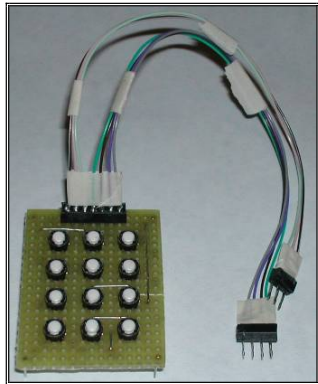
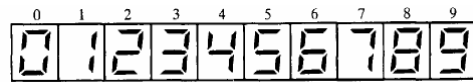
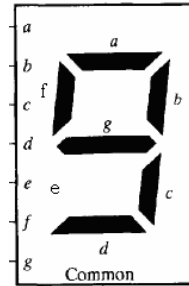


Fig. 2. Teclado matricial de 4 x 3.

Las cosas versátiles son aquellas que poseen mayor acogida, y dentro de la electrónica esto es prácticamente un dogma, pues no tendría sentido construir un hardware de carácter cerrado, es decir que cumpla una y solo una función, lo que pretendo dar a entender, es que nuestro teclado no nos servirá únicamente para implementar el SIVDN, pues se lo podría emplear en un sistema de seguridad, en la ya mencionada “calcu”, quizás como un mouse para computador o en todo aquello que se nos ocurra. Habiendo hecho las aclaraciones respectivas sobre un hardware de carácter abierto, debemos continuar con la construcción de la etapa visualizadora, para ello vamos a hacer uso de los ultrafamosos displays de 7 segmentos, siendo estos un simple arreglo de leds en forma de 8, que polarizados adecuadamente nos permiten observar números decimales, hexadecimales y una que otra letra del alfabeto castellano.

En la Fig. 3 es fácil distinguir las letras asignadas a cada segmento, y además la codificación para obtener los números decimales en el display.



	a	b	c	d	e	f	g
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1
0	1	1	1	1	1	1	0

Fig. 3. Display de 7 segmentos y su codificación.

Cabe aclarar que la codificación presentada es para un display con cátodo común, si fuese de ánodo común se intercambiarían ceros y unos, recalcando lo anteriormente dicho, es ideal que un hardware sea versátil, por lo tanto para la construcción del visualizador trataremos de integrar los componentes necesarios, a fin de reducir el cableado en trabajos futuros, cuando hablo de componentes necesarios me refiero sobretodo a resistores para limitar la corriente en el display, y transistores para control de cátodos, una vez puntualizados estos detalles, únicamente nos queda soldar los componentes de acuerdo al circuito presentado en la Fig. 4.

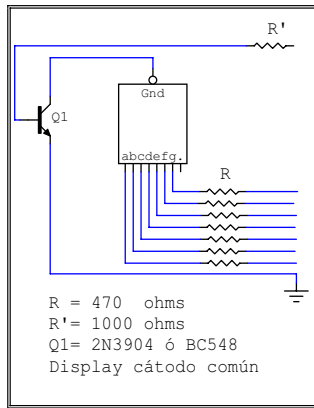


Fig. 4. Circuito para manejo de display.

Una vez realizado el trabajo de soldadura, el producto final tendrá una apariencia similar a la presentada en la Fig. 5, para ser preciso es el circuito de la Fig. 4 pero repetido 3 veces, y lo pude construir tan pequeño gracias a que encontré un módulo con tres displays, de los cuales en el SIVDN utilizaré únicamente uno de ellos.

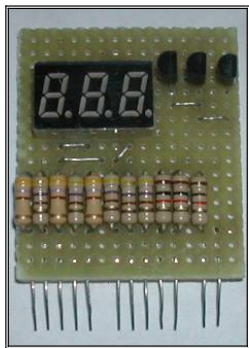


Fig. 5 Módulo para manejo de displays.

Como paso último debemos integrar las partes construidos, para lo cual voy a utilizar el microcontrolador [ $\mu C$ ] PIC18F6520, perteneciente a la familia de PICs 18FXX, fabricados por la casa **Microchip**. El motivo para escoger este  $\mu C$  es la disponibilidad de información y equipos en el laboratorio de la Escuela de Electrónica y Telecomunicaciones [EET].

Por sobre todo lo mencionado en líneas anteriores, lo que más rescato son las características de los equipos, técnicamente conocidos como **“C Stamp™ microcontroller module”**

y desarrollados por la firma comercial **A-WIT Technologies Inc.**, el mérito radica en la facilidad que ofrece al usuario para descargar el soft en el  $\mu C$ , y aún más en el lenguaje que ofrece para desarrollar el soft, hago hincapié en esta última parte debido a que no es indispensable conocer las instrucciones que maneja a bajo nivel el  $\mu C$ , ya que la programación es realizada en lenguaje **C-Stamp** muy similar al C convencional.

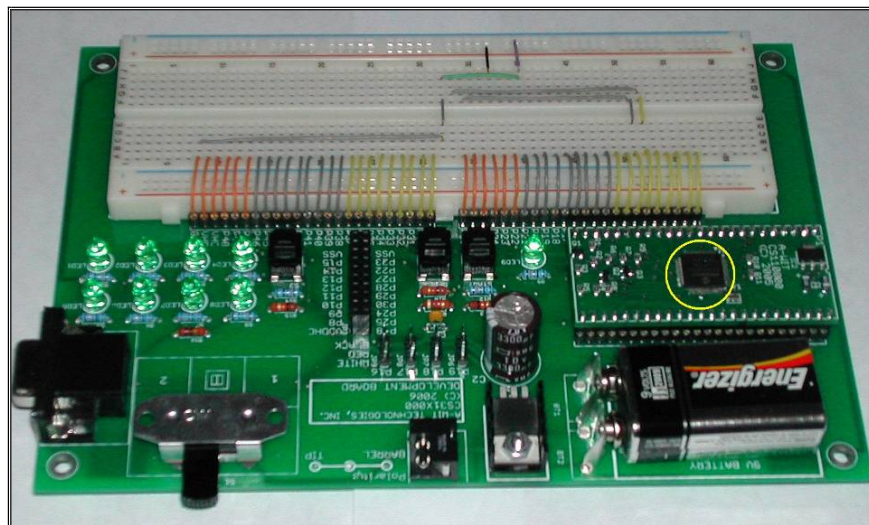


Fig. 6. C Stamp™ microcontroller module.

En la Fig. 6 es posible apreciarlo en todas sus dimensiones al equipo, el mismo que a grandes rasgos dispone de un puerto serial para descargar el programa, 8 leds para visualización de estados lógicos, una fuente de energía, un protoboard para montar hardware externo al módulo, y por supuesto el  $\mu C$  con sus 48 pines a la derecha de la imagen. En cuanto al soft que ejecuta el  $\mu C$ , lo codifiqué en el siguiente orden: a) inicia

con un while para evaluar el estado lógico de los pines a los cuales se ha anclado las filas y columnas del teclado, b) de acuerdo al estado detectado el  $\mu C$  enviará los unos y ceros necesarios para comandar el display, c) en caso de presionar 2 teclas a la vez por más de 3 segundos el  $\mu C$  interpretará esta acción como un OFF, y consecuentemente finalizará el while con lo que terminará el programa.

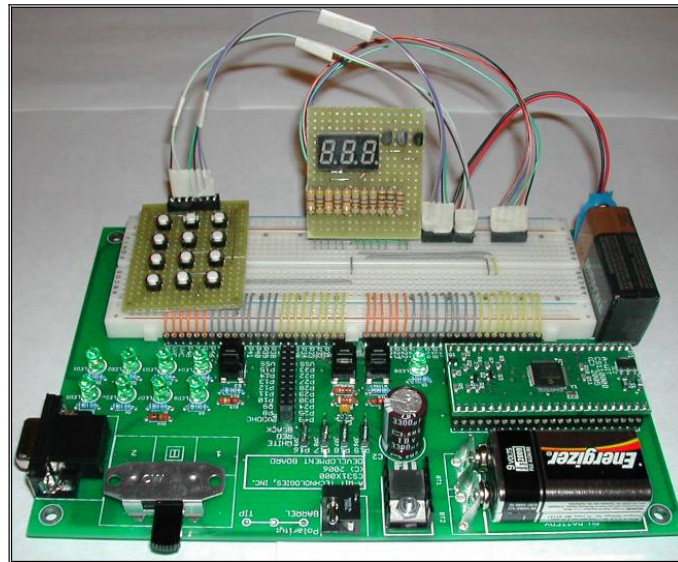


Fig. 7. SIVDN armado en el C Stamp™ microcontroller module.

A lo largo del artículo he sido muy claro en cuanto a un hardware de tipo abierto, y en honor a esa característica me es necesario señalar nuevamente que tanto el teclado, como la pequeña tarjeta para manejo de display no únicamente son utilizables a través del PIC18F6520, lo pueden ser con otro tipo de  $\mu C$  de mayores o menores prestaciones, es más, antes de tomar en serio la utilización del PIC, probé el SIVDN en un PC a través del puerto paralelo con un soft desarrollado bajo LABVIEW.

Para todo tipo de montaje, siempre es necesario contar con un estimado de cuánto se va a gastar, en vista de eso finalizo detallando un cuadro con los componentes y precios en dólares de c/u, como no podía ser de otra manera en la Fig. 7 se lo puede observar al SIVDN totalmente armado,

mientras que en la Fig. 8 esta funcionando según lo previsto.

Ha sido un placer haber compartido nuevamente estas líneas con ustedes, como de costumbre las sugerencias, correcciones, interrogantes sobre los circuitos o programas utilizados en este artículo, agradecería me las remitan al correo. Hasta la siguiente edición.

Componentes	Can	Pre Uni.	Pre To.
Resis 470 $\Omega$	8	0.05	0.4
Resis 1K $\Omega$	3	0.05	0.15
Baquela perfo.	1	0.75	0.75
Display	1	0.8	0.8
Pulsador	12	0.15	1.8
Transis 3904	3	0.1	0.3
<b>Precio total</b>			<b>4.2</b>

Cuadro 1. Precios de componentes para el SIVDN.

