

**Fuente de alimentación variable a pequeña escala controlada por puerto
serial o *bluetooth***



**Proyecto de Trabajo de Grado
TG. 1332**

William David Riaño

Director: Ing. Gabriel Perilla M.Sc

**Pontificia Universidad Javeriana Bogotá
Facultad De Ingeniería Electrónica
2014**

1	INTRODUCCIÓN	5
2	OBJETIVO GENERAL Y OBJETIVOS ESPECÍFICOS	6
2.1	OBJETIVO GENERAL	6
2.2	OBJETIVOS ESPECÍFICOS	6
3	MARCO CONCEPTUAL	7
3.1	INTRODUCCIÓN A SISTEMAS EMBEBIDOS.....	7
3.2	PROTOCOLOS DE COMUNICACIÓN	7
3.2.1	<i>Introducción a I²C</i>	7
3.2.2	<i>Introducción a UART</i>	8
3.3	FUENTES DE CONMUTACIÓN	9
3.3.1	<i>Fuentes Buck</i>	9
4	ESPECIFICACIONES	10
4.1	HARDWARE.....	10
4.2	SOFTWARE.....	11
5	DESARROLLOS	12
5.1	HARDWARE.....	12
5.1.1	<i>AD5282 Potenciómetro digital</i>	12
5.1.2	<i>LM2577 (Boost) Y LM2576 (Buck)</i>	14
5.1.3	<i>Limitador de corriente</i>	16
5.1.4	<i>Potenciómetro Digital + Limitador de Corriente + Fuente Buck</i>	19
5.2	SOFTWARE.....	20
5.2.1	<i>UART</i>	20
5.2.2	<i>I²C</i>	22
5.2.3	<i>GUI (Interfaz Gráfica)</i>	25
5.2.4	<i>Circuito Impreso</i>	27
5.2.5	<i>Criterio de ubicación de componentes PCB</i>	30
5.2.6	<i>Programación de micro controlador en PCB final</i>	30
5.2.7	<i>Tabla de pines de cada componente</i>	31
6	ANÁLISIS DE RESULTADOS	34
6.1	CIRCUITO IMPRESO.....	34
6.2	TRANSMISIÓN DE DATOS POR BLUETOOTH.....	35
6.3	CÁLCULO DE TIEMPOS	36
6.4	PROTOCOLO UART	37
6.5	PROTOCOLO I2C.....	38
6.6	FUENTES BUCK-BOOST	38
6.7	DISEÑO DE CAJA	41
6.8	COSTOS	43
7	CONCLUSIONES	44
8	BIBLIOGRAFÍA	45

Índice de Figuras.

Figura 1. Sistema embebido general. [1].....	7
Figura 2. Condiciones de Start y Stop I ² C.....	8
Figura 3. Topología simple Buck[7].....	9
Figura 4. Diagrama de Bloques.....	10
Figura 5. Prueba protocolo I2C con Potenciómetros digitales. (A: dspic33fj128mc802; B: Modulo de comunicación Bluetooth; C: Potenciómetros Digitales AD5282).....	12
Figura 6. Linealidad caracterización potenciómetro digital.....	14
Figura 7. Implementación fuente Boost.....	14
Figura 8. Montaje en Protoboard LM2577.....	15
Figura 9. Esquemático de Fuente Buck.....	15
Figura 10. Diferenciador con ganancia de 51.....	17
Figura 11. Limitador de corriente con amplificadores operacionales.....	18
Figura 12. Divisor de voltaje variable [8].....	18
Figura 13. Integración total de una salida.....	19
Figura 14. Integración total con una fuente en protección.....	20
Figura 15. Condiciones de I ² C para potenciómetro digital [8].....	22
Figura 16. Información datasheet potenciómetros digitales. [8].....	23
Figura 17. Máquina de estados lectura I ² C.....	24
Figura 18. Máquina de estados escritura I ² C.....	25
Figura 19. Apariencia de la interfaz gráfica GUI.....	26
Figura 20. Herramienta GUI Builder para implementación de interfaz gráfica [9].....	26
Figura 21. Gráfico para cálculo de ancho de caminos [12].....	27
Figura 22. PCB final.....	29
Figura 23. Circuito final para montaje.....	30
Figura 24. Programación de micro controlador final.....	31
Figura 25. Distribución de pines AD5282[8].....	31
Figura 26. Distribución de pines LM2577 [20].....	32
Figura 27. Distribución de pines LM2576 [11].....	33
Figura 28. Distribución de pines DSPic33FJ128MC802[13].....	33
Figura 29. Peso obtenido del circuito final.....	34
Figura 30. Montaje final de componentes.....	35
Figura 31. Hyperterm – Recepción de datos por UART.....	36
Figura 32. Transmisión de UART.....	37
Figura 33. Tiempo de transmisión UART.....	37
Figura 34. Señal I ² C SDA.....	38
Figura 35. Gráfica regulación de carga con Vout 3 [v].....	39
Figura 36. Gráfica regulación de carga Vout 4,7 [v].....	39
Figura 37. Vista general diseño de caja.....	42

Índice de Tablas

<i>Tabla 1. Caracterización de potenciómetro digital</i>	<i>13</i>
<i>Tabla 2. Comportamiento fuente Buck</i>	<i>16</i>
<i>Tabla 3. Consideraciones para valor resistivo de variación en fuente Buck-boost.....</i>	<i>16</i>
<i>Tabla 4. Muestra de corriente en términos de voltaje.....</i>	<i>17</i>
<i>Tabla 5. Datos salida del amplificador diferenciador</i>	<i>18</i>
<i>Tabla 6. Valores de limitación con Potenciómetro Digital.....</i>	<i>19</i>
<i>Tabla 7. Distribución de pines AD5282</i>	<i>32</i>
<i>Tabla 8. Distribución de pines LM2577.....</i>	<i>32</i>
<i>Tabla 9. Distribución de pines LM2576.....</i>	<i>33</i>
<i>Tabla 10. Distribución de pines DSPIC33FJ128MC802</i>	<i>34</i>
<i>Tabla 11. Características circuito impreso.....</i>	<i>34</i>
<i>Tabla 12. Regulación de carga a 3 [v] Vout</i>	<i>38</i>
<i>Tabla 13. Regulación de carga a 4,7 [v] Vout.....</i>	<i>39</i>
<i>Tabla 14. Regulación de línea con 4,7 [v] Vout.....</i>	<i>40</i>
<i>Tabla 15. Costos de producción.....</i>	<i>43</i>

1 *Introducción*

Este proyecto surgió con la idea de solucionar un problema muy frecuente de los estudiantes de ingeniería Electrónica, brindar una fuente de voltaje portátil de un tamaño pequeño dando la facilidad de trabajar en diseños simples y de poco consumo en la comodidad de su hogar y no únicamente en la facultad de ingeniería, ya que todos los estudiantes se enfrentan a la limitación de trabajar sus diseños en las instalaciones educativas

Además de esto, también ofrece la posibilidad de incentivar el desarrollo de otros proyectos para así ir ofreciendo diferentes soluciones a los estudiantes tales como: generador de funciones, osciloscopios, *DVM*, entre otros.

El desarrollo de este proyecto involucra conocimiento de programación de micro controladores, trama de datos, protocolos de comunicación (*UART*, *I²C*), comunicación *Bluetooth*, desarrollo de interfaz gráfica, integración de software con hardware como lo es fuentes *Buck*, fuentes *Boost*, potenciómetros digitales, limitador de corriente.

Así mismo, este proyecto expande el conocimiento de la universidad Javeriana en el campo de sistemas embebidos, en este caso especialmente en el área de potencia lo cual abre las puertas a futuros proyectos e investigaciones que se busquen plantear en áreas similares.

2 *Objetivo general y objetivos específicos*

2.1 *Objetivo general*

- Desarrollar un sistema embebido de fuentes de conmutación móviles y su respectiva interfaz gráfica (*GUI*¹) el cual permita modificar la configuración del voltaje y la corriente a la salida.

2.2 *Objetivos específicos*

1. Diseñar una fuente de pequeña escala con dimensiones de (200 mm “largo” X 50 mm “alto” X 150 mm “profundo”).
2. Obtener una eficiencia en el mejor caso de 90%. Un rango de variación del voltaje a la entrada de 3-12 [v]. Una máxima potencia a la salida de 12 [watts]. Un rango de voltaje a la salida de la fuente entre 3-12 [v].
3. Desarrollar una *GUI* con la cual se pueda modificar el voltaje con una resolución de 0.1 [v] y que permita variar la limitación de corriente a la salida por puerto serial o por comunicación inalámbrica *Bluetooth*.

¹ GUI siglas para Graphic Unit Interface

3 *Marco Conceptual*

3.1 *Introducción a sistemas embebidos*

Un sistema embebido es la combinación de software y hardware en un sistema, incluyendo hasta partes mecánicas u otras, diseñadas para ejercer una función específica. Un buen ejemplo de un sistema embebido es un horno microondas, muy pocas personas son conscientes que un procesador y software está involucrado en la preparación de la comida.[2]

Todo sistema embebido posee una memoria en la cual se guarda el código ejecutable, para esto los microprocesadores cuentan con dos tipos de memoria *ROM*² y *RAM*³ si solo se requiere una pequeña cantidad de memoria esta se encuentra dentro del mismo chip, pero también está la opción de acceder desde chips de memoria externa como se conoce en los computadores de hoy en día.

La siguiente Figura 1. muestra un sistema embebido general, representado por sus entradas, salidas y memoria.

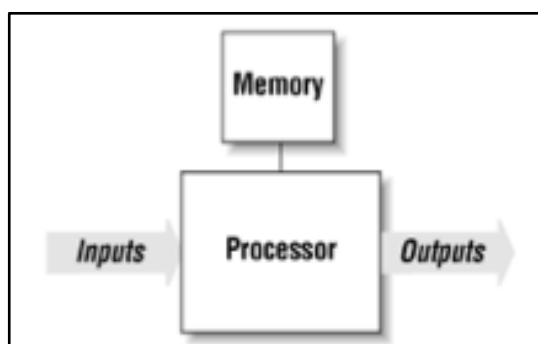


Figura 1. Sistema embebido general. [1]

3.2 *Protocolos de comunicación*

A continuación se hará una breve introducción a cada uno de los protocolos de comunicación implementados en el desarrollo de este proyecto.

3.2.1 *Introducción a I²C*

I²C es un protocolo de comunicación creado por Phillips en los 80's, permitiendo una fácil comunicación entre componentes en un mismo circuito.

Originalmente, el bus I²C fue diseñado para conectar un pequeño número de componentes en una sola tarjeta, ya sea para manejar el radio de un carro o TV. La capacitancia máxima permitida que se puede conectar a la línea de datos fue establecida en 400 pF esto con el fin de permitir los flancos de subida y bajada apropiados para la integración óptima entre la señal de reloj y data. Todos los componentes que operan bajo el protocolo I²C están diseñados para comunicarse entre ellos por el mismo bus de datos

² Siglas en Inglés de Read Only Memory

³ Siglas en Inglés de Random Access Memory

de dos cables, la arquitectura de funcionalidad está limitada solamente por la imaginación del diseñador

Las características más significativas de I²C son:

- Cada componente conectado al bus de datos cuenta con una dirección única programable por software y relación simple entre master/slave; un componente implementado como master puede tanto transmitir como recibir.
- Serial, orientado a 8 bits, bidireccional, la transferencia se puede hacer a 100 kbit/s en standard-mode, 400 kbit/s en fast-mode y hasta 3.4 Mbit/s en High-speed mode.
- Solo dos líneas de bus son requeridas SDA y SCL
- No requiere estrictas configuraciones de Baud rate, el master genera su reloj de bus
- Cada componente en la línea de datos tiene una única dirección.[4]

El protocolo de comunicación I2C ofrece la posibilidad de conectar diferentes componentes en el mismo bus de datos, esto puede implicar el uso de diferentes voltajes para cada componente y teniendo en cuenta que este protocolo funciona de forma booleana se define que para poder garantizar un 0 lógico este debe estar por debajo del treinta por ciento (30%) de VCC y el nivel alto 1 lógico debe estar por encima del setenta por ciento (70%).

Los datos en la línea SDA (Serial Data) deben permanecer estables durante un periodo alto del reloj. Los estados de alto o bajo de la línea SDA solo pueden cambiar cuando la señal de reloj SCL está en bajo, así como lo muestra la siguiente figura 2:

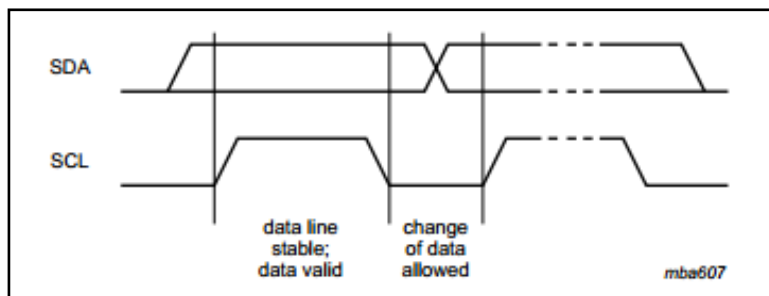


Figura 2. Condiciones de Start y Stop I²C

3.2.2 Introducción a UART

UART⁴ es un canal de comunicación bilateral asíncrono entre periféricos y computadores, usando protocolos tales como RS-232, RS-485, LIN 1.2 y IrDa. [5].

El propósito de UART es convertir los bytes del PC del bus paralelo a una trama de datos serial. Los datos se enviarán en dos cables cada uno para cada dirección de flujo de datos. El puerto serial envía una trama de bits, un bit a la vez.

Gracias a UART es posible crear una comunicación entre la interfaz gráfica y el sistema embebido.

Gracias a los registros del modulo UART el micro controlador puede leer y escribir datos a medida que se van recibiendo o enviando los datos por UART, en el caso de este proyecto los datos se van enviando por UART y se almacenan en un buffer el cual una vez se encuentre lleno aplicará un algoritmo para poder enviarlos por I²C a los respectivos componentes.

⁴ UART: Universal Asynchronous Receiver Transmitter

Es importante establecer la tasa de envío de datos Baud rate y con esto se puede conocer el tiempo que demora en enviar un bit, el cual es el inverso de Baud rate.

Es importante mantener el bit más significativo de cada palabra en 0 para que el protocolo UART entienda que un nuevo dato se va a enviar y pueda ser convertido en datos seriales para ser enviado.

El protocolo UART soporta el envío de 1 bit hasta 23, pero normalmente se hace en trama de 8, esto permite el uso de código ASCII para descifrar los datos enviados.

3.3 Fuentes de conmutación

Una fuente de conmutación regula el flujo de corriente interrumpiendo el voltaje de entrada y controlando el promedio de la corriente en términos de ciclo útil. Cuando la carga incrementa su solicitud de corriente el porcentaje de tiempo en “On” (encendido) se aumenta y se acomoda al cambio.[6]

3.3.1 Fuentes Buck

Dentro de las fuentes que no poseen transformador, sino que se cuenta con un paso previo para suministrar un voltaje DC hay tres topologías básicas, buck (reducción), Boost (Amplificación), y Buck-Boost (inversora). Cada topología genera y regula la salida de voltaje por encima o debajo del voltaje de entrada.

La siguiente imagen (Figura 3) muestra una topología simple de una fuente Buck.

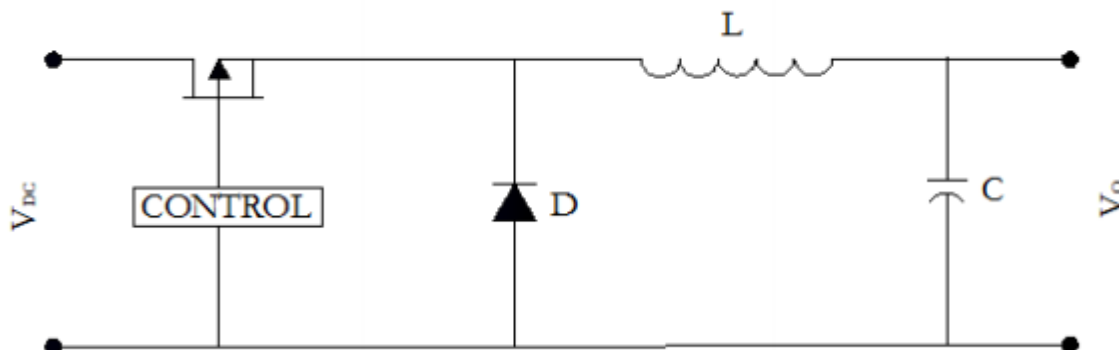


Figura 3. Topología simple Buck[7]

El transistor conmuta entre dos estados, el primero cuando se encuentra circulando corriente por la inductancia y en este momento el diodo se encuentra polarizado inversamente lo que impide el paso de corriente a la salida, cuando el transistor se apaga la inductancia deja de recibir corriente y con la almacenada que obtuvo polariza el diodo de manera que este permite el paso de corriente a través del circuito.

4 *Especificaciones*

La fuente BSE⁵ está conformada por 1 fuente *Boost* fija a 15 [v] la cual se encarga de la polarización de todos los integrados del sistema embebido, un regulador de voltaje de 3,3 voltios el cual alimenta el micro controlador y bluetooth y 3 fuentes *Buck* variables y configurables gracias al divisor resistivo de realimentación.

El siguiente diagrama en bloques (Figura 4) muestra como está conformada una salida de la fuente de voltaje BSE, para las otras dos salidas la misma estructura se aplica.

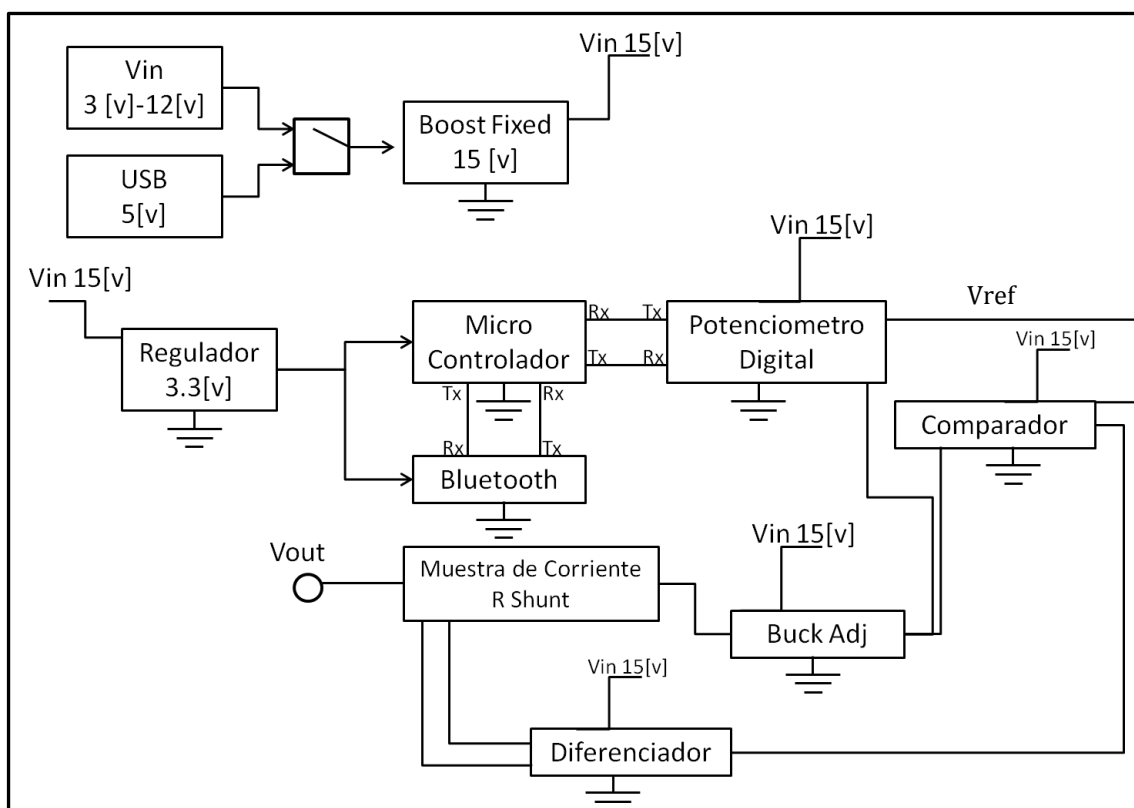


Figura 4. Diagrama de Bloques

Como resultado se obtuvo una fuente de voltaje de las siguientes características:

- Voltaje de entrada V_{in} : 3-12 [v]
- Voltaje de salida V_{out} : 3-12.5 [v]
- Resolución 0.1 [v]
- Potencia máxima: 18 [w]
- Eficiencia máxima: 76.9%

Los siguientes componentes y desarrollos de software fueron los implementados para el desarrollo del proyecto:

4.1 *Hardware*

- 1 fuente Boost fija a 15 [v] LM2577
- 3 fuentes Buck variables LM2576
- 3 potenciómetros digitales AD5282
- Regulador de voltaje 3.3 [v]

⁵ BSE Siglas para la fuente de este proyecto (Bench Supply Equipment)

- 1 dspic33fj128MC802
- 3 Amplificadores operacionales LM358

4.2 *Software*

- Protocolo de comunicación UART
- Protocolo de comunicación I²C
- GUI Interfaz gráfica para la configuración de cada puerto

5 *Desarrollos*

En este ítem se explicará cada uno de los procedimientos desarrollados en el transcurso de ejecución del proyecto, tanto hardware (Fuentes Buck, Boost, Potenciómetros digitales y limitador de corriente). como de software (protocolos de comunicación, programación de GUI y algoritmo de funcionamiento)

5.1 *Hardware*

En este ítem se explicará el proceso para cada componente usado en el desarrollo del proyecto.

5.1.1 *AD5282 Potenciómetro digital*

Contando con el protocolo de comunicación I2C se hicieron las pruebas con un solo integrado conectado al bus de datos de I2C del micro controlador, el siguiente comando escribe los tres tramos de datos para configurar el potenciómetro digital, en este caso a la dirección 0-0 (la cual está definida con dos pines de dirección y con su valor lógico), opción A o B (potenciómetro 1 o 2) y un valor entre 0 y 255.

- `I2Cwrite(ADD00W,0b00000000,1);`

La siguiente figura 5 muestra el montaje que se implementó para las pruebas del potenciómetro digital.

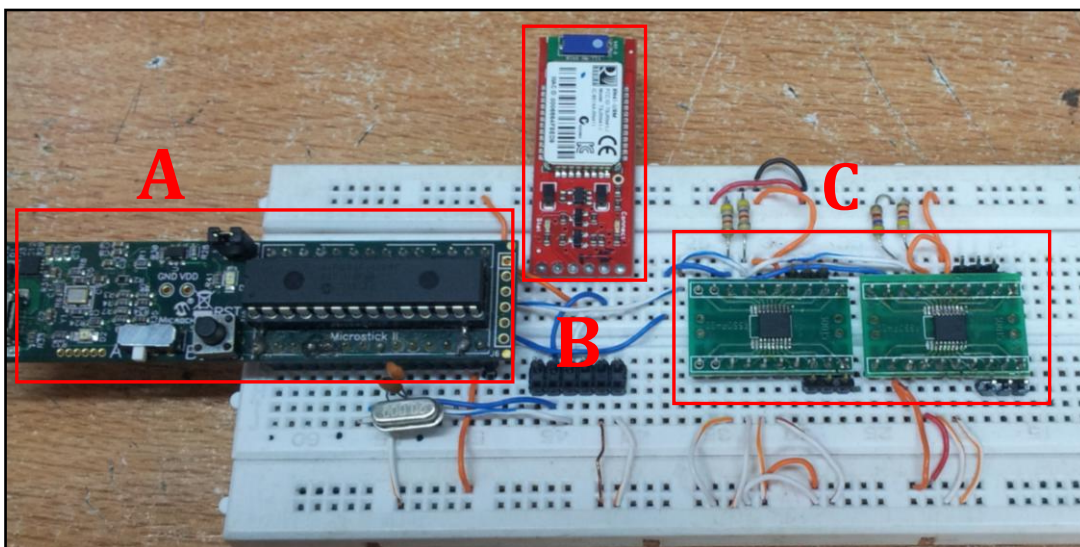


Figura 5. Prueba protocolo I2C con Potenciómetros digitales. (A: dspic33fj128mc802; B: Modulo de comunicación Bluetooth; C: Potenciómetros Digitales AD5282)

Una vez se comprueba que el protocolo funciona correctamente y la configuración de los potenciómetros se realiza con éxito, se caracterizó el potenciómetro digital comparando los valores teóricos vs los valores medidos y se obtiene:

Rw [Ω]	Rab [Ω]	Rwb [Ω]	n	Rwb medido [Ω]	Error [%]
60	20000	138.125	1	139.25	0.8%
60	20000	450.625	5	449.5	0.2%
60	20000	841.25	10	842.25	0.1%
60	20000	1231.875	15	1237.5	0.5%
60	20000	1622.5	20	1624	0.1%
60	20000	2013.125	25	2013.25	0.0%
60	20000	2403.75	30	2410	0.3%
60	20000	2794.375	35	2798.25	0.1%
60	20000	3185	40	3188.75	0.1%
60	20000	3575.625	45	3575.25	0.0%
60	20000	3966.25	50	3970.25	0.1%
60	20000	4356.875	55	4359.25	0.1%
60	20000	4747.5	60	4749.25	0.0%
60	20000	5138.125	65	5140.25	0.0%
60	20000	5528.75	70	5550.25	0.4%
60	20000	5919.375	75	6001.2	1.4%
60	20000	6310	80	6402.25	1.5%
60	20000	6700.625	85	6725.25	0.4%
60	20000	7091.25	90	7102.25	0.2%
60	20000	7481.875	95	7490.2	0.1%
60	20000	7872.5	100	7882.25	0.1%
60	20000	8263.125	105	8270.2	0.1%
60	20000	8653.75	110	8660.7	0.1%
60	20000	9044.375	115	9044	0.0%
60	20000	9435	120	9460	0.3%
60	20000	9825.625	125	9840.25	0.1%

Tabla 1. Caracterización de potenciómetro digital

Teniendo:

$$Rwb(D) = \frac{D}{256} Rab + Rw \quad [8]$$

Siendo D un valor entre 0 y 255 de la configuración, Rab el valor máximo, en este caso 20 k Ω , Rwb la resistencia variable entre los terminales de salida y Rw la resistencia más pequeña configurada con D=0 (60 Ω).

Como se puede observar los valores tienen una pequeña variación con respecto a lo esperado, sin embargo, este pequeño error no afecta el funcionamiento ni el propósito del potenciómetro digital

Haciendo una gráfica de linealidad de los datos y comparándolos con lo teórico se obtiene la siguiente figura 6:

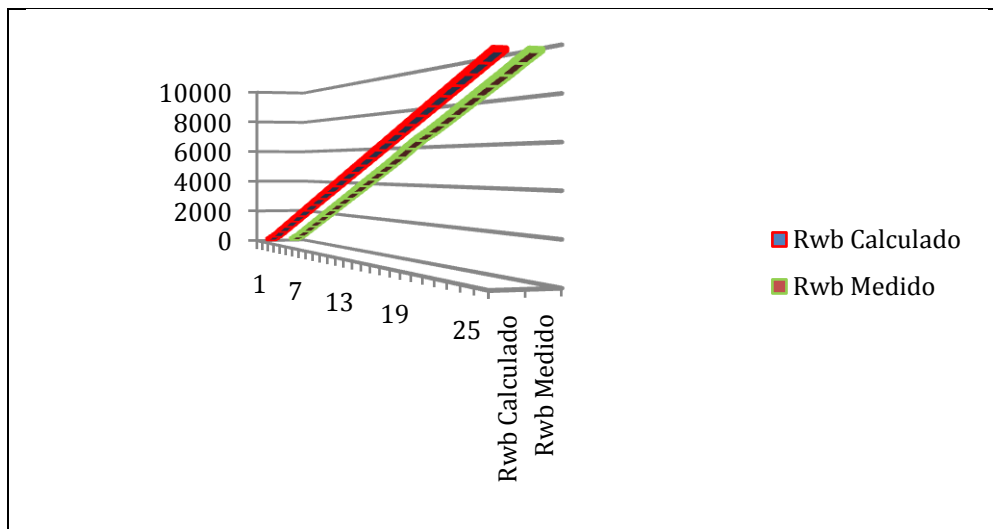


Figura 6. Linealidad caracterización potenciómetro digital

Este gráfico da una idea del comportamiento del potenciómetro digital, mostrando su linealidad y su comportamiento muy similar a lo esperado teóricamente.

5.1.2 LM2577 (Boost) Y LM2576 (Buck)

- LM2577 (Boost)

Una de las razones por la cual se escogió esta fuente Boost fue por su entrega a la salida de hasta 3 [A], este componente viene en versiones ajustables; Para el desarrollo de este proyecto se usa la versión fija a 15[v] la cual se encarga de polarizar todo el sistema embebido.

El siguiente esquemático figura 7 y montaje en protoboard figura 8 muestran el desarrollo de la fuente boost:

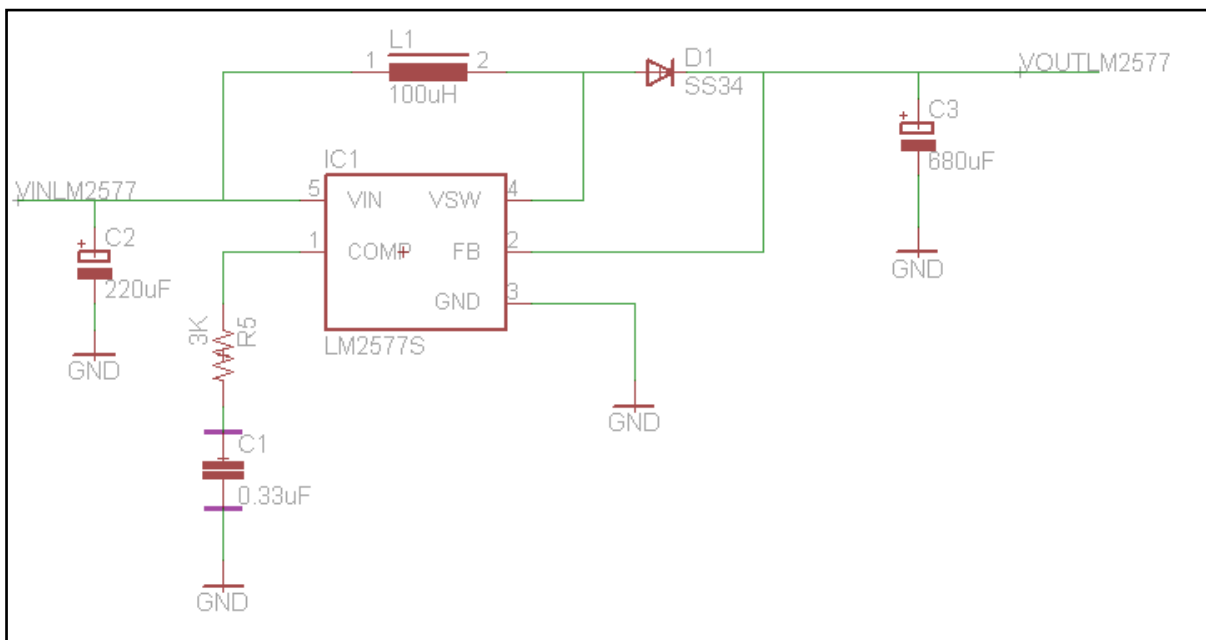


Figura 7. Implementación fuente Boost.

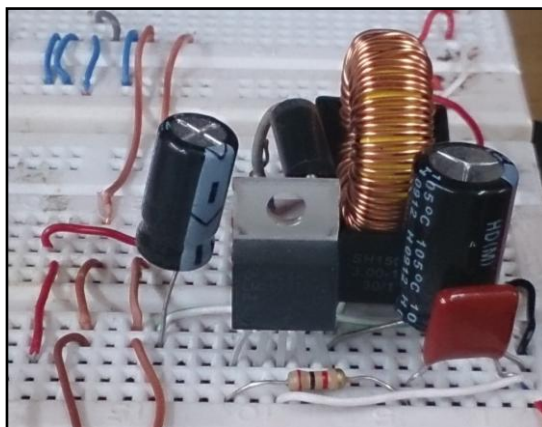


Figura 8. Montaje en Protoboard LM2577

Una vez montado y comprobado el funcionamiento de la fuente boost se procede a diseñar cada salida a partir de fuentes Buck.

- **LM2576 (Buck)**

Una vez se tiene un voltaje estable de 15 voltios para el circuito, se procede a diseñar las fuentes encargadas de cada una de las salidas de la fuente BSE, en este caso se usa el componente LM2576 (buck) de la misma familia del LM2577 (boost), este se consigue en diferentes presentaciones, entre estas, unos voltajes fijos de salida y uno variable el cual depende del divisor resistivo de la realimentación del componente.

La primera prueba con este componente es usando resistencias discretas y fijas, esto con el fin de comprobar su correcto funcionamiento y su estabilidad, se calculan las resistencias del divisor resistivo de acuerdo con la siguiente fórmula

$$V_{out} = V_{ref} \left(1 + \frac{R2}{R1} \right)$$

Donde $V_{ref} = 1.23 [v]$

Y se monta el siguiente circuito en protoboard para evaluar su funcionamiento:

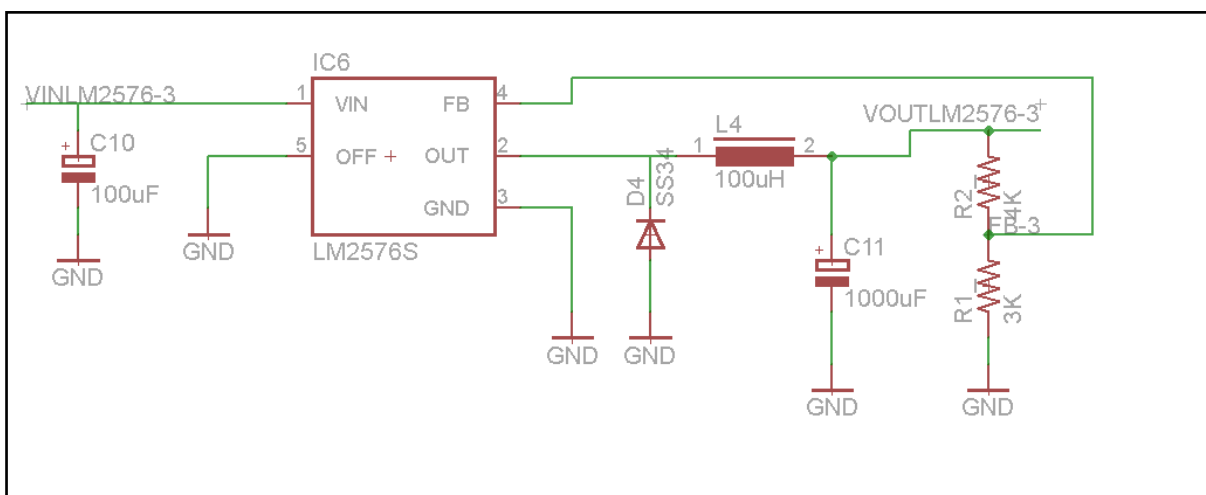


Figura 9. Esquemático de Fuente Buck.

El siguiente cuadro nos muestra el comportamiento de la fuente Buck con diferentes juegos de resistencias discretas en el divisor de realimentación:

Siendo R2 la resistencia de VOUT al pin de realimentación FB y R1 la de FB a tierra, las cuales se encargan del ajuste del voltaje siendo el divisor resistivo de realimentación.

Vin [v]	R1[Ω]	R2 [Ω]	Vout Teorico [v]	Vout medido [v]	Error [%]
15	2193,0	3305	3,08	3,10	0,65
15	2192,5	4700	3,87	3,91	1,02
15	2192,5	5160	4,12	4,18	1,44
15	2192,5	6930	5,12	5,16	0,78

Tabla 2. Comportamiento fuente Buck

Es importante aclarar que una vez se configuraba un voltaje este no variaba y a pesar de que mostraba un error con respecto al valor teórico esperado, este puede ser compensado ya que sigue comportándose como se espera y mantiene la linealidad en su configuración

El siguiente paso es la integración con el potenciómetro digital. Para esto se hicieron ciertos cálculos primero con el fin de poder usar todo el rango de variación del potenciómetro digital, en este caso 20kΩ

Los siguientes son los cálculos implementados para los valores resistivos del divisor de realimentación de la fuente:

Se plantearon dos casos para la fuente Buck, uno en la que el voltaje a la salida sea 3 [v] y otro 12 [v] (tabla 3) y conociendo la fórmula de operación de la fuente buck se tiene:

$$V_{out} = 1.23 * \left(\frac{R_2}{R_1} + 1 \right) \quad [11]$$

Vout = 3 [v]	Vout = 12 [v]
$1.439 = \frac{R_1}{R_2}$	$8.756 = \frac{R_1}{R_2}$
$1.439 R_1 < R_2 < 8.756 R_1$ Si $R_1 = 2000[k\Omega]$ $2878[\Omega] < R_2 < 17512[\Omega]$	

Tabla 3. Consideraciones para valor resistivo de variación en fuente Buck-boost.

Gracias a estos cálculos se definen las resistencias necesarias en el divisor de realimentación y de este modo variar el voltaje según lo planeado.

Como podemos ver en este caso los pasos que daría el potenciómetro digital son de 78.125 Ω, esto significa que el voltaje tendrá una variación de 0.04 [v] por cada una de las 256 posiciones del potenciómetro digital.

5.1.3 Limitador de corriente

El primer paso para poder limitar la corriente a la salida es muestrearla, para esto se cuenta con una resistencia Shunt, la cual permite el paso de la corriente presentando caída de voltaje proporcional a esta, el primer experimento fue conocer el comportamiento de esta R shunt en términos de voltaje para representar la corriente, para esto se conectó la fuente en serie a la R shunt y se fue variando la carga a la salida y se obtuvo lo siguiente:

lout [A]	Vrshunt [v]
0.5	0.05
0.8	0.08
1	0.1
1.5	0.15
2	0.2

Tabla 4. Muestra de corriente en términos de voltaje.

Como se observa la caída de voltaje por la resistencia shunt es proporcional a la corriente entregada a la carga

Conociendo la diferencia de potencial que pasa por la resistencia shunt se diseña un amplificador diferenciador el cual toma como muestra el voltaje diferencial por la resistencia y lo amplifica en un factor deseado. Para este diseño la ganancia del amplificador esta en 51 y se define así para obtener 10.2 voltios cuando haya una corriente de 2 A de esta forma obtener un rango amplio de variación pero sin ir hasta el límite de polarización de los amplificadores, la resistencia shunt es de 0,1 Ω .

El siguiente esquemático muestra el montaje del diferenciador.

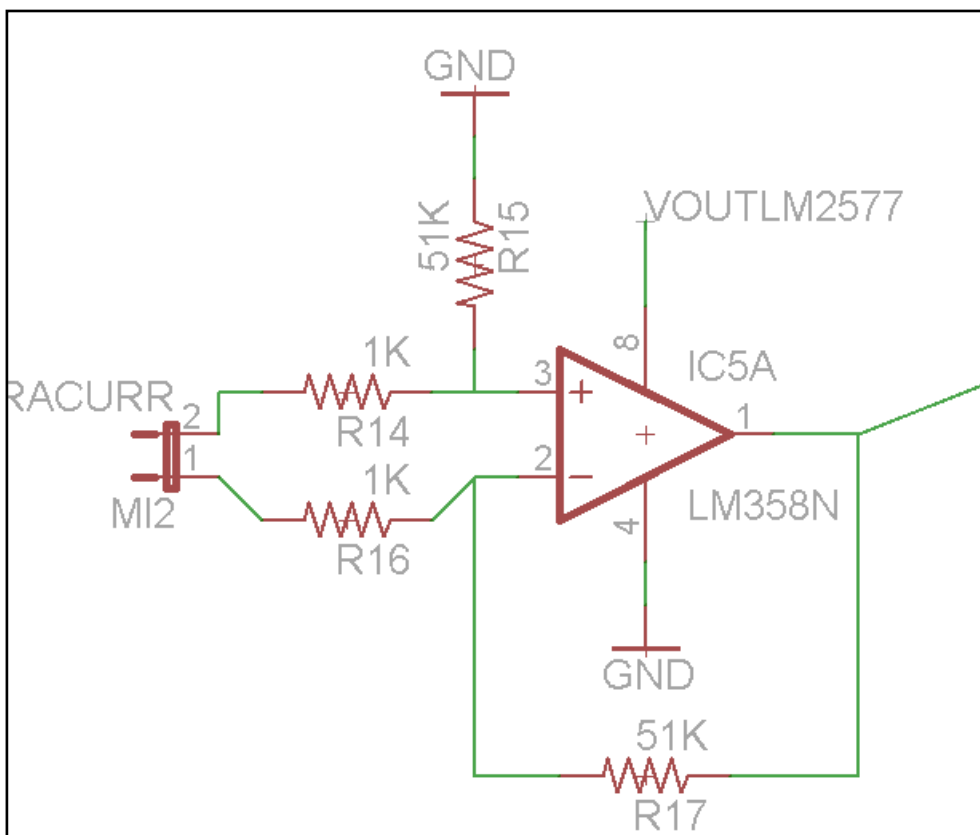


Figura 10. Diferenciador con ganancia de 51

Contando con la ganancia del amplificador de 51 la siguiente tabla resume lo obtenido:

lout [A]	Vrshunt [v]	Vout amp [v]
0.5	0.05	2.55
0.8	0.08	4.08

I_{out} [A]	V_{rshunt} [v]	V_{out} amp [v]
1	0.1	5.1
1.5	0.15	7.65
2	0.2	10.2

Tabla 5. Datos salida del amplificador diferenciador

Una vez se tiene el voltaje proporcional a la corriente amplificado se hace un comparador, en el cual se establece como voltaje de referencia el voltaje equivalente a la corriente que se desea limitar, de este modo si el voltaje muestreado de la resistencia R shunt supera el voltaje de referencia a la salida del amplificador se obtendrá un alto y mientras este voltaje no se supere estará en bajo.

El siguiente esquemático muestra la integración de las dos etapas del amplificador y su salida la cual va al pin de realimentación (FB) de la fuente *Buck*

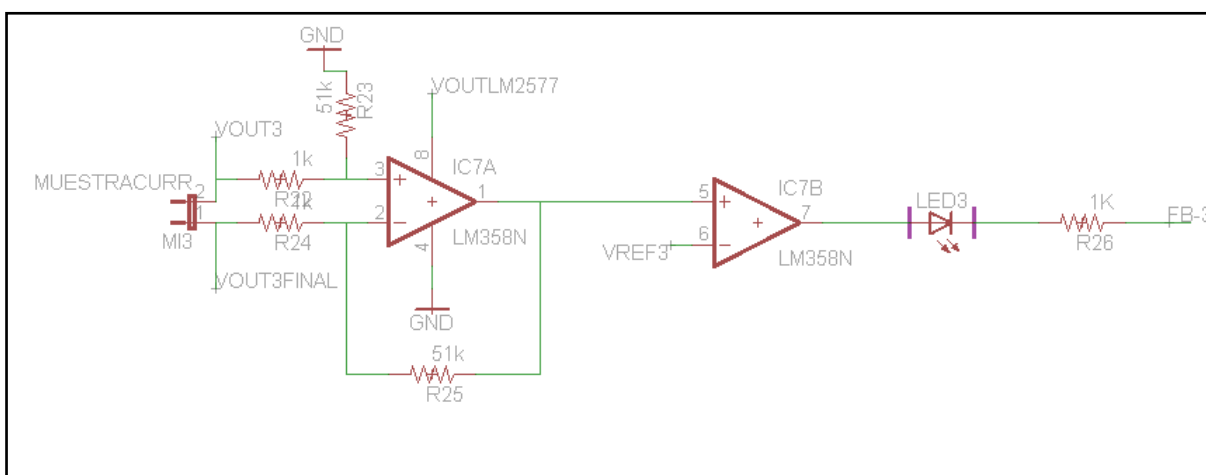


Figura 11. Limitador de corriente con amplificadores operacionales

Un paso importante es establecer el voltaje de referencia el cual define la corriente máxima permitida, para esto se usa un potenciómetro digital usado en modo divisor variable, el cual conecta el terminal A a +Vcc y el B a tierra dejando una variación en el terminal W de 256 pasos desde A (+Vcc 15 [v]) hasta B tierra, la siguiente figura ilustra el uso del potenciómetro digital.

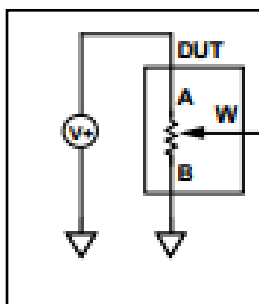


Figura 12. Divisor de voltaje variable [8]

El voltaje a la salida del terminal W esta dado por la siguiente fórmula:

$$V_w = \frac{D}{256} V_a \quad [8]$$

Donde D es un valor entre 0 y 256. Teniendo esto simplemente se ajusta el voltaje deseado para la limitación de corriente de acuerdo con la muestra de corriente en términos de voltaje dada por la R shunt.

La siguiente tabla resume los valores necesarios de D para cada limitación de corriente

Iout [A]	Vrshunt [v]	Vout amp [v]	D	Vref [v]
0.2	0.02	1.02	17.408	1.02
0.3	0.03	1.53	26.112	1.53
0.4	0.04	2.04	34.816	2.04
0.5	0.05	2.55	43.52	2.55
0.6	0.06	3.06	52.224	3.06
0.7	0.07	3.57	60.928	3.57
0.8	0.08	4.08	69.632	4.08
0.9	0.09	4.59	78.336	4.59
1	0.1	5.1	87.04	5.1
1.1	0.11	5.61	95.744	5.61
1.2	0.12	6.12	104.448	6.12
1.3	0.13	6.63	113.152	6.63
1.4	0.14	7.14	121.856	7.14
1.5	0.15	7.65	130.56	7.65
1.6	0.16	8.16	139.264	8.16
1.7	0.17	8.67	147.968	8.67
1.8	0.18	9.18	156.672	9.18
1.9	0.19	9.69	165.376	9.69
2	0.2	10.2	174.08	10.2

Tabla 6. Valores de limitación con Potenciómetro Digital

5.1.4 Potenciómetro Digital + Limitador de Corriente + Fuente Buck

Después de probar y entender el funcionamiento de cada una de las partes fundamentales del proyecto se unificó una salida completa con el micro controlador con los potenciómetros digitales y estos conectados a la limitación de corriente y el voltaje de la fuente.

La siguiente imagen (Figura 13) muestra la integración total de dos etapas de salida

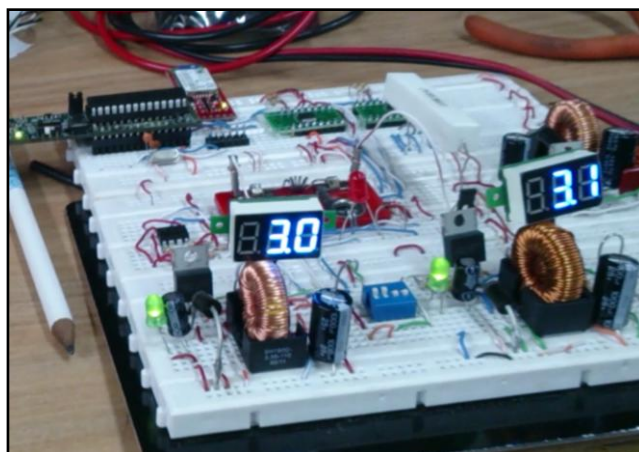


Figura 13. Integración total de una salida

En esta etapa de desarrollo el usuario cuenta con una interfaz gráfica para variar el voltaje y limitar la corriente, al definir una configuración la trama de datos es enviada por bluetooth al micro controlador

William David Riaño

el cual recibe por puerto UART los datos y los transmite por I2C a cada uno de los potenciómetros digitales, estos se configuran de acuerdo con los valores recibidos y de esta forma modifican el voltaje a la salida de cada fuente y limitan la corriente.

Si la carga a la salida de la fuente pide más corriente que la establecida, el comparador tendrá a la salida un alto el cual llega a la realimentación de la fuente Buck engañándola de tal forma que baje el voltaje para compensar con respecto a la limitación .

La siguiente imagen nos muestra una configuración de 3.1 [v] para ambas salidas y una de estas entrando en protección

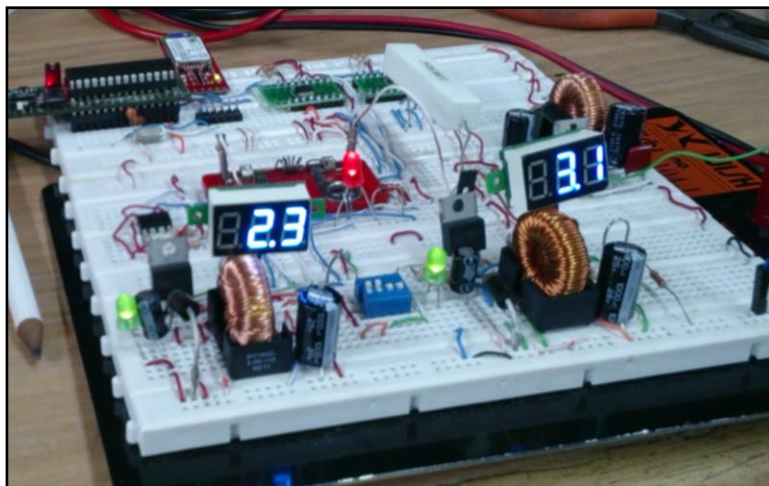


Figura 14. Integración total con una fuente en protección

Como se aprecia en la figura 20, cuando se conecta una carga que excede la limitación de corriente configurada el voltaje cae hasta llegar a compensar la corriente límite y el led indicador rojo se enciende informando que entró en dicha protección.

5.2 *Software*

Para la programación de software, primero fue necesario tener claro los protocolos de comunicación necesarios, el primer protocolo fue UART con el cual se establece comunicación desde la interfaz gráfica al micro controlador entregando los datos de cada configuración escogida por el usuario. El segundo protocolo fue I2C este fue necesitado por los potenciómetros digitales con la ventaja de usar solo dos líneas de datos y asignar a cada potenciómetro con una dirección única.

5.2.1 *UART*

Para la implementación de este periférico fue necesaria una extensa búsqueda del funcionamiento de cada registro, el datasheet del micro controlador DSPIC suministra la información necesaria del modo de operación de UART y el efecto de cada registro.

Para la implementación del protocolo UART se programó el siguiente Header en el cual se define la frecuencia Baud rate y los prototipos de cada función:

UART.h

```

#if !defined(__UART_H)
#define __UART_H
#include <p33FJ128MC802.h>
#define FOSC()                (20000000ul)
#define FCY()                 (FOSC()/2)
#define BAUDRATE              9600ul
#define MYBAUDRATE            (((FCY())-(4*BAUDRATE))/(4*BAUDRATE))
#define SETBAUDRATE           ((FCY())/(4*(MYBAUDRATE+1)))

// ----- Prototipo de Función -----

void UART_Init();

void UART_Put_Char(char c);

void UART_Put_String(char *s);

char UART_Get_Char(void);

```

El siguiente código es el compilador de UART en el cual se configuran los registros necesarios, para iniciar el puerto de transmisión UART y establecer sus pines de salida. Cada programa puede necesitar configurar más o menos registros esto depende solamente del tipo de comunicación necesaria entre los programas por sus respectivos requerimientos.

UART.c

```

#include "UART2.h"
void UART_Init(void){
IEC1bits.U2RXIE = 0; // Solicitud de interrupción no activa pg 109
U2MODEbits.UARTEN = 0; //Uart2 desactivo pg 242
U2BRG = MYBAUDRATE;
U2MODE &= 0b00000000;
U2MODEbits.BRGH = 1; //BRG genera 16 ciclos por periodo de bit 16x pg 243
U2MODEbits.UARTEN = 1; // Uart2 Activo, Todos los pins son controlados por UARTx pg 242
U2STAbits.UTXEN = 1; // Transmision activa pg 244
RPINR19bits.U2RXR = 0b00010; Input seleccionado RP2 (Pin 6)
RPOR1bits.RP3R = 0b00101; // Transmisión ligado a pin RP3 pg 167
};
void UART_Put_Char(char c){
while(U2STAbits.UTXBF); //mientras transmite, Activo
U2TXREG = c;
IFS1bits.U2TXIF = 0;
};
char UART_Get_Char(void){
while(!U2STAbits.URXDA); //Mientras no esté completo, al menos un carácter mas
IFS1bits.U2RXIF = 0;
return U2RXREG;
};

```

Los cálculos realizados para el funcionamiento del protocolo UART fueron los siguientes:

```

#define FOSC()                (20000000ul)
#define FCY()                 (FOSC()/2)
#define BAUDRATE              9600ul
#define MYBAUDRATE            (((FCY())-(4*BAUDRATE))/(4*BAUDRATE))
#define SETBAUDRATE           ((FCY())/(4*(MYBAUDRATE+1)))

```

$$F_{cy} = \frac{F_{osc}}{2}$$

$$F_{cy} = 10\text{Mhz}$$

$$\text{Baud rate deseado} = 9600$$

$$UxBRG = \frac{(F_{cy}/\text{Baud rate} - \text{deseado})}{4} - 1$$

$$UxBRG = \frac{10\text{Mhz}/9600}{4} - 1$$

$$UxBRG = 259.417$$

Estos valores son definidos como condiciones del micro controlador para poder iniciar la transmisión del puerto UART, en estos cálculos se definen las frecuencias de operación y la velocidad de transmisión de datos Baud Rate.

El valor UxBRG es el que se le aplica al registro que controla el periodo del temporizador de 16 bits.

Corroborando el valor obtenido se tiene:

$$\text{Baud rate} = \frac{F_{cy}}{4 (UxBRG + 1)}$$

$$\text{Baud rate} = \frac{10\text{Mhz}}{4 (259.417 + 1)} = 9600$$

5.2.2 I²C

Para la implementación del protocolo I²C se estudió el funcionamiento del potenciómetro digital. El *datasheet* del potenciómetro digital nos da la siguiente información (figura 15):

S	0	1	0	1	1	AD1	AD0	R/W	A	A/B	RS	SD	O ₁	O ₂	X	X	X	A	D7	D6	D5	D4	D3	D2	D1	D0	A	P
Slave Address Byte						Instruction Byte						Data Byte																

where:
 S = start condition
 P = stop condition
 A = acknowledge
 X = don't care
 AD1, AD0 = Package pin programmable address bits. Must be matched with the logic states at Pins AD1 and AD0.
 R/W = Read enable at high and output to SDA. Write enable at low.
 A/B = RDAC subaddress select; 0 for RDAC1 and 1 for RDAC2.
 RS = Midscale reset, active high.
 SD = Shutdown in active high. Same as SHDN except inverse logic.
 O₁, O₂ = Output logic pin latched values
 D7, D6, D5, D4, D3, D2, D1, D0 = data bits.

Figura 15. Condiciones de I²C para potenciómetro digital [8]

A continuación se muestra cómo se tienen que enviar los datos para una correcta configuración y teniendo en cuenta que cada integrado cuenta con dos potenciómetros digitales:

1. Bit de comienzo.
2. 8 bits de la dirección del componente + leer o escribir.
3. 8 bits con la opción de configurar potenciómetro A o B y salidas lógicas.

4. 8 bits con la configuración de un valor entre 0 y 255.

Con esta información se crean las siguientes funciones prototipo

```
//Iniciar periférico I2C
void i2c_init(void);

// Escribir a una dirección
void I2Cwrite(char addr, char subaddr, char value);

//leer de una dirección
char I2Cread(char addr);
```

Con estas funciones se desarrolla todo el protocolo de comunicación con los potenciómetros digitales, en los anexos se encuentra el programa en su totalidad, es importante aclarar que si deseo leer los datos configurados de cada potenciómetro digital hay que tener en cuenta que solo se puede hacer seguido de una escritura, es decir, la función de lectura de *I²C* se hace a la escritura previa, una vez se configura el segundo potenciómetro de cada integrado es imposible hacer una lectura a la primera configuración.

5.2.2.1 Frecuencia de operación

La frecuencia de operación máxima sobre la cual se comunican los tres dispositivos conectados, se calcula de la siguiente manera:

$$F_{clk} = 1/t_{low(min)} + t_{high(min)} + t_{r(actual)} + t_{f(actual)}$$

Siendo $t_{low(min)}$ el periodo en bajo del reloj, $t_{high(min)}$ el periodo en alto de reloj, $t_{r(actual)}$ el periodo de subida para las señales de reloj y data y $t_{f(actual)}$ el tiempo de caída para las señales de reloj y data.

Esta información la da el datasheet de los potenciómetros digitales AD5282, como se muestra en la siguiente figura 16:

INTERFACE TIMING CHARACTERISTICS (APPLIES TO ALL PARTS ^{5, 9})						
SCL Clock Frequency	f_{SCL}		0	400	kHz	
Bus Free Time Between Stop and Start, t_{BUF}	t_1		1.3		μ s	
Hold Time (Repeated Start), $t_{HD, STA}$	t_2	After this period, the first clock pulse is generated	600		ns	
Low Period of SCL Clock, t_{LOW}	t_3		1.3		μ s	
High Period of SCL Clock, t_{HIGH}	t_4		0.6	50	μ s	
Setup Time for Repeated Start Condition, $t_{SU, STA}$	t_5		600		ns	
Data Hold Time, $t_{HD, DAT}$	t_6			900	ns	
Data Setup Time, $t_{SU, DAT}$	t_7		100		ns	
Rise Time of Both SDA and SCL Signals, t_r	t_8			300	ns	
Fall Time of Both SDA and SCL Signals, t_f	t_9			300	ns	
Setup Time for Stop Condition, $t_{SU, STO}$	t_{10}					

Figura 16. Información datasheet potenciómetros digitales. [8]

Mirando la tabla obtenemos que:

$$F_{clk} = \frac{1}{1,3\mu s + 0,6\mu s + 300ns + 300ns}$$

Y se puede concluir, al igual que lo dice el datasheet, la frecuencia de operación

$$F_{clk} = 400 \text{ kHz}$$

5.2.2.2 Máquina de Estados

Los siguientes diagramas de flujo describen la máquina de estados de I2C (figura 17 y 18):

- Lectura por I²C

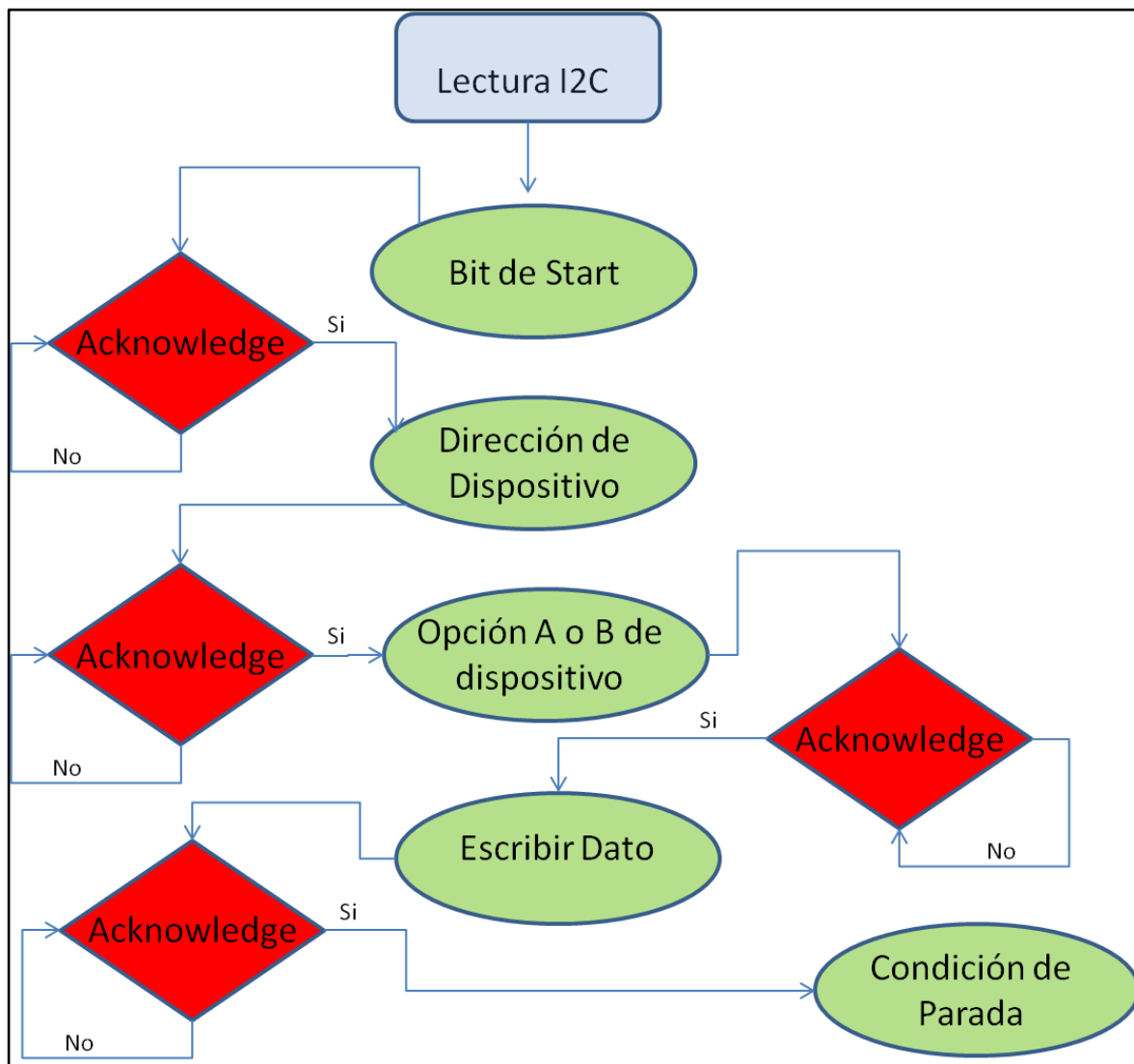


Figura 17. Máquina de estados lectura I²C

- Escritura por I^2C

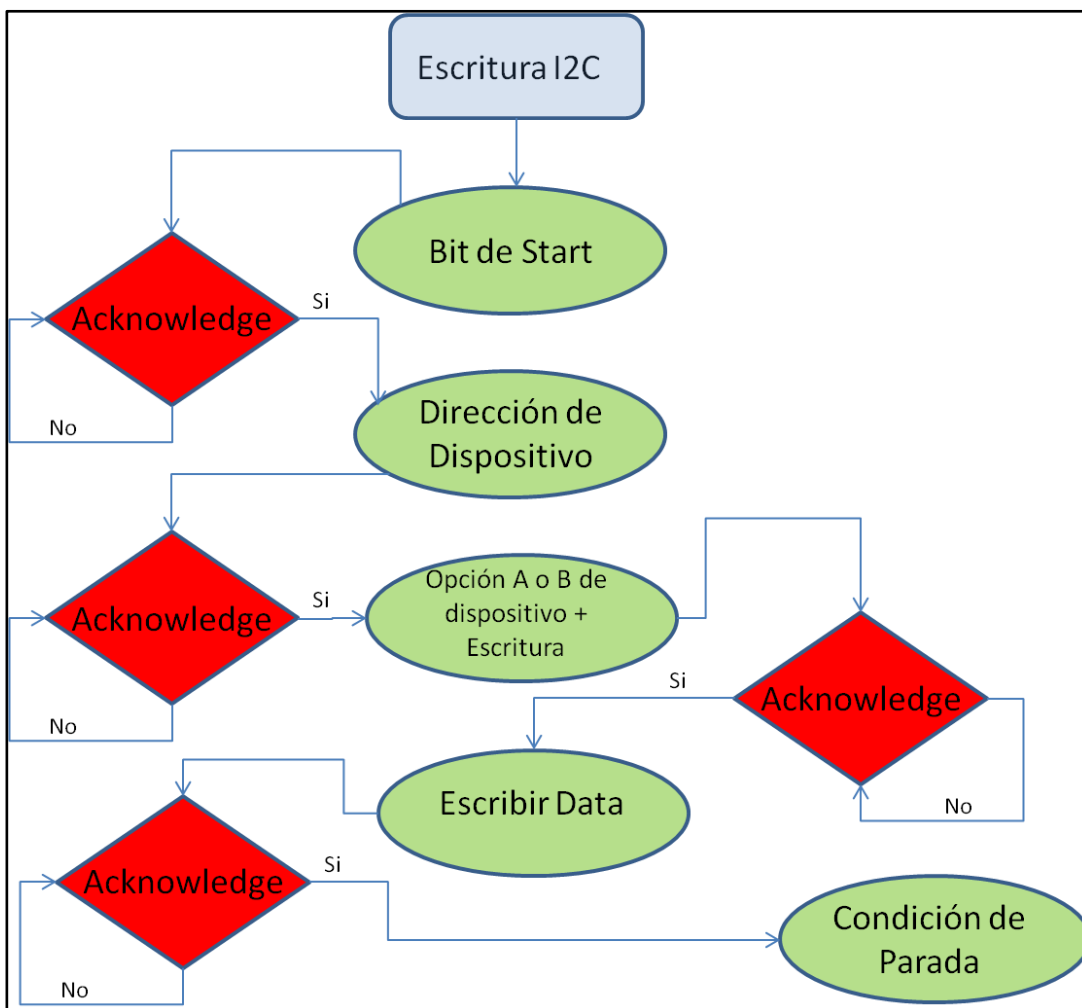


Figura 18. Máquina de estados escritura I^2C

Estas maquinas de estados nos ilustran el proceso por el cual pasan los datos por el micro controlador para escribir o leer datos de $I2C$.

5.2.3 GUI (Interfaz Gráfica)

La interfaz gráfica se desarrolló en la plataforma Processing [21] la cual es gratuita y ofrece la facilidad de la interacción entre el micro controlador y la plataforma desarrollada, en el caso de este proyecto y para cumplir con el protocolo de comunicación deseado se hace una condición inicial que cada configuración va a enviar por UART 16 bits de ceros (0) y en el micro se establece que solamente después de recibir 16 ceros se cumple la condición para poder configurar, esto se hace con el fin de evitar configuraciones indeseadas.

A continuación las siguientes imágenes muestran la interfaz gráfica y su implementación (figura 19 y 20):

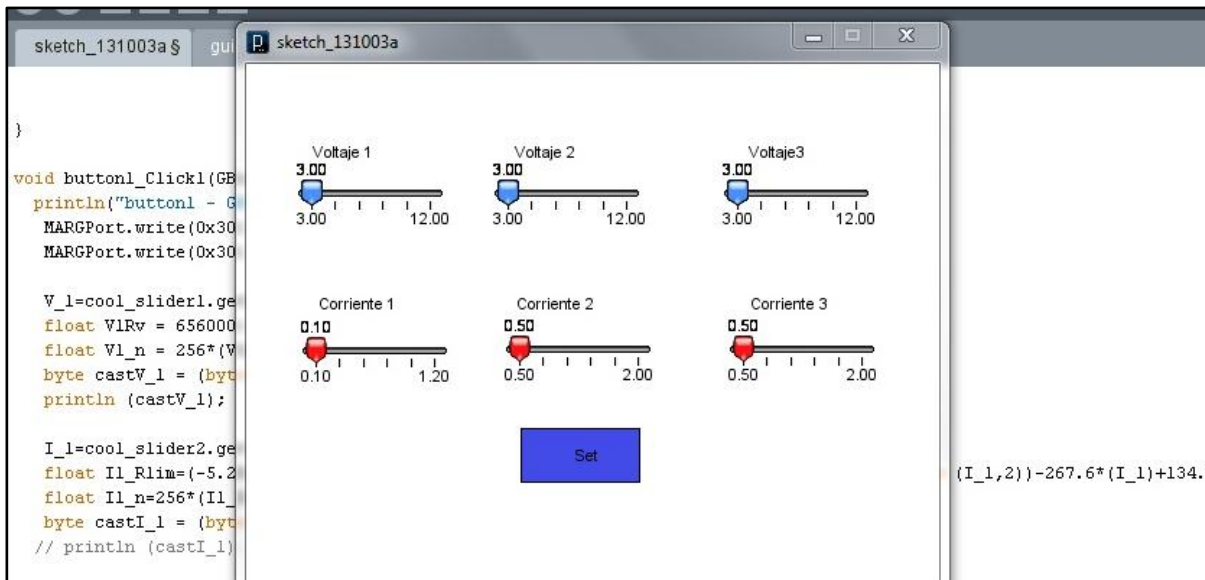


Figura 19. Apariencia de la interfaz gráfica GUI

Como podemos ver en la imagen anterior se cuenta con 3 barras de configuración de voltaje y 3 de limitación de corriente, esto para cada salida. Una vez se configura el valor deseado y se presiona el botón de configurar el programa lee los datos de cada barra y los envía por I2C a cada potenciómetro para configurarlo

Se contó con la ayuda de una herramienta llamada GUI Builder Tool la cual da la opción de crear barras, botones y diseño simple de una interfaz gráfica, generando el código básico sobre el cual se trabajó y se implementó lo necesario para el funcionamiento, la siguiente figura muestra el diseño de la herramienta

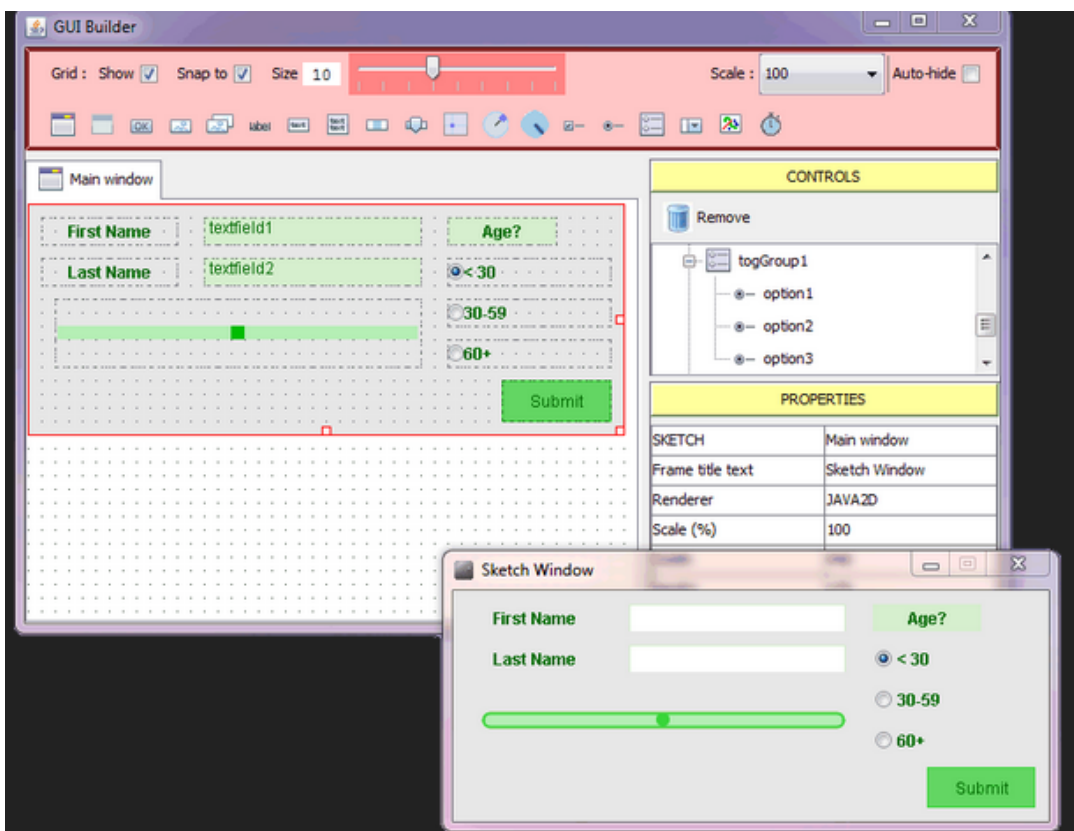


Figura 20. Herramienta GUI Builder para implementación de interfaz gráfica [9]

El código implementado para la interfaz gráfica se encuentra en los anexos.

5.2.4 Circuito Impreso

Para el circuito impreso final se tuvieron en cuenta tanto las recomendaciones de cada fabricante de cada integrado, como lo son, condensadores de desacople, proximidad de componentes que le brindan estabilidad al comportamiento del integrado entre otros, además de esto es importante tener claridad cuales caminos van a suministrar alta corriente ya que de esto depende el ancho de los caminos de pista del PCB, de acuerdo con la siguiente figura se concluye lo siguiente:

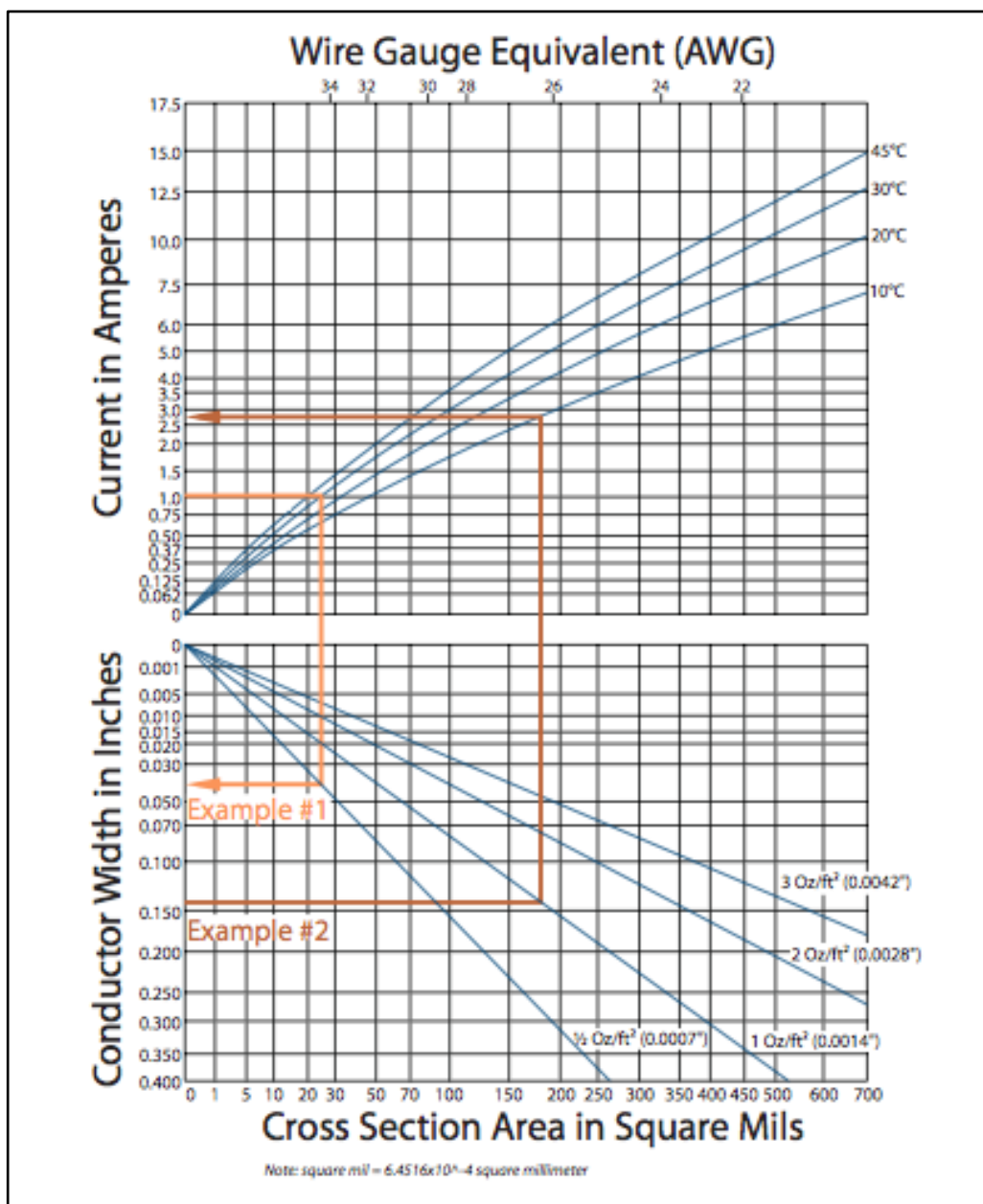
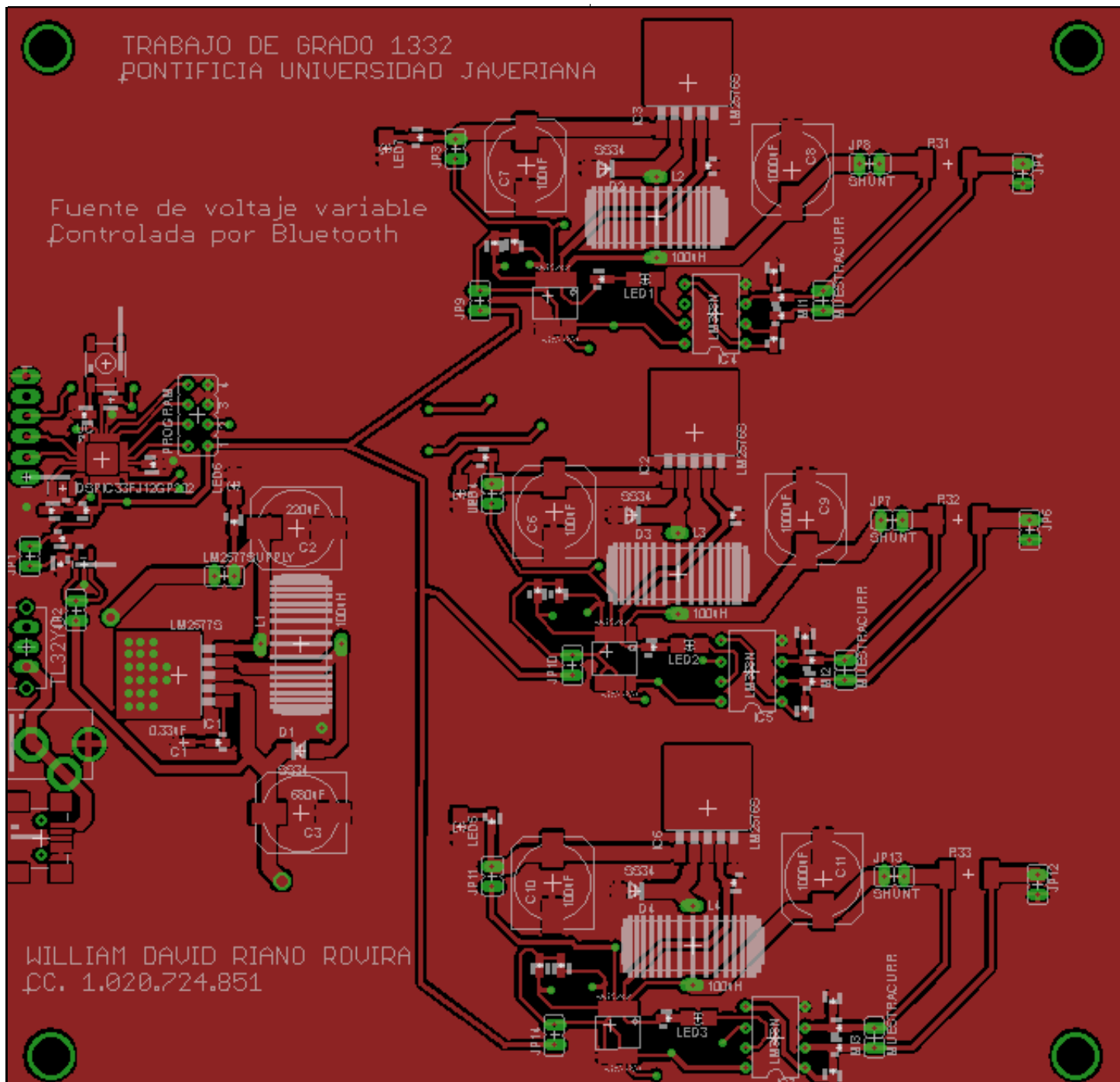


Figura 21. Gráfico para cálculo de ancho de caminos [12]

Analizando el gráfico se observa que manejando una corriente máxima de 3 [A], permitiéndole una elevación de temperatura de 10 grados centígrados y ya que el fabricante ofrece 2 Oz/ft² se necesitaría un ancho de pista de 0,07 pulgadas lo que es equivalente a 0,2 [cm].

William David Riaño

El camino más ancho que se implemento es por el que pasa más corriente, el cual viene desde la alimentación general Vin hasta Vout llegando a la salida de cada puerto, los siguientes archivos gerber muestran el diseño final de BSE (figura 22) el cual se mando a producción y se montó:



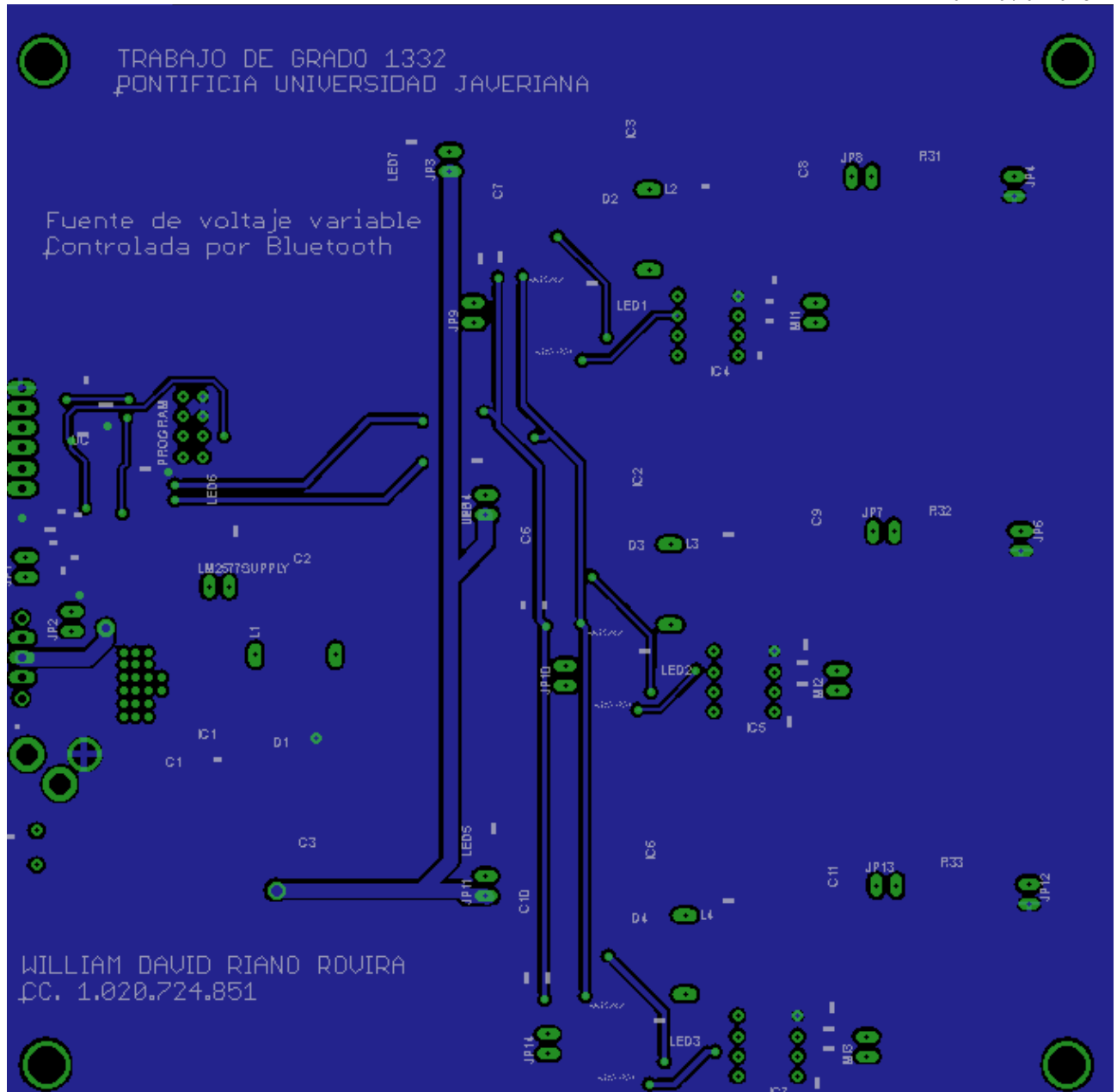


Figura 22. PCB final

La siguiente imagen (Figura 23) muestra el circuito obtenido después de su producción, en el cual se pueden observar la ubicación de los componentes

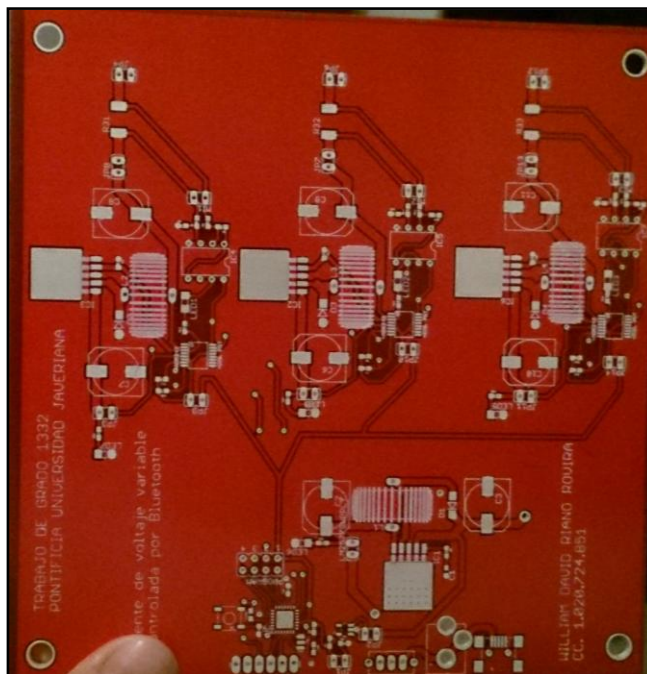


Figura 23. Circuito final para montaje

5.2.5 Criterio de ubicación de componentes PCB

Para este diseño se tuvieron en cuenta los siguientes criterios al momento de la ubicación de los componentes:

- La ubicación del micro controlador tenía que estar cerca a un borde para poder tener acceso a él y poder instalar el programa implementado
- Puerto de carga USB, Jack 1.5 mm, ya que estos puertos son la alimentación principal de la fuente también tienen que estar a un borde.
- Debido a que se implementó un módulo *bluetooth* y ya que este ocupa un tamaño de 4,4 x 1,6 cm este se ubica por encima del micro controlador aprovechando de esta manera el espacio y teniendo en cuenta que no dificulte el acceso a la programación del micro controlador.

Las tres fuentes *Buck* de cada salida se implementaron cerca del extremo de las conexiones de salida de la fuente manteniendo todos los componentes de cada fuente cerca y los potenciómetros digitales cerca de la realimentación (Divisor resistivo) para evitar caminos largos que afecten su funcionamiento.

5.2.6 Programación de micro controlador en PCB final

La programación se efectúa a través de un programador de referencia *Microstick II*, que está en capacidad de programar dsPICs y algunos PICs de la empresa Microchip®. Este sistema posee un socket para poder ubicar los dispositivos con empaque “DIP”, del cual se tomaron las cinco líneas necesarias para la programación:

- El *TARGET_DATA* es el pin a través del cual ingresa el código convertido en un tren de pulsos seriales par ser almacenado en la memoria flash del dsPIC.
- El *TARGET_CLOCK* establece el “timing” o reloj con el cual se programa el dispositivo.

William David Riaño

- *TARGET_MCLR* habilita el proceso de programación. Este pin borra la memoria del dispositivo poniéndose en bajo(lógica negada) y luego en alto durante un periodo que se realiza el proceso de programación.
- VCC y tierra (GND) polarizan el sistema durante el proceso de programación.

Para la programación en el circuito impreso se dejaron estos pines libres a los cuales se les pone un conector y contando con el Microstick II se sacan las señales correspondientes como se observa en la imagen

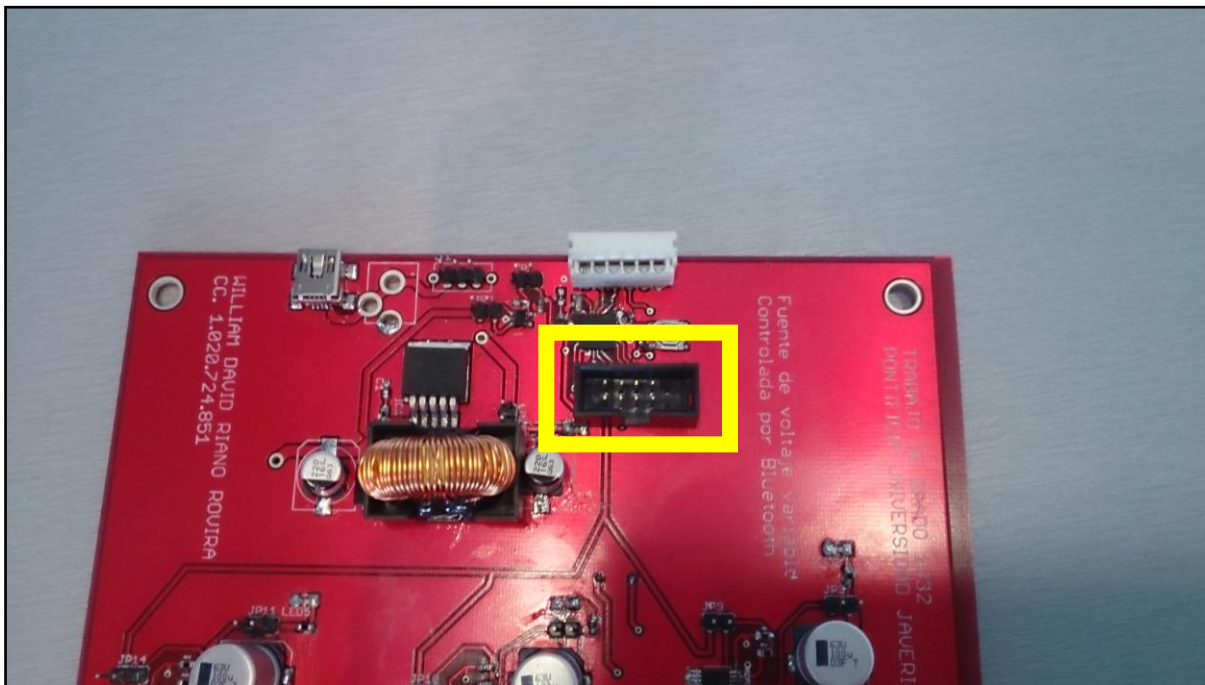


Figura 24. Programación de micro controlador final

La programación del micro controlador se maneja en el lenguaje C a través del programa MPLAB, este cuenta con las librerías de microchip para que reconozca el micro controlador implementado.

5.2.7 Tabla de pines de cada componente

Las siguientes tablas muestran los pines usados por cada componente y su función correspondiente

- AD5282 Potenciómetro digital

La siguiente figura 25 muestra la distribución de pines del AD5282 el cual viene en un empaque TSSOP-16

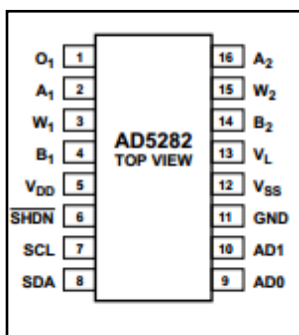


Figura 25. Distribución de pines AD5282[8]

Los pines se implementaron de la siguiente manera Tabla 7:

PIN	Función	PIN	Función
1	O1: Salida lógica programable (No en uso)	9	AD0: Conexión lógica de dirección de dispositivo
2	A1: Terminal de Resistencia programable	10	AD1: Conexión lógica de dirección de dispositivo
3	W1: Divisor de Resistencia entre pin A1 y B1	11	GND: Tierra común
4	B1: Terminal B de Resistencia programable	12	VSS: Polarización negativa entre -2.7 y 0 [v]
5	VDD: Fuente 5[v]	13	VL: Voltaje lógico
6	SHDN: Activo en Bajo, Conexión asincrona entre W y B y circuito abierto de A (Conectado a VDD)	14	B2: Terminal B de resistencia programable
7	SCL: Serial Clock	15	W2: Divisor de Resistencia entre pin A2 y B2
8	SDA: Serial Data	16	A2: Terminal de Resistencia programable

Tabla 7. Distribución de pines AD5282

Ya que se desea usar el potenciómetro digital como una resistencia simple el terminal A se deja como no conectado y la variación se da entre los pines W y B.

- LM2577 Boost

Este es el empaque usado en la implementación de la fuente Boost en un empaque TO220

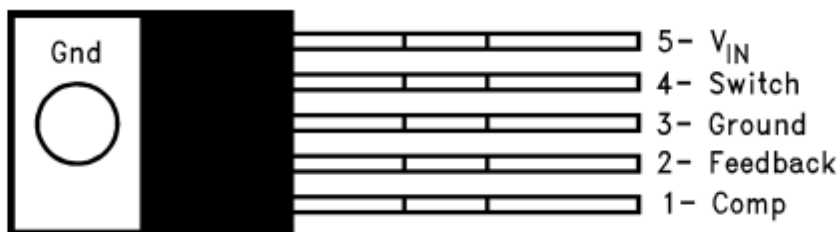


Figura 26. Distribución de pines LM2577 [20]

La siguiente tabla 8 hace una descripción de cada uno de los pines

PIN	Descripción
1	Comp: Red de compensación que estabiliza el regulador
2	Feedback: es la realimentación del regulador
3	Ground: Tierra común
4	Switch: el switcheo a una frecuencia de 52 khz
5	Voltaje de entrada

Tabla 8. Distribución de pines LM2577

- LM2576 Buck

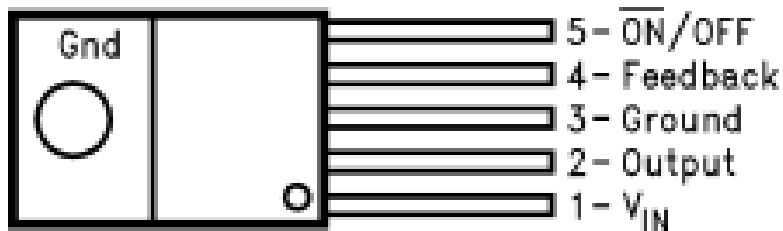


Figura 27. Distribución de pines LM2576 [11]

Al igual que la fuente *Boost*, este componente se monta en un empaque TO220, a continuación una descripción de cada uno de los pines del componente

PIN	Descripción
1	Voltaje de entrada
2	Voltaje de salida del regulador
3	Ground: Tierra común
4	Feedback: es la realimentación del regulador
5	On / off

Tabla 9. Distribución de pines LM2576

- Micro controlador DSPic33FJ128MC802

La siguiente imagen (Figura 28) es la vista física del micro controlador con su correspondientes pines

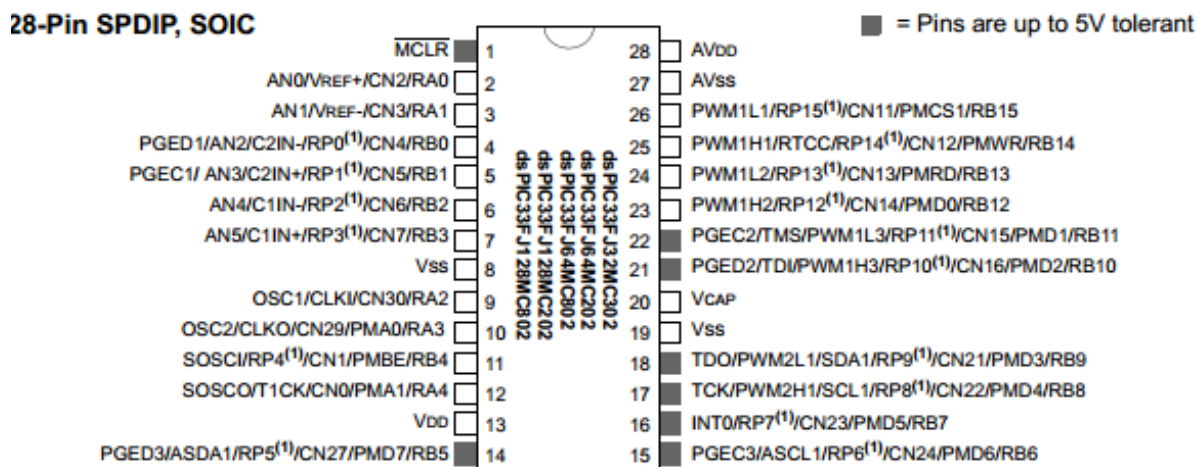


Figura 28. Distribución de pines DSPic33FJ128MC802[13]

PIN	Descripción
6	UART RX
7	UART TX
8	GND
9,10	Oscilador
13	VDD
17	SCL I2C
18	SDA I2C

Tabla 10. Distribución de pines DSPIC33FJ128MC802

6 Análisis de Resultados

En este ítem se analizarán los datos obtenidos en el desarrollo y los resultados finales.

6.1 Circuito Impreso

Como resultado del circuito impreso se tienen las siguientes principales características:

Característica	Valor
Peso	157 gramos
Dimensiones	14,5 cm x 14,3 cm x 3 cm

Tabla 11. Características circuito impreso

La siguiente imagen (Figura 29) muestra el peso obtenido del circuito final con los componentes montados.

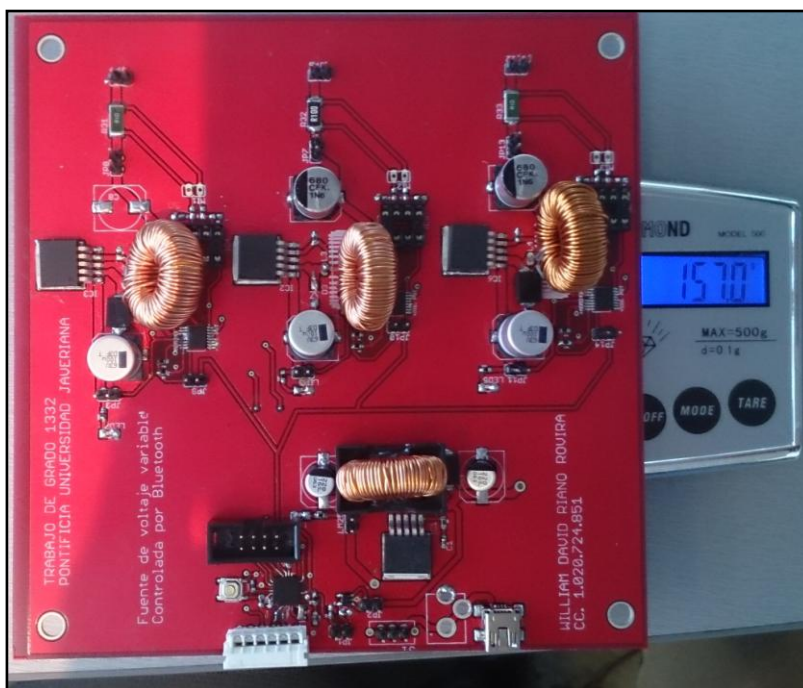


Figura 29. Peso obtenido del circuito final

En la siguiente imagen (Figura 30) se hace una descripción de cada componente usado y su ubicación

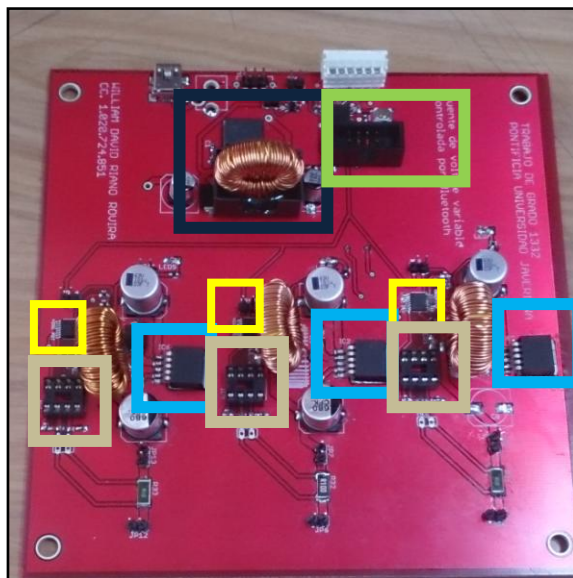


Figura 30. Montaje final de componentes

De esta imagen vemos los componentes implementados:

- En negro la fuente principal que se encarga de polarizar todo el sistema a 15 [v] *Boost*
- En azul las tres fuentes *Buck* de cada salida del sistema
- En amarillo los 3 potenciómetros digitales AD5242
- En pastel los 3 amplificadores operacionales LM358
- En verde el micro controlador DSPIC33GJ128MC802

6.2 Transmisión de datos por Bluetooth

Una de las ventajas con las que cuenta la plataforma es la capacidad de transmitir datos de forma inalámbrica por medio del Bluetooth, la interfaz gráfica cuenta con la opción de definir el puerto que se quiere usar al momento de transmitir por UART, en el caso del computador siendo COM5 transmisión por USB o COM40 transmisión por Bluetooth. En un principio se hicieron las pruebas de funcionamiento por USB con un FTDI el cual es un modulo de sparkfun que permite el envío y recepción de datos vía USB, el siguiente paso fue conectar el *bluetooth* y enviar la misma trama de datos y verificar la configuración. Como un paso adicional se envió datos por USB y se reenviaron por el *bluetooth* para tener una realimentación y gracias a un programa llamado Hyperterm el cual permite observar los datos recibidos a través de un puerto se observa lo siguiente (Figura 31):

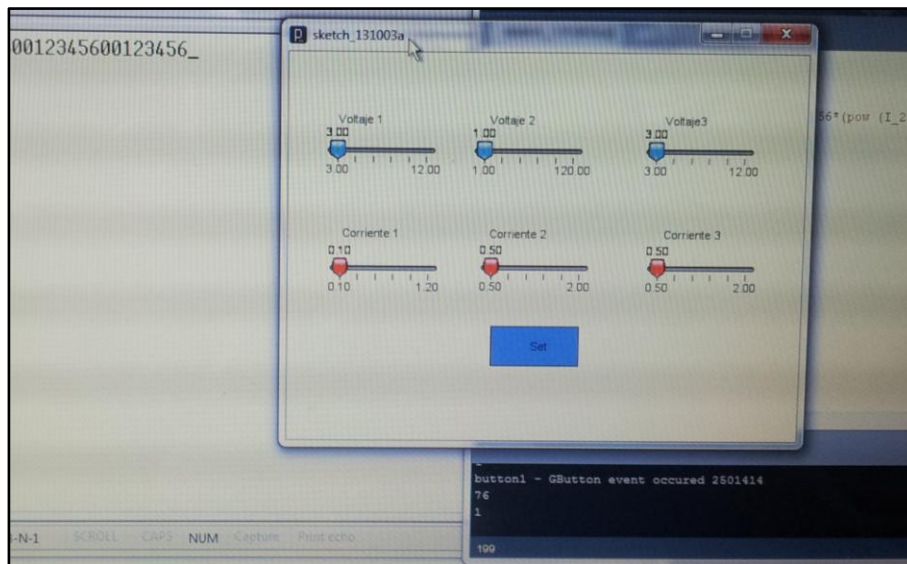


Figura 31. Hyperterm – Recepción de datos por UART

Como se puede observar se están transmitiendo los dos “0” de condición de inicio y 1,2,3,4,5,6 por UART, la realimentación muestra que los datos se envían de manera correcta.

6.3 Cálculo de Tiempos

Para el cálculo de tiempos es necesario implementar un código el cual recorra todo un ciclo de trabajo y de la información de cuanto demoró en este proceso, esto está fuera del alcance del proyecto, sin embargo, es posible conocer con los datos del tiempo requerido de envío de cada bit el tiempo total de operación, estos cálculos se hacen de la siguiente manera:

- UART

En el momento que el usuario establezca una configuración se van a enviar 8 datos en tramas de 8 bits es decir 64 bytes, y si se sabe que UART está transmitiendo a 9600 baudios entonces:

$$Tiempo\ de\ bit = \frac{1}{Baudrate}$$

$$Tiempo\ de\ bit = 104\mu s$$

Entonces para enviar 64 bytes correspondería a **6,6µs**

- I2C

Para la transmisión de I2C se conoce que el periodo de un solo bit es de 400KHz lo que corresponde a un tiempo de 2.5µs, este valor se multiplica por la cantidad de bits transmitidos lo que corresponde a:

- 8 bits de dirección de componente
- 8 bits de orden de lectura o escritura y pines lógicos
- 8 bits de data

Esto es para cada configuración, se están manejando 6 configuraciones las que corresponden a cada voltaje y cada limitación de corriente, entonces la cantidad total de bits es:

$$Cantidad\ total\ de\ bits = 24 * 6$$

$$\text{Cantidad total de bits} = 144$$

El tiempo requerido en enviar estos datos es: $144 * 2.5\mu\text{s} = 360\text{ms}$

Esto significa que el tiempo total que demora en efectuar todas las operaciones necesarias para configurar el dispositivo es de: 366.6 ms

6.4 *Protocolo UART*

Después de implementar los respectivos códigos del protocolo UART y programar con éxito el micro controlador, con la ayuda de un osciloscopio se observan las señales de transmisión por este puerto, la siguiente imagen muestra lo observado



Figura 32. Transmisión de UART

Se puede observar que los pulsos se están transmitiendo como se espera, la siguiente imagen (Figura 33) amplifica un pulso

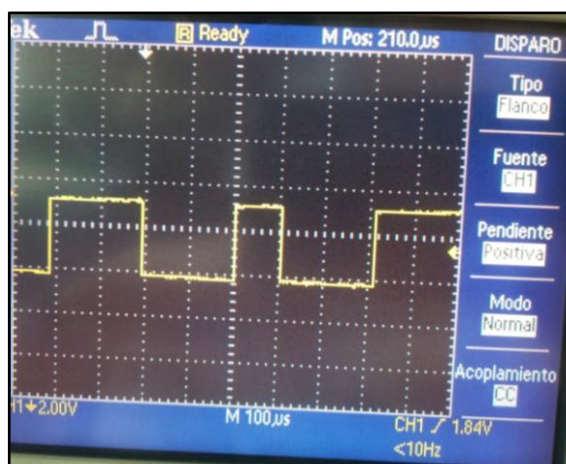


Figura 33. Tiempo de transmisión UART

De esta imagen se puede observar que se tiene un tiempo de 100 µs el cual se aproxima a los 104 µs calculados para el tiempo de transmisión de un bit.

6.5 Protocolo I2C

Con el uso de un osciloscopio se observa la señal SDA de I2C para poder observar los pulsos que se están enviando, la siguiente figura 34 muestra lo observado

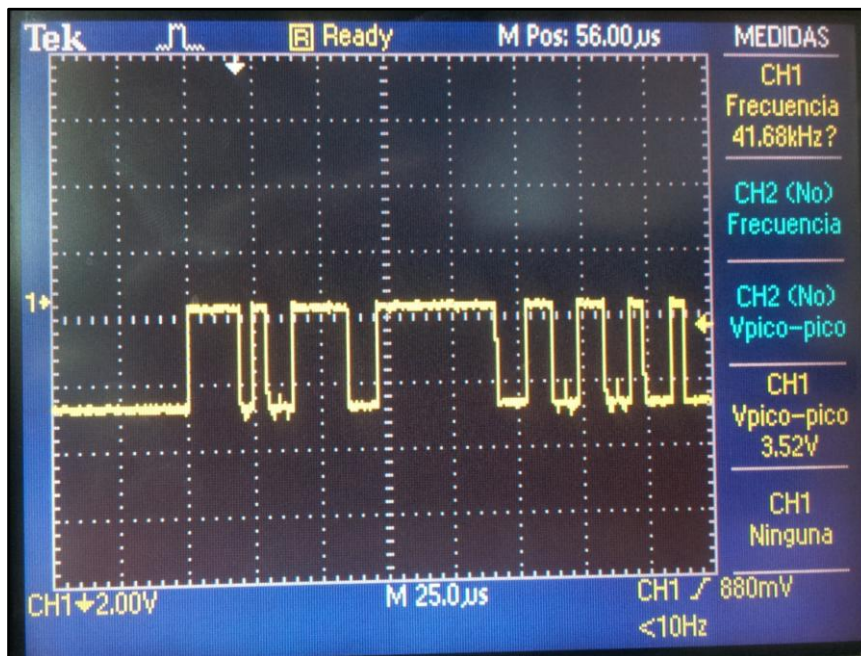


Figura 34. Señal I²C SDA

De esta imagen se puede concluir que la comunicación entre el micro controlador y los potenciómetros digitales es exitosa, operando con un voltaje lógico de 3.52 [v] el cual corresponde al 70% del voltaje de operación del micro controlador (5[v]).

6.6 Fuentes Buck-Boost

Regulación de carga

Una característica importante de la fuente es su caída de voltaje a medida que la carga cambia pidiendo más o menos corriente, las siguientes tablas nos muestran una configuración de voltaje a la salida y cuatro análisis con diferentes cargas para cada configuración.

- Configuración de salida a 3 [v]

Vout [v]	Rload [Ω]	Iout [A]	Vout_Rload [v]
3.0320	47	0.064	3.0224
3.0320	6.8	0.444	3.0237
3.0320	3.9	0.774	3.0293
3.0320	2.7	1.119	3.0355

Tabla 12. Regulación de carga a 3 [v] Vout

Graficando la corriente de salida vs el voltaje a la salida se tiene

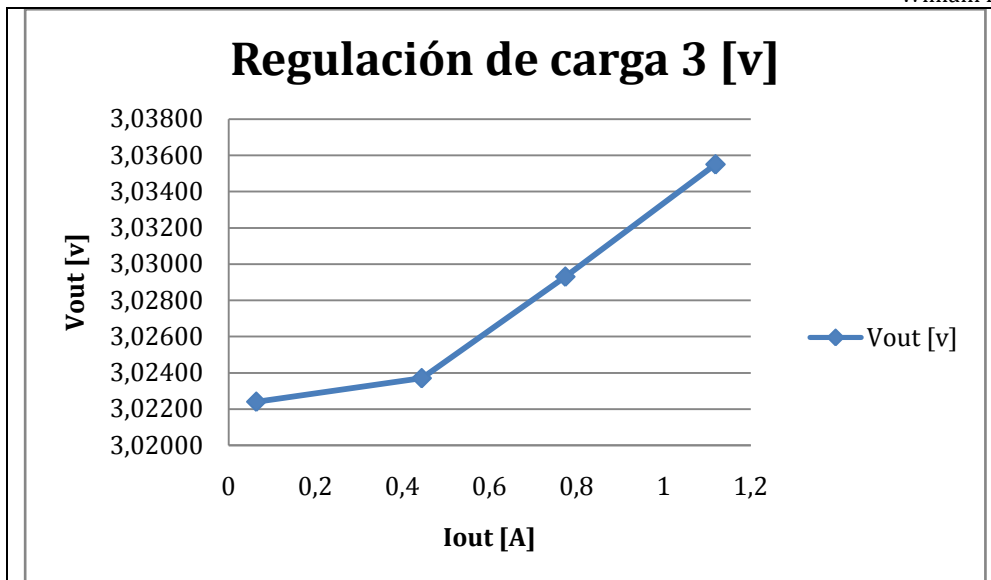


Figura 35. Gráfica regulación de carga con Vout 3 [v]

Analizando estos datos se puede concluir que la fuente *Buck* LM2576 presenta una regulación de carga del 0.4%.

- Configuración a la salida de 4,7 [v]

Vout [v]	Rload [Ω]	Iout [A]	Vout_Rload [v]
4.787	47	0.086	4.770
4.787	6.8	0.593	4.787
4.787	3.9	1.034	4.798
4.787	2.7	1.494	4.798

Tabla 13. Regulación de carga a 4,7 [v] Vout

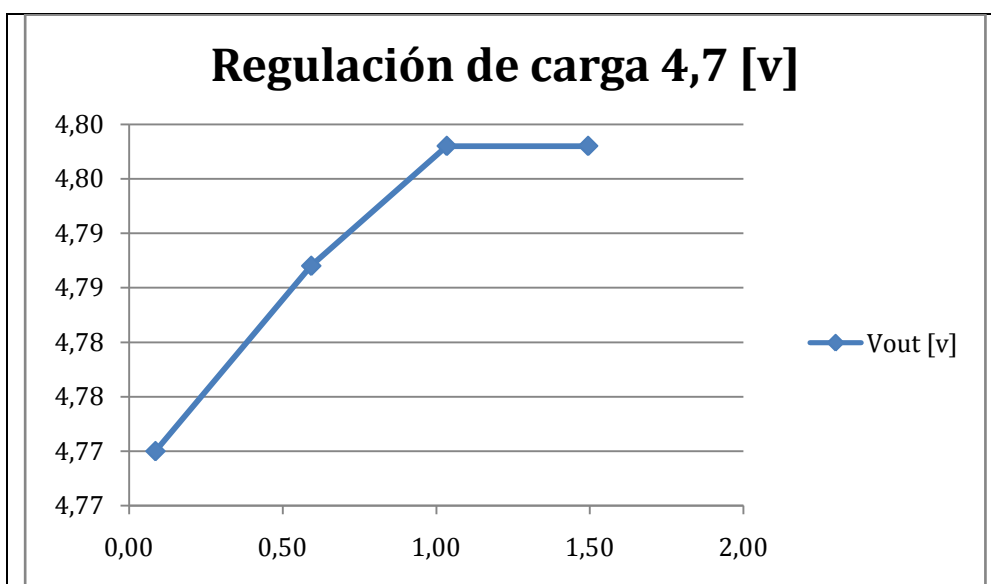


Figura 36. Gráfica regulación de carga con Vout 4,7 [v]

Para este caso con una configuración de 4,7 [v] a la salida se presenta un porcentaje de regulación de carga del 0,5%.

Como se observa a medida que se pide más corriente a la fuente Buck esta presenta una caída de voltaje a la salida, este es un valor importante a considerar al momento de usar la fuente para un consumo alto de corriente, ya que este podría brindar un voltaje menor al deseado.

Regulación de línea:

Vin [v]	Vout 4,7 [v]
15	4,786
10	4,784
8	4,782
6	4,776
5	4,761

Tabla 14. Regulación de línea con 4,7 [v] Vout

Para una configuración de 4,7 voltios se observa que la regulación de línea es del 0,5% significando una variación mínima a la salida independiente del voltaje de entrada.

Eficiencia de la fuente:

A continuación las siguientes tablas muestran el comportamiento de la fuente con diferentes cargas a la salida, midiendo parámetros de entrada vs parámetros de salida para hacer un análisis de su eficiencia:

TABLA 1								
Parmetros de entrada			Parametros de salida					Eficiencia [%]
V in [v]	I in [A]	Pin [w]	Vout Teorico [v]	Rload [Ω]	Vout Real [v]	Iout [A]	Pout [w]	Eficiencia [%]
5	0.290	1.45	3	0		0.000	0	0.0%
5	0.370	1.85	3	38	3.06	0.080	0.2448	13.2%
5	0.400	2	3	27	3.06	0.111	0.33966	17.0%
5	0.540	2.7	3	12	3.05	0.250	0.7625	28.2%
5	1.130	5.65	3	3.9	3.06	0.714	2.18484	38.7%

TABLA 2								
Parmetros de entrada			Parametros de salida					Eficiencia [%]
V in [v]	I in [A]	Pin [w]	Vout Teorico [v]	Rload [Ω]	Vout Real [v]	Iout [A]	Pout [w]	Eficiencia [%]
5	0.290	1.45	4	0	4	0.000	0	0.0%
5	0.410	2.05	4	38	3.99	0.103	0.41097	20.0%
5	0.460	2.3	4	27	3.97	0.144	0.57168	24.9%
5	0.700	3.5	4	12	3.97	0.326	1.29422	37.0%
5	2.600	13	4	3.9	3.98	0.924	3.67752	28.3%

TABLA 3								
Parmetros de entrada			Parametros de salida					Eficiencia [%]
V in [v]	I in [A]	Pin [w]	Vout Teorico [v]	Rload [Ω]	Vout Real [v]	Iout [A]	Pout [w]	Eficiencia [%]
12	0.185	2.22	12.1	0	12.1	0.000	0	0.0%
12	0.520	6.24	12.1	38	12.00	0.315	3.78	60.6%
12	0.660	7.92	12.1	27	12.00	0.440	5.28	66.7%
12	1.300	15.6	12.1	12	12.00	1.000	12	76.9%
12	1.900	22.8	12.1	8.3	12.00	1.417	17.004	74.6%

Tabla 15. Eficiencia de la fuente con su regulación

Como se puede apreciar en las tablas anteriores, se obtiene una eficiencia en el mejor caso del 76.9%, esto se debe al consumo del micro controlador, modulo *bluetooth* y el consumo de cada fuente.

Es importante resaltar que al tener dos fuentes en serie (una boost que eleva a 15 [v] y una buck para cada salida) la eficiencia en el mejor caso teorica va a ser de 65%, esto sin tener en cuenta el consumo de cada componente adicional como lo son bluetooth, potenciómetros digitales, amplificadores operacionales y modulo bluetooth.

6.7 Diseño de caja

Cuando se tienen las tres fuentes funcionando se diseña una caja para darle presentación y estilo al proyecto, esta caja se diseña en Corel Draw [22] y se contrata un servicio de corte laser de acrílico, como resultado se obtienen las caras de una caja las cuales se arman, se ensamblan y se acoplan al circuito obteniendo lo siguiente:



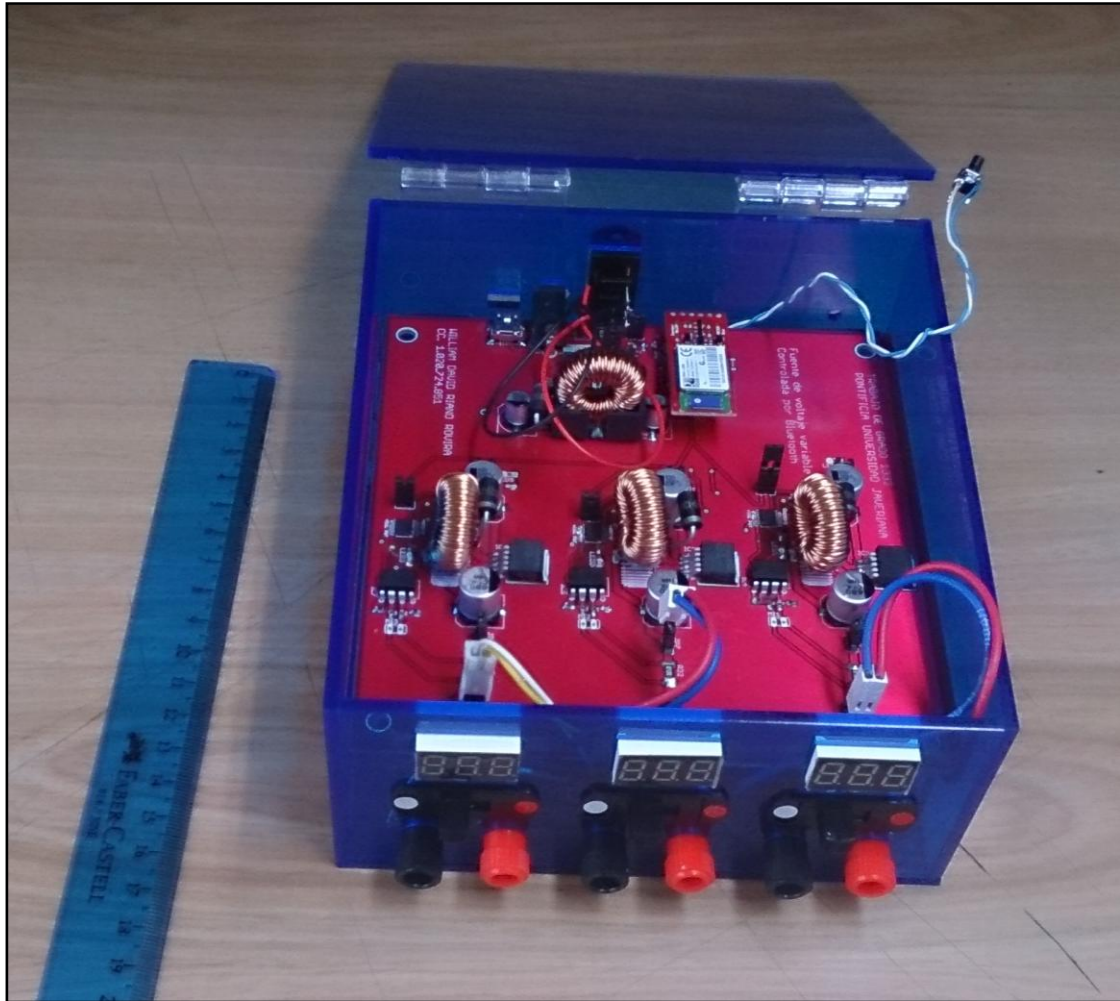


Figura 37. Vista general diseño de caja

6.8 Costos

Los costos más importantes y más significativos son la fabricación del impreso y el montaje de los componentes, para la fabricación del circuito impreso se contó con la empresa Microcircuitos ubicada en Cali la cual es bastante conocida por tanto estudiantes como ingenieros por su alta calidad y su excelente servicio, para el montaje se solicita ayuda para el micro controlador el cual es un empaque difícil de soldar y el resto se suelda con mucho cuidado.

La mayoría de los componentes fueron importados de Estados Unidos, el modulo bluetooth y FTDI fueron comprados en (www.sparkfun.com), el oscilador, resistencias, condensadores, fuentes buck-boost, amplificador LM358, potenciómetro digital AD5282, Microstick II starter kit, inductancias y conectores en Digikey (www.digikey.com), la siguiente tabla resume los componentes y los valores de cada uno

COMPONENTE	VALOR (USD)
Programador	30
Dspic33	3.73
Rn-42	16
Conectores	1.5
Resistencias y condensadores	5
Manufactura y Montaje	150
Envíos	100
Oscilador	1.3
Fuente Buck-boost	50
AD5282	3
LM358	9
Caja	50
TOTAL 1 EQUIPO	419.53

Tabla 15. Costos de producción

El costo total optimizado de la producción de la fuente BSE 419.53

7 Conclusiones

- Es de suma importancia tener claro los requerimientos de los componentes que van a ser controlados por algún protocolo de comunicación, por más que se use el mismo protocolo en componentes diferentes las condiciones pueden cambiar, una vez se definen estas condiciones se configura el protocolo, esto con la ayuda del datasheet del micro controlador. Es necesario observar esta transmisión o corroborarla con algún programa para poder ir asegurando un correcto funcionamiento.
- Una vez se implementa el sensado de corriente, se observa que la regulación de la fuente se está haciendo en un punto anterior al deseado, por este motivo es necesario hacer un muestreo remoto y establecer la regulación al conector de salida de cada fuente.
- La eficiencia máxima de la fuente es del 76.9% la cual se presenta cuando el voltaje de entrada de la fuente es el voltaje máximo al igual que la configuración de salida siendo V_{in} 12[v] y V_{out} 12[v] con una carga de 12[Ω].

8 Bibliografía

- [1] M. Barr and P. O. Reilly, *Programming Embedded Systems in C and C++*, 1999, pp. 1–187.
- [2] P. D. A. P. A and D. A. P. A, “Sistemas Embebidos y Sistemas Operativos Embebidos Sistemas Embebidos y Sistemas Operativos Embebidos,” 2009.
- [3] M. “What is a M. P. 1999. Gilliland, “What is a Microcontroller,” “*What is a Microcontroller*”. Parallax., 1999.
- [4] Telos, “I2C-Bus: What’s that? , Telos, Available FTP <http://www.i2c-bus.org/>.”
- [5] “Microchip.”, Microchip Technology Inc. All rights reserved, 1998-2014 [Online]. Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/en026583.pdf>.
- [6] M. Brown, *Practical switching power supply design*. San Diego; New York; Boston, 1990, pp. 1–256.
- [7] “Guía de Diseño y Análisis de Convertidores”, 2008, Universidad de Costa Rica, José Luis Gamboa Quesada, Available FTP: <http://eie.ucr.ac.cr/uploads/file/proybach/pb0820t.pdf>.
- [8] Analog Devices, I²C Compatible devices 256 positions digital potentiometer, AD5282, disponible en: http://www.analog.com/static/imported-files/data_sheets/AD5280_5282.pdf.
- [9] Peter Lager, “GUI Builder Tool”, 2013, Available [online]: <http://www.lagers.org.uk/g4ptool/index.html>
- [10] On Semiconductors, Fixed/Adjustable Current- Limiting Power- Distribution Switches, “NCP380”, Available [online]: http://www.onsemi.com/pub_link/Collateral/NCP380-D.PDF
- [11] Texas Instruments simple switcher step down 3 [A], DC/DC Converter, “LM2576”, Available [online]: <http://www.ti.com/lit/ds/symlink/lm2576.pdf>
- [12] Minco’s General Capabilities, Conductor width nomograph, Available [online]: http://sccontent.minco.com/sitecore/shell/Controls/Rich%20Text%20Editor/~/_media/36BEAAE50B4B4A2F84F79CEFB97A9979.ashx
- [13] DSPIC33fj128mc Microcontroller [online]: <http://ww1.microchip.com/downloads/en/DeviceDoc/80443G.pdf>
- [14] Practical switching power supply design, Marty Brown, Motorola, 1990.
- [15] K.Matsui ,I.Yamamoto ,T.Kishi ,M.Hasegawa, H.Mori and EUeda, IEEE, A Comparison of Various Buck-boost Converters and Their Application to PFC.
- [16] Huang Wentian, Li Jinping, IEEE, Design of Buck DC-DC Power Supply Based on TPS54310.
- [17] Understanding Buck-Boost Power Stages in Switch Mode Power Supplies, Texas Instruments, March 1999 – Revised November 2002, Application Report.

- [18] E. Rogers, “Understanding Buck-Boost Power Stages in Switch Mode Power Supplies,” *Power*, vol. 18, no. November, pp. 1–32, 2002.
- [19] B. Sahu, S. Member, G. A. Rincón-mora, and S. Member, “Supply IC for Portable Applications,” pp. 18–21, 2005.
- [20] Texas Instruments simple switcher step up, voltage regulator, DC/DC Converter, “LM2576”, Available [online]: <http://www.ti.com/lit/ds/symlink/lm1577.pdf>
- [21] Processing org, 2001, Ben Fry, Casey Reas Available FTP <http://www.processing.org/>
- [22] Corel Draw, 2014, Corel Corporation, Available FTP <http://www.coreldraw.com/us/product/graphic-design-software/>

ANEXOS

Esquemático DsPic33FJ128MC802

