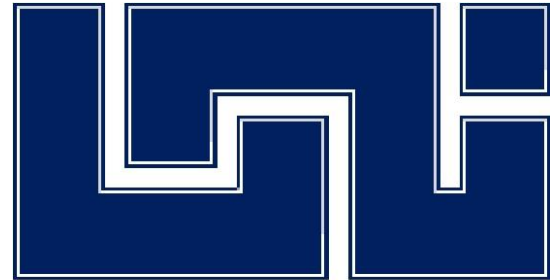




SEMAFORO INTELIGENTE CON CONEXIÓN BLUETOOTH

[Uso Permitido para oficiales de tránsito]



Líder en Ciencia y Tecnología

UNIVERSIDAD NACIONAL DE
INGENIERIA

Electrónica Aplicada

Profesor: Ing. Oscar Pérez

Integrantes

- Keller Jirón Ruiz
- Yaser Alfredo Morales

Índice

Introducción.....	2
Reseña Histórica.....	3
Descripción del problema.....	4
Criterios de desempeño.....	5
Alcances.....	5
Análisis de descripción del problema.....	6
Objetivos Generales y específicos.....	6
Especificación de la solución.....	7
Diagrama de bloques.....	9
Especificación de componentes.....	10
Esquemático Final.....	15
Diseño del software.....	16
Aplicación Celular.....	17
Funcionamiento.....	19
Implementación.....	20
Conclusión.....	21
Recomendaciones.....	22
Bibliografía.....	23
Anexos.....	24

Introducción

En nuestro país se presenta un problema en las calles que muchos conductores quisieran evitar, este es el tráfico cogestionado que se presenta principalmente en los cruces de las calles más transitadas y muchas veces el problema se da porque una de las vías esta con más afluencia vehicular que la otra y el semáforo no posee la capacidad de dar pase a la vía más transitada. La policía de tránsito buscando resolver el congestionamiento vehicular ha decidido designar oficiales de tránsito que controle el paso vehicular.

El policía de tránsito al realizar su labor se expone al sol, lluvia e incluso el riesgo de un accidente por un conductor desprevenido por esta razón se decidió tratar de mejorar o acabar con este problema y evitar esas horas perdidas en el tráfico de una manera más inteligente sin someter al oficial de tránsito a peligros y al conductor garantizarle la seguridad que este desea al conducir mejorando el estilo de cruce en la vía ahorrándonos más tiempo el cual se podrá ocupar de manera más productiva

En base a estos ciertos problemas que se encontraron en las intersecciones de calles muy transitadas se decidió presentar esta solución la cual consiste en un “semáforo Automático inteligente con comunicación bluetooth”,

Reseña Histórica.

El 9 de diciembre de 1868 se instaló el primer semáforo en Londres. Fue diseñado por el ingeniero ferroviario John Peake Knight, quién se basó en las señales ferroviarias de la época. El primer semáforo tenía dos brazos que se levantaban para indicar el sentido que tenía que detenerse, además usó lámparas de gas de colores rojo y verde para su uso nocturno. Sin embargo, este primer semáforo era manual por lo que requería que un policía lo controlase todo el tiempo.



Figura1. Técnico instala un semáforo en San diego EEUU, 1940

En 1910, Ernest Sirrine mejoró el semáforo volviéndolo automático. El sistema de Sirrine usó las palabras no iluminadas «stop » (detenerse) y «proceed» (proceder) Aunque era manual, la innovación fue usar luces eléctricas y un zumbador para advertir del cambio de estado. Además permitía a las estaciones de policía y bomberos cambiar el estado del semáforo en caso de emergencias.

Hoy los semáforos son de muy bajo consumo y por tanto ahorran energía, mayor vida útil de las lámparas incandescentes, mínimo mantenimiento, respeto por el medio ambiente, unidad óptica con protección UV y alto contraste durante el día, proyección luminosa uniforme, se evita el fundido total del panel luminoso, de forma que el semáforo nunca se apagará totalmente, mayor seguridad vial, se pueden cambiar la imágenes fácilmente, su bajo consumo permite que funcionen autónómicamente mediante una batería durante cierto tiempo, precaución a los peatones, incorporación de sonidos intermitentes cuando el muñeco verde esté parpadeando para ponerse en rojo. Actualmente se utiliza una voz grabada con el nombre de la calle para que un peatón ciego no se pueda confundir con otros semáforos cercanos, como sucedía con los primeros semáforos con este sistema con trino de canarios.

Descripción del Problema.

Muchas veces nosotros como conductores quisiéramos llegar de forma más rápida a nuestro lugar de destino pero esto se nos dificulta por el tráfico excesivo que hay en las carreteras de Managua principalmente en las “horas pico” y esto se debe a que en los cruces hay una irregularidad de paso ya que los semáforos al ser automáticos estos siguen una secuencia y muchas veces hay vías que no tienen presencia vehicular pero el semáforo sigue en verde en vez de cambiar a otra vía en la que se encuentren más vehículos y de esa forma acelerar el paso inteligentemente.



Figura2. Transito Managua Rotonda Jean Paul Genie

En los últimos meses se ha observado que la policía Nacional ha intentado mejorar este problema de forma que en cada cruce de las calles de Managua hay presencia de Policías De Tránsito que están dándole paso a diferentes vías para acelerar el paso debido a la constante crecimiento de vehículos en las calles capitalinas, pero a cómo podemos observar estos policías deben de estar a la mitad de la calle para ejercer su labor lo cual se les dificulta principalmente por la hora en la que están trabajando ya que el sol está muy fuerte y les puede provocar algún tipo de desgaste físico, a su vez se presenta el problema de que los conductores al ver el semáforo y no lograr ver al policía de tránsito quieren seguir la señal del semáforo por lo que podrían provocar algún tipo de accidente de tránsito.



Figura3. Oficial de Tránsito.

Por estas razones se planteó este problema para poder darle una solución que beneficie tanto a los conductores como a los agentes de tránsito que están ejerciendo su labor ahorrándole tanto tiempo como trabajo.

“Managua Mejor” se denomina el plan de tránsito que la Policía Nacional y la Alcaldía de Managua están implementando para mejorar la circulación vehicular durante las horas pico (entre las 6:30 y las 8:30 a.m., y las 4:00 y 7:00 p.m.) en las principales vías de la capital.

Figura4. Screenshot Noticia El Nuevo Diario (09 de junio de 2015)

Criterios de desempeño

Desarrollar un sistema confiable y robusto para la ciudadanía que cruza una intersección muy transitada permitiendo al policía tener el control del semáforo sin presentar riesgo alguno cuando este desee conectarse al sistema.

Alcance

- Se hará una selección de los componentes necesarios para cumplir con las especificaciones.
- Se diseñará y calculará el circuito electrónico que realice la función del objeto de este proyecto.
- Se programará el software del Microcontrolador y la aplicación Android.

- Se construirá un primer prototipo, probándolo en la tabla de nodos.
- Se calculará el presupuesto.
- Se confeccionará el manual de usuario.
- Dar seguimiento al proyecto.

Análisis de la descripción del problema

Llevar a cabo la solución de este problemática que representa el cruce de una intersección tiende a darle confianza al conductor donde este no tendrá señales erróneas o dobles solo seguirá la señal habitual permitiendo agilizar con seguridad el tráfico vehicular.

Objetivos

General

Implementar Un sistema Automático Electrónico que permita controlar el paso vehicular en los cruces de las carreteras vía bluetooth y ayudar al paso peatonal con personas no videntes.

Específicos

- El policía de tránsito pueda comunicarse con el sistema vía bluetooth para controlar el semáforo.
- Facilitarle a los conductores un tráfico más ligero para ahorrar tiempo al transitar en una intersección.
- Implementar el prototipo y darle seguimiento para corregir errores.

Especificación de la solución

El sistema que se presenta a continuación es una solución de prototipo que debe someterse a revisiones que verifiquen su confiabilidad en un cruce de vehículos ya que presentaría riesgos muy graves si este falla. Como estudiantes comprometidos con la innovación y la capacidad de dar repuestas a problemas que son detectados y representan la posibilidad de poner en práctica los conocimientos adquiridos como estudiantes de ingeniería electrónica proponemos este designio valiéndonos de los métodos de ejecución indispensables al abordar una problemática real.

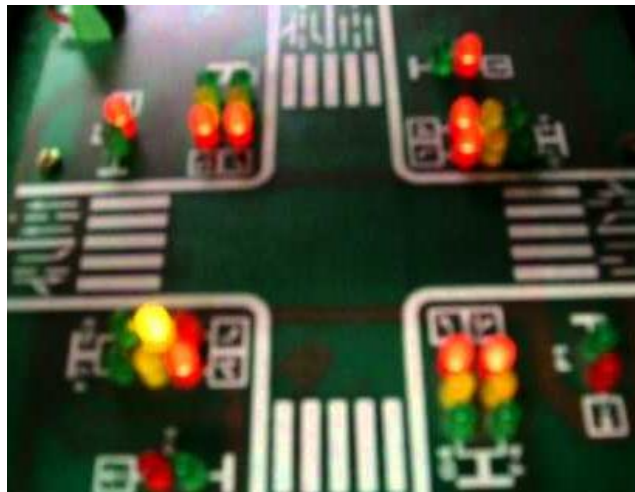


Figura5. Prototipo Semáforo

Un semáforo automático habitual hace transiciones de 30 segundos la luz verde, la amarilla por un periodo de 5 segundos la luz roja el tiempo necesario establecido es igual a la suma de la espera de las transiciones de las 3 calles restantes un aproximado de 105 segundos y así sucesivamente para las demás calles, un programador de esta intersección buscaría el balance en las cuatro vías y esto se ha mantenido a lo largo de años en la capital.

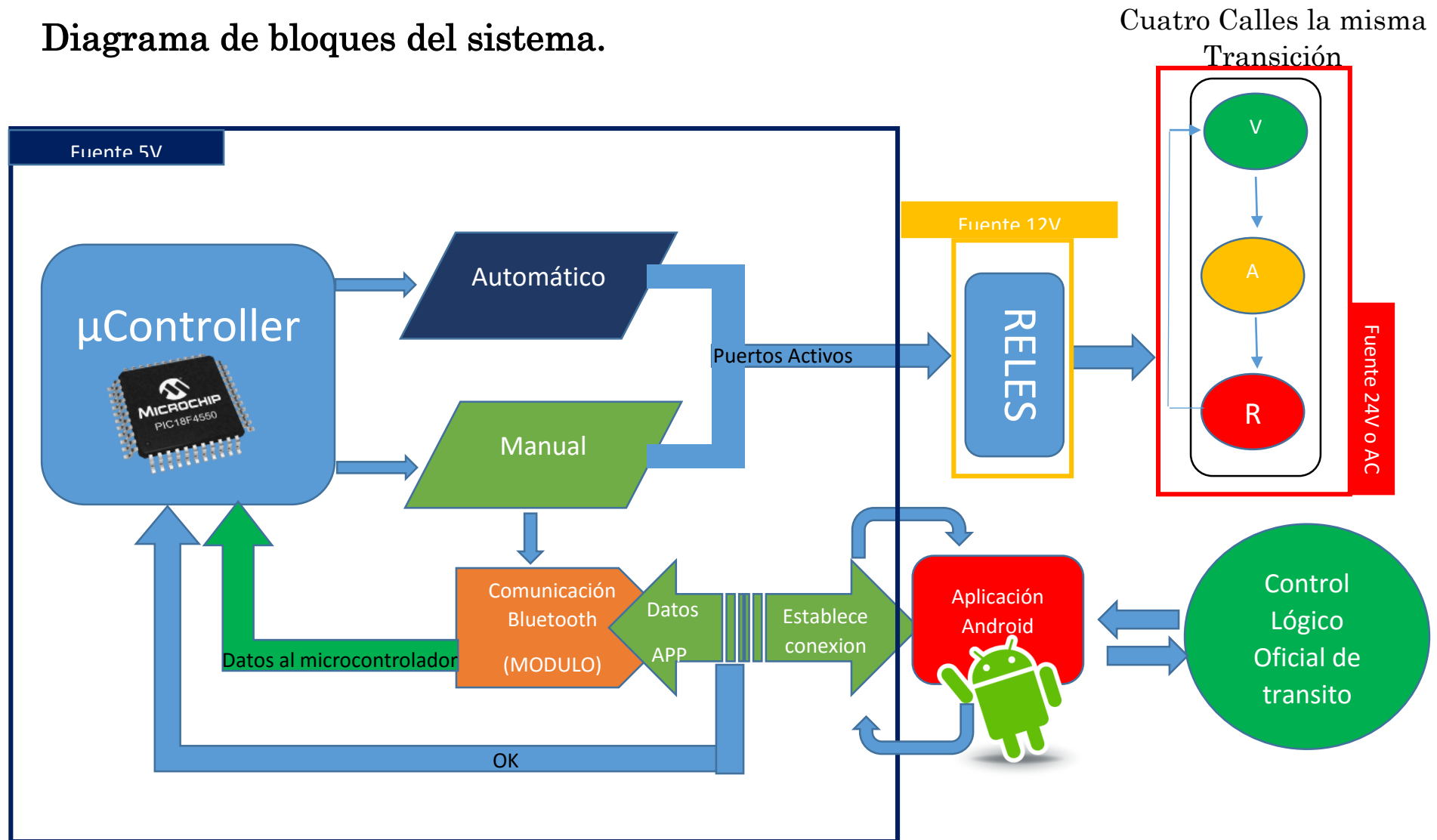
Dar prioridad valiéndonos del personal que ofrece la policía de tránsito para la regulación del tráfico es la piedra angular de esta solución. Se tomara los componentes pasivos del semáforo como las

luces señalizadores y se usara otra programación para el semáforo que cumplirá con más funciones eficientemente.

El uso de microcontroladores representa gran auge en la industria electrónica desde hace varios años permitiéndonos el ahorro de dinero y ofreciéndonos una amplia gama de dispositivos disponibles para el fin del desarrollador, el PIC18F4550 es el que se ha decidido elegir para esta solución posee una amplia capacidad de memoria para las líneas de programación y nos facilita varios protocolos de comunicación del cual el protocolo RS232 será usado para establecer una comunicación con un módulo Bluetooth y una terminal Android con una aplicación exclusiva para la policía de tránsito.

El policía designado a cubrir la intersección en las horas picos se establecerá a un rango de distancia no mayor a 35 metros que es el alcance máximo del módulo bluetooth ejecutara la aplicación, se le pedirá la contraseña de seguridad el sistema detectara que está conectado y esperara las instrucciones del policía permitiendo así designar que calle y el tiempo que esta estará permitido el pase de vehículos.

Diagrama de bloques del sistema.



Especificación de componentes

1. PIC18F550

Especificaciones básicas de la serie PIC18F:

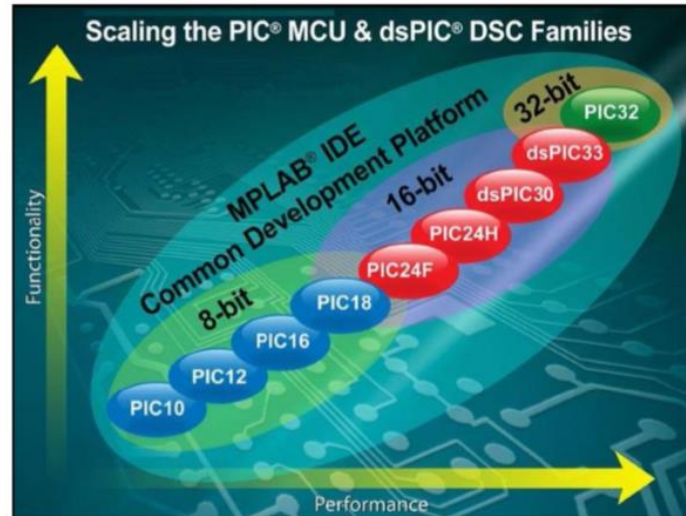


Figura6. Gama de los microcontroladores

Características

- 83 instrucciones.
- Direccionamiento de memoria de programa de hasta 2Mbytes.
- Direccionamiento de memoria de datos de hasta 4Kbytes.
- Velocidad de funcionamiento hasta 40MHz.
- Multiplicador por hardware de 8x8 bits.
- Dos niveles de prioridad de interrupción.
- Instrucciones de 16 bits.
- Bus de datos de 8 bits.
- Hasta 20 interrupciones externas.
- Módulo de puerto serie síncrono (modos SPI y I2C).
- Hasta dos módulos USART.
- Hasta 13 convertidores analógico-digitales.
- Programación In-Circuit.

El PIC18F4550 comparte las especificaciones básicas de la serie PIC18F descritas anteriormente y cuenta también, con una memoria flash de programa de 32Kb, una RAM de 2048 bytes bytes y una EEPROM de 256 bytes. En cuanto a puertos de E / S, tiene 5 (puerto A, B, C, D y E) sumando un total de 36 pines de entrada y salida en su versión de 40 pines

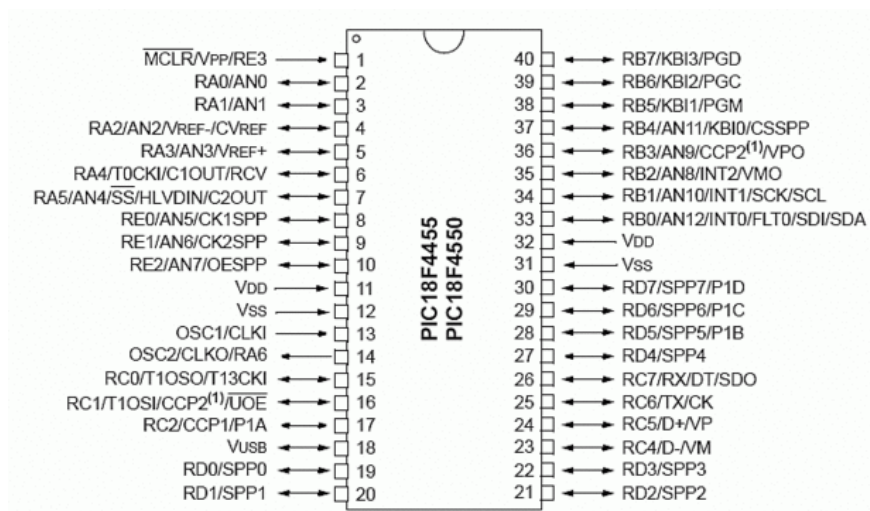


Figura7. Descripción de pines Microcontrolador.

1.1 Esquema de conexión y descripción de los componentes

EL μC necesita de varios componentes externos para su funcionamiento Independientemente de los que se quieran controlar a través de las entradas o salidas, estos componentes conforman un sistema oscilador externo y dos condensadores de bypass en los terminales de alimentación, mostrados en la figura 8.

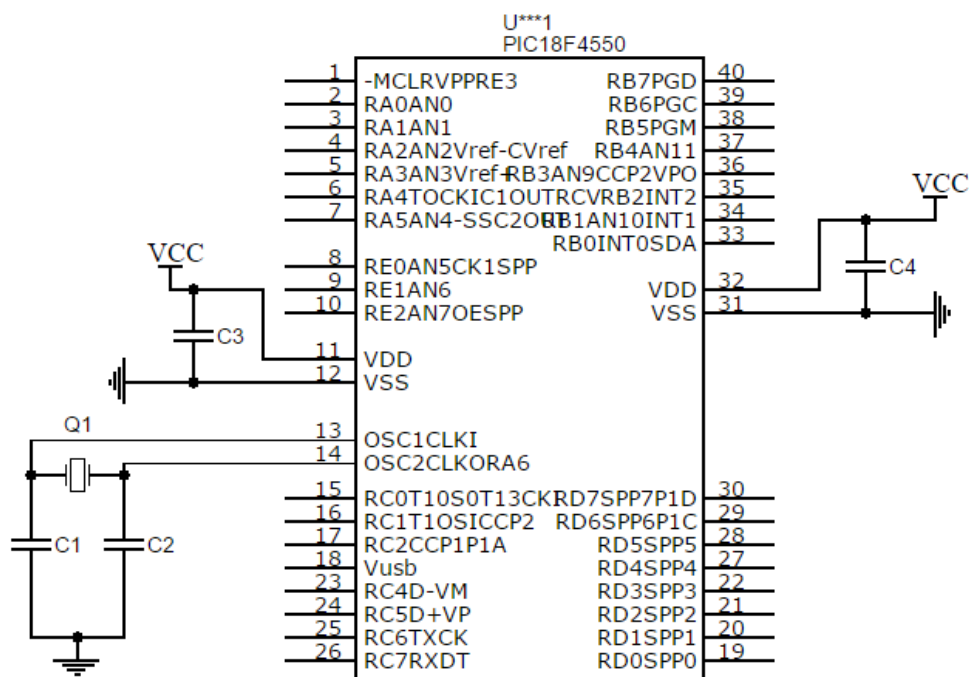


Figura8. Esquema de conexión para el correcto funcionamiento del PIC18F4550.

2. Modulo Bluetooth HC05

El módulo bluetooth HC-05 viene configurado de fábrica para trabajar como maestro o esclavo. En el modo maestro puede conectarse con otros módulos bluetooth, mientras que en el modo esclavo queda a la escucha peticiones de conexión. Con este módulo en un proyecto podrás controlar a distancia desde un celular o una laptop todas las funcionalidades que desees.

El modulo Bluetooth HC-05 utiliza el protocolo UART RS 232 serial. Es ideal para aplicaciones inalámbricas, fácil de implementar con PC, microcontrolador o módulos Arduinos.

La tarjeta incluye un adaptador con 6 pines de fácil acceso para uso en protoboard.

Los pines de la board correspondientes son:

- EN
- VCC
- GND
- TX
- RX
- STATE

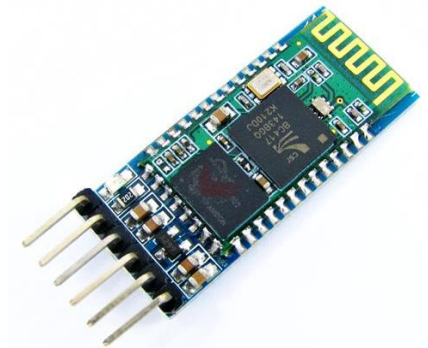


Figura9. Modulo bluetooth HC05

Características

- Compatible con el protocolo Bluetooth V2.0.
- Voltaje de alimentación: 3.3VDC – 6VDC.
- Voltaje de operación: 3.3VDC.
- Baud rate ajustable: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
- Baud rate por defecto: 9600

2.1 Conexión del módulo con el microcontrolador.

El microcontrolador posee dos puertos TXD para transmitir y RXD para recibir los datos via serial desde el Modulo el modulo HC05 cuenta con estos dos mismos puertos.

Para una correcta conexión el TXD del microcontrolador se conecta al RXD del Modulo y el RXD del microcontrolador al TXD del microcontrolador.

Los voltajes de nivel logico de la transmision son niveles TTL para ambos beneficiandonos en no acondicionar la señal, ya que este microcontrolador puede ser usado para establcer una comunicación serial con un computador y para este fin es necesario usar un acondicionador de señal el mas conocido en la industria es el MAX232 que convierte los niveles logicos TTL en Niveles apropiados para el computador.

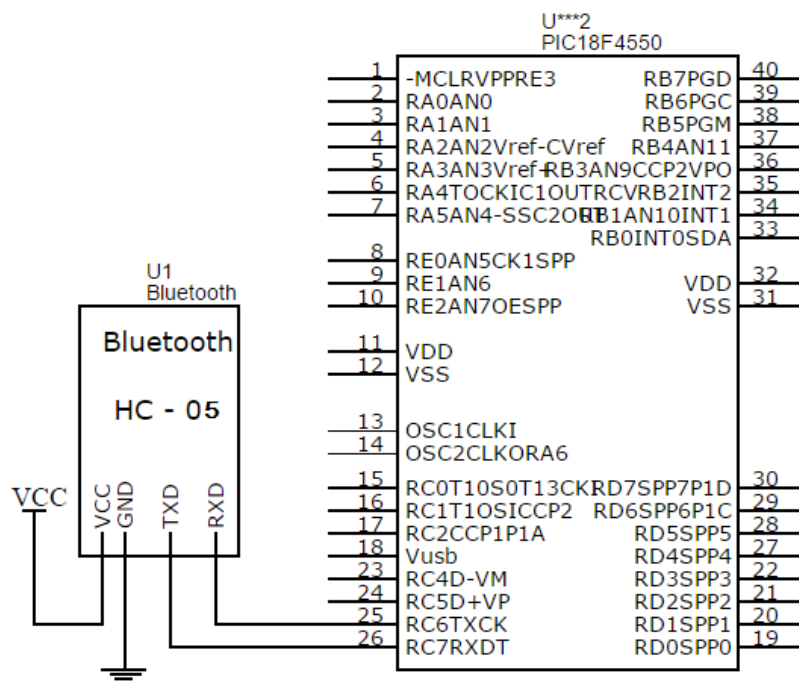


Figura10. Conexión Microcontrolador Y Modulo

3. Transistores NPN como Switch + Reles.

Para poder controlar los voltajes de niveles más altos usaremos la configuración del transistor como switch con la resistencia que nos permitirá delimitar el paso máximo de la corriente que necesita suplir el microcontrolador y el transistor para operar una vez excitada su base. Es importante conocer el valor de la Resistencia para no exigirle más corriente al microcontrolador ya que podría dañarse el puerto o incluso el microcontrolador.

Los cálculos para esta operación se basan en conocer los siguientes datos.

- V_{max} del microcontrolador = 5V
- $V_{BE} = 0.7V$
- I_{max} a controlar = $I_C = 500mA$
- β (Ganancia) = 50

$$R = \frac{V_{max} - V_{BE}}{I_{max} / \beta} \quad R = \frac{5v - 0.7v}{500mA / 50} = 430\Omega \approx 470\Omega$$

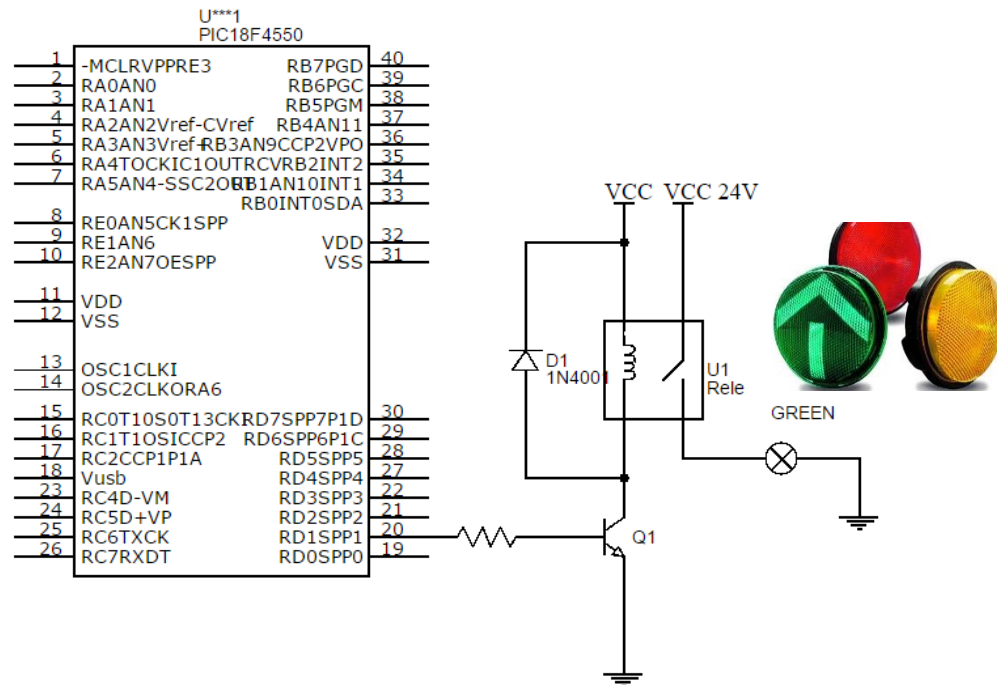


Figura11. Conexión Microcontrolador Y Etapa de potencia

El siguiente esquemático muestra la asignación del pin RD2 – 20 a la etapa de potencia donde se controla mediante este puerto la luz Verde del semáforo de la calle 1, esta misma configuración coicidira con los demás puertos.

Esquemático final

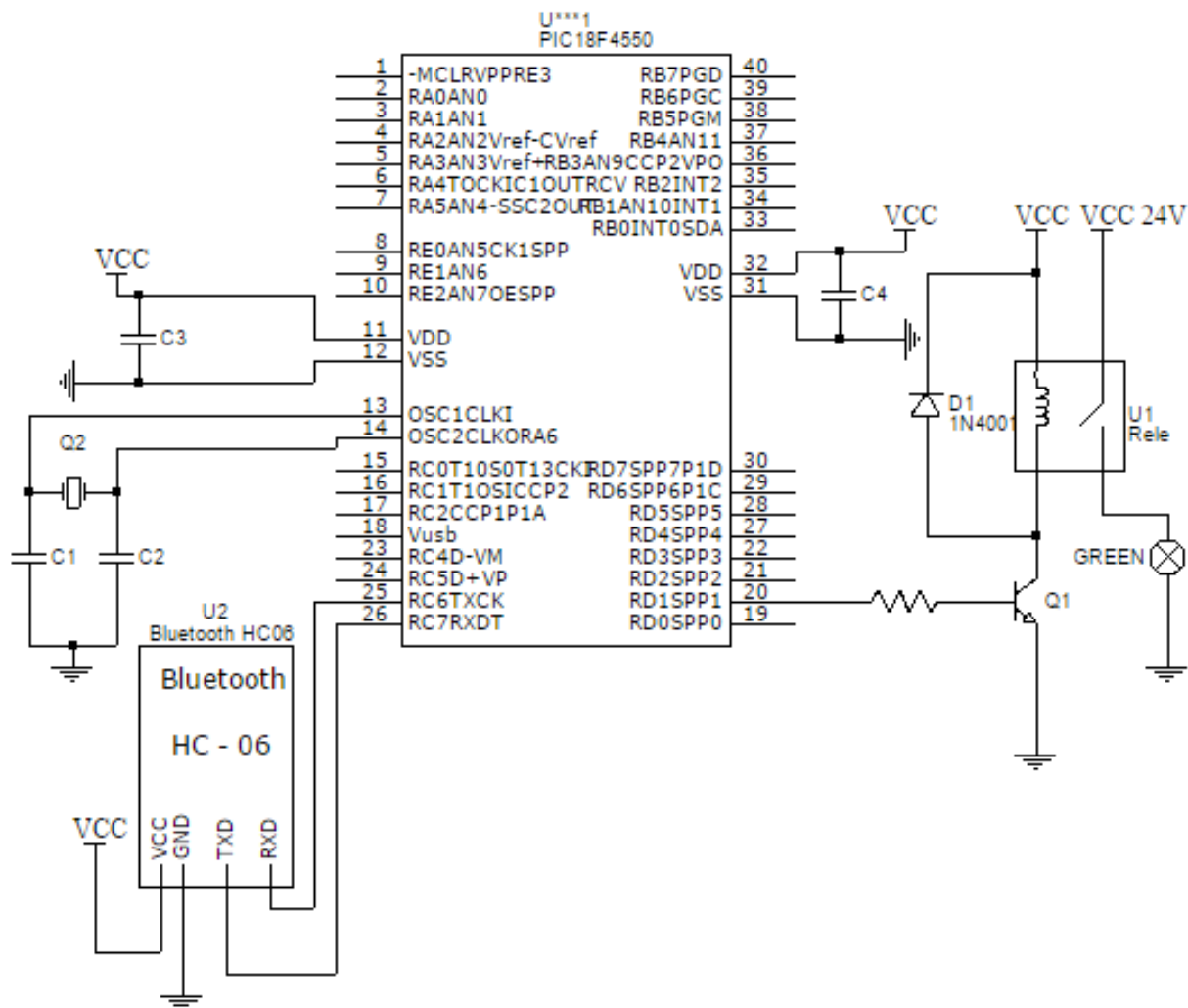


Figura12. Esquema conexión

Aca se muestra solo una configuración de Switch – Rele en total son 12 por motivos de visualización no aparecen.

Los pines excluidos son (21 – 22 – 27 – 28 – 29 – 34 – 35 – 36 – 37 – 38 – 39)

Diseño del software

El lenguaje de programación usado, es el lenguaje para programación de PIC's MikroC del desarrollador MikroElektronika. La mayor parte de las acciones son prácticamente iguales que en cualquier lenguaje de la familia aunque existen palabras clave específicas para la asignación los pines del PIC o para configurar los diferentes registros de funciones especiales tales registros y características de importancia para una excelente respuesta del microcontrolador las podemos consultar en la hoja de datos.

La líneas de código que componen el funcionamiento principal del programa está compuesto de máquina de Estado Finito que nos permite una respuesta del sistema más rápida permitiéndonos movernos entre los datos vinculados en la programación y sus condiciones

```

    .
    . struct State fsm_sema[] =
    .     {
    .         //LSB      MSB
    .         { 0x04, 0x01, 0x04, 0x04, 3000, {1}},
    .         { 0x04, 0x02, 0x04, 0x04, 800, {2}},
20    .         { 0x04, 0x04, 0x04, 0x04, 500, {3}},
    .         { 0x01, 0x04, 0x04, 0x04, 3000, {4}},
    .         { 0x02, 0x04, 0x04, 0x04, 800, {5}},
    .         { 0x04, 0x04, 0x04, 0x04, 500, {6}},
    .         { 0x04, 0x04, 0x01, 0x04, 3000, {7}},
    .         { 0x04, 0x04, 0x02, 0x04, 800, {8}},
    .         { 0x04, 0x04, 0x04, 0x04, 500, {9}},
    .         { 0x04, 0x04, 0x04, 0x01, 3000, {10}},
    .         { 0x04, 0x04, 0x04, 0x02, 800, {11}},
    .         { 0x04, 0x04, 0x04, 0x04, 500, {0}},
30    .     };
    .

```

Figura13. Líneas de código Máquina de estado finito.

Debido a que estas líneas de código están en la memoria temporal del circuito esta consume más RAM y es por tal razón que se usó el pic18f4550.

Aplicación Celular

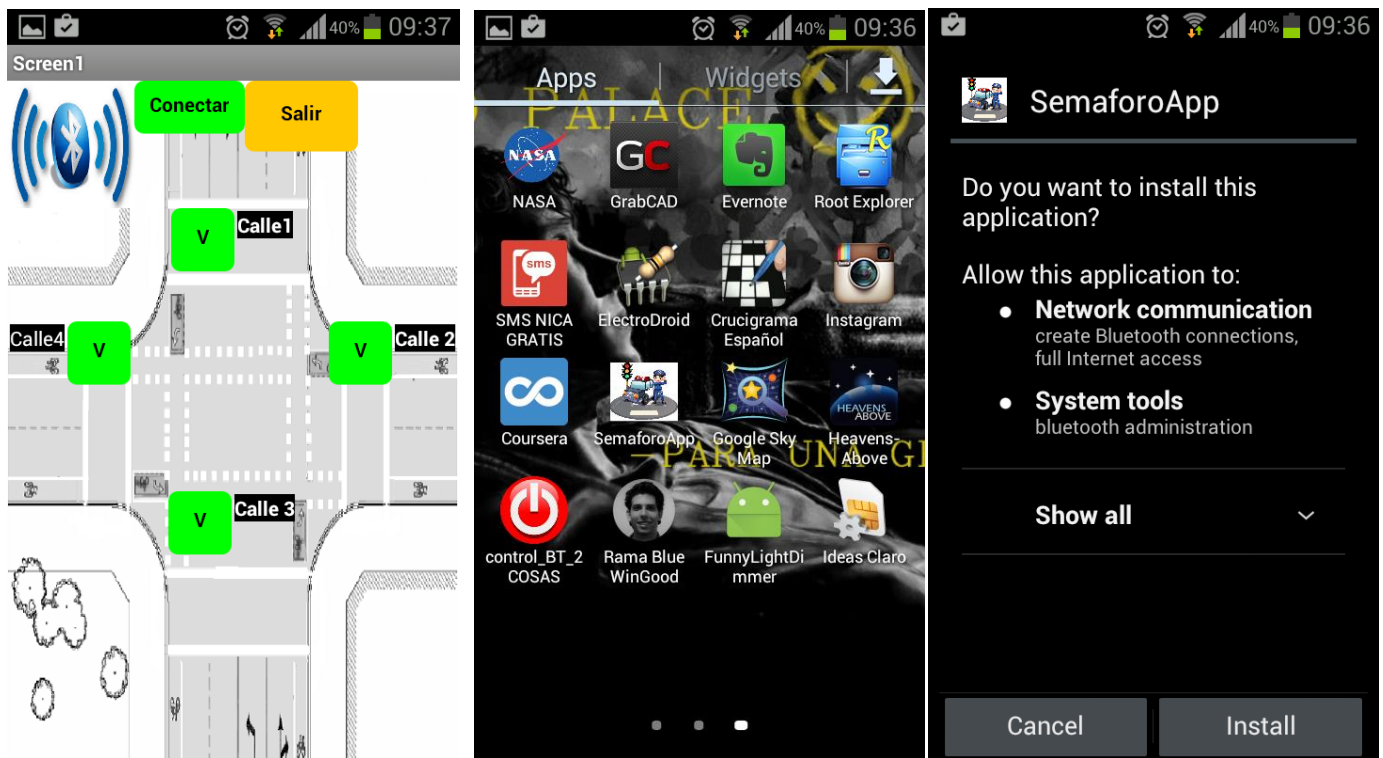


Figura14. Screenshots de la app Semáforo

Esta aplicación fue diseñada en appinventor entorno de programación desarrollada por el MIT posee un método de bloques que son interconectados gráficamente que después es compilado a lenguaje JAVA los sensores disponibles para el desarrollo de una app son.

Sensores

- Acelerómetro.
- Lector de códigos de barras y QR.
- Geolocalización GPS.
- Brújula.
- Gestión de contactos, llamadas, etc.
- Conectividad mediante Bluetooth

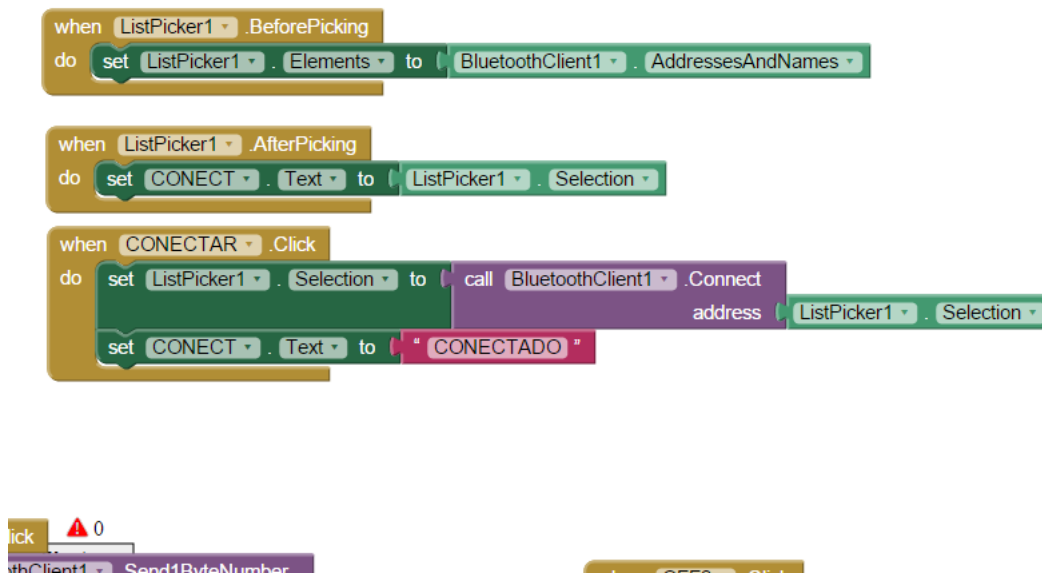


Figura15. Screenshots programación APP

Este método de programación agiliza el desarrollo de la creación de la aplicación es indispensable conocer la descripción de los bloques esta programación utiliza sentencia lógicas que son comunes de encontrar en otros lenguajes de programación : IF , ELSE, WHEN WHILE etc.

Funcionamiento

El sistema de control posee instrucciones de conectividad y estados lógicos direccionados a pines específicos de acuerdo a la tarea a ejecutar, el PIC18F4550 posee puertos dedicados en la comunicación RS232 (Puerto serial) tales puertos van conectados al módulo Bluetooth para recibir y enviar datos desde la aplicación. En el dispositivo móvil esta instrucción de lectura y escritura de datos en la programación es de vital importancia ya que el sistema está programado para detectar el estado de conectividad de la aplicación hacia el microcontrolador, así podemos definir el estado automático o manual del sistema.

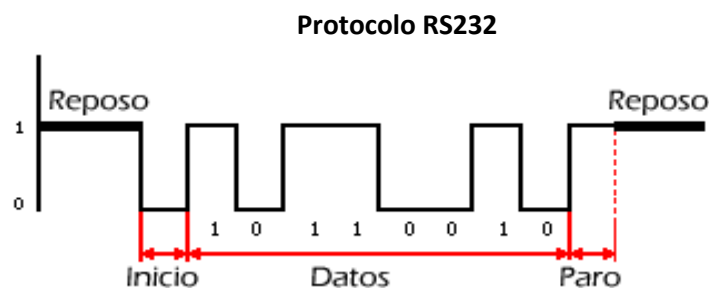


Figura16. Datos RS232

Estado Manual: permite al oficial de tránsito activar transiciones dirigidas a su elección ya que este prioriza el pase a la vía con más afluencia vehicular. El control que posee en la aplicación envía datos que identifican el estado del semáforo en las correspondientes vías seleccionadas.

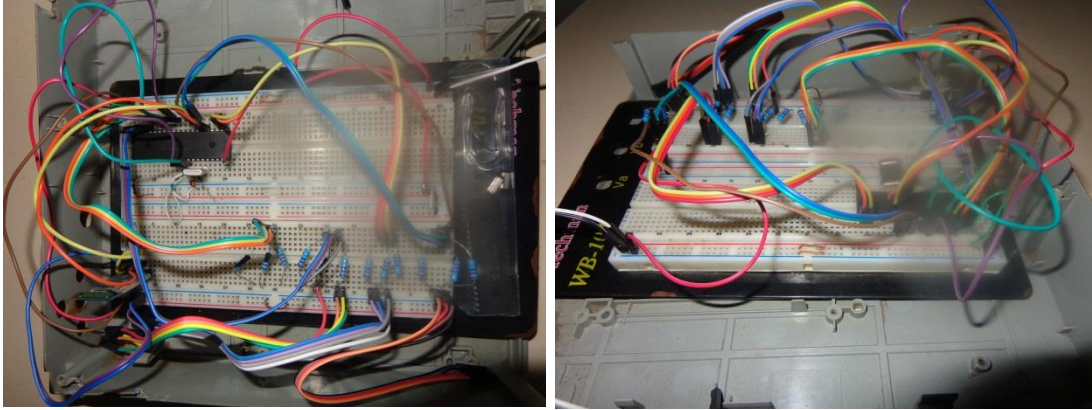
Estado automático: Cuando el sistema detecta que la aplicación ya ha sido cerrada o no está en uso las instrucciones que han sido programadas siguen el estado típico de un semáforo en una intersección.

Cada una de las salidas que son activadas tiene que ser acondicionadas con circuitería externa al microcontrolador ya sea el típico circuito de un transistor como switch con la carga de un

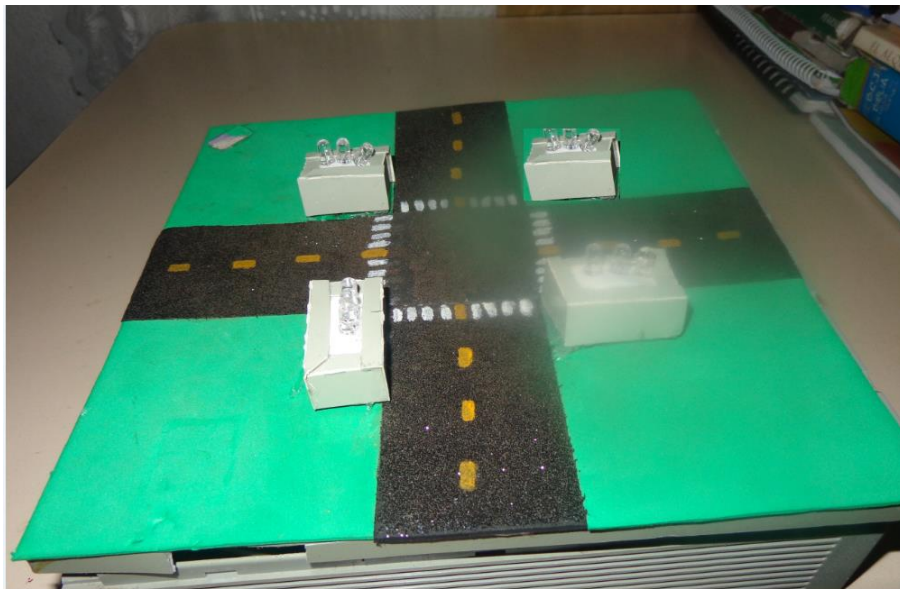
RELE que permite que los niveles de voltaje del microcontrolador accionen cargas con niveles de voltajes más altos como un Arreglo de Leds en un semáforo.

IMPLEMENTACION

Desarrollo del experimento propuesto en la protoboard:



Se implemento el diseño del circuito propuesto anteriormente para comprobar su funcionamiento y de igual forma se realizo una maqueta demostrativa.



Conclusión

El desarrollo de este sistema Semáforo inteligente con conexión Bluetooth nos ha permitido abordar conceptos de programación para el comportamiento de los señalizadores del semáforo donde se ha implementado una máquina de estado finito en la parte automática que responde efectivamente ante las interrupciones de las instrucciones del policía cabe destacar que abordando este método es posible agilizar el tiempo de respuesta de este. En la programación de la aplicación nos abra otra puerta de lógica programada para resolver problemas donde podemos vincular nuestro propio Smartphone esta observación nos sirve de referencia de la visión de los proyectos futuros.

La Conexión bluetooth y la APP nos permite resolver de manera Eficiente la problemática ya que no solo resolvemos el cambio de vías si no que la damos movilidad al uniformado para situarse en lugares estratégicos donde este pueda observar el comportamiento del tráfico.

Además de la creación de la programación se ha desarrollado la etapa de potencia donde se conectaran las luces indicadoras cabe destacar que no se propuso el tipo de luces ya que pueden ser leds o bombillos, de tal forma se ha dejado un sistema genérico que puede conectarse cualquier semáforo que se requiera reutilizar

Recomendaciones

El sistema ofrece resolver una problemática que es observada a diario por los habitantes de la capital ya que las horas picos el tráfico se vuelve muy congestionado. El oficial de tránsito en turno resuelve y nos permite la opción de no tener señales contradictorias en una intersección, este sistema es rentable pero no eficiente ya que solo si el oficial está controlando el paso este funciona inteligentemente como grupo recomendamos la elaboración de un sistema híbrido que nos permite tener un semáforo que sea capaz de darle la conectividad al oficial de tránsito y después que el oficial deje de manipular este sea capaz de darle el paso con ciertos parámetros previamente analizados estadísticamente de las afluencia de vehículos en cada intersección.

Existe ciertos sensores de presencia de automóviles ya sea por inducción electromagnética o ultrasónicos pero su instalación y sistemas de control son muy costosos dificultando la detección inteligente de autos y permitir al sistema darle el paso a la vía con más tráfico.



Figura17. Sensor de detección de vehículo.

En muchas ocasiones no basta con solo tener los detectores y permitir que el sistema trabaje automáticamente ya que el campo de visión de un oficial de tránsito es más amplio y eficiente y es por tal razón que en la horas con demasiado trafico dar paso a las más transitadas con el sistema híbrido.

Un ejemplo seria si el sistema solo posee 8 sensores ubicados estratégicamente en cada vía para cubrir un sensor por automóvil pero si hay más de 8 automóviles quizás puedan haber 16 automóviles pero en la otra vía solo hay 8 automóviles activando los sensores y el sistema detectaría que las dos vías tienen la misma afluencia de vehículos y este no daría el paso a las de las 16 si no que entraría al estado de semáforo con transiciones normales.

Por lo tanto es recomendable la elaboración de un sistema híbrido para la optimización del pase de vehículos por una intersección.

Biografías

<http://electronilab.co/tienda/modulo-bluetooth-hc-05-serial-rs232/>

<http://ww1.microchip.com/downloads/en/DeviceDoc/39632e.pdf>

<http://www.elnuevodiario.com.ni/nacionales/361916-policia-poner-orden-vias/>

<http://rdedatos.tripod.com/rs232.htm>

<http://developer.android.com/intl/es/sdk/index.html>

<http://www.mikroe.com/chapters/view/7/chapter-6-serial-communication-modules/>

<http://appinventor.mit.edu/explore/>

*Diseño y simulación de sistemas microcontroladores en lenguaje C

Juan Ricardo Clavijo Mendoza

Anexos

Programa en C del microcontrolador PIC18F4550

```
//----- Semáforo Control Policiaco ----- //
// ELABORADO POR YESSER MORALES y KELLER JIRON      //
// Maquina de estado                                //
// 07-12-2015 //

struct State
{
    unsigned char St1;
    unsigned char St2;
    unsigned char St3;
    unsigned char St4;
    unsigned int delay;
    int next[4]; // NEXT [11] [10] [01] [00]

}; // MSB LSB

struct State fsm_sema[] =
{
    //LSB MSB
    { 0x04, 0x01, 0x04, 0x04, 3000, {1}},
    { 0x04, 0x02, 0x04, 0x04, 800, {2}},
    { 0x04, 0x04, 0x04, 0x04, 500, {3}},
    { 0x01, 0x04, 0x04, 0x04, 3000, {4}},
    { 0x02, 0x04, 0x04, 0x04, 800, {5}},
    { 0x04, 0x04, 0x04, 0x04, 500, {6}},
    { 0x04, 0x04, 0x01, 0x04, 3000, {7}},
    { 0x04, 0x04, 0x02, 0x04, 800, {8}},
    { 0x04, 0x04, 0x04, 0x04, 500, {9}},
    { 0x04, 0x04, 0x04, 0x01, 3000, {10}},
    { 0x04, 0x04, 0x04, 0x02, 800, {11}},
    { 0x04, 0x04, 0x04, 0x04, 500, {0}},
};

unsigned char cs;
char uart_rd;
char uart_rdl;
unsigned int i;
unsigned int get,CNT;

void main(void)
{
    ADCON = 0x0F;
    INTCON = 0x90;
    TRISC = 0x80;
    TRISB = 0x00; //Salida
    TRISA = 0xFF; //Entrada
    TRISD = 0x00; //Salida
    PORTB = 0x00; // Estado inicial del puerto PORTB
    PORTA= 0x00;
    PORTD= 0x00;
```

```

UART1_Init(9600);          // Initialize UART module at 9600 bps
Delay_ms(100);             // Wait for UART module to stabilize

UART1_Write_Text("Start");
UART1_Write(10);
UART1_Write(13);
//uart_rd ='0';
//uart_rd = UART1_Read0;
while(1)
{
    uart_rd = UART1_Read0;
    Error:

    while(uart_rd == 'A' || 'B' || 'C' || 'D')
    {

        // If data is received,
        // read the received data,
        // uart_rd1 = uart_rd;

        while (uart_rd=='A') {

            uart_rd = UART1_Read0;
            // Calle 1 con pase 2,3 y 4 en ALTO
            PORTB = (0x04 << 4) | 0x04 << 1;
            PORTD = (0x01 << 4) | 0x04 << 1;
            delay_ms (500);
            UART1_Write_Text("I have not ");
            if (uart_rd !='A') {
                PORTB = (0x04 << 4) | 0x04 << 1;
                PORTD = (0x02 << 4) | 0x04 << 1;
                delay_ms (800);
                goto Error;
            }

        }

        while (uart_rd=='B') {
            uart_rd = UART1_Read0;
            PORTB = (0x04 << 4) | 0x04 << 1;
            PORTD = (0x04 << 4) | 0x01 << 1;
            delay_ms (500);
            UART1_Write_Text("I have all ");
            if (uart_rd !='B') {
                PORTB = (0x04 << 4) | 0x04 << 1;
                PORTD = (0x04 << 4) | 0x02 << 1;
                delay_ms (800);
                goto Error;
            }

        }

        while (uart_rd=='C') {
            uart_rd = UART1_Read0;

```

```

    PORTB = (0x01 << 4) | 0x04 << 1;
    PORTD = (0x04 << 4) | 0x04 << 1;
    delay_ms (500);
    UART1_Write_Text("I have maximun ");
    if ( uart_rd !='C') {
    PORTB = (0x02 << 4) | 0x04 << 1;
    PORTD = (0x04 << 4) | 0x04 << 1;
    delay_ms (800);
    goto Error;
}

    }

while (uart_rd=='D') {
    uart_rd = UART1_Read();
    PORTB = (0x04 << 4) | 0x01 << 1;
    PORTD = (0x04 << 4) | 0x04 << 1;
    delay_ms (500);
    UART1_Write_Text("I have less ");
    if ( uart_rd !='D') {
    PORTB = (0x04 << 4) | 0x02 << 1;
    PORTD = (0x04 << 4) | 0x04 << 1;
    delay_ms (800);
    goto Error;
}

    }

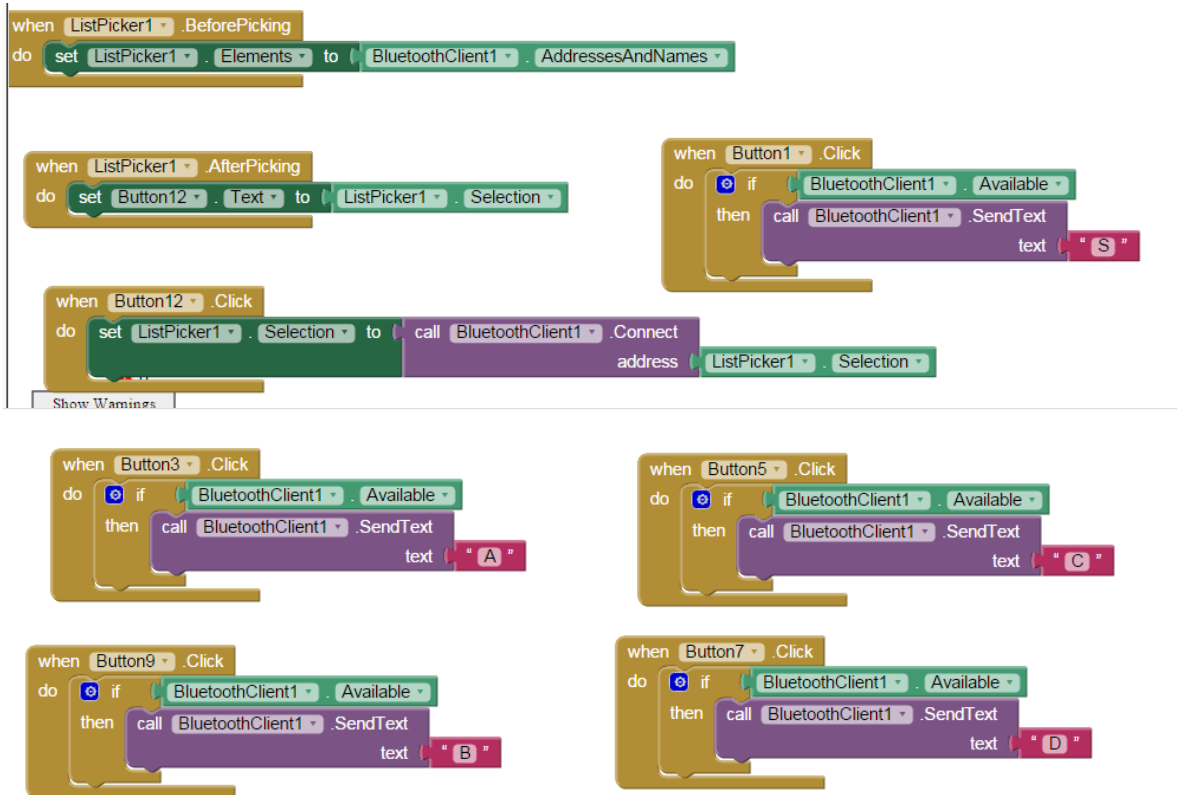
    break;
    continue;
    }

    cs = next[0];
    PORTB = (fsm_sema[cs].St3 << 4) | fsm_sema[cs].St4 << 1;
    PORTD = (fsm_sema[cs].St2 << 4) | fsm_sema[cs].St1 << 1;
    Vdelay_ms (fsm_sema[cs].delay);

}
}

```

AppInventor



PCB

Debido a que se necesita una etapa para acondicionar el circuito anteriormente mencionado en el esquemático se propone el uso del siguiente circuito para su posterior instalación no se implementó esta placa PCB debido a que solo se entregó en protoboard.

Schematic En el software EAGLE (software exclusivo para la elaboración de PCB)

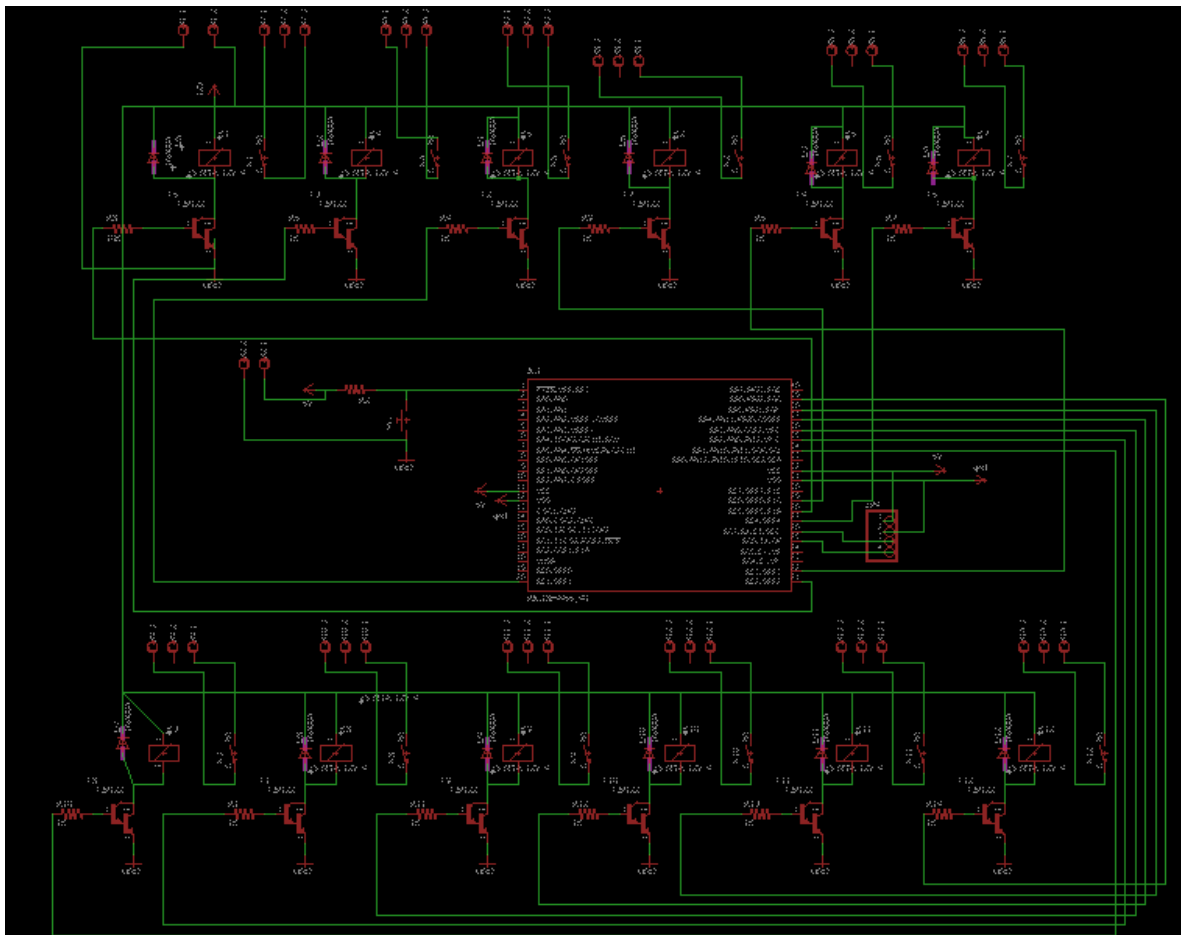


Figura18. Schematic de componentes Eagle Diseño PCB

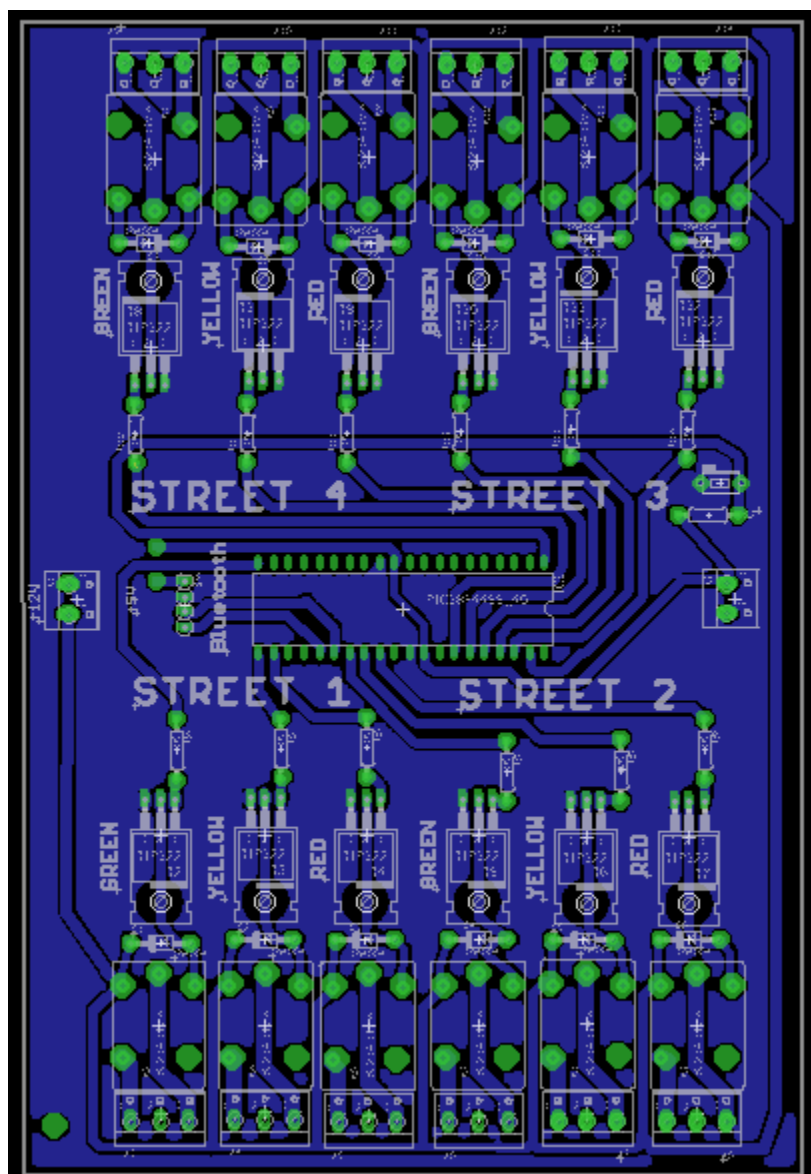


Figura19. Se muestra la pcb del microcontrolador conectado con la etapa de potencia

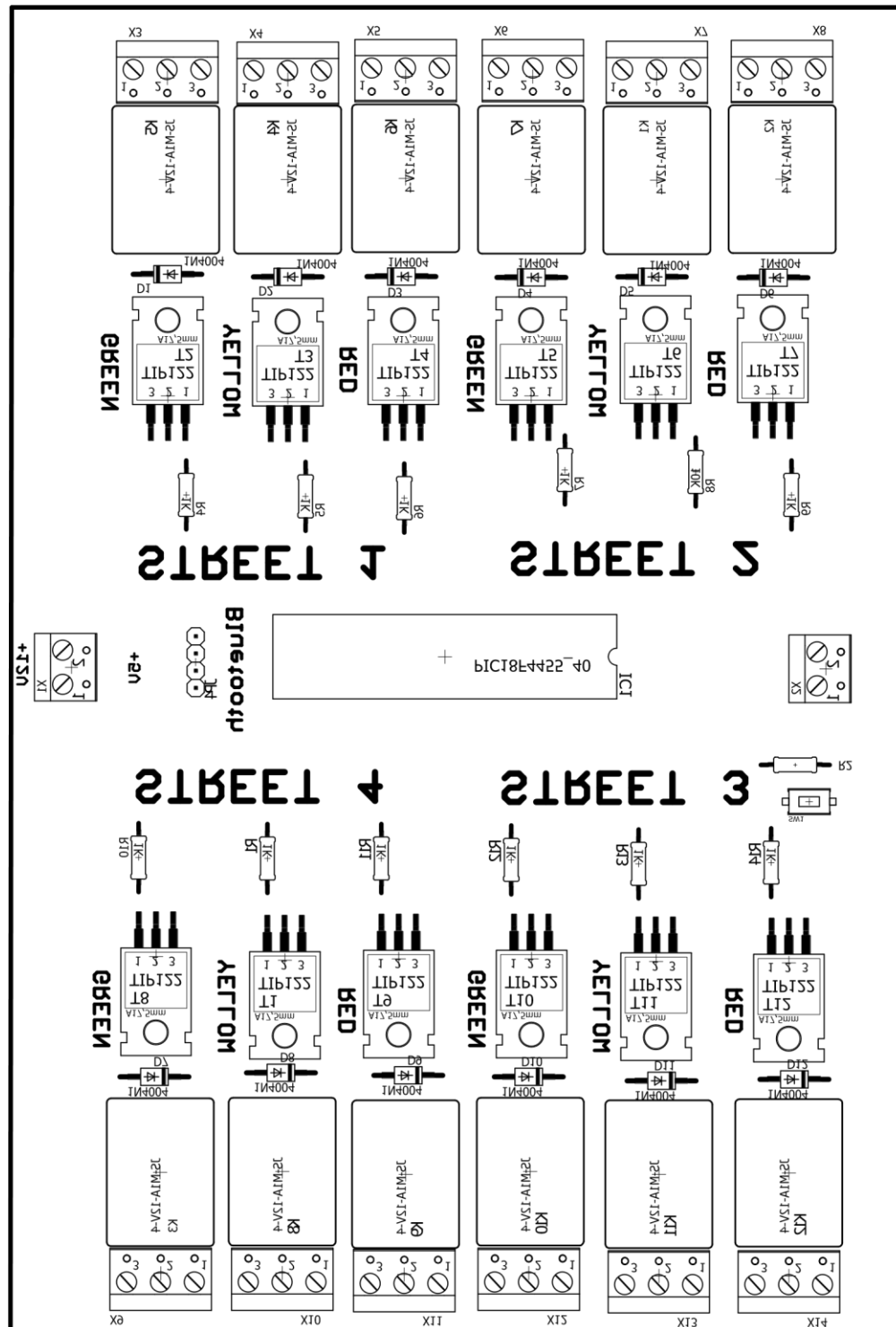


Figura20. Top Mirror de los componentes escala 100%

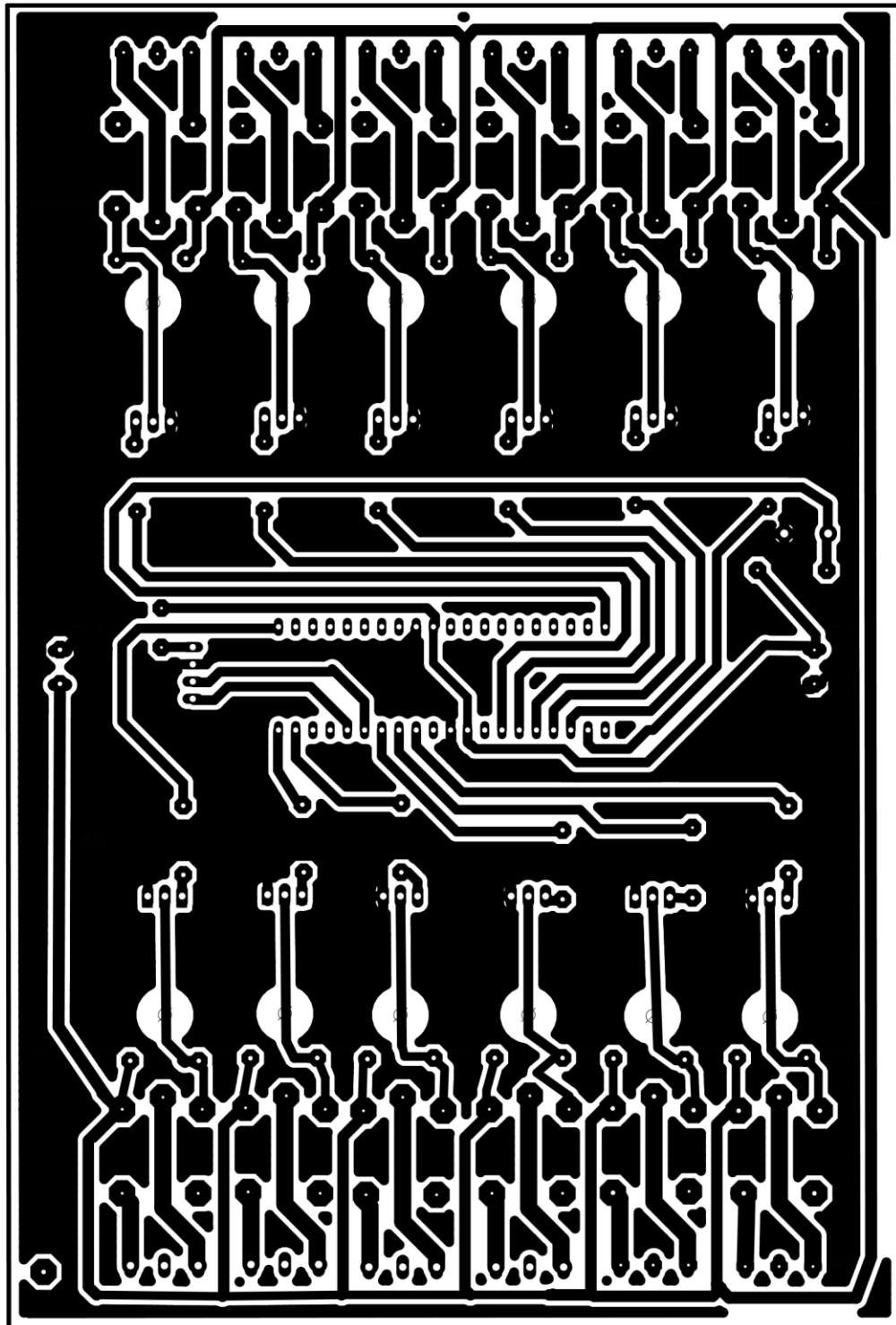


Figura21. Bottom de los componentes escala 100%