

Diseño e implementación de un prototipo de chaleco inteligente, para seguridad vial mediante computación portátil.

Andrés Eduardo Dávila Miranda

Departamento de Ciencias de la Energía y Mecánica
Carrera de Ingeniería Mecatrónica
Sangolquí, Ecuador
aedavila2@espe.edu.ec

Byron Patricio Torres Merino

Departamento de Ciencias de la Energía y Mecánica
Carrera de Ingeniería Mecatrónica
Sangolquí, Ecuador
bptorres@espe.edu.ec

Road Safety and improving mobility in Ecuador has a problem, one of the reasons is that the reflective vest now used forwarding agents, only allows the display and does not provide any safety case because need external light to be seen, being limited to be indicative element and no control mobility so to fix some part design and implement a prototype vest using laptop computer, which will have different types of materials, as conductive fibers that can easily attach to clothing for electrical circuits in it and for the management of different traffic signs and the location and sending emergency messages glove controller attached to vest is designed to facilitate the simple operation thereof by sensors (buttons), traffic signals going to the left or right, stop, caution will be displayed with high brightness LEDs, low power consumption, designed for dress or e-textiles. The signal acquisition, management and control will be performed by a microcontroller device designed for e-textiles and garments called LilyPad low power consumption which is easily programmed using the Arduino IDE. Furthermore, this project will include a GPS module that allows the location of users and a GSM module that can send text messages.

Keywords: Laptop Computer, Conductive Fiber, e-textiles, GPS, GSM, LilyPad.

I. INTRODUCCIÓN

El chaleco inteligente para seguridad vial es un sistema compuesto por dispositivos electrónicos y leds de alto brillo diseñados para textiles inteligentes, de bajo consumo energético y sus circuitos eléctricos unidos mediante fibras conductoras. Posee un guante que controla las funciones del chaleco inteligente.

Las funciones que posee el prototipo son las siguientes: la localización del usuario mediante la utilización de dispositivos electrónicos GPS y GSM, y la aplicación APK para el sistema operativo Android, se puede visualizar la ubicación en google maps, los datos de posición son almacenados en una memoria SD. La activación de luces indicatoras leds para mejorar la movilidad de tránsito y la seguridad vial, las señales indicatoras son de prevención de color amarillo, de alto de color rojo y direccionales de color verde.

En la actualidad la seguridad vial es ineficiente ya que se ha incrementado el número de vehículos motorizados en los últimos años aumentando el riesgo de accidentes y el tráfico vehicular, siendo la movilidad un tema de gran importancia para el desarrollo del buen vivir del Ecuador, por lo tanto para resolver este problema se plantea el uso de chalecos con implementación de computación portátil por parte de ciclistas, conductores de motocicletas y agentes de tránsito, de forma que este chaleco innovador y tecnológico permita una correcta visualización y fácil comprensión de señales de tránsito como el de prevención, alto y direccionales, ayudando a manejar el flujo de movilidad y evitar los accidentes.

II. SELECCION DE MATERIALES Y COMPONENTES ELECTRONICOS

A. Selección de materiales para la implementación del chaleco y los circuitos eléctricos

El chaleco inteligente debe ser una prenda a prueba de agua no transpirable, los textiles a prueba de agua son diseñadas para ambientes de extremo frío e intensidad de lluvia, ya que la impermeabilidad también ayuda a mantener una temperatura cálida en el usuario, debido a que no posee la característica de transpirabilidad no se recomienda su uso en actividades que demanden gran esfuerzo físico, por lo tanto se ha seleccionado un tipo de tela que sea lo suficientemente impermeable pero a las ves transpirables y la tela seleccionada es Taslan como se muestra en la figura 1.



figura 1. Tela Taslan impermeable.

El hilo conductor seleccionado para este proyecto está manufacturado a partir de fibras de acero inoxidable 316L, por tal motivo se garantiza una buena conductividad eléctrica, y resistencia al fuego porque a diferencia de otros hilos conductores no posee nylon en su núcleo interno.

Este tipo de hilo conductor no presenta buenas características para ser soldado con otros materiales entonces una alternativa para conectar componentes eléctricos y electrónicos mediante esta fibra es mediante costura manual o sino con la utilización de máquinas de coser. Debido a la forma no tan compacta de este tejido en algunos tramos del mismo se producen pelusas, por lo tanto se debe engrasar el hilo antes de utilizarlo.

El proceso mediante al cual se elabora los hilos conductores se denominada estiramiento en haz, en cual consiste en estirar una probeta circular de acero 316L hasta que adquiera un diámetro se encuentre en el rango de 1 a 40 micras.

Este hilo viene enrollado en una bobina de plástico, y la longitud que alcanza es de 30 pies, esto se puede apreciar en la figura 2.



Figura 2: Hilo conductor [1].

B. Selección de los componentes electrónicos

Selección del controlador

Entre todas las variedades y tipos de arduinos existentes, se selecciona el arduino LilyPad que reúne todas las especificaciones y características necesarias y sus especificaciones técnicas se puntualizan en la tabla 1.

Tabla 1

Características LilyPad Arduino 328 Main Board [2].

Microcontrolador	ATmega328V
Tensión de funcionamiento	2.7 a 5.5 V
Voltaje de entrada	2.7 a 5.5 V
Pines digitales I/O	14(de los cuales 6 proporciona PWM)
Pines de entrada analógica	6
Corriente DC por Pin I/O	40 mA
Memoria Flash	16KB(de los cuales 2 KB utilizados por el gestor de arranque)

SRAM	1 KB
EEPROM	512 bytes
Velocidad de reloj	8 MHz
Diámetro exterior	50 mm
Espesor	0.8 mm de la PCB
Lavable	Si
Peso	10 g

Esta placa electrónica está diseñada para wearables y e-textiles y puede ser unida o cosida a una tela o a una malla plástica y además unida a otros dispositivos electrónicos mediante hilo conductor, posee 22 pines que se encuentran descritos en la figura 3.N, posee un puerto de comunicación serial por hardware en Rx y Tx en el pin 0 y 1 respectivamente, Pin 13 SCK, Pin 12 MISO, Pin 11 MOSI, CS puede ser cualquier pin digital excepto los pines 11, 12 y 13, el requerimiento necesario también para la implementación del prototipo de chaleco es de que sea liviano y ergonómico por lo que es necesario que el peso del controlador sea lo menor posible en el caso del LilyPad arduino tan solo pesa 10 g por lo que cumple las exigencias requeridas.

En la figura 3 observamos el controlador LilyPad Arduino.

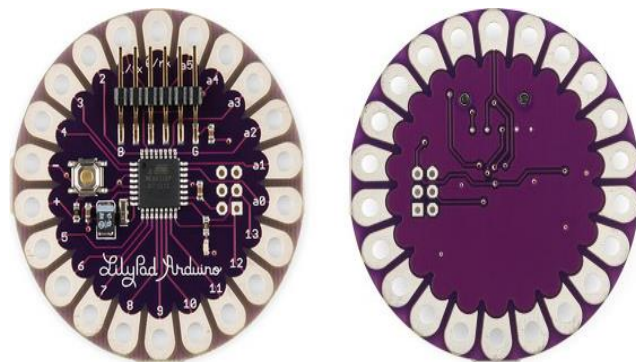


Figura 3: Controlador LilyPad [2].

Selección de los sensores para el guante de control.

Para la activación de las múltiples funciones del prototipo de chaleco se selecciona los sensores de contacto que son sensores digitales que entrega a su salida una señal de tipo discreta o binario (0 o 1). Los sensores de contacto a utilizar para el prototipo son tipo pulsador.

En la figura 4 se muestra lo pulsadores tipo botón a utilizar en el prototipo para el control de las funciones del sistema ya que es económico, fácil de adquirir, sencillo de utilizar e intercambiable.



Figura 4: Pulsadores [3].

Selección del módulo GSM.

Se selecciona un escudo GPRS V1.0 con el chip SIM 900 ya que las prestaciones necesarias que ofrece el escudo son suficientes en la aplicación de la construcción del prototipo ya que solo utilizaremos el uso de SMS para enviar los datos de la ubicación mediante comandos AT y cuya velocidad de comunicación es de 19200 a través de comunicación serial la cual es soportada fácilmente por el controlador Lilypad, además que se puede encender el modulo mediante software conectando a uno de sus pines, es accesible, económico y de bajo de consumo energético, utiliza un chip GSM o tarjeta SIM de cualquier operadora y en cualquier país, el único inconveniente del escudo GSM/GPRS es la antena para la recepción de la señal que es del tipo antena de mástil lo que conlleva a requerir un espacio mayor al construir el prototipo, lo ideal para la construcción del prototipo sería una antena tipo PCB (Printed Circuit Board o Tarjeta de Circuito Impreso) ahorrando mayor espacio y disminuyendo la rigidez en la construcción del circuito.

Las características del escudo GPRS V1.0 son las siguientes [4]:

- Basado en SIMCom módulo SIM900.
- Quad-Band 850/900/1800/1900 MHz - funcionaría en redes GSM en todos los países del mundo.
- Control a través de comandos AT - Comandos estándar: GSM 07.07 y 07.05 | mejoradas Comandos: Comandos AT SIMCOM.
- Servicio de mensajes cortos - de modo que usted puede enviar pequeñas cantidades de datos a través de la red (ASCII o hexadecimal en bruto).
- TCP Embedded / pila UDP - le permite cargar datos a un servidor web.
- Tomas de los altavoces y auriculares - por lo que puede enviar señales DTMF o reproducir la grabación como un contestador.
- Zócalo de la tarjeta SIM y la antena GSM - presente a bordo del escudo.
- 12 GPIO, 2 PWM y un ADC (toda lógica 2,8 voltios) - para una conexión con Arduino.

- Bajo consumo de energía - (modo de ahorro) 1.5 mA.
- Rango industrial de temperatura de - 40 ° C a 85 ° C. (seedstudio, GPRS Shield V1.0, 2014)

Las especificaciones técnicas del SIM900 se detallaran en la tabla 3-

Tabla 3

Especificaciones del SIM900 [4].

MODULO GSM/GPRS	SIM900
Tensión de funcionamiento	3.1 a 4.8 V
Voltaje de entrada	3.1 a 4.8 V
Corriente DC	50 mA
Control	Vía comando AT
Dimensiones con la antena	110x58x19 mm
Peso Neto	76±2 g

Un modelo compacto que reúne las funciones antes descritas en la se muestra en la figura 5.



Figura 5: Módulo GSM [4].

Selección del módulo GPS.

Para el desarrollo del prototipo se selecciona un módulo GPS el cual es fabricado y diseñado para e-textiles o para vestimentas construido alrededor del chipset MTK3339. Es ligero, pequeño, delgado y posee una antena de parche interno, fácil de unir a una tela o malla plástica mediante hilo conductor a otros dispositivos electrónicos, consume pequeñas cantidades energía eléctrica y es un excelente receptor de alta sensibilidad GPS, el nombre del pequeño pero potente receptor GPS es modulo GPS ULTIMATE.

Las características del módulo GPS se detallan a continuación [5]:

- Satélites: 22 de seguimiento, 66 de búsqueda
- Antena Patch Tamaño: 15mm x 15mm x 4mm

- Velocidad de actualización: 1 a 10 Hz
- Precisión de la posición: <3 metros (toda la tecnología GPS tiene unos 3 metros de exactitud)
- Precisión de la velocidad: 0,1 m / s
- Cálido / arranque en frío: 34 segundos
- Sensibilidad de adquisición: -145 dBm
- Seguimiento de sensibilidad: -165 dBm
- Altitud máxima para PA6H: probado en 27.000 metros
- Velocidad máxima: 515m / s
- Gama Vin: 3.0-5.5VDC
- Actual MTK3339 de funcionamiento: 25 mA de seguimiento, 20 mA de corriente durante la navegación
- Salida: NMEA 0183, 9600 baudios por defecto
- DGPS / WAAS / EGNOS apoyado
- Cumplimiento y soporte AGPS (Modo Offline: EPO válida hasta 14 días) FCC E911
- Hasta 210 canales PRN
- La detección y la reducción de Jammer
- Detección multi-ruta de acceso y compensación
- Historial de revisiones
- 07 de octubre 2014 - que ahora lo venden con v5516 firmware que corrige algunos cálculos a gran altitud (más precisión)
- Detalles del tablero del desbloqueo:
- Peso: 5,43 g
- Dimensiones: 30,5 mm de diámetro x 5.98mm de espesor / 1.2 "de diámetro x 0.24" de espesor.

En la figura 6 se muestra el modulo GPS a utilizar.

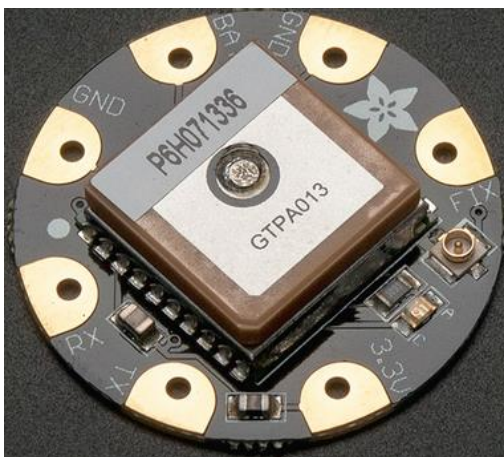


Figura 6: Modulo GPS [5].

Selección del módulo de lectura y escritura SD.

Para transferir los datos receptados por el controlador LilyPad Arduino desde el GPS se utiliza una tarjeta SD de almacenamiento, la cual se manipula desde un módulo de lectura y escritura de tarjetas SD, el módulo es una tarjeta electrónica que poseen todos los componentes eléctricos y electrónicos para una fácil y rápida instalación con el controlador y un socket para ingresar la tarjeta SD.

EL módulo seleccionado para la realización del prototipo es una tarjeta lectora SD ARM MCU.

Las características del módulo de lectura y escritura SD se detallan a continuación:

- Soporte para tarjetas SD.
- Pines de salida SD SPI: MOSI, SCK, MISO y CS.
- Voltaje de entrada de 3.3 a 5 V dc.
- Consumo de corriente 0.16 mA.
- Peso 6 g.

En la figura 7 se muestra el módulo de lectura y escritura SD, con el socket para la tarjeta de memoria SD y sus componentes eléctricos y electrónicos que permite la conexión directa al controlador LilyPad Arduino.

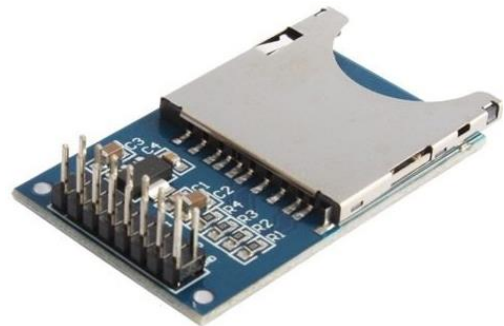


Figura 7: Modulo de lectura y escritura SD ARM MCU [6].

III. DESCRIPCIÓN DEL ALGORITMO DE CONTROL Y DESARROLLO DEL SOFTWARE APK PARA ANDROID

A. Descripción del Algoritmo de Control

El algoritmo del sistema de control se detalla a continuación: la primera función es la decodificación de los datos que provienen del módulo GPS utilizando la librería TinyGPS, y se procede a guardar los datos de latitud y longitud en la memoria SD.

Al iniciar el programa se asigna a la variable "var" la letra "e" y se realiza una sola vez. La siguiente interacción es leer los datos analógicos proporcionados por las entradas A0, A1, A2, A3, A4 del controlador LilyPad, que se encuentran conectados con los pulsadores del guante de control. Se verifica si la entrada A0 ha sido accionada, si es verdadero, se

asigna la letra “t” a una variable denominada “var”, pero en caso falso el programa la variable “var” se asigna la letra “z”. De forma similar para la entrada A1, pero en este caso a la variable “var” se asigna la letra “i”, para la entrada A2 se asigna la letra “d”, para la entrada A3, la letra “a”, para la entrada A4, la variable “r”. Cuando se accionada las entradas A3 y A4 al mismo tiempo, la variable “var” se asigna la letra “O”.

Como siguiente interacción se tiene la revisión de la variable “var” que asocia a los diferentes casos para activar cada una de las funciones. Si la variable se asignó la letra “t”, en caso afirmativo se activa la función “SendMessage”, la cual envía un mensaje de texto con la información de ubicación hacia un teléfono móvil previamente seleccionado, en caso negativo se regresa a la función de decodificar variables de GPS. Si la variable se asignó la letra “e” se activa la función “powerUpDown” la cual enciende el módulo GSM, esta función se activa una sola ocasión. Si la variable se asignada con la letra “i”, en caso afirmativo, se activa la función lediz, para encender las luces de color verde, que tiene forma de flecha hacia la izquierda. Si la variable se asigna la letra “d”, si es así se activa la función “ledde”, que sirve para encender las luces de color verdad, cuya forma es de flecha hacia la derecha. Si la variable se asigna la letra “a”, activa la función “ledam” la cual enciende las luces indicadores de color amarillo, pero en caso negativo se regresa a la función de decodificar variables del módulo GPS. Si la variable se le asigna la letra “r” activa la función “ledroj” la cual enciende las luces indicadores de color rojo, pero en caso negativo se regresa a la función de decodificar variables del módulo GPS. Para desactivar cualquier función del prototipo la variable se asigna la letra “O” la que activa la función “ledoff” apagando todas las luces indicadores. Al finalizar cualquier función se asigna a la variable “var” con la letra “z”, en caso afirmativo sale de la verificación de los casos y se retorna con la lectura de las entradas analógicas para la asignación de la letra de la variable “var”.

B. Desarrollo del software APK para android

Para el desarrollo de la aplicación se usa el navegador web, se inicia la aplicación ingresando a <http://ai2.appinventor.mit.edu/> y con la cuenta de google se accede.

Una vez ingresado a la aplicación appinventor 2 se da clic en empezar un nuevo proyecto como se muestra en la figura 8.



Figura 8: Acceso a un nuevo proyecto para hacer la aplicación.

Una vez creado el proyecto en la parte superior derecha se encuentra dos botones en el que se debe seleccionar diseño. Se inicia con la realización de la interfaz gráfica de la aplicación, ingresando botones, línea donde se va a escribir nuestros datos, y el texto de la descripción y autoría como se muestra en la figura 9.

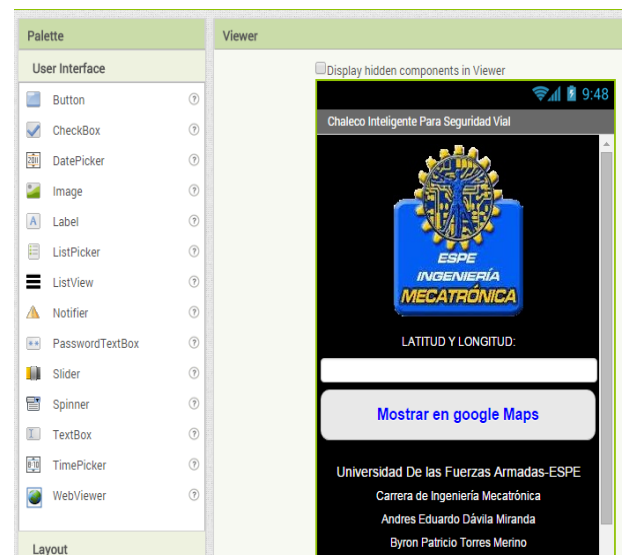


Figura 9: Entorno de diseño de la interfaz de la aplicación para (SO) Android.

Para iniciar la programación de la aplicación luego de hacer la interfaz se da clic en la parte superior derecha de la pantalla en el botón bloques, en la ventana de bloques es donde insertamos y nombramos a los componentes de nuestra aplicación como se muestra en la figura 10.

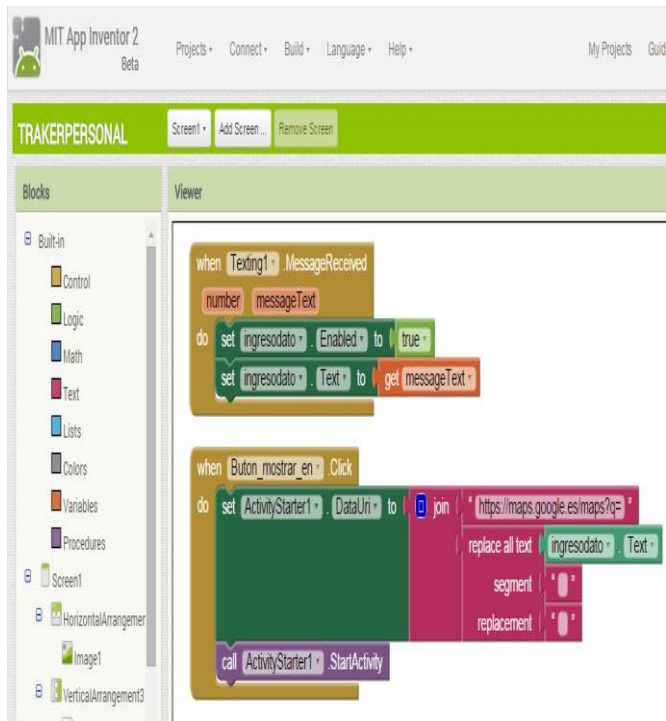


Figura 10: Entorno de programación por bloques de la aplicación para (S.O.) Android.

En la figura 10 básicamente se tiene los bloques de variables de programación y control, además existen bloques preestablecidos dependiendo de las aplicaciones propias para el sistema operativo Android.

En el desarrollo de la aplicación se utiliza un bloque llamado MessengerReceived como se indica en la figura 11, se encuentra en los bloques de control de mensajes, este bloque permite que la aplicación reconozca un número de teléfono y el texto que se encuentra al recibir un SMS de este número almacenando la información en las variables “number” y “messageText” respectivamente.

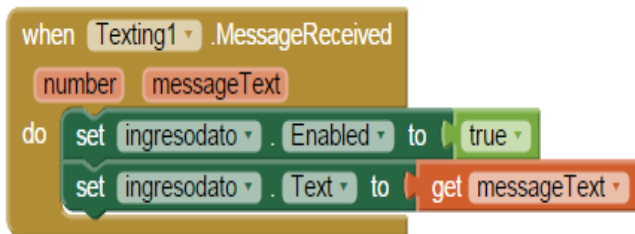


Figura 11: Bloque de programación MessengerReceived.

Ya con la información almacenada en variables hacemos la respectiva visualización con los bloques de control como se indica en la figura 12. Los bloques “set” del visualizador ingresado el primero sirve para confirmar la información que se ha reconocido al momento de llegar el SMS y el segundo bloque es para mostrar el texto del mensaje en la barra de búsqueda de nuestra aplicación.

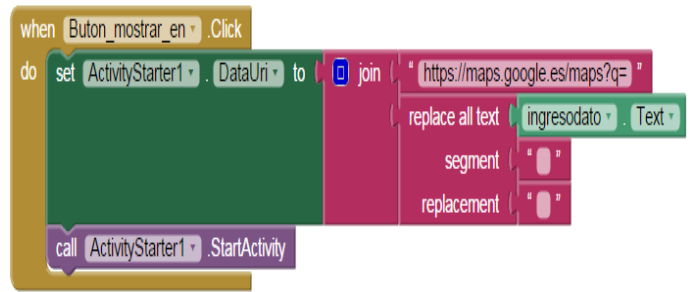


Figura 12: Bloque de programación para asociar a google maps.

En la figura 12 se observa la sentencia correspondiente al botón “mostrar en google maps” en el bloque ‘join’ ingresamos el direccionamiento de google maps en la aplicación maps de los teléfonos Android, reemplazando el texto del mensaje con lo que se encuentra en la barra de búsqueda en blanco.

En la figura 13 se muestra la aplicación en ejecución esperando el nuevo mensaje con la información de la ubicación del usuario.



Figura 13: Aplicación para Android en ejecución.

En la figura 14 se muestra cuando la aplicación recibe un SMS con los datos de la posición del usuario, para mostrar la ubicación con los datos recibido se da clic en el botón mostrar en google maps.



Figura 14: Aplicación recibiendo información de la ubicación.

En la figura 15 se visualiza la ubicación de los datos recibido por SMS en google maps.



Figura 15: Visualización de la ubicación con los datos recibidos por SMS

La aplicación actualiza la ubicación siempre y cuando se regrese a la pantalla principal y luego se realiza nuevamente la búsqueda con el botón buscar en google maps.

La aplicación se la puede instalar en cualquier teléfono con sistema operativo Android, con un ejecutable llamado trackerpersonal.APK, se lo copia en la memoria del celular

luego se lo hace clic y se cambia los permisos de instalación de aplicaciones de terceros. Esto se encuentra en configuración, seguridad señalar en cuadro de fuentes desconocida como se indica en la figura 16.

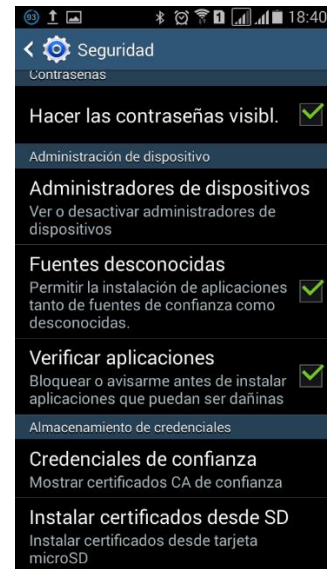


Figura 16: Configuración para la instalación del trackerpersonal.APK.

IV. IMPLEMENTACION Y FUNCIONAMIENTO DEL PROTOTIPO DE CHALECO INTELIGENTE

Implementación de los componentes eléctricos y electrónicos al chaleco de forma estratégica.

Se coloca las luces indicadoras led que muestran las direcciones de izquierda y derecha. Las luces indicadoras son de color verde y de forma de rombo se implementan en la parte frontal y en los dos costados del prototipo de chaleco inteligente sobre una cinta reflectiva como se demuestra en la figura 16.



Figura 16: Implementación de las luces indicadores de dirección.

El circuito de las luces indicadoras de dirección implementado en el prototipo se expone en la figura 17

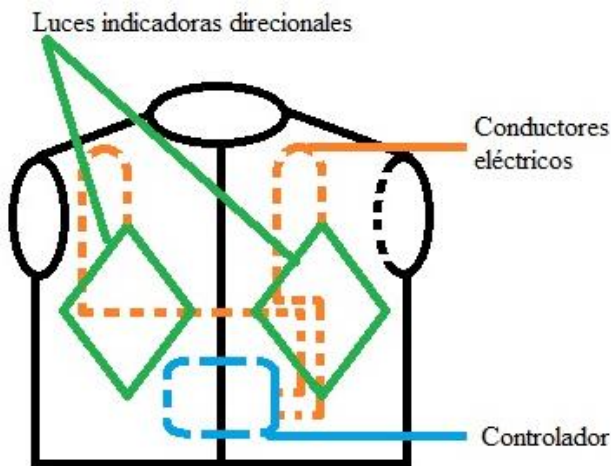


Figura 17: Circuito de las luces indicadoras de dirección implementado en el prototipo.

Para encender una sola dirección, la conexión de los conductores eléctricos está configurado para que vaya a un solo plug de dirección ya sea esté de izquierda o derecha, conectado al controlador.

Las luces indicadoras de paro (leds de color rojo) y de precaución (leds de color amarillo) se implementan en la parte posterior del prototipo como se muestra en la figura 18, en cintas reflectivas de forma horizontal y paralelas.



Figura 18: Implementación de las luces indicadores de paro y de precaución.

El circuito de las luces indicadoras de paro y precaución implementado en el prototipo se expone en la figura 19.

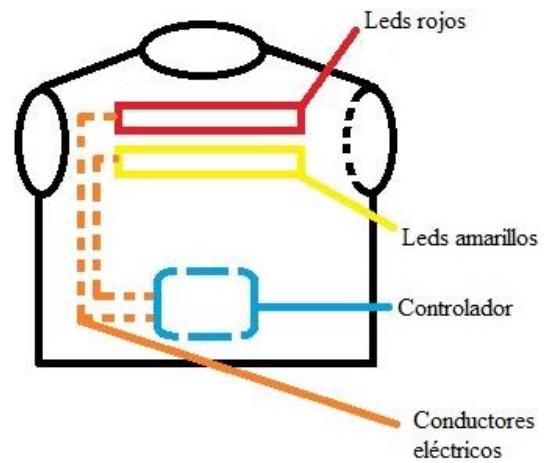


Figura 19: Circuito de las luces indicadoras de paro y precaución implementado en el prototipo.

Implementación del sistema de control.

El sistema de control está comando por un guante de control que posee pulsadores que envía señales al controlador para activar varias funciones. Las funciones que controlan son 5 y son las siguientes:

- Pulsador P1 controla el envío de la ubicación del usuario mediante el LilyPad, el módulo GPS y el módulo GSM.
- Pulsador P2 controla el encendido de las luces indicadoras led con dirección hacia la izquierda.
- Pulsador P3 controla el encendido de las luces indicadoras led con dirección hacia la derecha.
- Pulsador P4 controla el encendido de las luces indicadoras led de paro.
- Pulsador P5 controla el encendido de las luces indicadoras led de precaución.

Los pulsadores están dispuesto de izquierda a derecha desde P1 hasta P2, como se muestra en la figura 20.



Figura 20: Guante de control con los pulsadores de activación de funciones

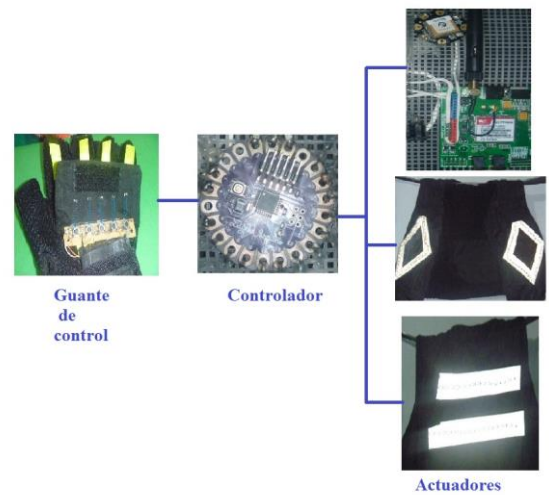


Figura 22: Sistema de control.

El guante de control en conjunto con los pulsadores, son conectados a los plugs mediante conductores eléctricos y además están recubiertos de un plástico que protege de ambientes hostiles. Lo que proporciona un fácil montaje y desmontaje al sistema de control como se indica en la figura 21.



Figura 21: Guante de control y conectores plugs.

El funcionamiento del sistema de control consiste en maniobrar cualquiera de los 5 sensores pulsadores, al accionarlos envía una señal a las entradas del controlador para que este la procese y ejecute cualquiera de las funciones ya antes mencionadas como se indica en la figura 22.

En la figura 23 se denota el sistema de control implementado en el prototipo de chaleco inteligente.



Figura 23: Sistema de control implementado

REFERENCES

- [1] Canakit. (10 de Diciembre de 2012). Canakit Conduvitive Thread Bobbin. Obtenido de Amazon Tri Prime: <http://www.amazon.com/gp/product/B007R9UA4S%3FSubscriptionId%3DA>.
- [2] ARDUINO. (2014). Introducción de Arduino. Obtenido de <http://arduino.cc/en/pmwiki.php?n=Guide/Introduction>.
- [3] Alavardo, C. (2012). *Space Electronic*. Obtenido de http://electronicavirtual-carlos.blogspot.com/2012_11_01_archive.html.
- [4] seeedstudio. (2014). *GPRS Shield V1.0*. Obtenido de http://www.seeedstudio.com/wiki/GPRS_Shield_V1.0
- [5] Adafruit Industries, L. (2013). *Flora Wearable Ultimate GPS Module*. Obtenido de <http://www.adafruit.com/products/1059>.
- [6] Admin. (5 de Agosto de 2012). *SD Card Module (HCARDU008)*. Obtenido de FORUM.HOBBY: <http://forum.hobbycomponents.com/viewtopic.php?f=25&t=5>.

