



---

# ***Carpeta Técnica***

---

**Camino sin obstáculos, camino seguro**

Explicación detallada técnica del sistema STAC

STAC 2021

2021



## Presentación del equipo de trabajo

Gabriel Balbi Trejo	Nicolas Alan Cabrera	Manuel Eduardo Torancio
		
7° 1° AVC Tel.: 1125890903 gabito.02@hotmail.com D.N.I.: 44545947	7° 1° AVC Tel.: 1125138803 nicocabreraaaa@gmail.com D.N.I.: 44890648	7° 1° AVC Tel.: 1134433987 manutorancio12@gmail.com D.N.I.: 44379519

Oriana Rebeca Alba	Agustín Joaquín Alfaro
	
7° 1° AVC Tel.: 1136823909 Oriialba8@gmail.com D.N.I.:	7° 1° AVC Tel.: 1125458119 agusalfarol28@gmail.com D.N.I.: 44461061

## Horas dedicadas:



CABRERA, Nicolas Alan. 216 Horas

TORANCIO, Manuel Eduardo. 220 Horas

BALBI TREJO, Gabriel Andrés. 203 Horas

ALFARO, Agustín Joaquín. 207 Horas

ALBA PONCE, Oriana Rebeca. 202 Horas

## Índice:



<b>1. Introducción</b>	
1.1 Sustento del proyecto .....	5
1.2 Descripción general .....	5
1.3 Descripción técnica .....	5
1.4 Destino y alcance .....	5
1.5 Partes significativas del proyecto .....	6
1.6 Diagrama en bloques del proyecto .....	7
1.7 Resultado conseguido .....	7
<b>2. Software</b>	
2.1 Lenguaje de programación usado .....	8
2.2 Captura del código .....	8
2.3 Estructura de datos .....	11
<b>3. Sistema embebido</b>	
3.1 Microcontrolador usado .....	12
3.2 Placas de desarrollo .....	13
3.3 Softwares utilizados para el desarrollo .....	14
3.4 Lenguajes de programación usados .....	14
3.5 Especificación de periféricos utilizados .....	15
<b>4. Electrónica</b>	
4.1 Diagrama de bloques .....	17
4.2 Software usado para el desarrollo de esquemático y PCB .....	17
4.3 Esquemático de cada bloque .....	17
4.4 PCB de cada bloque .....	18
4.5 Modelado 3D de cada bloque .....	19
4.6 Especificaciones de fuente de alimentación y potencias .....	20
4.7 Especificaciones técnicas de cada componente utilizado, datasheet .....	21
<b>5. Estructura</b>	
5.1 Diagrama en bloques de las partes .....	31
5.2 Software de diseño utilizado .....	31
5.3 Descripción de cada parte de la estructura .....	32
5.4 Imágenes del bastón completo con cada parte de la estructura .....	33
<b>6. Anexo</b>	
6.1 Resultado de investigaciones hechas .....	35
6.2 Bibliografía .....	36

## Introducción



## Sustento del proyecto

STAC se sustenta en la ayuda que sabemos que puede prestar a esas más de 280 millones de personas son ciegas en el mundo, también hay que fijarse y entender a estas personas, ya que hoy en día si bien cuentan con muchos objetos para su ayuda, a veces esto se convierte en algo poco práctico o molesto. Con STAC no solo le brindamos una ayuda a personas con ceguera, sino también autonomía en sus ambientes más complicados, como lo pueden ser la calle o un lugar con dificultad para transitar, sumando su ergonomía y practicidad, permitiendo manejarse autónomamente a la persona con ceguera y dar una sensación de libertad propia.

## Descripción general

El bastón consta con un sensor, el cuál indicará si hay algún obstáculo frente al usuario. Y además contará con un motor el cual alertará al usuario en caso de haber algún obstáculo frente a él.

Este se cargará mediante un cable micro USB, ya que esta funcionará con baterías recargables de litio, las cuales se encargarán de proveer la energía necesaria.

## Descripción Técnica

“STAC” utilizará baterías recargables de litio de 3,7V c/u, las cuales estarán conectadas en serie para lograr unos 7.4V para alimentar todas las funciones del bastón, con su correspondiente regulador de 5v para no quemar el micro. Aparte este mismo contará con un transistor TIP42 el cual se encargará de entregar la potencia necesaria al motor como al sensor, ya que la salida del regulador no nos basta directamente. Todo esto para abastecer por ejemplo el sensor ultrasónico ya nombrado (Hc-sr04), el cual se utilizará para saber e informar, respecto si hay o no un objeto delante del usuario a unos 2 metros, mediante un motor.

## Destino y Alcance

¿A quién va destinado STAC? ¿Qué alcance tiene?

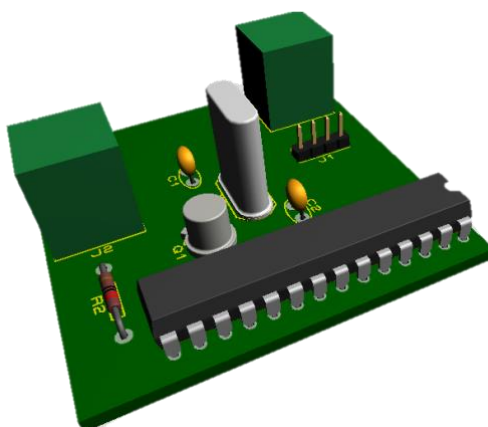
Cálculo que hasta esta altura ya se habrá notado que “STAC” es un proyecto que va abocado a personas con ceguera, esto además se ve reflejado en un entorno cotidiano bastante común, ya que hoy en día no es de extrañarse ver a una persona ciega en la calle, o transportes públicos, y al ser un problema tan común y con soluciones poco ortodoxas, ya que vimos con si bien hay soluciones que están muy buenas, muchas de ellas implican maniobrar o tener objetos que sean muy cargosos, y que terminan siendo capaz más una molestia que ayuda, por eso nosotros proponemos ayudar a las personas con ceguera de una manera fácil y sin causar problemas o molestias en su vida cotidiana.

Qué alcance tiene es una también de las preguntas, hasta dónde podría llegar, y podemos decir que llegaría hasta distintas personas con ceguera de diferentes ámbitos que podrían utilizar este mismo, ya que al ser un objeto práctico, ergonómico y accesible, es algo que a futuro podría verse bastante común, ya que si nos ponemos a pensar cada año va en aumento las personas con

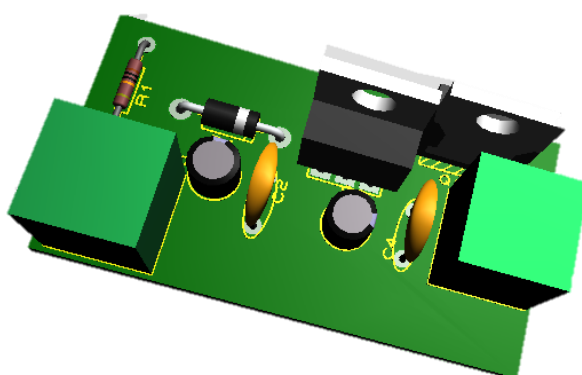
ceguera, por diferentes motivos, y las soluciones no son tantas como pueden llegar a parecer, es por eso que con esto ofrecemos una buena solución con este proyecto.

### Partes del proyecto

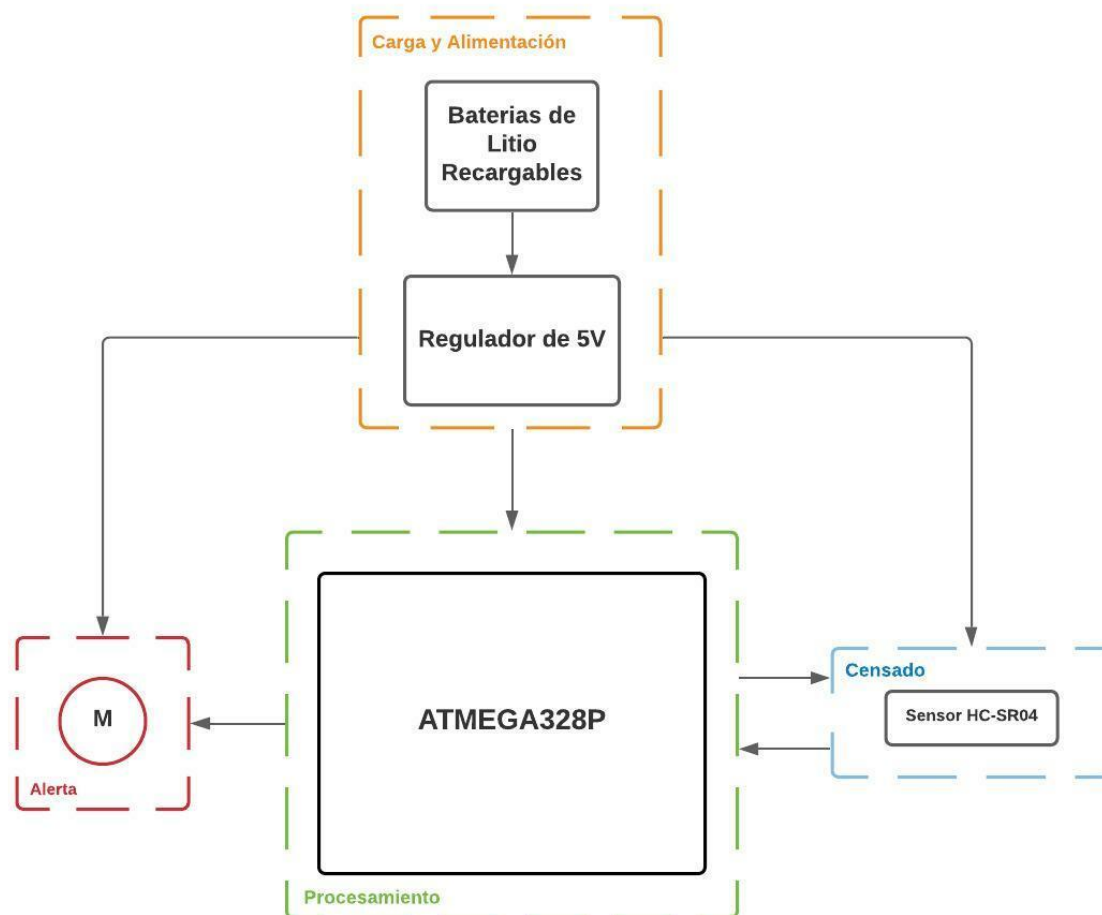
#### Placa del Bloque de procesamiento:



#### Placa del Bloque de alimentación:



### Diagrama en bloques del proyecto



## Resultado Conseguido

Los resultados que conseguimos en cuestión de este tiempo hasta ahora son todo el circuito de censado y la parte de la fuente o sea el regulador, porque en si la fuente son 2 baterías de litio, así que con el tema de PCB y esquemático de los mismo están finiquitados, en este mismo caso podemos hablar del diseño 3d del bastón, la verdad que hablando del proyecto en general se tuvieron mejores resultados de lo esperado en el tiempo usado y del tiempo que disponíamos. Así que hablando en rasgos generales del resultado acerca del proyecto nosotros podemos decir que es bueno, también esto cuenta con muchos errores y falla y prueba que estuvieron en el medio, por lo que se fue solucionando a medida que se fue avanzando.

## Software



## Lenguajes de programación usados

**Lenguaje C:** utilizamos este como lenguaje de programación principal por así llamarlo, ya que, con este, es con el cual se basó toda la programación del proyecto, ya sea del sensor como del motor, con este mismo también complementamos bien lo que teníamos planeado hacer, ya sea nuestro objetivo como expectativas, obviamente que hay que agregar que este es uno de los lenguajes más completos y con librerías más completos.



**Arduino:** se utilizó el lenguaje de programación Arduino que es basado en C++, para hacer pruebas al principio del proyecto, ya sea testeos, o mismo para probar el funcionamiento de determinados componentes u adentrarnos más en su funcionamiento y demás especificaciones, sin embargo, cabe destacar que se usó única y exclusivamente para esto, ya que como mencionamos anteriormente, nuestro lenguaje de programación principal y usado en el proyecto fue el C.



## Capturas de código y descripción

Habilitar interrupción en todo cambio del pin:

```
EIMSK |= (1<<INT0); // habilitar INT0
EICRA |= (1<<ISC00); // configurar el disparador de interrupcion en cualquier cambio
```

Con esta sección de código se habilita el uso de la interrupción externa INT0 en el puerto D2 del microcontrolador cuando pase cualquier cambio en ese pin gracias al Echo del Sensor Ultrasónico

Enviar pulsos de Trig cada 15us:



```
TRIG_PORT |= (1<<TRIG_PIN);  
_delay_us(15);           // activar el modulo ultrasonico durante 15us  
TRIG_PORT &= ~(1<<TRIG_PIN);
```

Se envían pulsos por el puerto PD3 cada 15us dirigidos al pin Trigger del Sensor Ultrasónico.

Cálculo de distancia:

```
distance = pulse_time * 0.0343 / 2;
```

Se realiza el cálculo de la distancia por medio de una formula proveída por el propio datasheet del Sensor Ultrasónico la cual es:

$$\text{Distancia} = \text{duración de pulso} * 0.0343 / 2$$

Condición de para el encendido del motor vibrador:

```
if (distance <= 200)  
{  
    PORTB |= (1<<PIND2);  
} else  
{  
    PORTB &=~ (1<<PIND2);  
}
```

En esta condición se estipulan los dos estados de motor según se cumpla la condición de que el obstáculo se encuentre a menos de 2 metros (Estado del motor encendido) o a más de 2 metros (Estado del motor apagado). Esto se hace mediante la definición del comportamiento del pin de control del motor PB2, el cual puede estar en ALTO o BAJO correspondientemente.

Llamado de Interrupción ISR en caso de cambio en el pin INTO (PD2):

```
// Este ISR se llama automaticamente cuando ocurre un cambio en el pin INT0 debido al cambio
// del pin de echo del modulo ultrasonico
ISR(INT0_vect)
{
    if ((ECHO_IN & (1<<ECHO_PIN)) == 0)
    {
        // El echo cambio de ALTO a BAJO
        TCCR1B = 0; // contador de desactivacion
        pulse_time = TCNT1; // la memoria de conteo se actualiza a un numero entero
        TCNT1 = 0; // resetear la memoria del contador
    } else {
        // echo cambio el cambio de BAJO a ALTO
        TCCR1B |= (1<<CS10); // habilitacion de contador
    }
}
```

Se llama a la ISR (Interrupt Service Routine) el cual como dice su nombre interrumpe el curso de ejecución actual de Procesador mientras cuando se cumpla un cambio interno o externo como en este caso. En esta condición se establece que cuando la lectura en el PD2 es que la señal del Echo pasa de ALTO a BAJO (interrupción activada por flanco de bajada) se realizan las siguientes acciones:

- Se desactiva el contador TCCR1B:  
$$\text{TCCR1B} = 0;$$
- Se guarda la duración del pulso como un entero:  
$$\text{pulse\_time} = \text{TCNT1};$$
- Se resetea la memoria del contador:  
$$\text{TCNT1} = 0;$$

La otra opción de la condición es que la señal de Echo pase de Bajo a ALTO y con esto se habilita nuevamente el contador TCCR1B:

$$\text{TCCR1B} \mid= (1<<\text{CS10});$$

Atmel Studio: es un Entorno de Desarrollo Integrado (IDE) para escribir y depurar el código del microcontrolador AVR/ARM en un PC con Windows. Atmel actúa como herramienta de gestión de proyectos, editor de archivos fuente, ensamblador, simulador y herramienta de programación y depuración en el chip. Fue utilizado por nosotros para hacer pequeñas correcciones de código en las pruebas debido a que lo utilizamos en las últimas instancias de la programación del proyecto.

## Estructura de datos:

<pre>#ifndef F_CPU #undef F_CPU #endif #define F_CPU 16000000u1</pre>	←	Definición de frecuencia de trabajo del microcontrolador
<pre>#define TRIG_PIN 3 #define ECHO_PIN 2</pre>	←	Definición de pines de señales Trig y Echo
<pre>#define TRIG_PORT PORTD #define TRIG_DDR DDRD #define ECHO_IN PIND #define ECHO_DDR DDRD</pre>	←	Definición de registros de puertos para señales Trig y Echo
<pre>#include &lt;avr/io.h&gt; #include &lt;avr/interrupt.h&gt; #include &lt;util/delay.h&gt; #include &lt;stdlib.h&gt; #include &lt;stdio.h&gt;</pre>	←	Inclusión de librerías

## Sistema embebido

### Microcontrolador usado:

#### Atmega328p



El microcontrolador utilizado es el Atmega328P en este caso, este mismo lo implementamos ya que al ser práctico en funcionalidad, de bajo consumo y bajo costo, fue lo que vimos más viable. A parte cabe aclarar que aquí se utilizaran diferentes entradas del Atmega que se aclararan más adelante, por ejemplo, unas de ellas es PB6, PB7, etc. Y respecto a su programación es algo que se explicara más adelante, pero se puede mencionar que se utilizó para programar tanto el sensor como el motor.

PARÁMETROS	VALORES
Flash	32 Kbytes
SRAM	2 Kbytes
Cantidad Pines	28
Frecuencia máxima de operación	20 MHz
CPU	852542-bit AVIR
Pines máximos de E/S	23
Interrupciones internas	24

Canales ADC	8
Resolución de ADC	10
Eeprom	1Kbytes
Canales PWM	6
Voltaje de operación	1.8-5.5 v
Timers	3

### Placas de desarrollo

**Regulador de 5v:** Utilizamos el regulador de 5v por lo ya mencionado, ya que las baterías proveen más de lo necesario, por lo cual nos vimos en la necesidad de utilizar este mismo regulador, además de que utilizaremos un transistor de potencia TIP42 ya que necesitamos desviar la mayor cantidad de corriente para que no se queme el regulador LM7805. Este mismo contará con estos componentes:

- 2 borneras
- 3 capacitores de 0,1 $\mu$ F
- 1 capacitor de 10 $\mu$ F
- 1 diodo Zener
- 1 LM7805
- 1 disipador
- 1 transistor TIP42

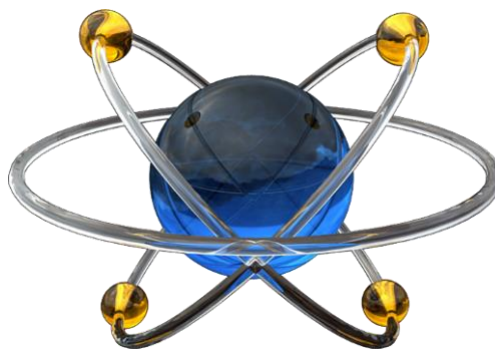
**Circuito del sensor:** Este sería el circuito utilizado para los 3 sensores y 3 motores, aparte de otros componentes, este mismo es la placa principal la cual contará con el atmega328p, este nos brindará la información acerca de la distancia del obstáculo mediante vibraciones de los motores vibradores y su sensor, obviamente que esto es gracias a la programación que se le aplicará al mismo. Este contará con:

- 3 motores vibradores

- 3 sensores ultrasónicos
- 1 cristal de 16MHz
- 2 capacitores
- 1 Atmega328p

### Softwares utilizados:

**Proteus:** Este fue utilizado para hacer la plaqueta del sensor, que vendría a ser la principal, aunque en el proceso se utilizaron varias versiones del mismo, desde la 8.9 a la 8.12, por simple razón que cada máquina contaba con una versión diferente del programa, aunque esto no nos limitó a la hora de usar herramientas del mismo. Este fue usado por conveniencia en cuanto a la facilidad que se logra obtener ciertas herramientas y componentes.



**WinAVR:** Es una suite compuesta de herramientas de desarrollo opensource para los microcontroladores AVR sobre plataforma Windows. Incluye el compilador GNU GCC para C y C++. Aunque nuestro uso de este fue solo como un compilador de C.



### Lenguajes de programación usados:

**C:** Utilizamos este como lenguaje de programación principal por así llamarlo, ya que con este, es con el cual se basó toda la programación del proyecto, ya sea de los sensores como de los motores, con este mismo también complementamos bien lo que teníamos planeado hacer, ya sea nuestro objetivo como expectativas, obviamente que hay que agregar que este es uno de los lenguajes más completos y con librerías más completos.



**Arduino:** Se utilizó el lenguaje de programación Arduino que es basado en c++, para hacer pruebas al principio del proyecto, ya sea testeos, o mismo para probar el funcionamiento de determinados componentes o mismo adentrarnos más en su funcionamiento y demás especificaciones, sin embargo cabe destacar que se usó única y exclusivamente para esto, ya que como mencionamos anteriormente, nuestro lenguaje de programación principal y usado en el proyecto fue el C.



### Periféricos utilizados

En el caso del Atmega 328p los pines que se utilizaron fueron el PD2, PD3, PB2, PB6 y PB7. A continuación especificaremos que hace cada pin en concreto de los utilizados:

PD2: INT0 (entrada de interrupción externa 0)/PCINT18 (interrupción de cambio de pin 18). El pin PD2 puede servir como fuente de interrupción externa.

PD3: INT1 (entrada de interrupción externa 1)/OC2B (salida del temporizador / contador 2 comparar salida B)/PCINT19 (interrupción de cambio de pin 19).

OC2B, salida de comparación de coincidencia: el pin PD3 puede servir como salida externa para la comparación de Timer / Counter0

El pin PD3 debe configurarse como una salida (DDD3 establecido (uno)) para cumplir esta función. El pin OC2B también es el pin de salida para la función de temporizador del modo PWM.



PB2: SS (selección de esclavo maestro de bus SPI) / OC1B (Salida de Timer / Contador1 Comparar Salida de Match B) / PCINT2 (Interrupción de cambio de pin 2). SS: Entrada de selección de esclavo. Cuando el SPI está habilitado como esclavo, este pin se configura como una entrada independientemente de la configuración de DDB2. Como esclavo, el SPI se activa cuando este pin se baja. Cuando el SPI está habilitado como maestro, la dirección de datos de este pin es controlada por DDB2. Cuando el pin es forzado por el SPI a ser una entrada, el bit PORTB2 aún puede controlar el pull-up.

OC1B, salida de comparación de salida: el pin PB2 puede servir como salida externa para el temporizador / contador1 Compare Match B. El pin PB2 debe configurarse como una salida (conjunto DDB2 (uno)) para cumplir esta función. El pin OC1B también es el pin de salida para la función de temporizador del modo PWM.

PCINT2: Fuente de interrupción de cambio de pin 2. El pin PB2 puede servir como una fuente de interrupción externa.

PB6/7: El puerto B es un puerto de E / S bidireccional de 8 bits con resistencias pull-up internas (seleccionadas para cada bit). Los búferes de salida del puerto B tienen características de unidad simétricas con capacidad de fuente y de sumidero alta. Como entradas, los pines del puerto B que se tiran hacia abajo externamente generarán corriente si se activan las resistencias pull-up. Los pines del Puerto B tienen tres estados cuando se activa una condición de reinicio, incluso si el reloj no está funcionando.

Dependiendo de la configuración del fusible de selección del reloj, PB6 se puede utilizar como entrada al amplificador del oscilador inversor y como entrada al circuito operativo del reloj interno.

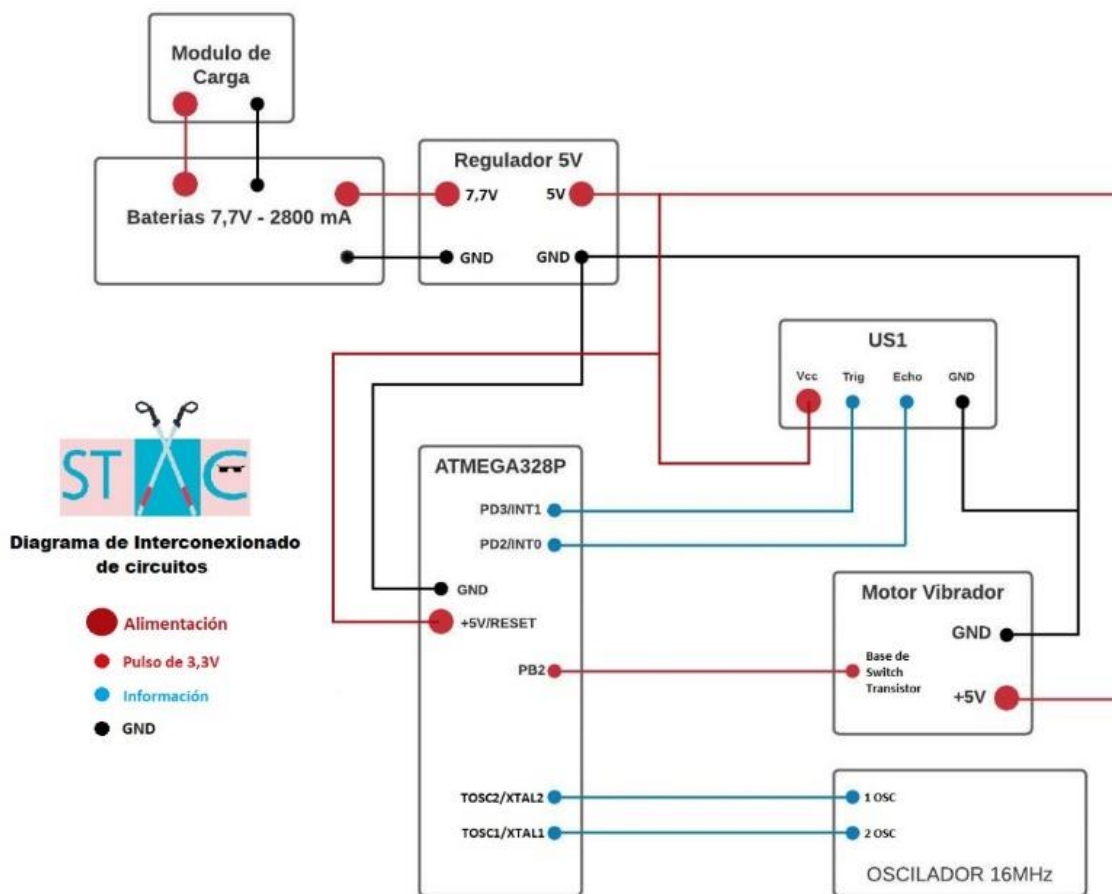
Dependiendo de la configuración del fusible de selección del reloj, PB7 se puede utilizar como salida del amplificador del oscilador inversor.

Si el oscilador RC calibrado interno se usa como fuente de reloj de chip, PB7.6 se usa como entrada TOSC2..1 para el temporizador / contador asíncrono2 si el bit AS2 en ASSR está establecido.



## Electrónica

### Diagrama en bloques:



### Software utilizado:

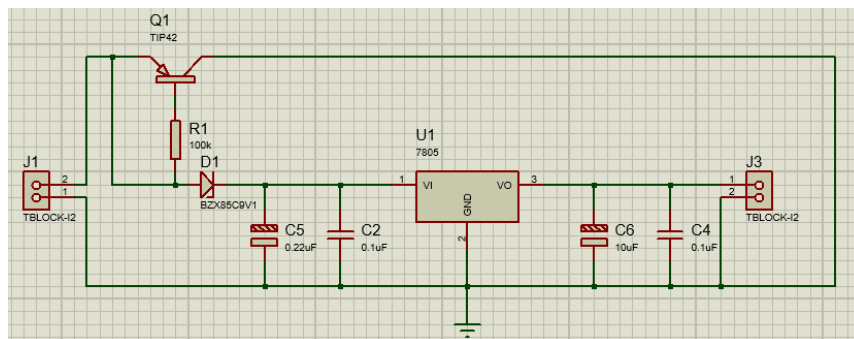
El software utilizado tanto para el desarrollo del esquemático como para el del PCB es:

Proteus:

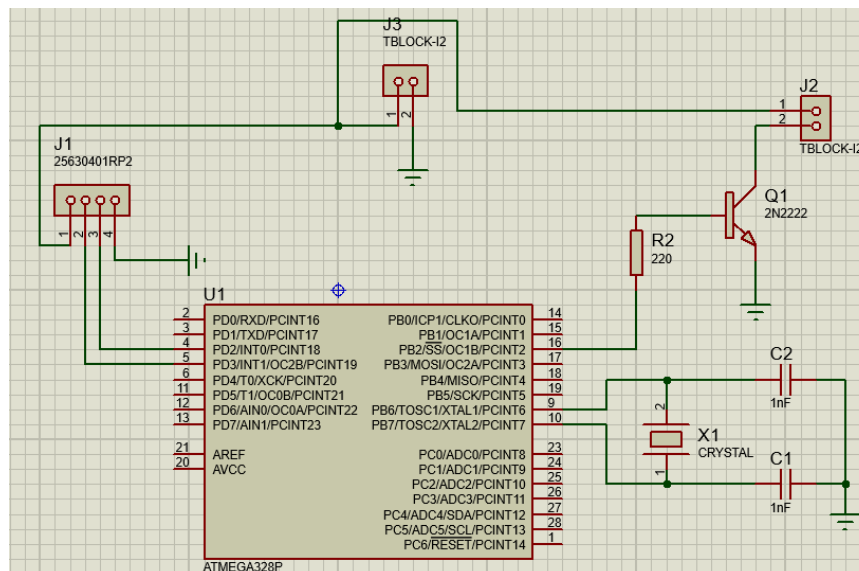
Se utilizó este software ya que nos sentíamos más familiarizados con este que con otros, además contando con la practicidad que nos ofrece este y sus herramientas, la versión utilizada fue la 8.12, además de que este software ya de por sí nos provee tanto la simulación, el modelo 3d de la placa, como el esquemático y el PCB, los cuales ayudan a una mayor rentabilidad a la hora de usarlo.

## Esquemáticos:

### Esquemático del regulador de 5v

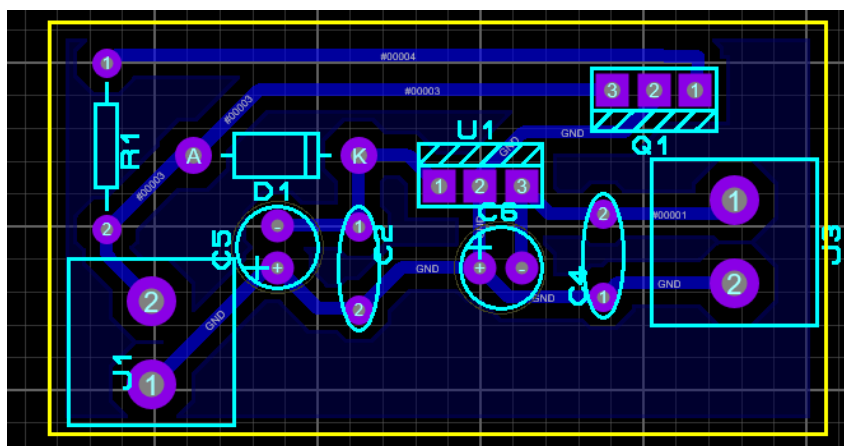


### Esquemático del sensor:

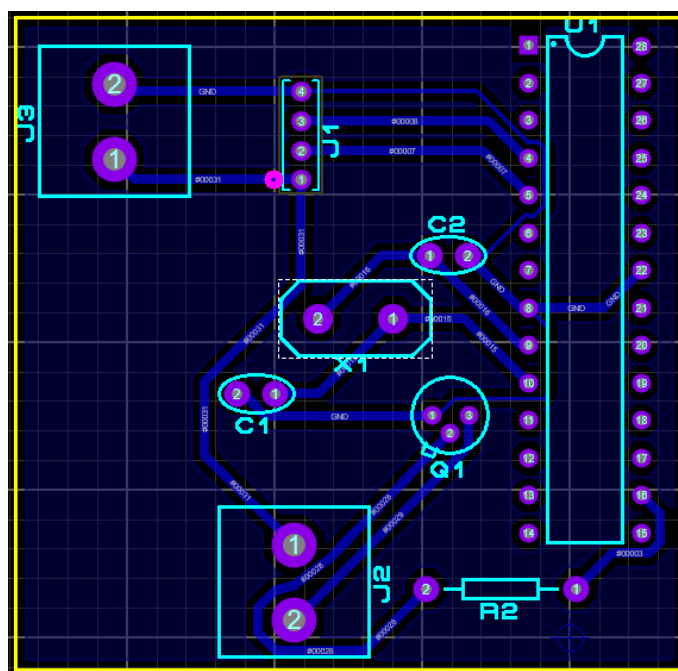


## PCB de cada bloque

PCB del regulador de 5v:

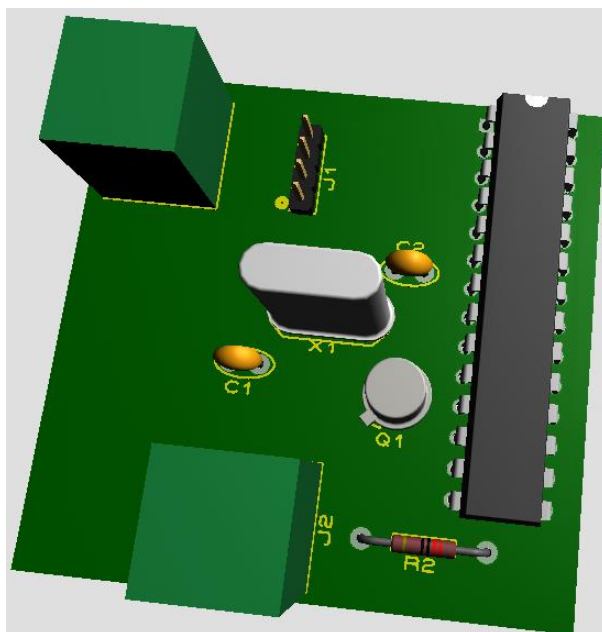


PCB del sensor:

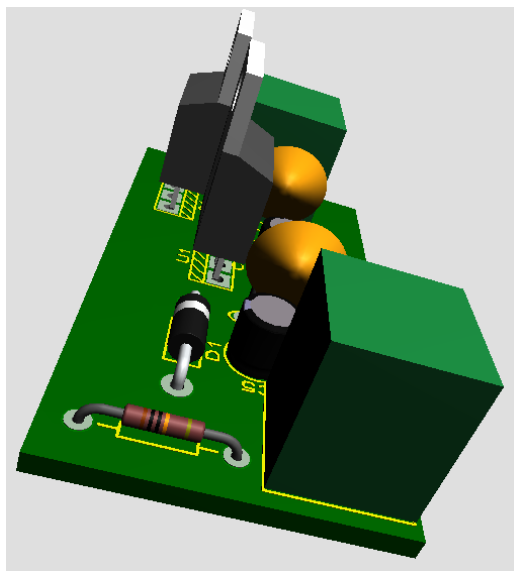


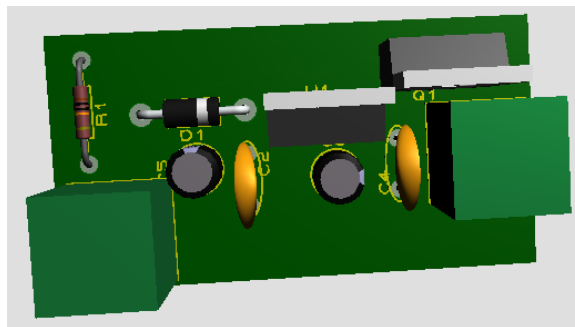
## Modelo 3D de cada PCB

Modelado 3D Sensor:



Modelado 3D del regulador de 5v:





## Fuente de alimentación

En nuestro caso para alimentar nuestro proyecto, utilizamos 2 baterías recargables de litio de 3,7v y 2800mA, que nos parecieron más que suficiente para lo que es STAC, aunque cabe aclarar que estas mismas baterías irían conectadas directamente a un regulador de voltaje de 5 volts, ya que tanto el amperaje como el voltaje que se maneja es superior al requerido, por lo que bastaría con lo que nos proporcionan, aunque tuvimos que agregarle un transistor TIP42 ya que si bien estos 2 son suficientes no pasa lo mismo con la potencia, por lo que tuvimos que desviar parte de la potencia de estas baterías, para alimentar directamente al sensor y al motor por separado del Atmega.

Por último, aclarar que la potencia total que manejamos en el bastón son la potencia del regulador unos 7,56W, más la potencia del motor 500mW y por último la del sensor 75mW. Dándonos un total de 8,135W necesarios, aclaremos mejor que si separamos las potencias a la salida del regulador, fue porque el Atmega como tolera una baja corriente entonces la potencia que genera es poco y no es la requerida por nosotros, por lo cual se utilizó el TIP42 para el uso optimo del sensor y motor. A continuación, se dejarán los cálculos que sacamos para averiguar la potencia total del circuito y de cada componente en específico:

PS (potencia del sensor):  $5v \cdot 15mA = 75mW$

PM (potencia del motor):  $5v \cdot 100mA = 500mW$

PR (Potencia del regulador):  $2,7V \cdot 2800mA = 7,56W$

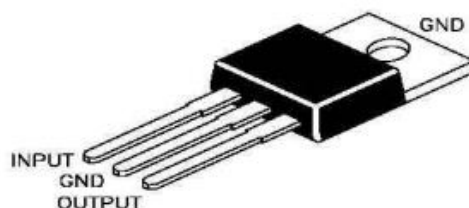
$P_{total} = PR + PM + PS = 7,56W + 500mW + 75mW = 8,135W$

## Especificación técnica sobre cada componente usado

Regulador de 5v

**LM7805:** Este se utilizará para poder regular nuestros 7,4V de las baterías a 5V, así que nada por eso se implementó en el circuito, contando con sus debidos capacitores, diodo, borneras y transistor.

Pines de este



Absolute maximum Ratings:

DESCRIPTION	SYMBOL	VALUE	UNIT
Input Voltage	$V_{IN}$	35	V
Continuous Total Dissipation at $T_a=25^{\circ}\text{C}$ free air Temperature	$P_D$	2.0	W
Continuous Total Dissipation at $T_c=25^{\circ}\text{C}$ case Temperature	$P_D$	15	W
Operating free-air, case, or Virtual Junction Temperature Range	$T_{OPR}$	0 to 150	$^{\circ}\text{C}$
Storage Temperature Range	$T_{slg}$	- 65 to +150	$^{\circ}\text{C}$
Lead Temperature 1.6mm (1/16 inch) from Case for 10 seconds	$T_L$	260	$^{\circ}\text{C}$

s

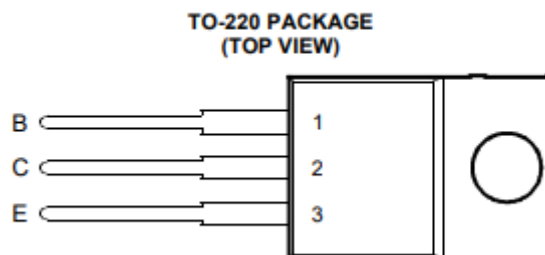
Especificaciones eléctricas ( $T_j = 25^{\circ}\text{C}$  a menos que se especifique lo contrario):

$V_I=10\text{V}$ ,  $I_O=500\text{mA}$

DESCRIPTION	SYMBOL	TEST CONDITION	MIN	TYP	MAX	UNIT
Output Voltage	$V_O$	$T_J=25^{\circ}\text{C}$	4.80		5.20	V
		$I_O=5\text{mA} \sim 1\text{A}$ $V_I=7\text{V} \sim 20\text{V}, P \leq 15\text{W}$ $T_J=0 \sim 125^{\circ}\text{C}$	4.75		5.25	V
Line Regulation	$R_{\text{REGV}}$	$V_I=7.0 \sim 25\text{V}$ $T_J=25^{\circ}\text{C}$			100	mV
		$V_I=8.0 \sim 12\text{V}$			50	mV
Ripple Rejection	$R_R$	$V_I=8.0 \sim 18\text{V}, f=120\text{Hz}$ $T_J=0 \sim 125^{\circ}\text{C}$	62			dB
Load Regulation	$R_{\text{EGL}}$	$I_O=5\text{mA} \sim 1.5\text{A}$ $T_J=25^{\circ}\text{C}$			100	mV
		$I_O=250\text{mA} \sim 750\text{mA}$			50	mV
Output Resistance	$R_O$	$f=1\text{KHz}$ $T_J=0 \sim 125^{\circ}\text{C}$		0.017		$\Omega$
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O=5\text{mA}$ $T_J=0 \sim 125^{\circ}\text{C}$		- 1.1		mV/ $^{\circ}\text{C}$
Output Noise Voltage	$V_{\text{NO}}$	$f=10\text{Hz} \sim 100\text{KHz}$ $T_J=25^{\circ}\text{C}$		40		$\mu\text{V}$
Dropout Voltage	$V_d$	$I_O=1\text{A}$ $T_J=25^{\circ}\text{C}$		2.0		V
Quiescent Current	$I_Q$	$T_J=25^{\circ}\text{C}$			8.0	mA
Quiescent Current Change	$\Delta I_Q$	$V_I=7.0 \sim 25\text{V}$ $T_J=0 \sim 125^{\circ}\text{C}$			1.3	mA
		$I_O=5\text{mA} \sim 1\text{A}$			0.5	mA
Short Circuit Output Current	$I_{\text{SC}}$	$T_J=25^{\circ}\text{C}$		750		mA
Peak Output Current	$I_{\text{PK}}$	$T_J=25^{\circ}\text{C}$		2.2		A

**Transistor TIP42:** Este mismo se utilizará ya que nosotros al necesitar más corriente de lo que podría llegar a aguantar el regulador elegimos usar este para transportar además de más corriente al sensor y motor, también brindar una mayor potencia a los ya nombrados.

Pines de este:



absolute maximum ratings at 25°C case temperature (unless otherwise noted)

RATING		SYMBOL	VALUE	UNIT
Collector-base voltage ( $I_E = 0$ )	TIP42	$V_{CBO}$	-80	V
	TIP42A		-100	
	TIP42B		-120	
	TIP42C		-140	
Collector-emitter voltage ( $I_B = 0$ )	TIP42	$V_{CEO}$	-40	V
	TIP42A		-60	
	TIP42B		-80	
	TIP42C		-100	
Emitter-base voltage		$V_{EBO}$	-5	V
Continuous collector current		$I_C$	-6	A
Peak collector current (see Note 1)		$I_{CM}$	-10	A
Continuous base current		$I_B$	-3	A
Continuous device dissipation at (or below) 25°C case temperature (see Note 2)		$P_{tot}$	65	W
Continuous device dissipation at (or below) 25°C free air temperature (see Note 3)		$P_{tot}$	2	W
Unclamped inductive load energy (see Note 4)		$\frac{1}{2}LI_C^2$	62.5	mJ
Operating junction temperature range		$T_j$	-65 to +150	°C
Storage temperature range		$T_{stg}$	-65 to +150	°C
Lead temperature 3.2 mm from case for 10 seconds		$T_L$	250	°C

Características eléctricas a una temperatura de la caja de 25 ° C:

PARAMETER	TEST CONDITIONS		MIN	TYP	MAX	UNIT
$V_{(BR)CEO}$ Collector-emitter breakdown voltage	$I_C = -30$ mA (see Note 5)	$I_B = 0$ TIP42 TIP42A TIP42B TIP42C	-40 -60 -80 -100			V
$I_{CES}$ Collector-emitter cut-off current	$V_{CE} = -80$ V $V_{CE} = -100$ V $V_{CE} = -120$ V $V_{CE} = -140$ V	$V_{BE} = 0$ $V_{BE} = 0$ $V_{BE} = 0$ $V_{BE} = 0$ TIP42 TIP42A TIP42B TIP42C			-0.4 -0.4 -0.4 -0.4	mA
$I_{CEO}$ Collector cut-off current	$V_{CE} = -30$ V $V_{CE} = -60$ V	$I_B = 0$ $I_B = 0$ TIP42/42A TIP42B/42C			-0.7 -0.7	mA
$I_{EBO}$ Emitter cut-off current	$V_{EB} = -5$ V	$I_C = 0$			-1	mA
$h_{FE}$ Forward current transfer ratio	$V_{CE} = -4$ V $V_{CE} = -4$ V	$I_C = -0.3$ A $I_C = -3$ A (see Notes 5 and 6)	30 15		75	
$V_{CE(sat)}$ Collector-emitter saturation voltage	$I_B = -0.6$ A	$I_C = -6$ A (see Notes 5 and 6)			-1.5	V
$V_{BE}$ Base-emitter voltage	$V_{CE} = -4$ V	$I_C = -6$ A (see Notes 5 and 6)			-2	V
$h_{fe}$ Small signal forward current transfer ratio	$V_{CE} = -10$ V	$I_C = -0.5$ A $f = 1$ kHz	20			
$ h_{fe} $ Small signal forward current transfer ratio	$V_{CE} = -10$ V	$I_C = -0.5$ A $f = 1$ MHz	3			

Características térmicas:

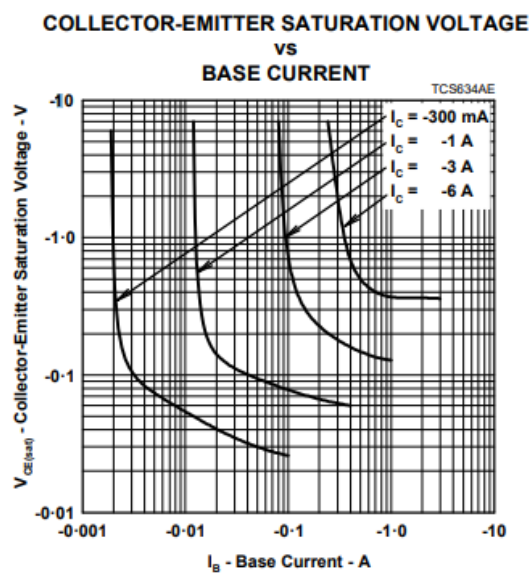
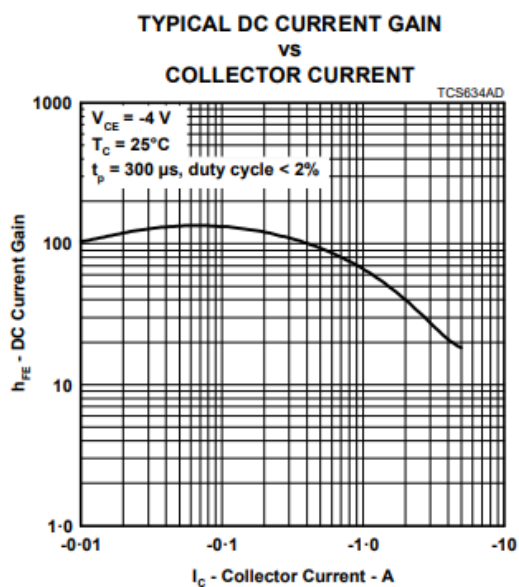
PARAMETER	MIN	TYP	MAX	UNIT
$R_{\theta JC}$ Junction to case thermal resistance			1.92	°C/W
$R_{\theta JA}$ Junction to free air thermal resistance			62.5	°C/W

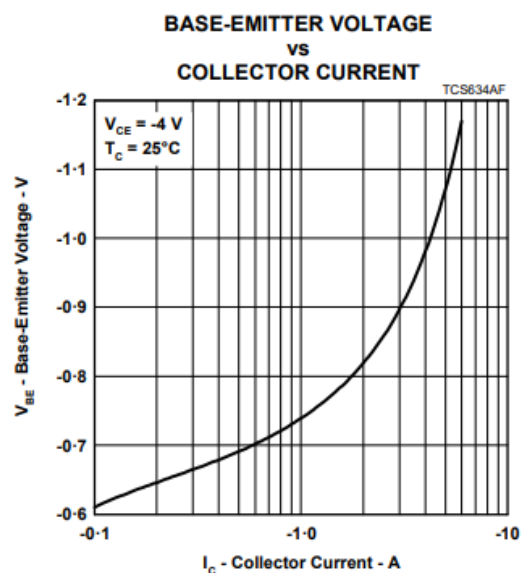
Resistive-load-switching characteristics at 25°C case temperatura:



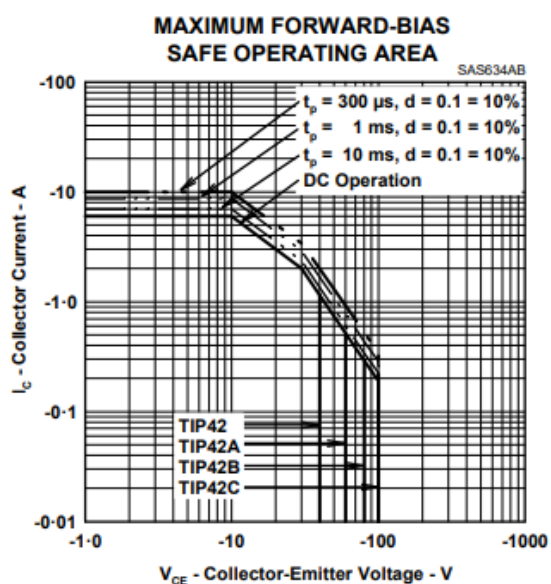
PARAMETER	TEST CONDITIONS <sup>†</sup>			MIN	TYP	MAX	UNIT
$t_{on}$ Turn-on time	$I_C = -6\text{ A}$	$I_{B(on)} = -0.6\text{ A}$	$I_{B(off)} = 0.6\text{ A}$		0.4		$\mu\text{s}$
$t_{off}$ Turn-off time	$V_{BE(off)} = 4\text{ V}$	$R_L = 5\ \Omega$	$t_p = 20\ \mu\text{s}$ , $dc \leq 2\%$		0.7		$\mu\text{s}$

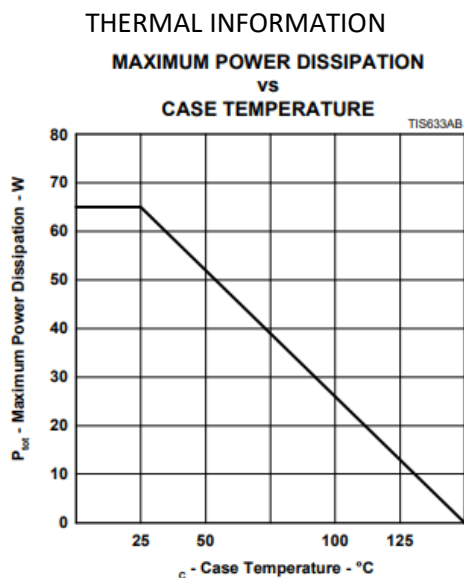
Typical characteristics:





Regiones operativas máximas seguras:





**2 Baterías recargables:** Su función ya explicada va a ser únicamente brindarle la alimentación al bastón necesaria y que se necesite, lo bueno es que al ser estas recargables ayudan mucho al medio ambiente y a la facilidad de cargarlas de pronto si se necesita utilizar al otro día y no tener que comprar unas baterías o pilas nuevas a último momento.



La Batería 18650 28EA es una batería recargable de Litio con una capacidad de 2800mAh, polo positivo plano, voltaje de salida de 3.65V y capacidad máxima de descarga de 35 Amp.

Especificaciones de las baterías:

- Modelo: 18650
- Corriente máxima de descarga: 25A
- Potencia: 2800 mAh 3.65V
- Recargable: Sí
- Celdas: Li-ion
- Largo: 6,5 cm



- Diámetro: 18 mm
- Peso: 35 g
- Tipo de pilas/baterías: 18650, Li-Ion, Recargable

### Parte del sensor:

**HC-SR04:** Este mismo lo utilizaremos para saber la distancia a la que se encuentra el sensor del objeto, esto nos dará una proximidad acerca del objeto en cuestión, lo cual nos ayudará para brindarle la información a la persona no vidente mediante la vibración en el mango del bastón.

### Especificaciones:

- Fuente de alimentación: + 5 V CC
- Corriente de reposo: <2 mA
- Corriente de trabajo: 15 mA
- Ángulo efectivo: <15 °
- Distancia de alcance: 2 cm - 400 cm / 1 "13 pies
- Resolución: 0,3 cm
- Ángulo de medición: 30 grados
- Ancho de pulso de entrada del Trigger: 10uS
- Dimensión: 45 mm x 20 mm x 15 mm

### Limitaciones y especificaciones:

Parameter	Min	Typ.	Max	Unit
Operating Voltage	4.50	5.0	5.5	V
Quiescent Current	1.5	2	2.5	mA
Working Current	10	15	20	mA
Ultrasonic Frequency	-	40	-	kHz

**Motor vibrador:** Este pequeño componente es un mini motor que, al momento de ser conectado, causa un efecto vibratorio. Es muy fácil de usar, solo hay que alimentarlo con un voltaje de 2 a 5 V, el cable rojo del motor vibrador es positivo y el cable negro es negativo, entre mayor sea el voltaje de alimentación el consumo de corriente será más alto, así como también la velocidad de vibración.

### Especificaciones

- Voltaje de alimentación mínimo: 2 V
- Voltaje de alimentación máxima: 5 V



- Consumo a 2 V: 40 mA
- Corriente a 3 V: 60 mA,
- Consumo a 4 V: 80 mA
- Consumo a 5 V: 100 mA
- Velocidad nominal del rotor: 11,000 RPM
- Dirección de rotación: Sentido horario y antihorario
- Posición del motor: Cualquier posición disponible
- Temperatura de operación mínima: -20 °C
- Temperatura de operación máxima: 70 °C

**Atmega328p:** El microcontrolador ATmega328P fabricado por ATMEL(ahora parte de Microchip®) pertenece a la familia de microcontroladores AVR® de arquitectura RISC 8-bit. Este además, posee características como, 32 KB de memoria FLASH con la capacidad de leer-mientras-escribe, 1 KB de memoria EEPROM, 2 KB de SRAM, 23 líneas de E/S de propósito general, 32 registros de proceso general, tres temporizadores flexibles/contadores con modo de comparación, interrupciones internas y externas, programador de modo USART, una interfaz serial orientada a byte de 2 cables, SPI e I2C, 6 canales 10-bit Conversor A/D, "watchdog timer" programable con oscilador interno, y cinco modos de ahorro de energía seleccionables por software.

#### **Peripheral Features:**

- Dos temporizadores / contadores de 8 bits con preescalador y modo de comparación independientes
  - Un temporizador / contador de 16 bits con preescalador, modo de comparación y modo de captura independientes
  - Contador en tiempo real con oscilador independiente
  - Seis canales PWM
  - ADC de 8 canales y 10 bits en paquetes TQFP y QFN / MLF Medición de temperatura
  - ADC de 6 canales y 10 bits en la medición de temperatura del paquete PDIP
  - USART serie programable
  - Interfaz serie SPI maestro / esclavo
  - Interfaz serial de 2 cables orientada a bytes (compatible con Philips I2C)
  - Temporizador de vigilancia programable con oscilador integrado en el chip
  - Comparador analógico en chip
  - Interrupción y activación al cambiar de pin
- I/O:
- 23 líneas I/O programables
- Voltaje de operación:
- 1.8 - 5.5V para ATmega328P
- Rango de temperatura:
- -40°C to 85°C



Grado de velocidad:

– ATmega328P: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V

### Configuración de pines

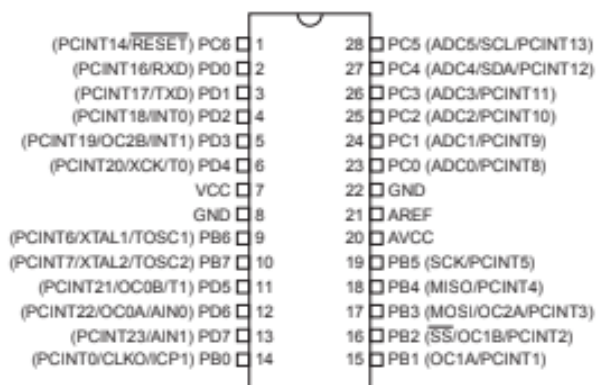
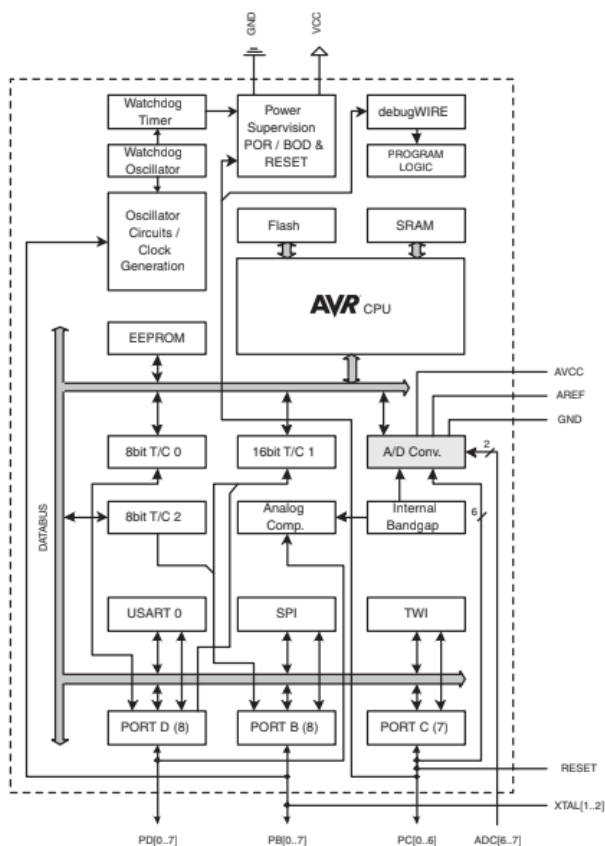
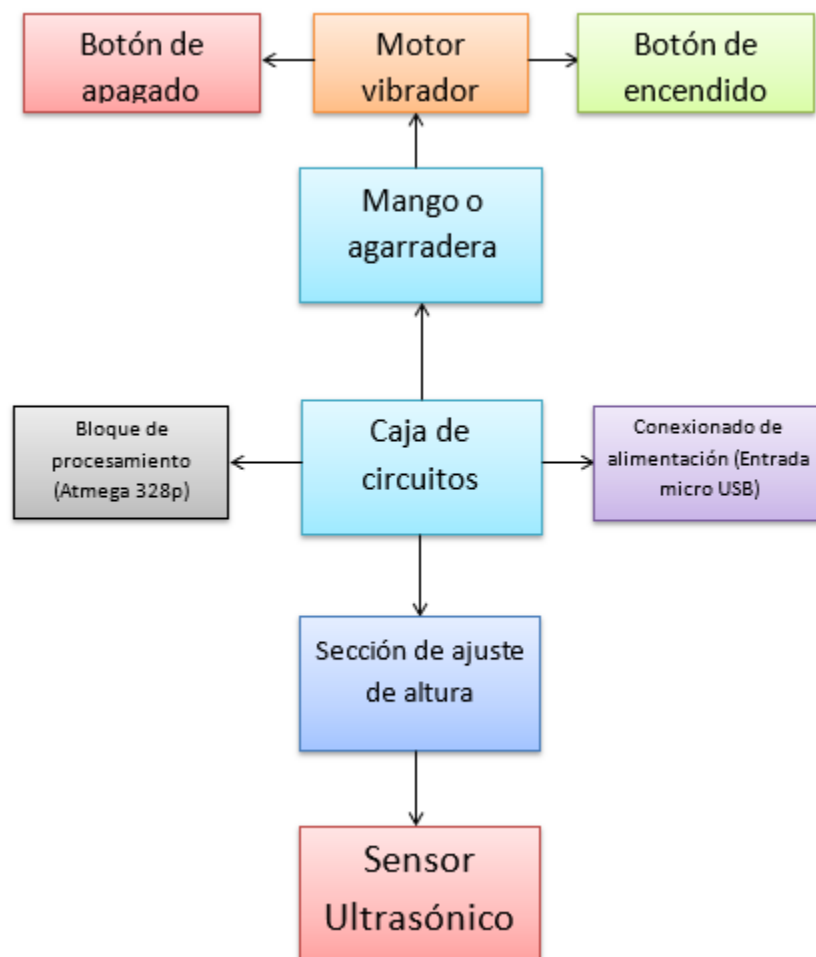


Diagrama de bloques:



## Estructura

### Diagrama de la estructura



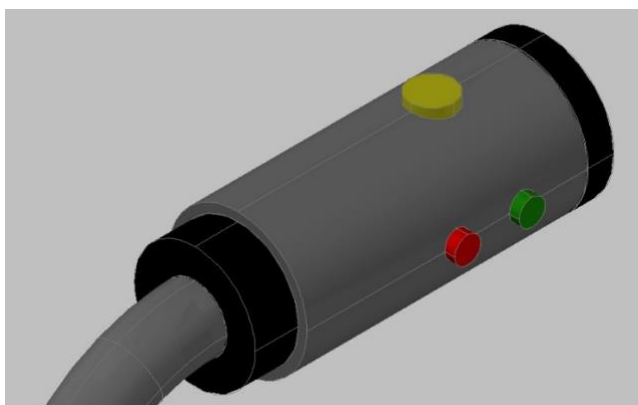
### Software utilizado:

**AutoCAD:** Este es un software de diseño asistido por computadora utilizado para dibujo 2D y modelado 3D. Actualmente es desarrollado y comercializado por la empresa Autodesk.

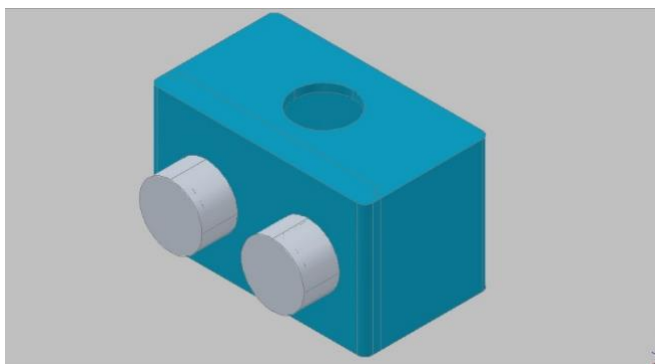
AutoCAD es un software reconocido a nivel internacional por sus amplias capacidades de edición, que hacen posible el dibujo digital de planos de edificios o la recreación de imágenes en 3D; es uno de los programas más usados por arquitectos, ingenieros, diseñadores industriales y otros. Este mismo como ya se da a ver lo utilizamos para hacer el bastón en 3d el cual fuimos armando las partes de este mismo, juntando estas y dando como resultado el bastón final.

## Descripción de sus partes

**Mango del bastón:** El bastón constara en su mango con 2 botones uno para apagar el sistema de censado del bastón con todo lo que conlleva esto obvio, y el botón de encendido que prendera el sistema. Esto se hizo por la simple razón de un ahorro de energía del mismo. Además, constara con un motor vibrador (el amarillo) como se aprecia en la imagen próxima, este será el encargado de avisar por medio de una vibración en el mango si hay un obstáculo.

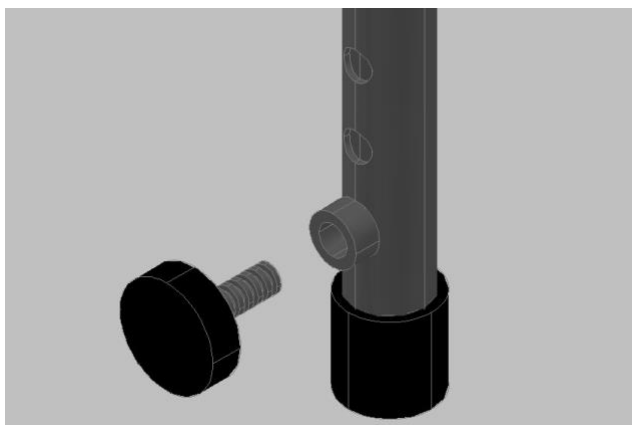


**Caja del sensor:** Esta caja se encontrará más abajo del bastón, y bueno como mismo se menciona es donde se encuentra el sensor ultrasónico que avisara a la persona con ceguera si hay un obstáculo en su camino, la posición del mismo es más dada por un tema de visibilidad de los objetos, ya que por ejemplo hay objetos demasiado altos que no hace falta censar, ya que no se vería como un obstáculo peligroso o alarmante para este, lo mismo en el caso de haber un obstáculo demasiado abajo.

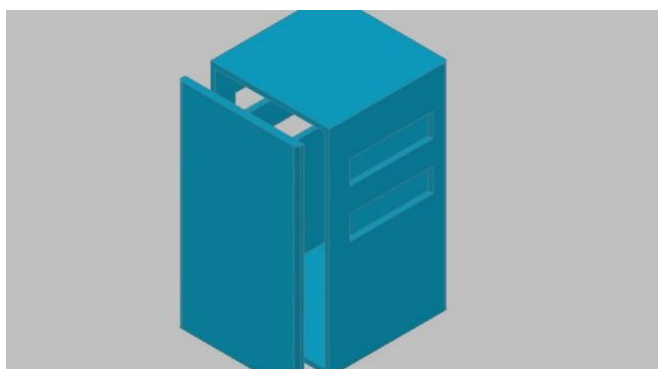


**Sección ajustable:** Esto servirá más que nada para justar el largo del bastón de acuerdo a la comodidad de la persona, es por eso que posee un tornillo y uno la ira posicionando según la comodidad o deseo de cada uno





**Caja de los circuitos:** En esta parte iría ubicado todos los circuitos que tiene el bastón, ya sea el regulador, las baterías, placa del sensor. Sirve como el almacenamiento de estos, y no cumple más otra función, y lo ubicamos en una parte elevada para que tampoco sea de mucho estorbo para el sensor o genere alguna incomodidad a la hora de caminar con el bastón.



**Goma de apoyo del bastón:** Esto se aplica más que nada para tener un mejor apoyo a la hora de querer dejar el bastón en algún sitio, ya que a veces querer apoyar el bastón en algún lugar se vuelve dificultoso, ya sea porque no hay buen apoyo, terminando en la caída de este, es por esto que se puso esto para un mejor agarre con el suelo a la hora de apoyarlo en cualquier sitio.

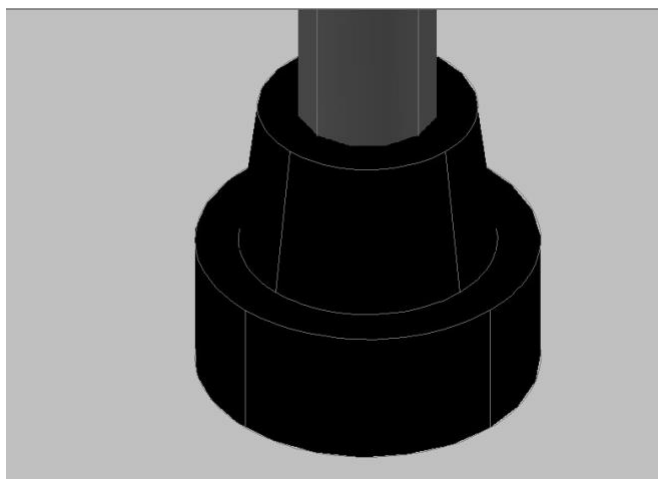
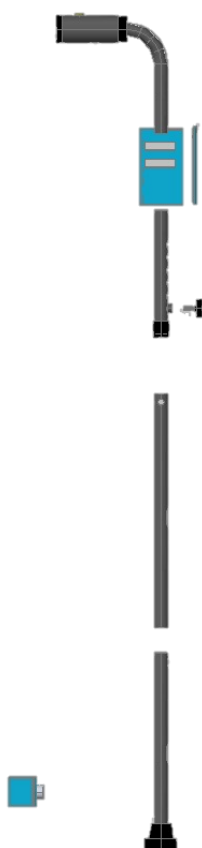
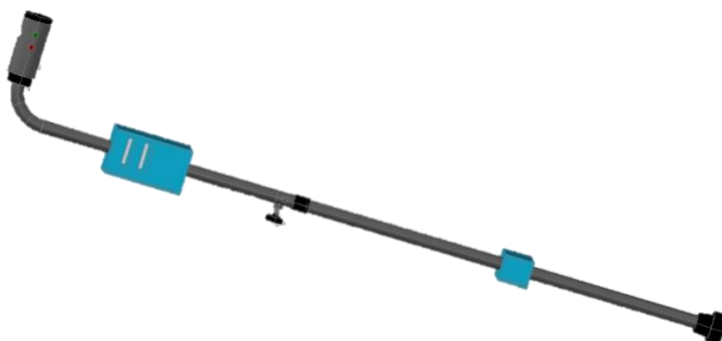


Imagen del bastón completo







## Anexo

### Resultados de la investigación

Hemos llegado con todo esto, a que si bien gran parte de la población es afectada, no hay mucha preocupación por ellos, y esto se puede ver en diferentes ámbitos, tanto laborales como cotidianos, y es por esto que decidimos realizar “STAC”, además de que no solo buscamos ayudar a las personas con ceguera, sino también planteamos una forma de brindarles comodidad y seguridad, porque también estuvimos investigando sobre que distancia sería mejor que detecte, que altura, la forma en que la persona prefiere comunicarse con el bastón, y tratamos de hacerlo lo más discreto posible en cuanto a su comunicación con el bastón, para que así no genere tampoco mucho ruido, o que en ambientes con mucha cantidad de ruido, se pueda identificar el obstáculo. Aparte de también ver su tiempo de autonomía para que no se descargue rápidamente o no llegue a durar lo suficiente o lo que uno necesita que dure en el lugar utilizado.



## Bibliografía:

### Electronica:

Datasheet del Hc-Sr04: <https://datasheetspdf.com/pdf-file/1291829/Cytron/HC-SR04/1>

Datasheet del TIP42: <http://www.datasheet.es/PDF/323658/TIP42-pdf.html>

Datasheet del LM7805: <http://j5d2v7d7.stackpathcdn.com/wp-content/uploads/2015/10/lm7805.pdf>

Datasheet del Atmega:  
<https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf>

Especificaciones del motor: <https://candy-ho.com/producto/motor-vibrador-celular-redondo-10mm-x-2-5mm-3v-arduino/>

Especificaciones de la batería: <https://www.todomicro.com.ar/investigacion-desarrollo-y-prototipado/1050-bateria-recargable-18650-2800mah.html>

### Software:

C: [https://es.wikipedia.org/wiki/C\\_\(lenguaje\\_de\\_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n))

Arduino: <https://www.arduino.cc/>

### Sistema embebido:

Proteus: <https://www.labcenter.com/>

WinAVR: <https://sourceforge.net/projects/winavr/>