



Master Universitario en Ingeniería de Sistemas Electrónicos.

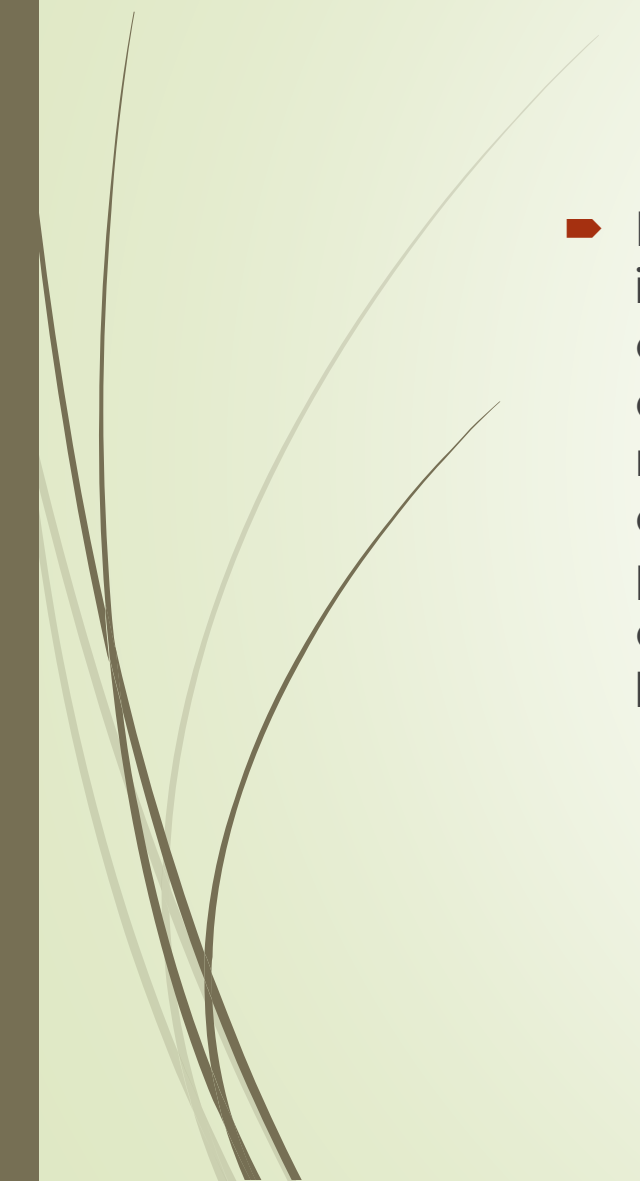
Sistema bajo coste basado en FPGA para la adquisición de datos de sensores con monitorización inalámbrica sobre dispositivos móviles

TUTOR: RICARDO COLOM PALERO.

ALUMNO: IGNACIO M. BLASCO LAHOZ.



INTRODUCCION.

- En el mundo de hoy en día se está extendiendo el uso de comunicaciones inalámbricas para todo tipo de usos. Esto usos, se pueden ver diariamente cuando vas a correr y tu pulsera cuenta pasos comunica a tu móvil la evolución de tu deporte diario o de una manera más profesional en la monitorización del regadío de una plantación agraria. Además, de este auge de las comunicaciones inalámbricas, también evolucionando positivamente el desarrollo de sistema integrados donde con un mismo dispositivo se llega a desarrollar o diseñar casi todo el sistema, tales como los sistemas basados en fpga.
- 



OBJETIVOS.

- Comunicar sistemas inteligentes, mediante tecnologías inalámbricas, a sistemas de monitorización.
- La realización de un TFM, con implementación hardware utilizando recursos al alcance de cualquiera por un precio bajo. Sin recurrir al uso de equipos de laboratorio costosos, a los cuales no todo el mundo tiene acceso. Esto entraría dentro de la filosofía americana, de las grandes empresas que surgieron en garajes con pocos recursos.

APLICACIONES.

- Domótica / Home Automation
- Salud
- Machine 2 / Machine (M2M).
- Smart Metering.
- Agricultura de Precisión.
- Ciudades Inteligentes.



MERCADO.

- Específicos:
 - Masimo.
 - Delta Ohm.
- Genéricos:
 - Microcontrolador.
 - MicroPc.
 - Fpga.
- Laboratorio:



DE0_Nano.

Featured device:

- Altera Cyclone® IV EP4CE22F17C6N FPGA
- 153 maximum FPGA I/O pins

Configuration status and set-up elements:

- On-board USB-Blaster circuit for programming
- Spansion EPCS64

Expansion header:

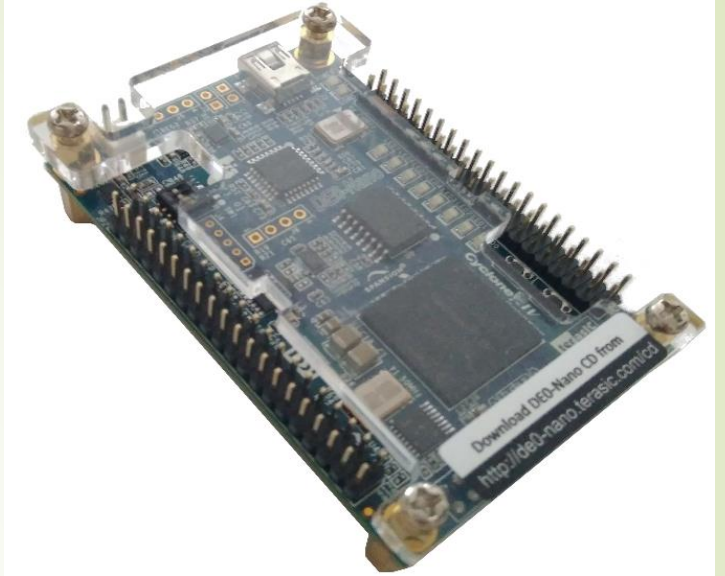
- Two 40-pin Headers (GPIOs) provide 72 I/O pins, 5V power pins, two 3.3V power pins and four ground pins

Memory devices:

- 32MB SDRAM
- 2Kb I2C EEPROM

General user input/output:

- 8 green LEDs
- 2 debounced pushbuttons
- 4-position DIP switch



G-Sensor:

- ADI ADXL345, 3-axis accelerometer with high resolution (13-bit)

A/D Converter:

- NS ADC128S022, 8-Channel, 12-bit A/D Converter
- 50 Ksps to 200 Ksps

Clock system:

- On-board 50MHz clock oscillator

Power Supply:

- USB Type mini-AB port (5V)
- DC 5V pin for each GPIO header (2 DC 5V pins)
- 2-pin external power header (3.6-5.7V)

RFS.

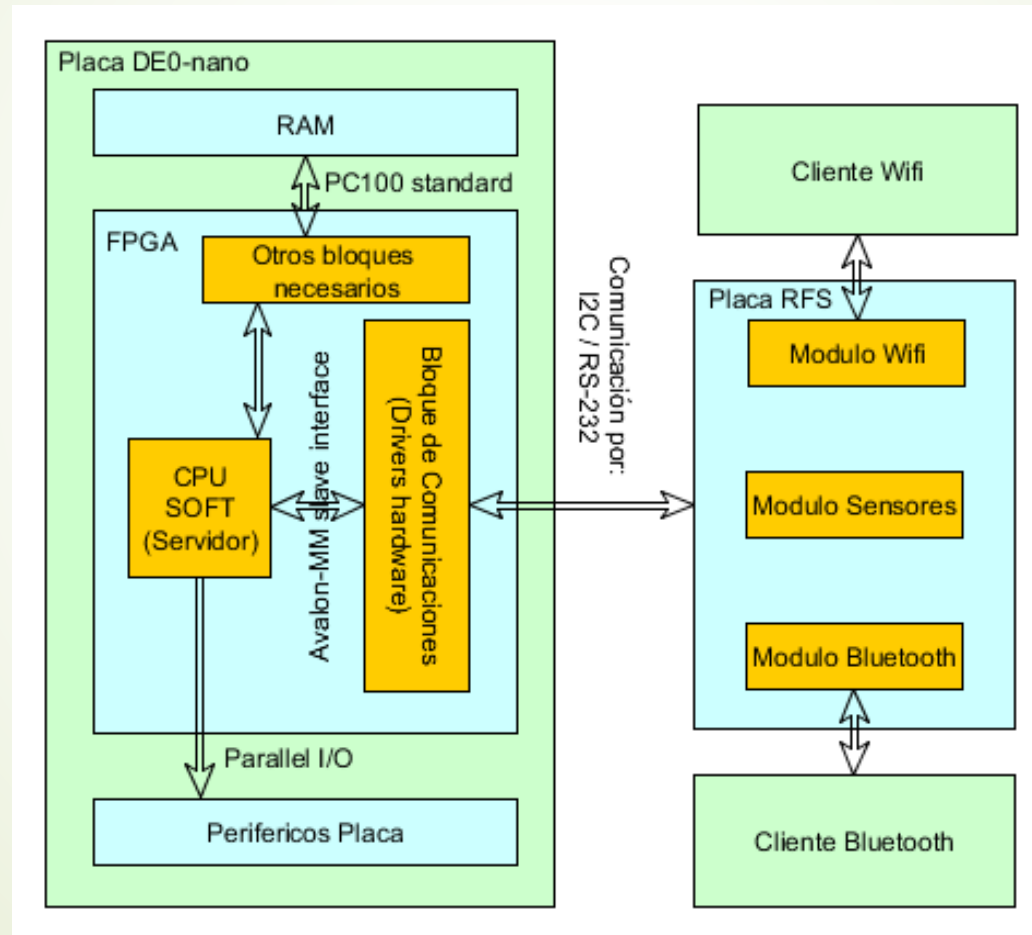
- 2x20 GPIO interface.
- CWi-Fi, using ESP- WROOM -02 module.
- Bluetooth SPP, using HC-05 module.
- Ambient Light Sensor.
- Temperature and humidity sensor.
- 9-axis sensor –accelerometer, magnetometer, and gyroscope.
- UARTto USB.
- 2x6 TMD GPIO Header.



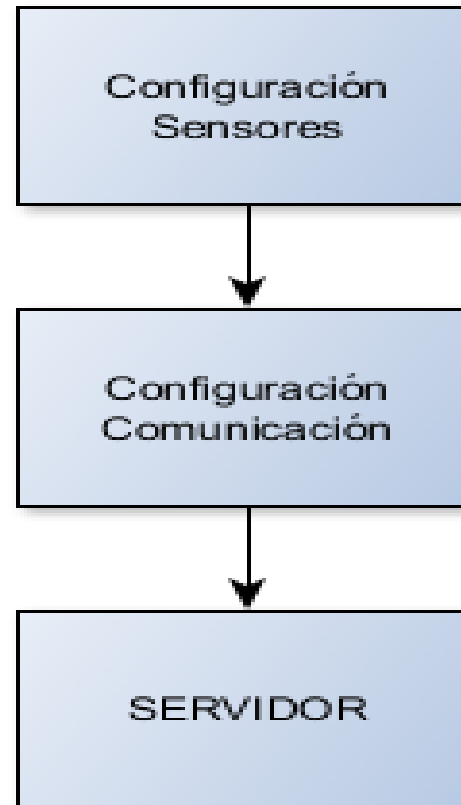
SOFTWARE USADO.



DISEÑO DEL SISTEMA.



DISEÑO SOFTWARE:





DISEÑO SOFTWARE:

Configuración de los sensores

- *Funciones proporcionadas por Terasic.*

- *Ejemplo:*

Inicialización y configuración:

- Temperatura (HDC1000):

- RH_Temp_Init(RH_TEMP_I2C_OPENCORES_BASE)

- RH_Temp_Sensor_Init().

Obtención de datos:

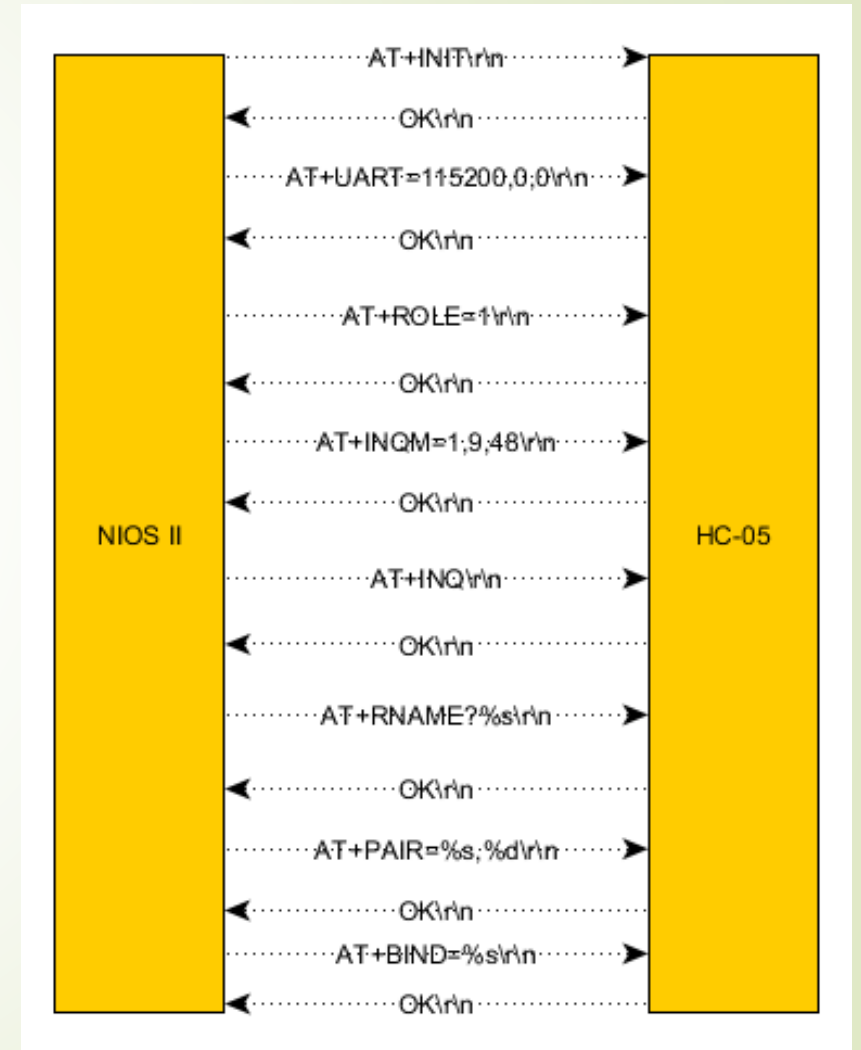
- HDC1000 (Temperatura):

- Bool temp_response(float *fTemperature, float *fHumidity)

SOFTWARE BLUETOOTH.

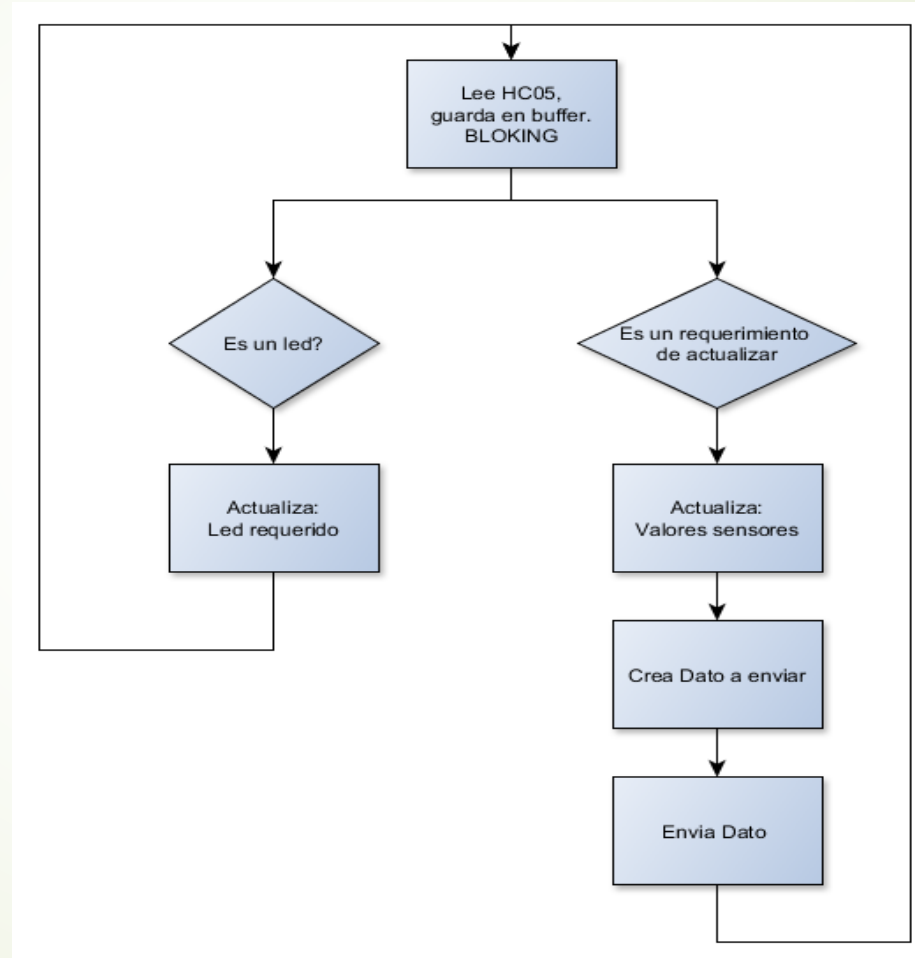
Configuración:

- CUart Uart;
- CHC05 HC05;
- Uart.Open(HC_05_UART_NAME).
- Una vez la Uart está operativa se tiene que entrar en el modo At, para ello se usa la siguiente función:
- SetConfig(true).
- Imagen config



SOFTWARE BLUETOOTH.

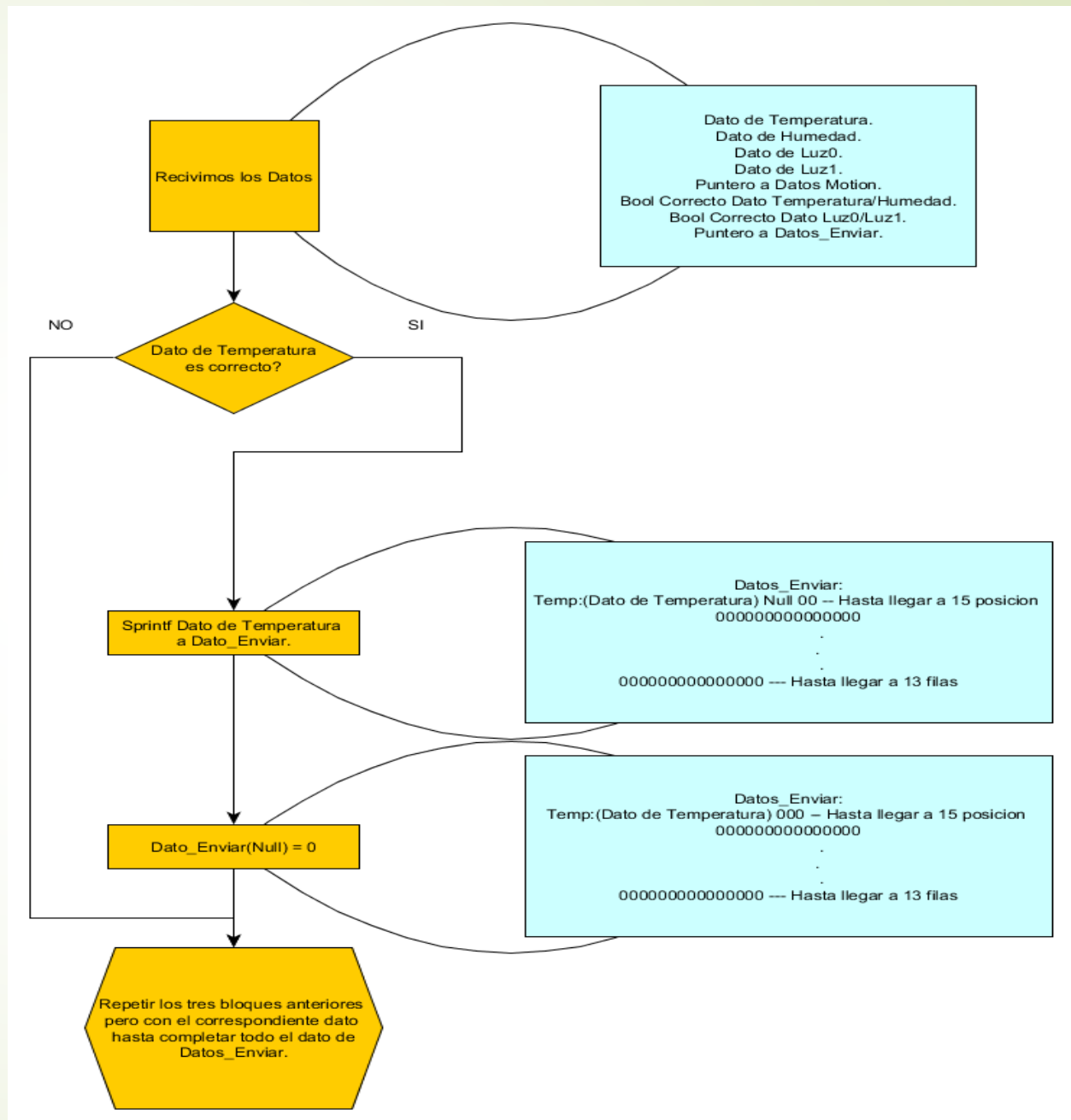
Servidor:



SOFTWARE BLUET.

Crear dato:

```
Datos_Enviar:  
Temp:(Dato de Temperatura) 000 -- Hasta llegar a 15 posicion  
000000000000000000  
.  
.  
.  
000000000000000000 -- Hasta llegar a 13 filas
```



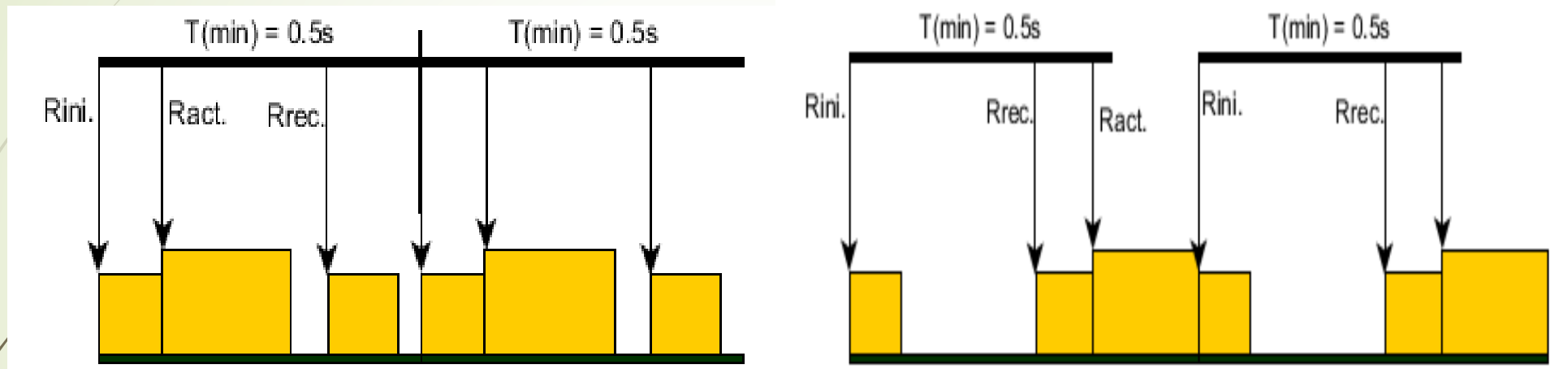
CLIENTE BLUETOOTH.

Interfaz Gráfica:



CLIENTE BLUETOOTH.

Descripción Tareas.



- Rini.: Inicio de envío de código al servidor.
- Ract.: Inicio de actualización datos en la interfaz gráfica, es la de mayor prioridad.
- Rrec.: Recibo de la información del servidor y tratamiento de esta.

PRUEBAS.

➤ Metodo:

➤ **#ifdef** DEBUG_BLUETOOTH

➤ start_time = alt_nticks();

➤ total_time = ((start_time) * 1000) /

➤ alt_ticks_per_second() ;

➤ **printf**("Tiempo absoluto en que se encuentra el sistema en ese momento %d ms\n\n\n", total_time);

➤ **#endif**

PRUEBA BLUETOOTH Y RESULTADOS.

```
Bluetooth_0 Nios II Hardware configuration - cable: USB-Blaster on localhost [USB-0]
Tiempo de la recepcion petición actualizar: 91750 ms
RX<--18
Tiempo respuesta del servidor 91750 ms

Tiempo de la recepcion petición actualizar: 92250 ms
RX<--18
Tiempo respuesta del servidor 92250 ms

Tiempo de la recepcion petición actualizar: 92750 ms
RX<--18
Tiempo respuesta del servidor 92750 ms

Tiempo de la recepcion petición actualizar: 93250 ms
RX<--18
Tiempo respuesta del servidor 93250 ms

Tiempo de la recepcion petición actualizar: 93750 ms
RX<--18
Tiempo respuesta del servidor 93750 ms

Tiempo de la recepcion petición actualizar: 94250 ms
RX<--18
Tiempo respuesta del servidor 94250 ms

Tiempo de la recepcion petición actualizar: 94750 ms
RX<--18
Tiempo respuesta del servidor 94750 ms
```

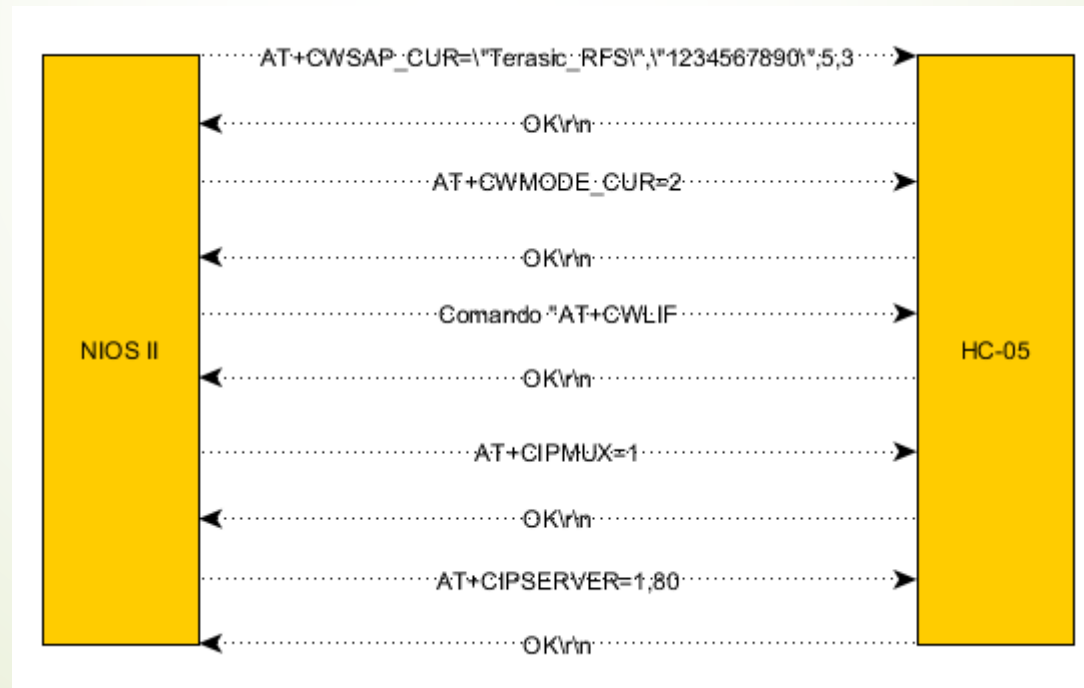
Resultados:

- Sincronía: Aceptable.
- Ancho Banda: 3.12 Kb/s.

SOFTWARE WIFI.

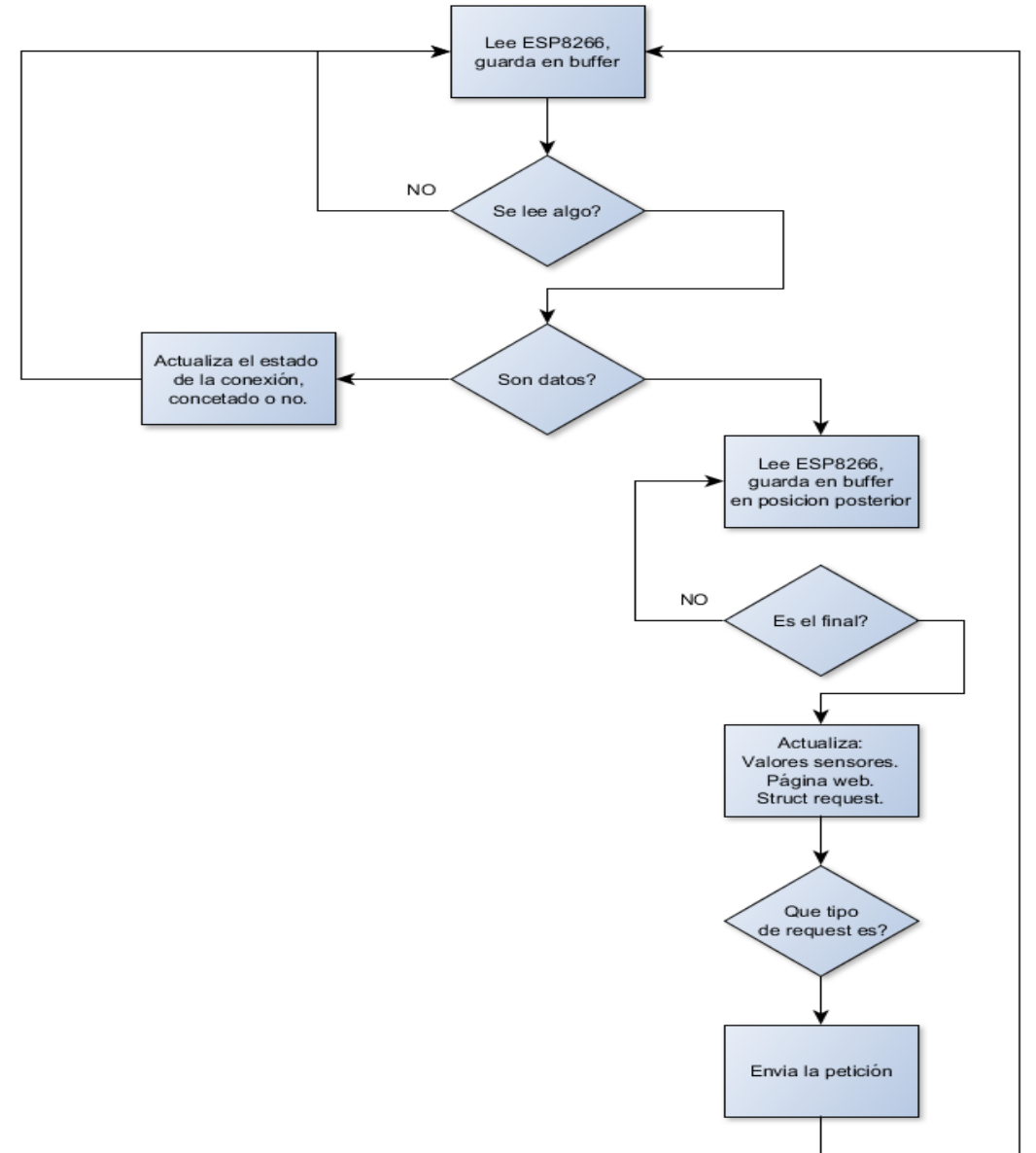
Configuración:

- **Configuración del ESP8266:**
- `esp8266_file = fopen(esp8266_uart, "rw+")`, modo ANSI C para inicializar la Uart, que se comunica con el módulo ESP8266.



SOFTWARE WIFI.

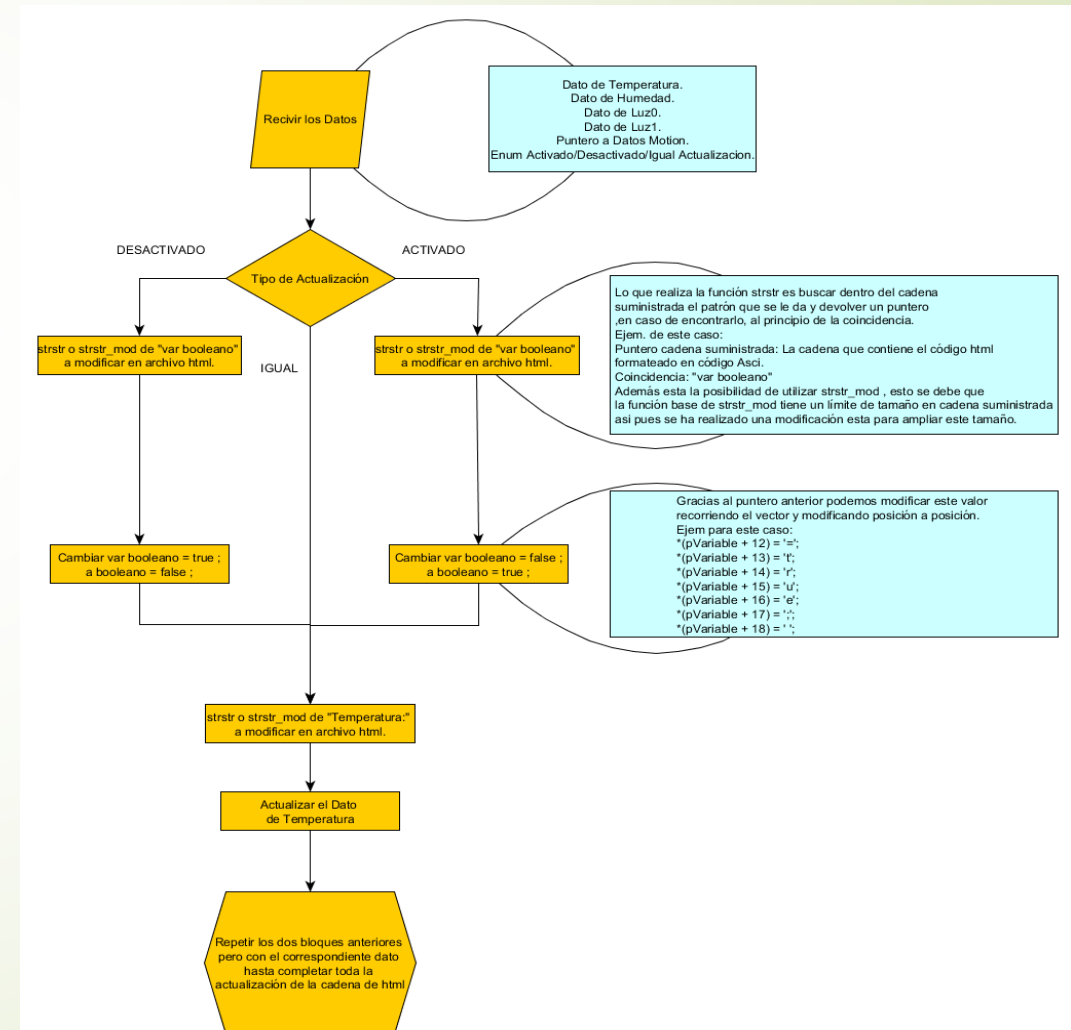
Servidor:



SOFTWARE WIFI.

Actualizar Página Html:

```
42     <table>
43         <tr>
44             <td>Temperatura: _____ </td>
45             <td>Humedad: _____ </td>
46         </tr>
47     </table>
48     <table>
49         <tr>
50             <td>Light0: _____ </td>
51             <td>Light1: _____ </td>
52         </tr>
53     </table>
54     <table>
55         <tr>
56             <td>Ax: _____ </td>
57             <td>Ay: _____ </td>
58             <td>Az: _____ </td>
59         </tr>
60     </table>
61     <table>
62         <tr>
```



CLIENTE WIFI.

Web / Tipo de datos.



RFS WiFi - LED

[led0 on](#) [led1 on](#) [led2 on](#) [led3 on](#) [led4 on](#) [led5 on](#) [led6 on](#) [led7 on](#)

[led0 off](#) [led1 off](#) [led2 off](#) [led3 off](#) [led4 off](#) [led5 off](#) [led6 off](#) [led7 off](#)

[ACTIVAR](#) [DESACTIVAR](#)

Temperatura: _____ Humedad: _____

Light0: _____ Light1: _____

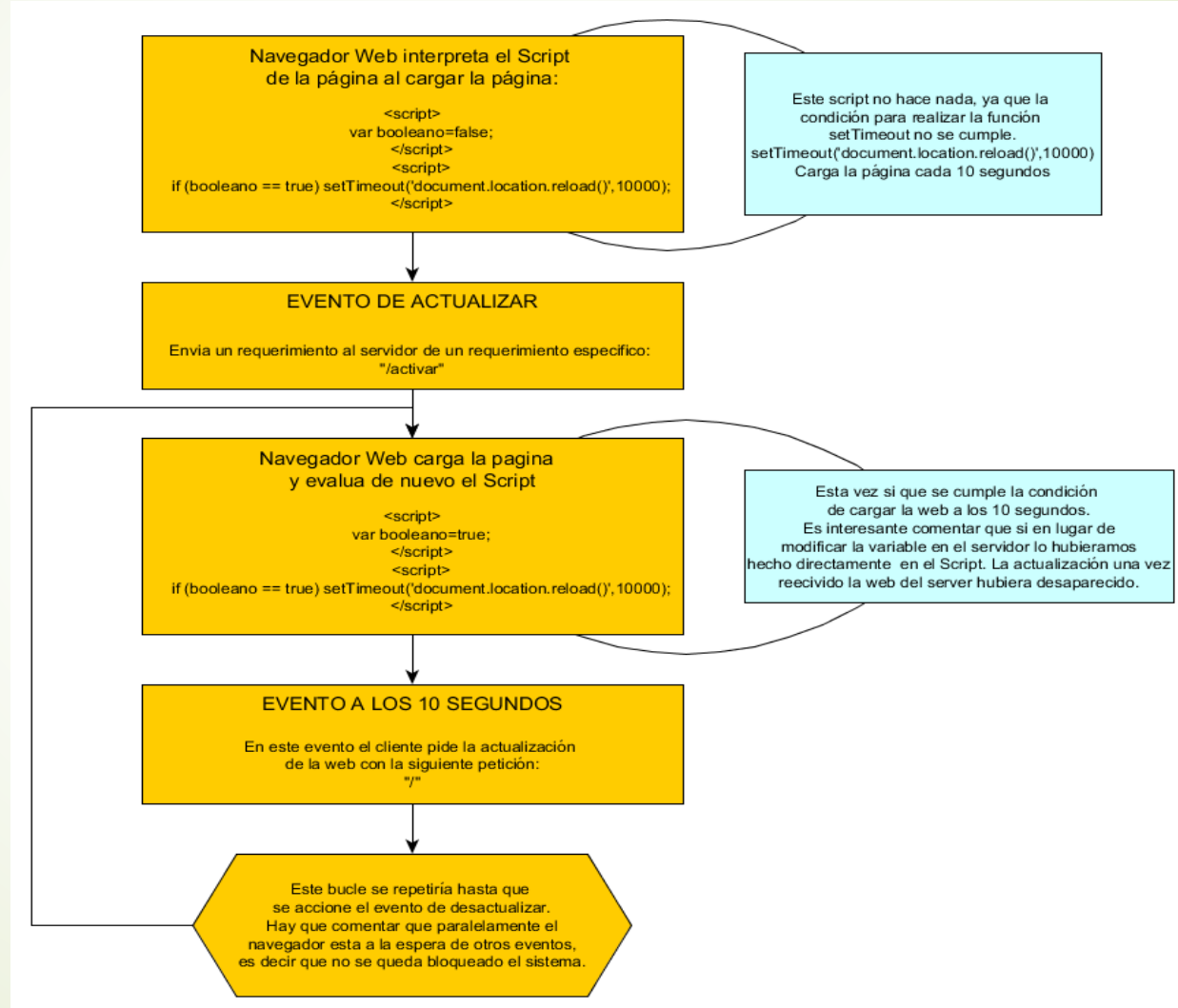
Ax: _____ Ay: _____ Az: _____

Gx: _____ Gy: _____ Gz: _____

Mx: _____ My: _____ Mz: _____

Resource type	Response		
	Status Code	Response Header	Body Message
/	200 OK	Content-Type: text/html	web_src/index.html
/favicon.ico	200 OK	Content-Type: text/plain	web_src/favicon.ico
/Logo_Terasic.jpg	302 Found	Content-Type: image/jpeg	web_src/Logo_Terasic.jpg
/led	302 Found	Location:/	---
/actualizar	302 Found	Location:/	---
/parar	302 Found	Location:/	---
Other	404 Not Found	Content-Type: text/html	web_src/404.html

CLIENTE WIFI. Bucle actualización:



PRUEBA WIFI Y RESULTADOS.

```
+IPD,0,441:GET / HTTP/1.1
```

```
Tiempo comienzo lectura petición: 34125 ms
```

```
Tiempo final lectura petición: 34250 ms
```

```
Tiempo final de envio: 34625 ms
```

```
0,CONNECT
```

```
+IPD,0,467:GET / HTTP/1.1
```

```
Tiempo comienzo lectura petición: 44625 ms
```

```
Tiempo final lectura petición: 44750 ms
```

```
Tiempo final de envio: 45000 ms
```

```
0,CONNECT
```

```
+IPD,0,467:GET / HTTP/1.1
```

```
Tiempo comienzo lectura petición: 55125 ms
```

```
Tiempo final lectura petición: 55125 ms
```

```
Tiempo final de envio: 55375 ms
```

Wi-Fi:

Ancho Banda: Insignificante.

Sincronización: Mejorable.

PRESUPUESTO.

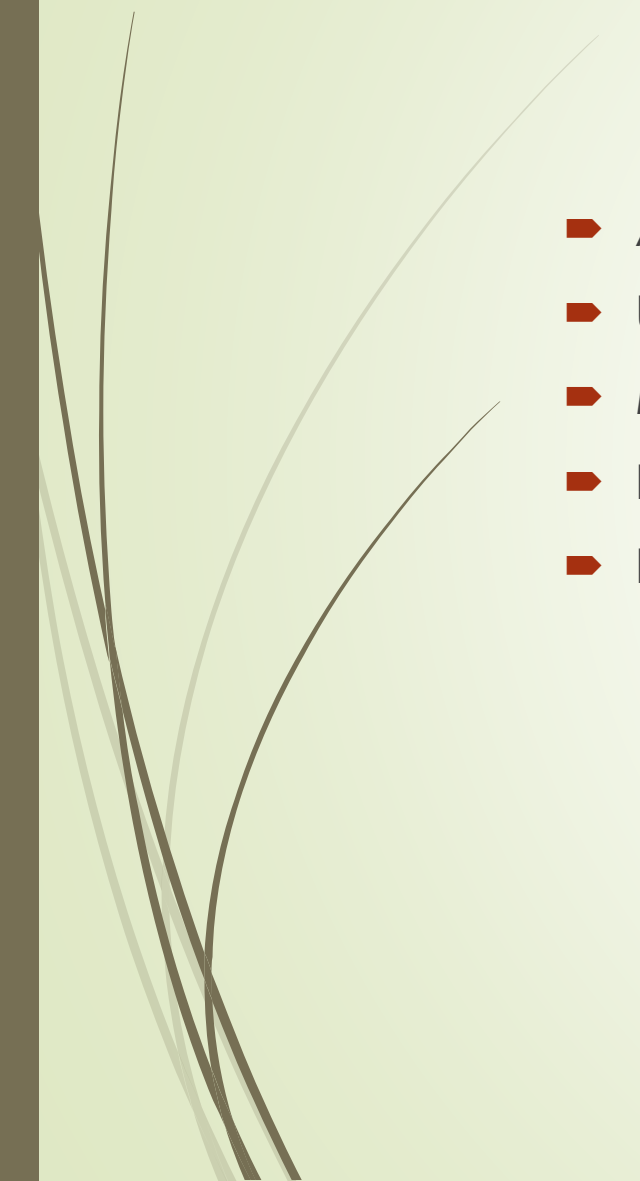
N o	Title	Weight(g)	Price	Qty.
1.	[DE0-Nano] DE0-Nano Development and Education Board (Academic) <ul style="list-style-type: none">Part Number: P0082	500	\$61	1

N o	Title	Feature	Weight(g)	Price	Qty.
2.	[RFS] The RFS Card <ul style="list-style-type: none">Part Number: P0499		300	\$59	1

Total Weight:	800(g)
Order Amount:	\$120
Shipping Cost:	\$38.49
Total Amount:	\$158.49



CONCLUSIONES.

- Aprendizaje transversal.
 - Uso de comunicación inalámbricas.
 - Método para realizar diseño completo de producto final.
 - Diseño de prototipo rápido.
 - Económico.
- 



MEJORAS FUTURAS.

- Mejorar el Timer en el Cliente Wi-Fi, obteniendo el tiempo global del Navegador Web y actuar en consecuencia para sincronizar el sistema correctamente.
- Implementar mediciones a la salida de la Uart, mediante ModelSim, para obtener una medición experimental del ancho de banda y no teórica.
- Integrar un sistema operativo dentro del microprocesador y dentro de éste un servidor ya diseñado.
- Analizar donde está el cuello de botella en el envío de datos mediante Bluetooth y Wifi e intentar mejorarlo mediante mejoras Hardware o Software, o una combinación de éstas.
- Mejorar la interfaz gráfica de la web y de la app.



VIDEOS.

- ▶ Bluetooth:

- ▶ <https://www.youtube.com/watch?v=vaMVgQrqGAo>

- ▶ Wi-Fi:

- ▶ https://www.youtube.com/edit?o=U&video_id=hpxFT5mcMGo