Departamento de Electrónica
Universidad Técnica Federico Santa María

Documento de Diseño
“WI-FI Shield”
Proyecto ELO-301

Fecha	Revisión	Autor	Comentarios
06/10/2023	1	Grupo 8	Revisión inicial
18/10/2023	2	Grupo 8	Cambiado sensor BME280 por APDS-9306-065 para ajustarse al presupuesto. Cambiada NUCLEO F401xx por NUCLEO L452R.

Índice

1 Introducción

Este proyecto consiste en desarrollar una shield que cumpla el propósito de actuar como modem Wi-Fi para cualquier placa de desarrollo que tenga la disposición de los pines igual a la de arduino. En este caso, para desarrollar y probar el dispositivo se utilizará la placa de desarrollo NÚCLEO L452RE.

Este dispositivo cuenta con dos módulos principales, que tienen comunicación directa con la placa de desarrollo. Uno de ellos es el que actúa como modem Wi-Fi por medio de una ESP32, además cuenta con dos LED que se utilizan para verificar correcta conexión y envío de datos respectivamente. El otro módulo consiste en un sensor de luminosidad, además de un botón. Esta parte del dispositivo permitirá hacer pruebas de envío de datos hacia una plataforma en internet, en la cual se mostrarán las mediciones en tiempo real cada vez que se presione el botón. Los datos enviados por internet serán presentados en una plataforma IoT (Internet of Things).

Estos dos módulos no están conectados entre sí directamente de ninguna forma, por lo que la placa de desarrollo es la encargada de, al momento de ser presionado el botón, consultar el valor actual del sensor con comunicación I2C, y por medio de comandos AT con comunicación UART, ordenar a la ESP32 que envíe los datos recopilados al servidor.

2 Funcionalidades identificadas

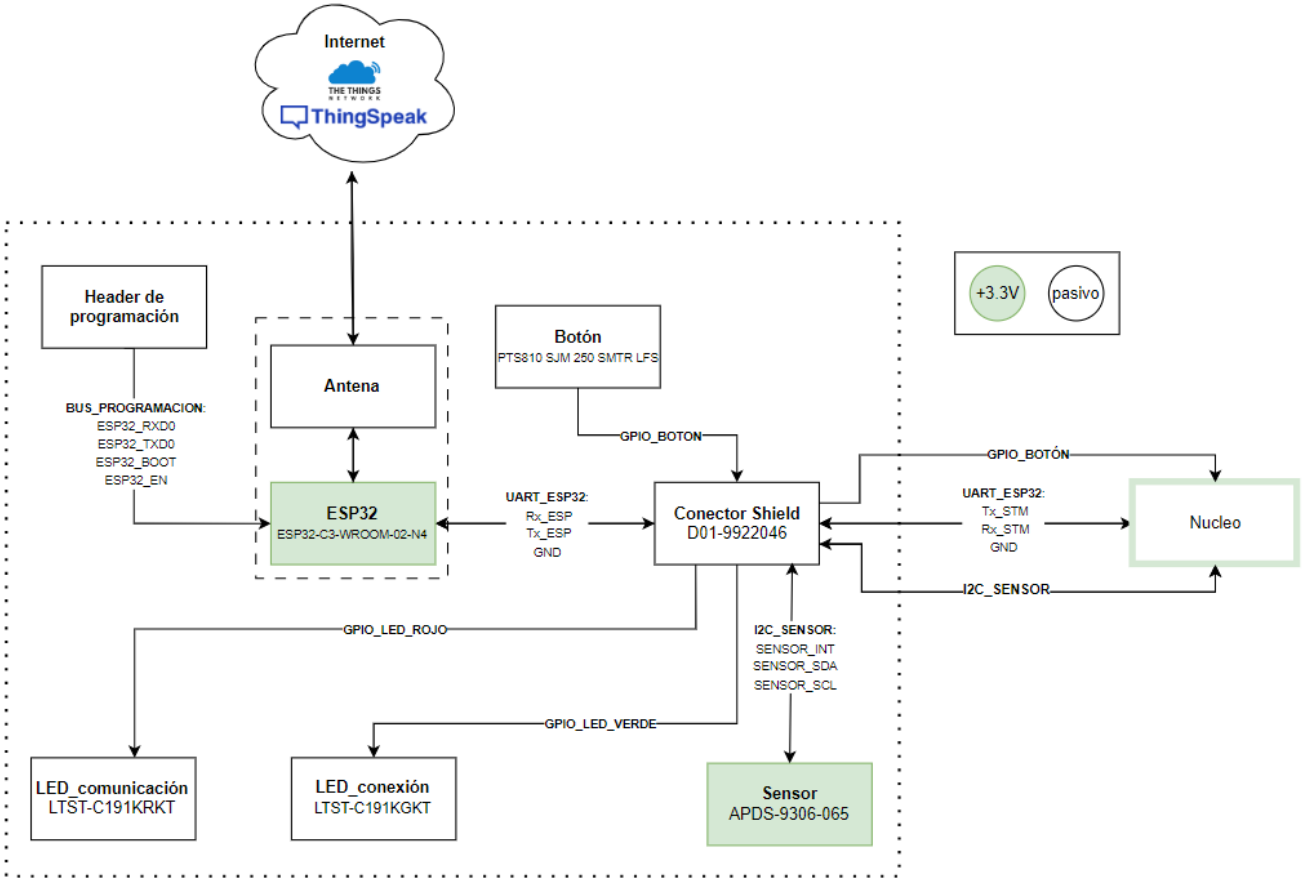
- Comunicación WI-FI.
- Medición de luminosidad.
- Envío de mediciones a una página web al presionar un botón.
- Reporte de conexión y envío de datos.

3 Elementos identificados

- ESP32
- Sensor de luminosidad
- Botón
- 2 LEDs
- Conectores tipo Shield
- Header de programación

4 Diseño de hardware propuesto

4.1 Diagrama de bloques



4.2 Detalle por bloque

Bloque 1: ESP32.

Part Number: ESP32-C3-WROOM-02-N4

Descripción: Módulo Wi-Fi y Bluetooth de propósito general. 18.0 x 20.0 x 3.2 [mm]. El voltaje de operación es entre 3.0 ~ 3.6 [V], típicamente 3.3 [V]. El consumo en transmisión puede variar entre 280 ~ 345 [mA], en recepción puede variar entre 82 ~ 84 [mA]. El consumo en estado *running* es entre 17 ~ 28 [mA], en estado *idle* es entre 13 ~ 21 [mA]. (En la práctica el consumo varía según los periféricos activados). Para bajo consumo, en *Light-sleep* consumo es 130 [µA], en *Deep-sleep* consumo es 5 [µA]. El estándar Wi-Fi es IEEE 802.11b/g/n. La potencia de transmisión varía entre 17.0 ~ 20.5 [dBm]. La sensibilidad de recepción está entre -98.0 ~ -74.4 [dBm].

Función: Unidad de microcontrolador. Se comunica de forma inalámbrica con un *Access Point*, para habilitar transferencia y recepción de datos desde la unidad STM32. Entrega señales sobre el estado de la red.

Señales:

GPIO_LED_COM: Señal de estado de comunicación.

GPIO_LED_CNX: Señal de estado de conexión.

UART_ESP32: Bus de comunicación serial entre STM32 y ESP32.

- Rx_ESP: Receptor UART
- Tx_ESP: Transmisión UART
- GND: Tierra común

BUS_PROGRAMACION: Bus usado para programar la ESP32

- ESP32_EN: Pin usado para habilitar el microcontrolador para ser usado (no debe quedar flotando)
- ESP32_BOOT: Pin usado para habilitar el *Booting Mode* del microcontrolador.
- ESP32_TXD0: Pin de transmisión hacia el programador de Firmware del microcontrolador.
- ESP32_RXD0: Pin de recepción desde el programador de Firmware del microcontrolador.

Bloque 2: Sensor.

Part Number: APDS-9306-065

Descripción: Sensor de luminosidad ambiental que convierte intensidad de luz a una salida I^2C. Tiene 6 pines, es de largo 2 ± 0.10 [mm], ancho de 2 ±0.10 [mm] y altura de 0.65 ±0.10 [mm]. Se alimenta con un voltaje entre 1.7 [V] y 3.6 [V], consumiendo una corriente típicamente de 85 [µA].

Función: Entregar mediciones de intensidad lumínica ambiental para su eventual envío.

Señales:

I2C_SENSOR: Comunicación entre Sensor y tarjeta núcleo mediante protocolo I2C. (Señal In and Out)

SENSOR_SCL: I^2C serial clock input

SENSOR_SDA: Serial data I/O para I^2C

SENSOR_INT : Pin de interrupción

Bloque 3: Conector macho.

Part Number: D01-9922046

Descripción: Paso 2.54 [mm] (distancia entre pines). El largo del conector es N x 2.54 [mm], donde N es 40. El diámetro del pin es 0.25 [mm]. A temperatura ambiente (25 °C) soporta una corriente máxima de 3 [A]. Se utilizará 1 de estos conectores separado en 3 grupos de pines: 1 de 6, 2 de 8 y 1 de 10 de la misma manera que están distribuidos los pines hembra de la tarjeta de desarrollo NÚCLEO L452RE.

Función: Conectar placa de desarrollo con placa a diseñar, en forma de shield.

Señales:

I2C_SENSOR: Comunicación entre Sensor y tarjeta núcleo mediante protocolo I2C. (Señal In and Out)

UART_ESP32: Comunicación serial entre tarjeta núcleo y ESP32 para dar instrucciones y paso de información mediante protocolo UART. (Señal In and Out)

GPIO_BOTON: Señal que indica a la tarjeta núcleo que debe recolectar datos del sensor, enviarlos a la ESP32 e indicarle que debe realizar el envío de datos mediante Wi-Fi.

Bloque 4: LED rojo.

Part Number: LTST-C191KRKT

Descripción: LED SMD con polaridad, conectado a la ESP32, de 0.8 [mm] de ancho, 1.6 [mm] de largo y 0.55 [mm] de alto, todas las dimensiones con ±0.1 [mm] de tolerancia. Voltaje de polarización típico 2 [V], máximo 2.4 [V] a 20 [mA] de corriente. Máximo peak de Forward Current 80 [mA] (1/10 Duty Cycle, 0.1ms Pulse Width), forward current DC 30 [mA]. Máxima corriente inversa de 10 [µA] y máximo voltaje inverso 5[V].

Función: Indicar cuando hay un envío de datos exitoso hacia el servidor.

Señales:

GPIO_LED_ROJO: Pin GPIO proveniente de la ESP32

Bloque 5: LED verde.

Part Number: LTST-C191KGKT

Descripción: LED SMD con polaridad, conectado a la ESP32, de 0.8 [mm] de ancho, 1.6 [mm] de largo y 0.55 [mm] de alto, todas las dimensiones con ±0.1 [mm] de tolerancia. Voltaje de polarización de 1.9 [V] a 2.4 [V] a 20 [mA] de corriente. Máximo peak de Forward Current 80 [mA] (1/10 Duty Cycle, 0.1ms Pulse Width), forward current DC 30 [mA]. Máxima corriente inversa de 10 [µA] y máximo voltaje inverso 5[V].

Función: se utilizará para indicar un estado importante (a definir)

Señales:

GPIO_LED_VERDE: Pin GPIO proveniente de la ESP32

Bloque 6: Botón.

Part Number: PTS810 SJM 250 SMTR LFS

Descripción: Botón ovalado de altura 2.5 [mm], con base rectangular de 4.6 [mm] de largo y 3.2 [mm] de ancho. Voltaje DC máximo de 16 [V] y corriente DC máxima de 50 [mA]. Tiempo de rebote menor a 10 [ms].

Función: se utilizará para indicar un estado importante (a definir)

Señales:

GPIO_BOTON: Pin GPIO de entrada a la tarjeta de desarrollo Nucleo

Bloque 7: Header de programación.

Part Number: TD-103-T-A

Descripción: Conector rectangular con 6 pines dispuestos en 2 corridas de 3 pines en paralelo.

Función: Conectar programador con esp32.

Señales:

ESP32_EN: Pin usado para habilitar el microcontrolador para ser usado (no debe ser dejado flotando)

ESP32_BOOT: Pin usado para habilitar el *Booting Mode* del microcontrolador.

ESP32_TXD0: Pin de transmisión hacia el programador de Firmware del microcontrolador.

ESP32_RXD0: Pin de recepción desde el programador de Firmware del microcontrolador.

Bloque 8: Núcleo

Part Number: NÚCLEO L452RE

Descripción: Placa de desarrollo programable, mediante el computador, con pines de I/O con soporte para protocolo I2C y UART. Con funcionalidad a temperatura ambiente (-40°C a 85°C), además soporta entradas de corriente y voltaje máximos en los pines de 5[mA] y 4[V] respectivamente.

Función: Obtener datos del sensor mediante protocolo I2C ,ante la presión del botón (GPIO), y posterior envío mediante UART a la ESP32, instruyendo el envío por Wi-Fi. Además de suministrar con 3.3 [V] a la ESP32 y el sensor.

Señales:

I2C_SENSOR: Comunicación entre Sensor y tarjeta núcleo mediante protocolo I2C. (Señal In and Out)

UART_ESP32: Bus de comunicación serial entre tarjeta núcleo y ESP32 para dar instrucciones y paso de información mediante protocolo UART. (Señal In and Out)

GPIO_BOTON: Señal que indica a la tarjeta núcleo que debe recolectar datos del sensor, enviarlos a la ESP32 e indicarle que debe realizar el envío de datos mediante Wi-Fi. (Señal In)

5 Diseño de firmware propuesto

5.1 Detalle de drivers

Driver I2C:

stm32l4xx_hal_i2c.c /.h

Descripción: Este driver provee funciones de firmware para manejar funcionalidades de I2C (Inter Integrated Circuit) como inicialización de funciones, operaciones de Input/Output, funciones de estado para el periférico.

Driver UART:

stm32l4xx_hal_uart.c /.h

Descripción: Este driver provee funciones de firmware para manejar funcionalidades de UART (Universal Asynchronous Receiver Transmitter), como inicialización de funciones, operaciones de Input/Output, funciones de estado para el periférico.

Driver interfaz usuario:

stm32l4xx_hal_gpio.c

Descripción: Este driver provee funciones de firmware para manejar funcionalidades de GPIO (General Purpose Input/Output), como inicialización de funciones, operaciones de Input/Output, funciones de estado para el periférico.

Controlador ESP32:

esp32.c /.h

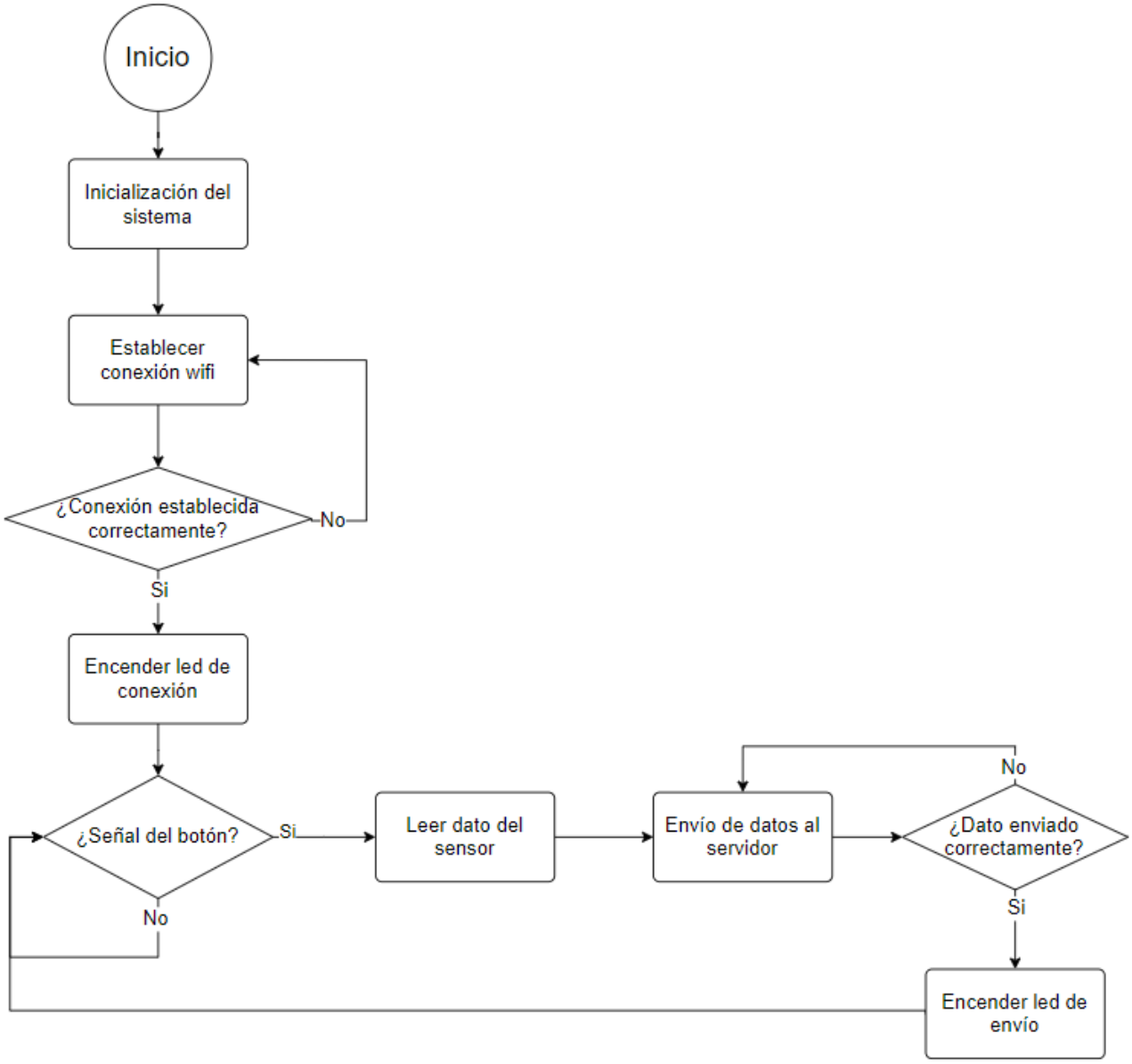
Descripción: Este controlador provee funciones de firmware para manejar funcionalidades del SoC ESP32, como inicialización de funciones, operaciones de comunicación y conectividad con *AT Commands*, mediciones de calidad de red y señal, transmisión y recepción de datos a través de red Wi-Fi.

Controlador APDS-9306-065:

apds9306.c/.h

Descripción: Este controlador contiene la API para manejar funcionalidades del módulo APDS-9306-065, como inicialización de funciones, consulta de datos, realización de mediciones, funciones de estado del módulo.

5.2 Diagrama de flujo



5.3 Detalle por módulo

Módulo 1: Inicialización del sistema.

Se inicializan todos los módulos: I2C, UART, GPIO, ESP32, interrupciones.

Módulo 2: Establecer conexión wifi.

Mediante comandos AT el microcontrolador se comunica con la esp32 para establecer la conexión Wi-Fi.

Módulo 3: Verificación de conexión.

Por medio de un pin de estado se verifica si la conexión fue exitosa. En caso de no tener éxito se re-intenta la conexión.

Módulo 4: Encender led de conexión.

En caso de tener una conexión wifi exitosa se enciende un led que indica el estado de conexión.

Módulo 5: Verificación señal botón:

Este módulo actúa como interrupción, activándose cada vez que se presiona el botón de envío de datos.

Módulo 6: Lectura de datos del sensor.

Por medio de pines ADC propios de la tarjeta de desarrollo se lee el valor que tiene el sensor en ese momento y se guarda en una variable temporal.

Módulo 7: Envío de datos al servidor.

El dato leído desde el sensor y guardado en la variable temporal se coloca en el payload del comando AT para ser enviado hacia el servidor desde la núcleo a través de la esp32.

Módulo 8: Verificación de envío de datos.

Por medio de un pin de estado de envío se verifica si el envío fue exitoso. En caso de no tener éxito se re-intenta el envío.

Módulo 9: Encender led de envío.

En caso de tener un envío de datos al servidor exitoso, se enciende un led que indica que se ha realizado un envío de datos.

6 Pruebas

6.1 Prueba sobre elementos

Prueba 1: Comprobar conducción de los conectores tipo Shield.

Soldar solo los conectores a la placa, conectarla a la núcleo y medir en cada pad de alimentación que el voltaje que se está entregando sea 3.3 [V].

Prueba 2: Polaridad de los LED

Antes de soldar los LED se debe medir con el tester en modo diodo la polaridad y comprobar que el indicador físico del componente está bien.

Prueba 3: Encendido de los LED

Encender mediante firmware los LED.

Prueba 4: Comprobar conducción del botón

Utilizar el tester para comprobar la conducción del botón. Si el botón funciona correctamente, conduce al pulsarlo y hace que el tester emita sonido, mientras que dejarlo libre no debería provocar este efecto.

Prueba 5: Interrupción del botón

Corroborar que se realicen interrupciones al presionar el botón.

Prueba 6: Comprobar mediciones del sensor (comunicación I2C)

Mediante firmware medir con el sensor y corroborar con la realidad. Probar en oscuridad y con luz directa para ver los límites inferior y superior. También probar leyendo registro Part_ID

Prueba 7: Programación de la ESP32

Comprobar que se puede programar el micro ESP32.

Prueba 8: Revisión de respuesta de la ESP32 a comando AT (comunicación UART)

Comprobar que la ESP32 responda correctamente a comandos AT enviados por la tarjeta núcleo.

6.2 Prueba sobre funcionalidades

Prueba 1: Conexión WIFI

Al enviar el comando AT que indique que la ESP32 debe conectarse a Wi-Fi, comprobar conexión

exitosa por medio de LED destinado para eso, respuesta de la ESP32 hacia la núcleo y verificar en los dispositivos conectados del celular en caso de que este esté actuando como access point.

Prueba 2: Envío de datos cualquiera.

Mediante comandos AT a la ESP32, enviar algo hacia el servidor para poder verlo desplegado en alguna plataforma web, ya sea texto, números, etc. Además comprobar que el LED que indica el envío exitoso se encienda.

Prueba 3: Captura de datos al presionar el botón.

Al presionar el botón, comprobar que dentro de la interrupción se están capturando de forma correcta los datos del sensor.

Prueba 4: Envío de datos al presionar el botón.

Comprobar que las dos pruebas anteriores funcionen simultáneamente. Además, que al presionar el botón, se genere una interrupción que capture los datos del sensor y los envíe mediante comando AT a la plataforma en internet.

7 Cronograma

Semana 1 comienza el lunes 02 de octubre de 2023

Semana	1	2	3	4	5	6	7	8
Diseño Industrial								
Diseño esquemático	x	x						
Diseño PCB		x	x	x				
Compra componentes			x	x	x			
Compra y envío de PCB				x	x	x		
Integración							x	
Desarrollo firmware					x	x		
Pruebas								x

Semana	Empieza el
1	02/10/2023
2	09/10/2023
3	16/10/2023
4	23/10/2023
5	30/10/2023
6	06/11/2023
7	13/11/2023
8	20/11/2023

Tabla 1: Cronograma

8 Costos

	Descripción	Precio por 1 unidad (USD$)	Precio x100 unidades (USD$)	Precio unitario x100 unidades (USD$)	Cantidad	Precio final (USD$)
Bloque 1: ESP32	Módulo Wi-Fi y Bluetooth	1,9	190	1,9	1	1,9
Bloque 2: Sensor	Sensor de intensidad lumínica	1.66	91,10	0,911	1	0,911
Bloque 3: Conector macho	Conector 40 pines	0,235	23,5	0,235	1	0,235
Bloque 4: LED rojo	LED rojo	0,26	6,63	0,0663	1	0,0663
Bloque 5: LED verde	LED verde	0,26	6,62	0,0662	1	0,0662
Bloque 6: Botón	Botón pulsador	0,39	30,64	0,3064	1	0,3064
Bloque 7: Programador	Conector 6 pines (3x2)	1,44	111,77	1,1177	1	1,1177
Componentes externos	Capacitancias y resistencias	-	-	-	-	3
PCB	JLCPCB	-	34,80	0,3480	1	0,3480
Ensamblado	2 horas hombre		0,859		2	1,719
Pruebas	4 horas hombre		0,859		4	3,439
Transporte componentes	Pedidos en digikey	-	100	1	-	1
Transporte PCB	JLCPCB por DHL Express Economy	-	-	-	1	0,8895
Otros	-	-	-	-	-	-
Total						15

Tabla 2: Costos