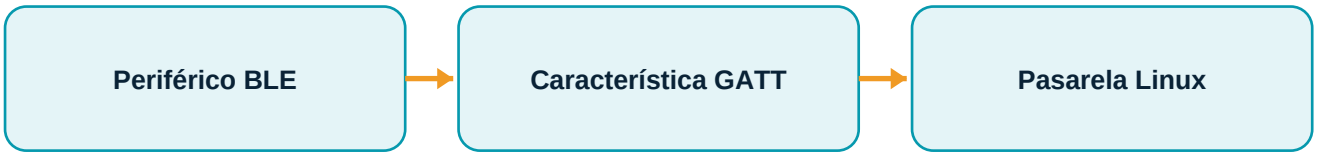
Capítulo 20 · Bluetooth Low Energy: GATT y pasarela BlueZ

Conceptos clave

Distingue anuncios, lecturas y notificaciones. Conserva las unidades y la escala al decodificar bytes.



GUÍA PASO A PASO DE LABORATORIO 06: NODOS INALÁMBRICOS E INTEGRACIÓN HÍBRIDA (WI-FI, BLE, LORA Y DEEP SLEEP)

Curso: Internet de las Cosas (IoT) con Arduino y Raspberry Pi — Ingeniería Mecatrobótica

Módulo 06 / Video 06: Nodos Inalámbricos Autónomos, Servidor BLE GATT (`0x181A`), Radio de Largo Alcance LoRa (`SX1278`) y Sueño Profundo (`Deep Sleep`) con Sensor Ambiental Bosch `BME280`

Plataformas: Arduino Nano ESP32 (`3.3 V` Nativo) + Raspberry Pi 4 Model B / 5 (`64-bit`)

---

1. Objetivo de Ingeniería del Laboratorio

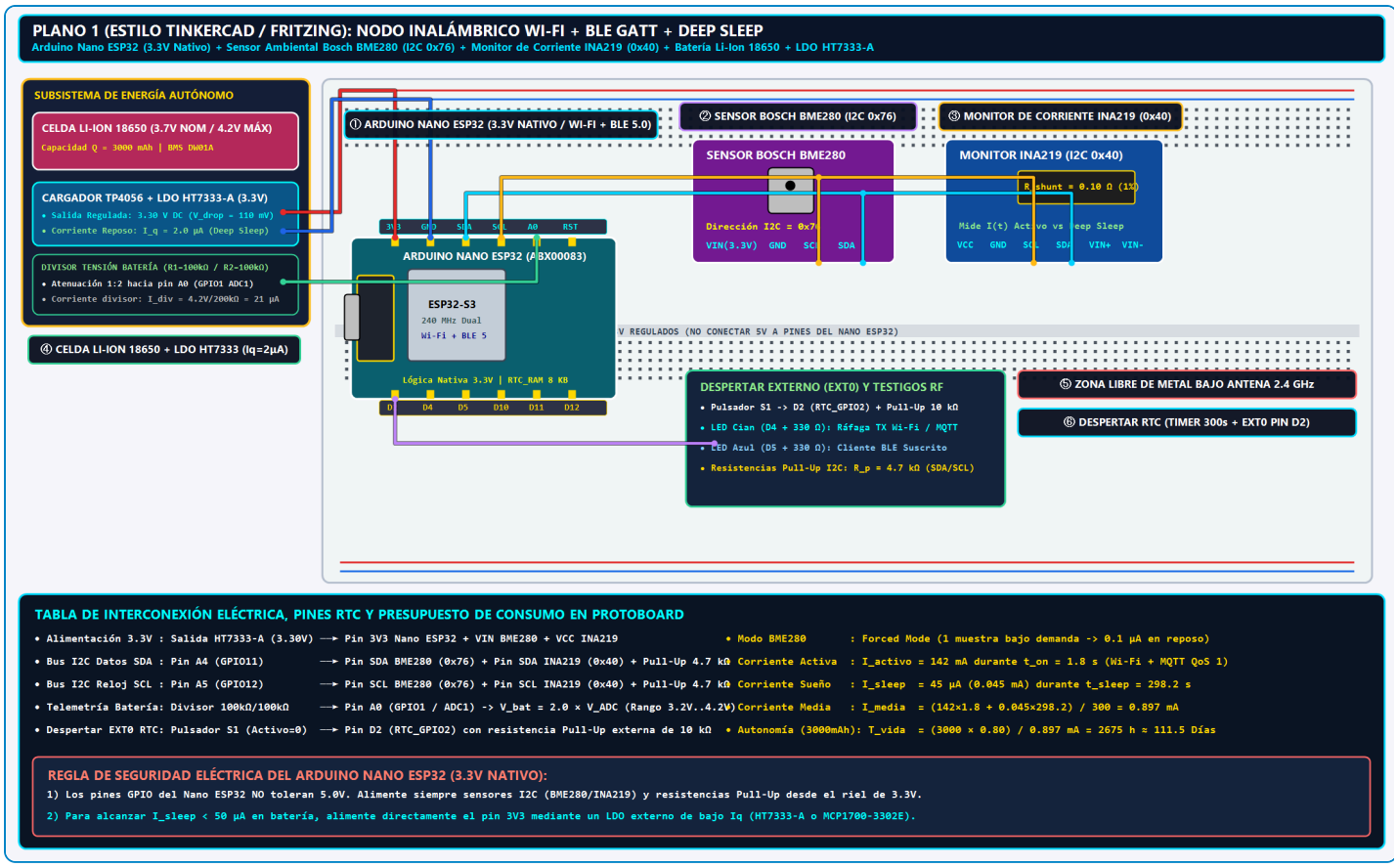
Diseñar, ensamblar en protoboard (estilo Tinkercad / Fritzing) y en placa de circuito impreso soldada (`PCB FR-4`), programar y auditar una **arquitectura inalámbrica híbrida de tres bandas/protocolos (`Wi-Fi` , `BLE 5.0` y `LoRa Sub-GHz`)** convergente hacia el Broker Eclipse Mosquitto en la Raspberry Pi:

- **Nodo Wi-Fi + MQTT de Ultrabajo Consumo (`Deep Sleep`)**: Basado en Arduino Nano ESP32 alimentado por celda `Li-Ion 18650 (3.7 V / 3000 mAh)` y regulador LDO `HT7333-A (I_q = 2.0 µA)` , leyendo temperatura, humedad y presión barométrica mediante el sensor Bosch `BME280` en **Modo Forzado (`Forced Mode`)** y reconectando con **Backoff Exponencial + Jitter**.
- **Nodo Periférico Bluetooth Low Energy (`BLE GATT 0x181A`)**: Exponiendo las características estándar `0x2A6E` (Temperatura), `0x2A6F` (Humedad) y `0x2A6D` (Presión) con notificaciones asíncronas (`CCCD 0x2902`) hacia el demonio Python `bleak` en la Raspberry Pi.
- **Concentrador de Largo Alcance** `LoRa SX1278 (433 / 915 MHz)` : Acoplado por bus `SPI0` e interrupción física `DIO0 (GPIO 25)` a la Raspberry Pi para enlaces rurales de `2 km` a `15 km` .

---

2. Montaje Físico en Protoboard (Estilo Tinkercad) y en Placa PCB Soldada

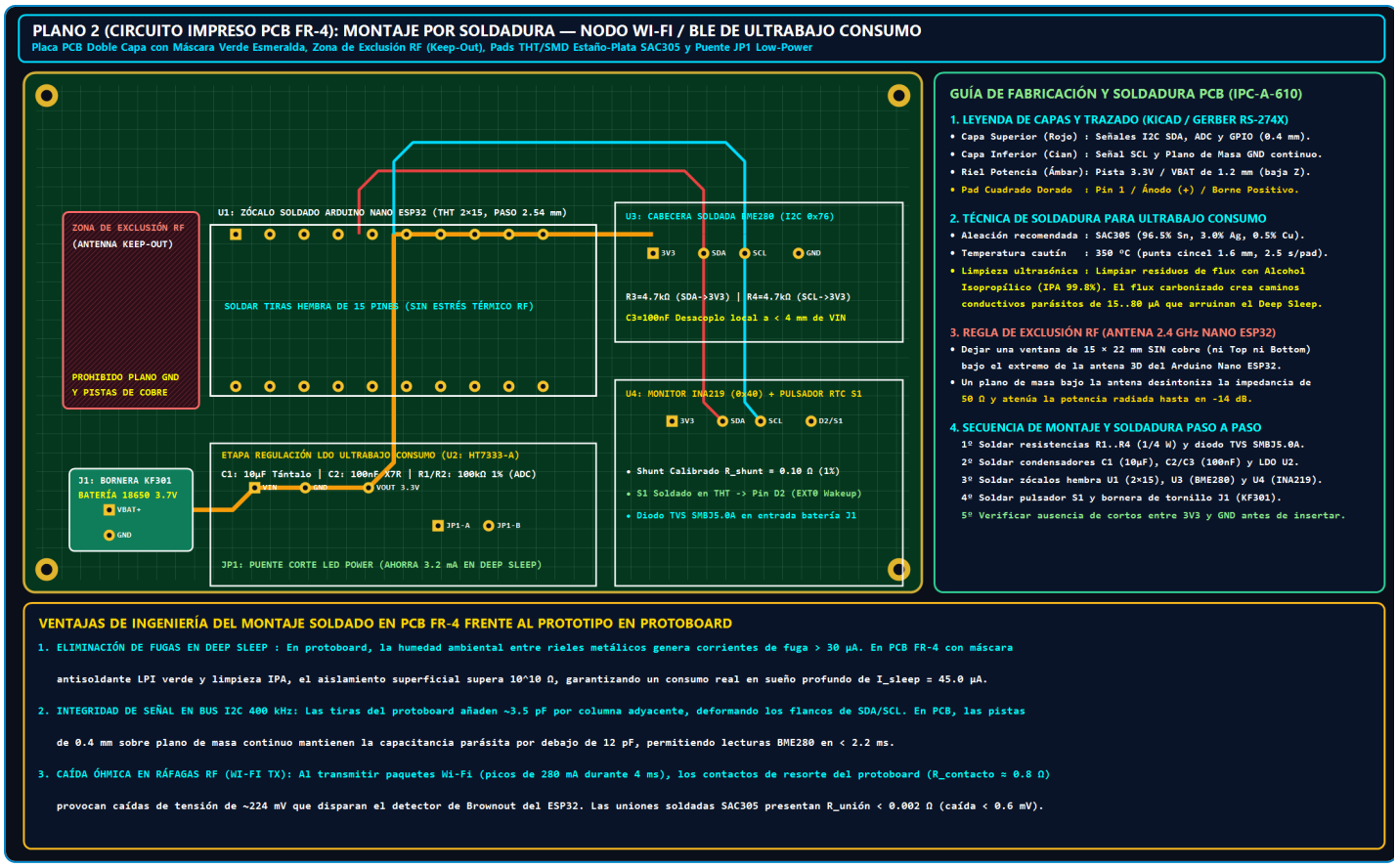
📐 **Plano 1: Montaje en Protoboard Estilo Tinkercad del Nodo Wi-Fi / BLE con BME280, INA219 y Batería 18650**



2.1 Tabla Maestra de Conexiones del Nodo Arduino Nano ESP32 (`Lógica Estricta 3.3 V`)

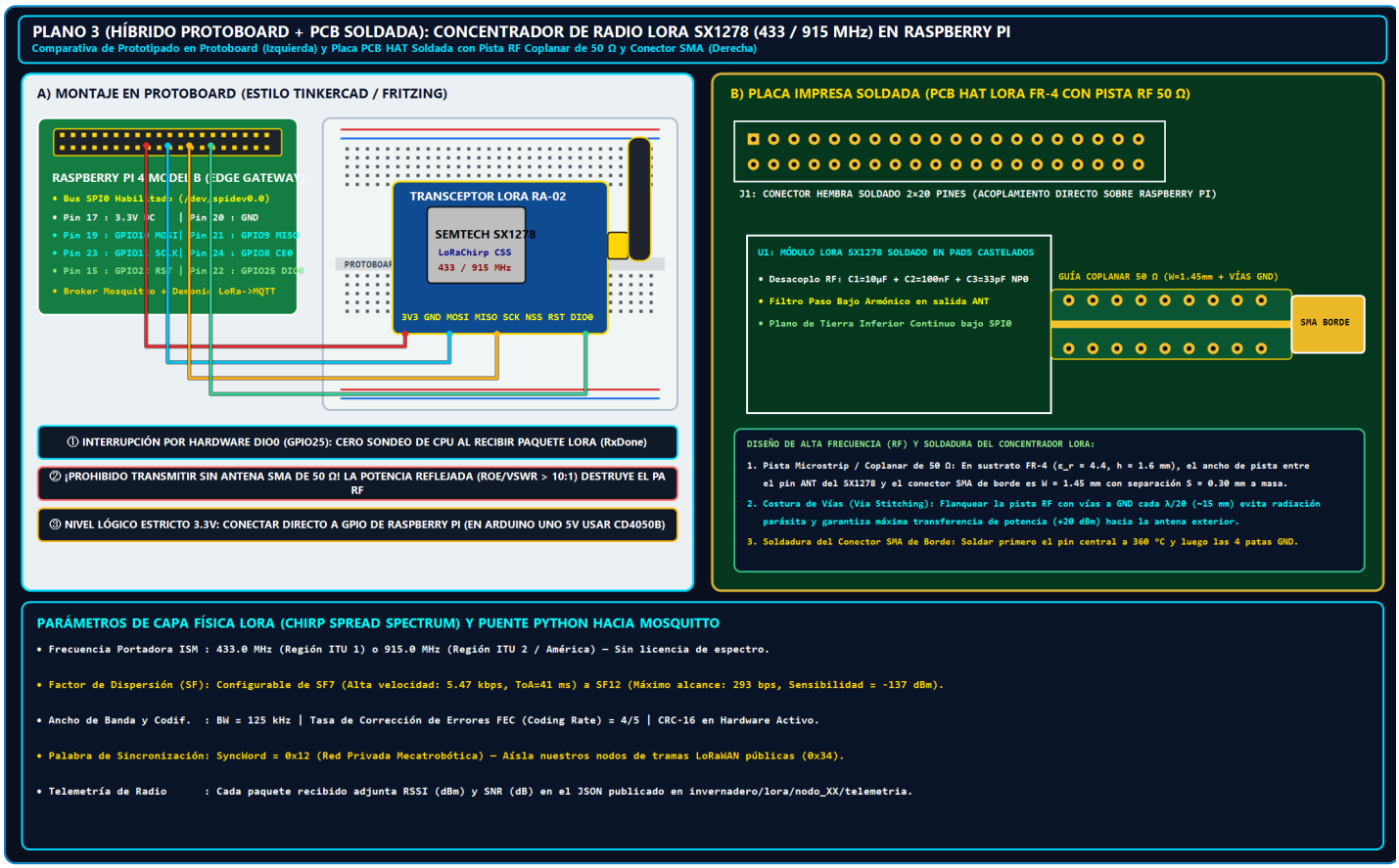
Subsistema / Componente	Pin del Componente	Pin en Arduino Nano ESP32	Función Eléctrica y Observaciones de Ingeniería
Regulador LDO HT7333-A	`VOUT (3.30 V)`	`Pin 3V3`	Alimentación directa regulada con corriente de reposo $I_q = 2.0\ \mu A$
Sensor Bosch BME280 (`0x76`)	`VIN / GND / SDA / SCL`	`3V3 / GND / A4 / A5`	Bus I^2C a $400\ kHz$ con resistencias Pull-Up de $4.7\ k\Omega$ a $3.3\ V$
Monitor Corriente INA219 (`0x40`)	`VCC / GND / SDA / SCL`	`3V3 / GND / A4 / A5`	Shunt de precisión $R_{shunt} = 0.10\ \Omega\ (1\%)$ en serie con el riel de $3.3\ V$
Divisor Tensión Batería	Punto medio `100kΩ/100kΩ`	`Pin A0 (GPIO1 / ADC1)`	Lectura calibrada con `analogReadMilliVolts(A0)` ($V_{bat} = 2.0 \times V_{ADC}$)
Pulsador Despertar RTC (`S1`)	Contacto NA a `GND`	`Pin D2 (RTC_GPIO2)`	Interrupción externa `EXT0` con resistencia Pull-Up externa de $10\ k\Omega$ a $3.3\ V$
Indicadores LED RF	Ánodo vía $330\ \Omega$	`Pin D4 (Wi-Fi)` / `D5 (BLE)`	Testigos luminosos activos únicamente durante la ventana despierta

📐 Plano 2: Placa de Circuito Impreso PCB FR-4 con Montaje por Soldadura del Nodo Inalámbrico



3. Montaje del Concentrador LoRa `SX1278` en Raspberry Pi (`SPI0` + `IRQ DIO0`)

📐 Plano 3: Interconexión en Protoboard y Placa PCB HAT Soldada del Transceptor LoRa SX1278



> **Regla Crítica de Radiofrecuencia (ROE / VSWR):** Jamás alimente ni ejecute el script de transmisión/recepción del módulo `SX1278` sin haber enroscado previamente su **antena helicoidal o dipolo de $50\ \Omega$ en el conector SMA**. Transmitir a $+20\ \text{dBm}$ con el puerto RF abierto refleja el 100% de la energía hacia el transistor de salida del amplificador de potencia (PA), destruyendo el chip en milisegundos.

Pin del Módulo LoRa `SX1278 (Ra-02)`	Pin Físico en Raspberry Pi (40 Pines)	GPIO BCM	Función en el Bus `SPI0` y Control de Radio
`VCC (3.3V)`	**Pin 17**	Riel `3.3V DC`	Alimentación limpia capaz de entregar picos de $120\ \text{mA}$
`GND`	**Pin 20**	Riel `GND`	Masa común de referencia digital y blindaje RF
`MOSI`	**Pin 19**	`GPIO 10 (SPI0_MOSI)`	Escritura de registros y carga de FIFO hacia el SX1278
`MISO`	**Pin 21**	`GPIO 9 (SPI0_MISO)`	Lectura de registros, RSSI, SNR y paquete recibido
`SCK`	**Pin 23**	`GPIO 11 (SPI0_SCLK)`	Reloj síncrono SPI a $4.0\ \text{MHz}$ (Modo `0,0`)
`NSS (CS)`	**Pin 24**	`GPIO 8 (SPI0_CE0_N)`	Selección de esclavo activa en nivel bajo
`RST`	**Pin 15**	`GPIO 22`	Reinicio físico por hardware del transceptor
`DIO0`	**Pin 22**	`GPIO 25`	Interrupción por flanco de subida (`RxDone` / `TxDone`)

---

4. Despliegue de Software y Verificación en Laboratorio

Paso 1: Habilitar `SPI0` y `Bluetooth BlueZ` en Raspberry Pi e instalar dependencias

```
sudo raspi-config nonint do_spi 0
sudo systemctl enable --now bluetooth
sudo apt update && sudo apt install -y python3-pip python3-spidev python3-rpi.gpio bluez
pip3 install --break-system-packages bleak paho-mqtt
```

Paso 2: Compilar y cargar los firmwares en Arduino Nano ESP32

- En Arduino IDE 2.x, instalar el paquete de placas **Arduino ESP32 Boards** y las librerías `PubSubClient`, `Adafruit BME280 Library` y `Adafruit INA219`.
- Para ensayar el nodo **Wi-Fi + MQTT con Deep Sleep**, abrir `codigo_fuente/nodo_wifi_mqtt_bme280_deepsleep.ino`, ajustar `WIFI_SSID`, `WIFI_PASSWORD` y `MQTT_BROKER_IP`, y subir al Nano ESP32.
- Para ensayar el nodo **Periférico BLE GATT (`0x181A`)**, subir `codigo_fuente/periferico_ble_gatt_ambiental.ino`.

Paso 3: Lanzar las pasarelas de borde en la Raspberry Pi y auditar en Mosquitto

```
# Terminal 1: Iniciar pasarela asíncrona BLE GATT -> MQTT
python3 codigo_fuente/pasarela_ble_a_mqtt_bleak.py

# Terminal 2: Iniciar concentrador LoRa SX1278 SPI0 -> MQTT
python3 codigo_fuente/transceptor_lora_sx1278_mqtt_bridge.py

# Terminal 3: Auditar en tiempo real la convergencia de las 3 tecnologías RF en el Broker
mosquitto_sub -h localhost -u scada_central -P Clave -t 'invernadero/#' -q 1 -v
```

LISTADO DE MATERIALES (BOM) — MÓDULO 06: NODOS INALÁMBRICOS E INTEGRACIÓN HÍBRIDA

Curso: Internet de las Cosas (IoT) con Arduino y Raspberry Pi — Ingeniería Mecatrobótica

Módulo 06 / Video 06: Nodos Inalámbricos (`Wi-Fi` , `BLE 5.0` , `LoRa SX1278`), Gestión Energética `Li-Ion 18650` y Montaje en Protoboard / PCB Soldada

---

1. Microcontroladores, Transceptores RF y Computadora de Borde

Ítem	Cant.	Designación PCB	Componente / Referencia Industrial	Especificación Técnica Clave	Función en el Módulo 06
01	2	`U1`	**Arduino Nano ESP32** (`ABX00083`)	SoC `ESP32-S3` Dual-Core Xtensa LX7 @ $240\ \text{MHz}$, $8\ \text{KB}$ RTC SRAM, Lógica nativa $3.3\ \text{V}$, Wi-Fi `802.11b/g/n` + `BLE 5.0`	Nodo 1: Wi-Fi + Deep Sleep; Nodo 2: Servidor Periférico BLE GATT (`0x181A`)
02	1	`GW1`	**Raspberry Pi 4 Model B (4 GB) o RPi 5**	Broadcom Quad-Core 64-bit, Bluetooth `BlueZ 5.x` (`/dev/hci0`), Bus `SPI0` (`/dev/spidev0.0`), Broker Mosquitto	Pasarela de Borde Híbrida (`BLE` + `LoRa` + `Wi-Fi` → `MQTT`)
03	2	`U_LORA`	**Módulo Transceptor LoRa** `Ra-02 (SX1278)`	Banda ISM Sub-GHz ($433\ \text{MHz}$ / $915\ \text{MHz}$), Modulación CSS (`SF7..SF12`), Sensibilidad $-137\ \text{dBm}$, SPI $3.3\ \text{V}$	Enlace de largo alcance rural ($2\ \text{km}$ a $15\ \text{km}$) con CRC-16 en hardware
04	2	`ANT1`	**Antena Dipolo / Helicoidal SMA** $50\ \Omega$	Impedancia característica $Z_0 = 50\ \Omega$, Ganancia $G = +3.0\ \text{dBi}$, ROE (VSWR) $< 1.5:1$, Cable pigtail IPEX a SMA	Radiación eficiente y protección del amplificador de potencia del `SX1278`

2. Sensórica Ambiental I^2C e Instrumentación de Consumo Energético

Ítem	Cant.	Designación PCB	Componente / Referencia Industrial	Especificación Técnica Clave	Función en el Módulo 06
05	2	`U3`	**Sensor Ambiental Bosch** `BME280`	Interfaz I^2C (`0x76`) / SPI $3.3\ \text{V}$, Temp ($\pm 0.5\ ^\circ\text{C}$), Humedad ($\pm 3\%\ \text{RH}$), Presión ($\pm 1.0\ \text{hPa}$), $I_{sleep} = 0.1\ \mu\text{A}$	Adquisición meteorológica en Modo Forzado (`Forced Mode`) bajo demanda
06	1	`U4`	**Monitor High-Side Texas Instruments** `INA219`	Interfaz I^2C (`0x40`), ADC de $12\ \text{bits}$, resistencia Shunt $R_{shunt} = 0.10\ \Omega$ (1%), rango $\pm 3.2\ \text{A}$	Medición de perfil de corriente $I(t)$ en ráfaga RF frente a sueño profundo

3. Subsistema de Alimentación por Batería de Ultrabajo Consumo (`Deep Sleep`)

Ítem	Cant.	Designación PCB	Componente / Referencia Industrial	Especificación Técnica Clave	Función en el Módulo 06
07	1	`BAT1`	**Celda Recargable Li-Ion** `18650`	Tensión nominal $3.7\ V$ (carga plena $4.20\ V$, corte $3.0\ V$), Capacidad nominal $Q = 3000\ mAh$, química INR/NCR	Fuente primaria de energía para operación autónoma $> 110\ días$
08	1	`CHG1`	**Módulo Cargador Li-Ion** `TP4056` **con BMS**	Corriente de carga CC/CV hasta $1.0\ A$ más integrado de protección `DW01A` + doble MOSFET `FS8205A`	Carga segura por USB-C / panel solar de $5\ V$ y corte por sobredescarga
09	2	`U2`	**Regulador LDO** `HT7333-A` **o** `MCP1700-3302E`	Salida fija $3.30\ V$ ($I_{\max} = 250\ mA$), tensión de caída $V_{drop} = 110\ mV$, **corriente de reposo** $I_q = 2.0\ \mu A$	Sustituye reguladores `AMS1117` ($I_q > 5\ mA$) para lograr $I_{sleep} = 45\ \mu A$
10	2	`R1, R2`	**Resistencias de Película Metálica** $100\ k\Omega$	Tolerancia $\pm 1\%$, potencia $1/4\ W$, bajo coeficiente térmico ($\pm 50\ ppm/^\circ C$)	Divisor resistivo $1:2$ hacia el pin `A0 (ADC1)` para telemetría de batería
11	2	`C1, C2`	**Condensadores de Desacoplo** $10\ \mu F$ **y** $100\ nF$	$10\ \mu F$ Tántalo ($16\ V$, baja ESR) + $100\ nF$ Cerámico multicapa `X7R`	Absorción de picos transitorios de corriente ($240\ mA$) al transmitir en Wi-Fi

4. Insumos de Prototipado en Protoboard (Tinkercad) y Soldadura en Placa PCB FR-4

Ítem	Cant.	Designación PCB	Componente / Herramienta de Montaje	Especificación Técnica Clave	Función en el Módulo 06
12	2	`PROTO`	**Protoboard de 830 Puntos (** `MB-102` **)**	Contactos de bronce fosforoso niquelado, doble riel de alimentación lateral	Ensamble rápido sin soldadura (estilo Tinkercad / Fritzing)
13	2	`PCB1`	**Placa Virgen o Fabricada** `PCB FR-4` **Doble Capa**	Fibra de vidrio epoxi `FR-4` ($h = 1.6\ mm$, cobre $1\ oz/ft^2 = 35\ \mu m$), máscara antisoldante verde esmeralda	Montaje definitivo por soldadura con zona de exclusión RF y pista de $50\ \Omega$
14	4	`U1, U3, U4`	**Tiras de Pines Hembra Paso** $2.54\ mm$ **(** `THT` **)**	Contactos dorados de $15\ pines$ (para Nano ESP32) y $4/6\ pines$ (para `BME280` / `INA219`)	Permite insertar y retirar los módulos sobre la PCB sin aplicar calor directo
15	2	`J1, J2`	**Borneras de Tornillo Soldables** `KF301-2P`	Paso $5.08\ mm$, capacidad $300\ V\ /\ 10\ A$, fijación mecánica por jaula de latón	Fijación firme de los cables de la celda `18650` sobre la placa PCB
16	1	`SOLD`	**Soldadura** `SAC305` **+ Alcohol Isopropílico** `IPA`	Aleación sin plomo ($96.5\% Sn, 3.0\% Ag, 0.5\% Cu$) + `IPA 99.8%` para limpieza de flux	Eliminación de corrientes de fuga superficiales entre pads durante el Deep Sleep

MANUAL DE INGENIERÍA DE RADIOFRECUENCIA: WI-FI, BLE GATT, LORA Y SUEÑO PROFUNDO (DEEP SLEEP)

Curso: Internet de las Cosas (IoT) con Arduino y Raspberry Pi — Ingeniería Mecatrobótica

Módulo 06 / Video 06: Fundamentos Matemáticos de Presupuesto de Enlace RF, Arquitectura GATT Bluetooth SIG, Modulación Chirp Spread Spectrum (`LoRa`) y Modelado Energético de Baterías

---

1. El Trilema de las Comunicaciones Inalámbricas en IoT

En ingeniería de telecomunicaciones industriales, ningún protocolo de radiofrecuencia puede maximizar simultáneamente **tres variables físicas contrapuestas**:

- **Alcance Geográfico (d)**
- **Caudal de Datos o Ancho de Banda (R_b)**
- **Autonomía Energética en Batería (T_{vida})**

Parámetro de Ingeniería	Wi-Fi (`IEEE 802.11n`)	Bluetooth Low Energy (`BLE 5.0`)	Radio `LoRa Sub-GHz (SX1278)`
Banda de Frecuencia ISM	$2.4\ GHz$ / $5.0\ GHz$	$2.4\ GHz$ (40 canales de $2\ MHz$)	$433\ MHz$ / $868\ MHz$ / $915\ MHz$
Velocidad Efectiva (R_b)	$20\ Mbps$ a $72\ Mbps$	$125\ kbps$ a $2.0\ Mbps$	$293\ bps$ (`SF12`) a $5.47\ kbps$ (`SF7`)
Alcance Típico en Campo	$30\ m$ a $90\ m$	$15\ m$ a $60\ m$	$2.0\ km$ **a** $15.0\ km$
Sensibilidad del Receptor ($S_{\min}$)	$-75\ dBm$	$-95\ dBm$	$-123\ dBm$ **(** `SF7` **)** **a** $-137\ dBm$ **(** `SF12` **)**
Corriente Pico en Transmisión	$180\ mA$ a $260\ mA$	$12\ mA$ a $18\ mA$	$90\ mA$ a $120\ mA$ ($+17..20\ dBm$)
Pila TCP/IP Nativa	Sí (Directo a Broker MQTT)	No (Requiere Pasarela GATT → IP)	No (Requiere Concentrador SPI → IP)

---

2. Presupuesto de Enlace RF (Link Budget) y Ecuación de Friis

La potencia electromagnética recibida P_{rx} (expresada en dBm, decibelios referidos a $1\ mW$) en la antena del servidor de borde viene determinada por el **balance logarítmico de potencias, ganancias y pérdidas**:

$$P_{rx}\ (dBm) = P_{tx}\ (dBm) + G_{tx}\ (dBi) + G_{rx}\ (dBi) - FSPL(d, f)\ (dB) - L_{misc}\ (dB)$$

Donde la **Pérdida de Propagación en el Espacio Libre de Friis ($FSPL$)** en función de la distancia $d\ (km)$ y la frecuencia portadora $f\ (MHz)$ es:

$$FSPL\ (dB) = 20 \cdot \log_{10}(d_{km}) + 20 \cdot \log_{10}(f_{MHz}) + 32.44$$

2.1 ¿Por qué la banda Sub-GHz ($433\ MHz$) penetra mucho mejor que $2.4\ GHz$?

A igual distancia d, la diferencia de atenuación en espacio libre entre $2400\ MHz$ (`Wi-Fi` / `BLE`) y $433\ MHz$ (`LoRa`) es:

$$\DeltaFSPL = 20 \cdot \log_{10}\left((2400) / (433)\right) = 20 \cdot \log_{10}(5.5427) \approx +14.87 \text{ dB}$$

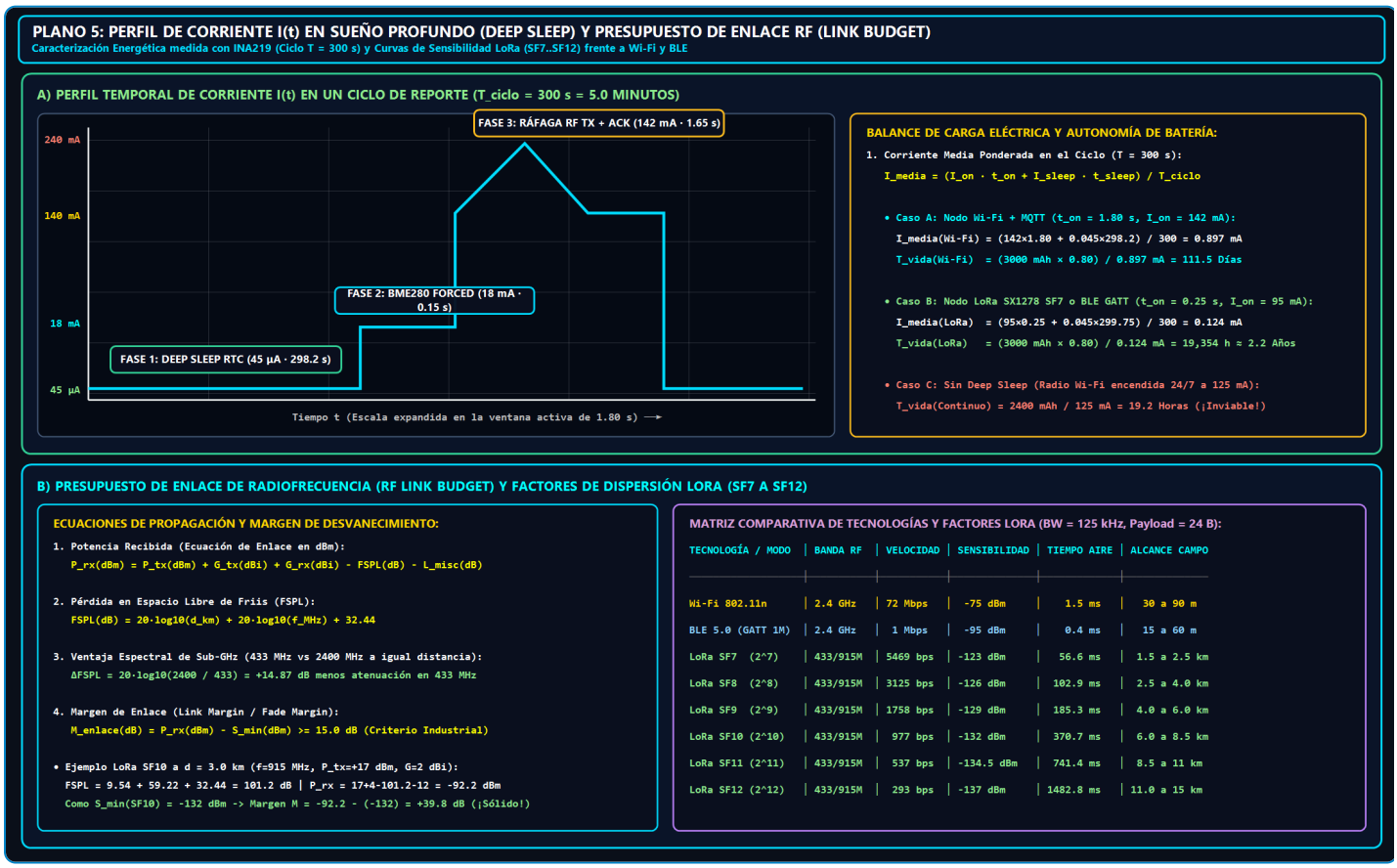
Una ventaja de $14.87\ \text{dB}$ equivale a multiplicar el alcance lineal en espacio libre por un factor de $5.54\times$ con la misma potencia de transmisión, además de sufrir menor absorción dieléctrica al atravesar follaje húmedo en invernaderos o muros de hormigón.

2.2 Margen de Desvanecimiento (Link Margin)

Para que un enlace industrial sea confiable ante lluvia, niebla o interferencia multitrayecto, exigimos un margen mínimo de seguridad de $15\ \text{dB}$ sobre la sensibilidad límite $S_{\min}$ del demodulador:

$$M_{\text{enlace}}\ (\text{dB}) = P_{\text{rx}}\ (\text{dBm}) - S_{\min}\ (\text{dBm}) \geq 15.0\ \text{dB}$$

📐 Plano 5: Perfil Temporal de Corriente en Deep Sleep y Presupuesto de Enlace RF



---

3. Reconexión Wi-Fi y MQTT con Backoff Exponencial y Jitter Estocástico

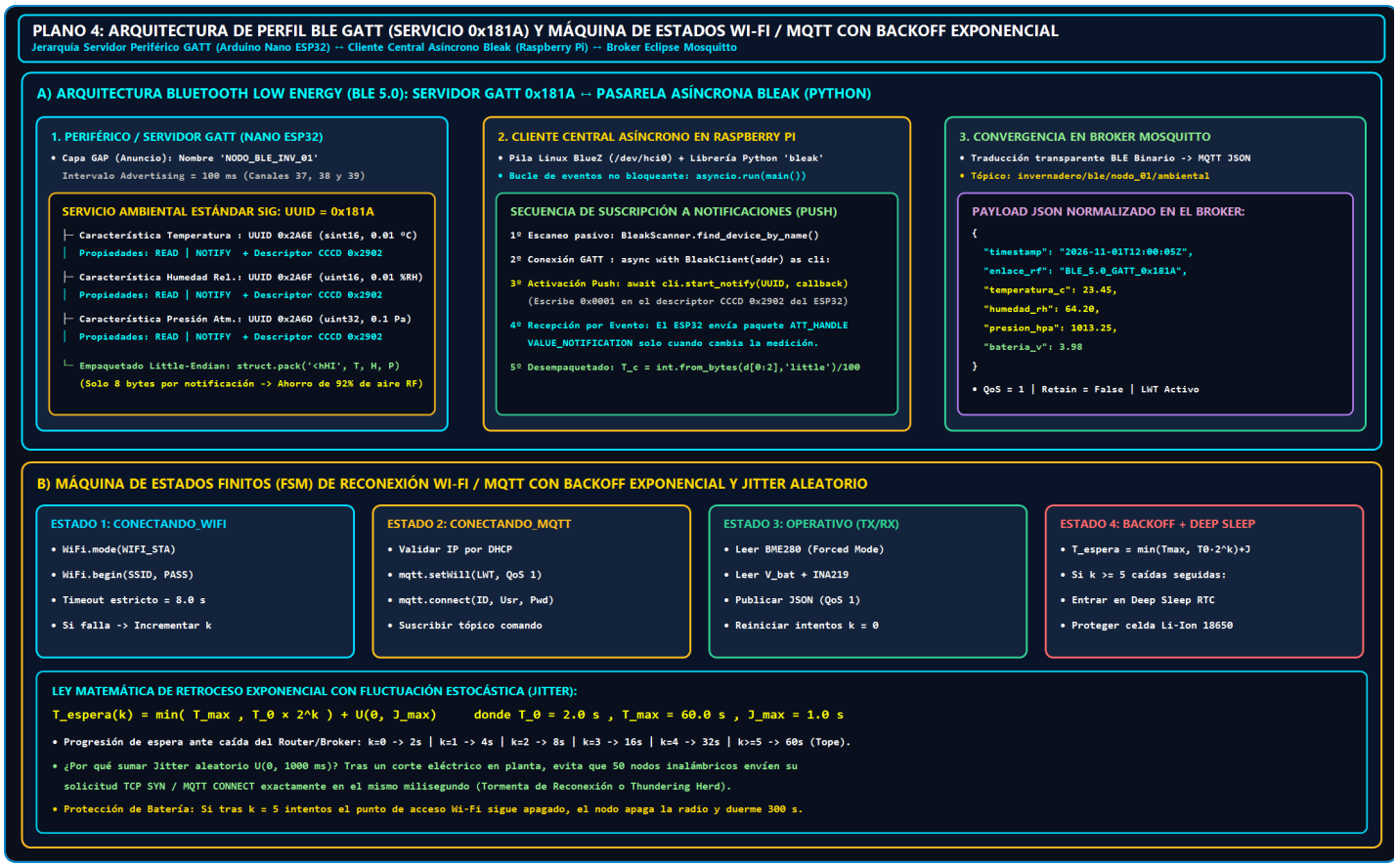
Cuando se restaura la energía eléctrica tras un apagón en planta, los microcontroladores arrancan en menos de $200\ \text{ms}$, mientras que el router Wi-Fi industrial y el servidor Linux tardan entre $30\ \text{s}$ y $60\ \text{s}$ en levantar sus servicios. Si N nodos reintentan conectarse en un bucle fijo cada $500\ \text{ms}$, provocan una *Tormenta de Reconexión (*Thundering Herd*)* y agotan sus baterías.

La solución de grado industrial implementa *Retroceso Exponencial Truncado con Fluctuación Aleatoria (*Jitter*)*:

$$T_{\text{espera}}(k) = \min\left(T_{\max},\ T_0 \cdot 2^k\right) + U(0, J_{\max})$$

Con $T_0 = 2.0\ \text{s}$, $T_{\max} = 60.0\ \text{s}$ y $J_{\max} = 1.0\ \text{s}$, la espera crece geométricamente ($2\ \text{s}, 4\ \text{s}, 8\ \text{s}, 16\ \text{s}, 32\ \text{s}, 60\ \text{s}$) y el término aleatorio uniforme $U(0,\ 1000\ \text{ms})$ desincroniza las ráfagas TCP SYN entre nodos.

📐 Plano 4: Arquitectura de Perfil BLE GATT 0x181A y Máquina de Estados de Reconexión con Backoff



---

4. Modelado Energético en Sueño Profundo (`Deep Sleep`) y Vida Útil de Batería

En un nodo cíclico con periodo total $T_{ciclo} = t_{activo} + t_{sleep}$, la **corriente media ponderada** drenada de la batería es:

$$I_{media} = \frac{I_{activo} \cdot t_{activo} + I_{sleep} \cdot t_{sleep}}{t_{activo} + t_{sleep}}$$

Aplicando un factor de descarga segura $\eta_{bat} = 0.80$ para preservar la química de la celda `Li-Ion 18650` ($Q_{bat} = 3000\ mAh$), la autonomía en días es:

$$T_{vida\ (días)} = \frac{Q_{bat\ (mAh)} \cdot \eta_{bat}}{24 \cdot I_{media\ (mA)}}$$

- **Sin Deep Sleep ($I_{media} = 125\ mA$ continuo):** $T_{vida} = (3000 \times 0.80) / (24 \times 125) = 0.80$ días $= 19.2$ horas.
- **Con Deep Sleep en Wi-Fi ($t_{activo} = 1.8$ s a $142\ mA$, $t_{sleep} = 298.2$ s a $45\ \mu A$):** $I_{media} = 0.897\ mA \implies T_{vida} = \mathbf{111.5\ días}$.
- **Con Deep Sleep en LoRa `SF7` o `BLE` ($t_{activo} = 0.25$ s a $95\ mA$, $t_{sleep} = 299.75$ s a $45\ \mu A$):** $I_{media} = 0.124\ mA \implies T_{vida} = \mathbf{806\ días\ (\approx 2.2\ años)}$.

PLANO Y ESQUEMÁTICO DEL CIRCUITO — MÓDULO 06: NODOS INALÁMBRICOS Y PLACAS PCB SOLDADAS

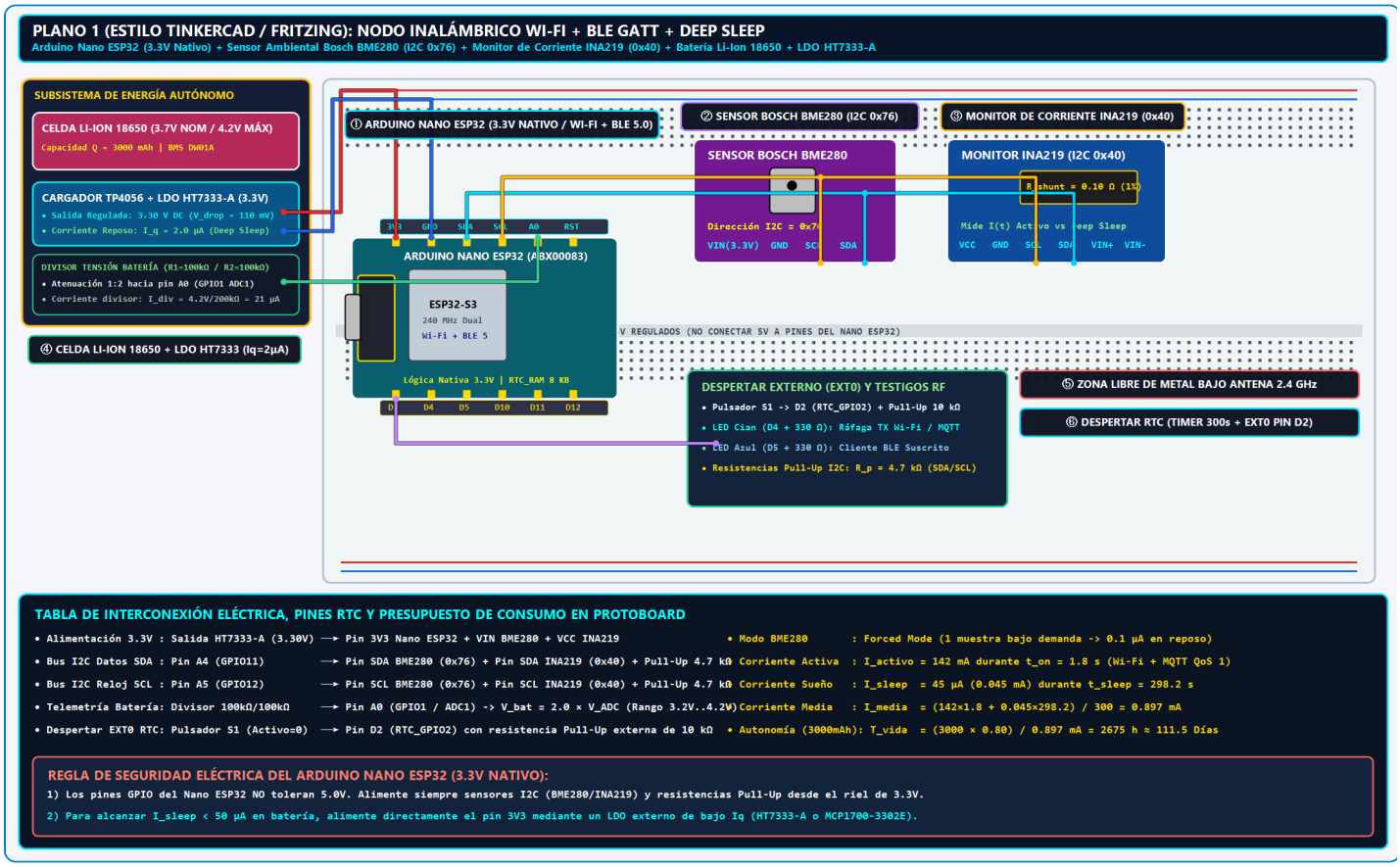
Curso: Internet de las Cosas (IoT) con Arduino y Raspberry Pi — Ingeniería Mecatrobótica

Módulo 06 / Video 06: Planos de Montaje en Protoboard (Estilo Tinkercad / Fritzing) de Alta Legibilidad y Planos de Circuito Impreso (`PCB FR-4`) con Montaje por Soldadura

---

1. Plano 1 (Estilo Tinkercad / Fritzing): Nodo Inalámbrico Wi-Fi + BLE + Deep Sleep en Protoboard

📐 Plano 1: Montaje en Protoboard Estilo Tinkercad del Nodo Wi-Fi y BLE con Sensor BME280, Monitor INA219 y Batería Li-Ion 18650

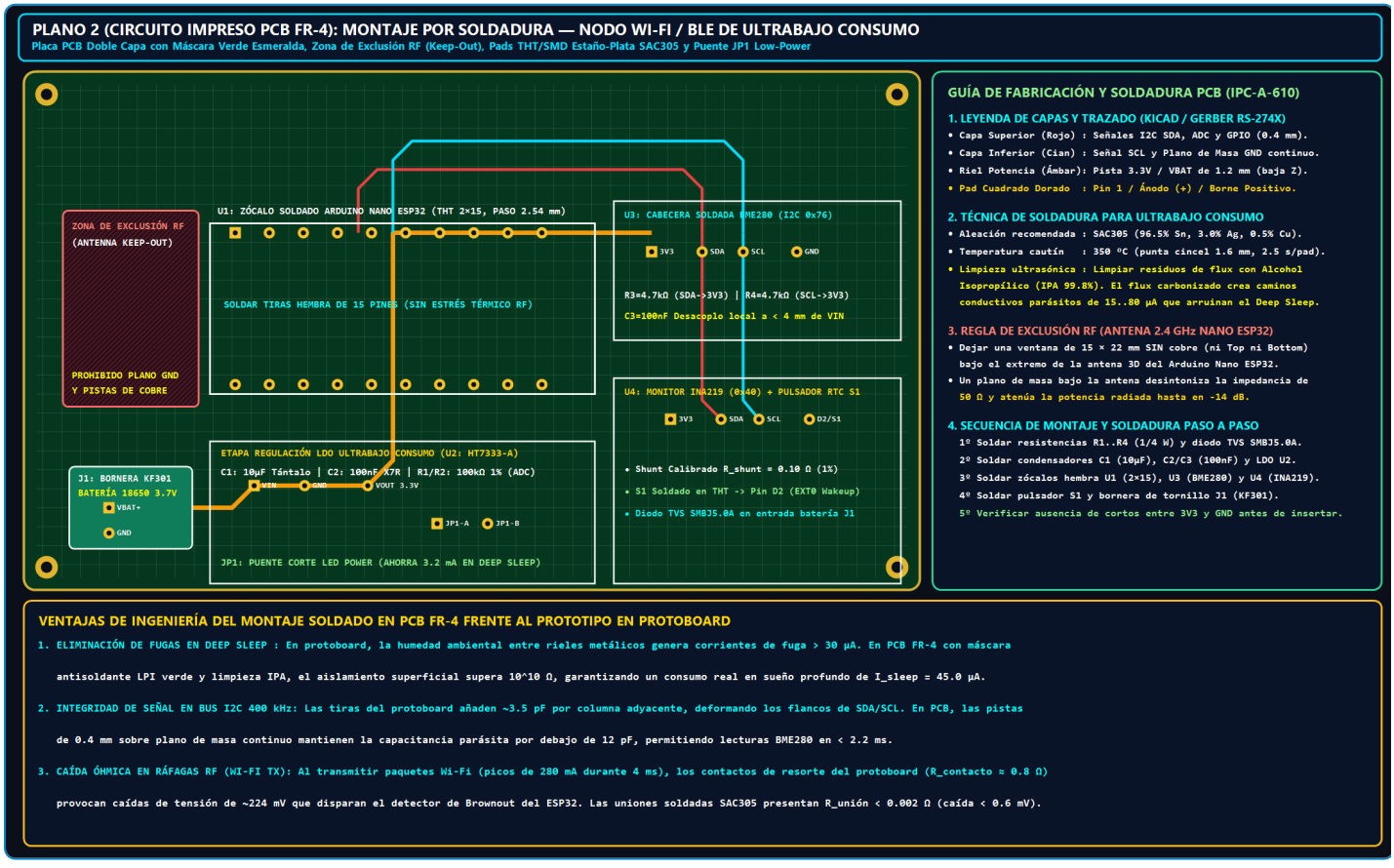


1.1 Resumen de Asignación de Pines del Arduino Nano ESP32 (`3.3 V` Nativo)

Callout en Plano	Subsistema / Módulo	Pines en Protoboard	Parámetros Eléctricos y de Firmware
①	**Arduino Nano ESP32 (** `ABX00083` **)**	Riel `3V3` y `GND`	SoC `ESP32-S3` con $8\ KB$ de memoria RTC (`RTC_DATA_ATTR`)
②	**Sensor Ambiental Bosch** `BME280`	`SDA -> A4` / `SCL -> A5`	Dirección I^2C `0x76` , operado en `MODE_FORCED` ($0.1\ \mu A$ en reposo)
③	**Monitor de Corriente** `INA219`	`SDA -> A4` / `SCL -> A5`	Dirección I^2C `0x40` , resistencia Shunt $R_{shunt} = 0.10\ \Omega\ (1\%)$
④	**Batería** `18650` **+ LDO** `HT7333-A`	Salida `3.30 V` → `Pin 3V3`	Corriente de reposo $I_q = 2.0\ \mu A$ y divisor $100\ k\Omega/100\ k\Omega$ hacia `A0`
⑤	**Antena Cerámica / 3D** `2.4 GHz`	Extremo izquierdo del módulo	Libre de cables puente y planos metálicos a $15\ mm$ a la redonda
⑥	**Despertar RTC (** `Timer` **+** `EXT0` **)**	Pulsador `S1` → `Pin D2`	`esp_sleep_enable_timer_wakeup(300e6)` + `esp_sleep_enable_ext0_wakeup(GPIO_NUM_2, 0)`

2. Plano 2 (Circuito Impreso `PCB FR-4`): Montaje por Soldadura del Nodo de Ultrabajo Consumo

📐 Plano 2: Placa de Circuito Impreso PCB FR-4 Doble Capa con Montaje por Soldadura del Nodo Inalámbrico

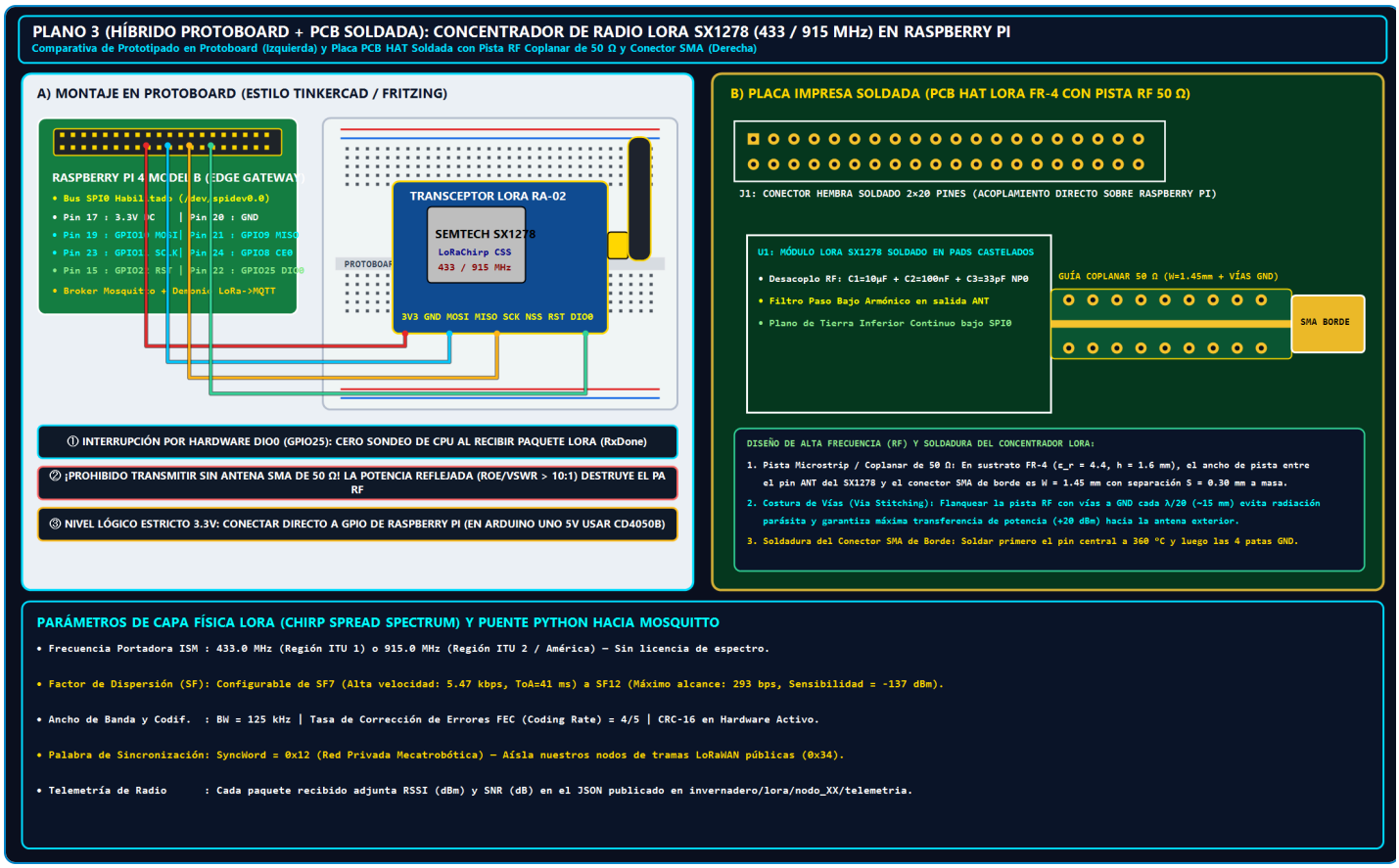


2.1 Especificaciones de Fabricación PCB (`KiCad` / `IPC-2221`) y Soldadura (`IPC-A-610`)

Elemento en la Placa PCB	Especificación Geométrica y Metalúrgica	Razón de Ingeniería (RF y Ultrabajo Consumo)
Zona de Exclusión RF (`Keep-Out`)	Ventana de 15×22 mm **sin cobre** en `Top` ni `Bottom` bajo la antena del Nano ESP32	Evita desintonizar la antena de 2.4 GHz (un plano GND debajo atenúa hasta -14 dB)
Puente Soldable `JP1` (Corte LED)	Dos pads dorados de 2.0×2.0 mm en serie con el LED indicador de encendido	Abrir `JP1` en despliegue final elimina el consumo fijo de 3.2 mA del LED Power
Limpieza de Flux con `IPA 99.8%`	Lavado con cepillo antiestático tras soldar con aleación `SAC305` a $350\ ^\circ$C	Los residuos de flux forman resistencias parásitas superficiales que drenan $15..80\ \mu$A
Zócalos Hembra `THT` (`U1, U3, U4`)	Agujeros metalizados $\varnothing 1.0$ mm con anillo anular dorado $\varnothing 1.8$ mm	Permite sustituir el Nano ESP32 o el `BME280` en campo en 5 segundos sin desoldar

3. Plano 3 (Híbrido Protoboard + `PCB HAT Soldada`): Concentrador `LoRa SX1278` en Raspberry Pi

📐 Plano 3: Montaje en Protoboard y Placa PCB HAT Soldada con Guía de Onda Coplanar de 50 Ohmios del Concentrador LoRa SX1278



4. Plano 4: Arquitectura de Perfil BLE GATT (`0x181A`) y Máquina de Estados con Backoff Exponencial

📐 Plano 4: Jerarquía del Servidor BLE GATT 0x181A y Máquina de Estados de Reconexión Wi-Fi/MQTT

PLANO 4: ARQUITECTURA DE PERFIL BLE GATT (SERVICIO 0x181A) Y MÁQUINA DE ESTADOS WI-FI / MQTT CON BACKOFF EXPONENCIAL

Jerarquía Servidor Periférico GATT (Arduino Nano ESP32) ↔ Cliente Central Asíncrono Bleak (Raspberry Pi) ↔ Broker Eclipse Mosquitto

A) ARQUITECTURA BLUETOOTH LOW ENERGY (BLE 5.0): SERVIDOR GATT 0x181A ↔ PASARELA ASÍNCRONA BLEAK (PYTHON)

1. PERIFÉRICO / SERVIDOR GATT (NANO ESP32)

- Capa GAP (Anuncio): Nombre 'NODO_BLE_INV_01'
  Intervalo Advertising = 100 ms (Canales 37, 38 y 39)

SERVICIO AMBIENTAL ESTÁNDAR SIG: UUID = 0x181A

- ├ Característica Temperatura : UUID 0x2A6E (sint16, 0.01 °C)
  | Propiedades: READ | NOTIFY + Descriptor CCCD 0x2902
- ├ Característica Humedad Rel.: UUID 0x2A6F (uint16, 0.01 %RH)
  | Propiedades: READ | NOTIFY + Descriptor CCCD 0x2902
- ├ Característica Presión Atm.: UUID 0x2A6D (uint32, 0.1 Pa)
  | Propiedades: READ | NOTIFY + Descriptor CCCD 0x2902
- └ Empaquetado Little-Endian: struct.pack('<hHI', T, H, P)
  (Solo 8 bytes por notificación -> Ahorro de 92% de aire RF)

2. CLIENTE CENTRAL ASÍNCRONO EN RASPBERRY PI

- Pila Linux BlueZ (/dev/hci0) + Librería Python 'bleak'
- Bucle de eventos no bloqueante: asyncio.run(main())

SECUENCIA DE SUSCRIPCIÓN A NOTIFICACIONES (PUSH)

1º Escaneo pasivo: BleakScanner.find_device_by_name()

2º Conexión GATT : async with BleakClient(addr) as cli:

3º Activación Push: await cli.start_notify(UUID, callback)
(Escribe 0x0001 en el descriptor CCCD 0x2902 del ESP32)

4º Recepción por Evento: El ESP32 envía paquete ATT_HANDLE VALUE_NOTIFICATION solo cuando cambia la medición.

5º Desempaquetado: T_c = int.from_bytes(d[0:2],'little')/100

3. CONVERGENCIA EN BROKER MOSQUITTO

- Traducción transparente BLE Binario -> MQTT JSON
- Tópico: invernadero/ble/nodo_01/ambiental

PAYLOAD JSON NORMALIZADO EN EL BROKER:

```
{
  "timestamp": "2026-11-01T12:00:05Z",
  "enlace_rf": "BLE_5.0_GATT_0x181A",
  "temperatura_c": 23.45,
  "humedad_rh": 64.20,
  "presion_hpa": 1013.25,
  "bateria_v": 3.98
}
```

- QoS = 1 | Retain = False | LWT Activo

B) MÁQUINA DE ESTADOS FINITOS (FSM) DE RECONEXIÓN WI-FI / MQTT CON BACKOFF EXPONENCIAL Y JITTER ALEATORIO

ESTADO 1: CONECTANDO_WIFI

- WiFi.mode(WIFI_STA)
- WiFi.begin(SSID, PASS)
- Timeout estricto = 8.0 s
- Si falla -> Incrementar k

ESTADO 2: CONECTANDO_MQTT

- Validar IP por DHCP
- mqtt.setWill(LWT, QoS 1)
- mqtt.connect(ID, Usr, Pwd)
- Suscribir tópico comando

ESTADO 3: OPERATIVO (TX/RX)

- Leer BME280 (Forced Mode)
- Leer V_bat + INA219
- Publicar JSON (QoS 1)
- Reiniciar intentos k = 0

ESTADO 4: BACKOFF + DEEP SLEEP

- T_espera = min(Tmax, T0·2^k)+J
- Si k >= 5 caídas seguidas:
- Entrar en Deep Sleep RTC
- Proteger celda Li-Ion 18650

LEY MATEMÁTICA DE RETROCESO EXPONENCIAL CON FLUCTUACIÓN ESTOCÁSTICA (JITTER):

$T_{espera}(k) = \min(T_{max} , T_0 \times 2^k) + U(0, J_{max})$ donde $T_0 = 2.0$ s , $T_{max} = 60.0$ s , $J_{max} = 1.0$ s

- Progresión de espera ante caída del Router/Broker: k=0 -> 2s | k=1 -> 4s | k=2 -> 8s | k=3 -> 16s | k=4 -> 32s | k>=5 -> 60s (Tope).
- ¿Por qué sumar Jitter aleatorio U(0, 1000 ms)? Tras un corte eléctrico en planta, evita que 50 nodos inalámbricos envíen su solicitud TCP SYN / MQTT CONNECT exactamente en el mismo milisegundo (Tormenta de Reconexión o Thundering Herd).
- Protección de Batería: Si tras k = 5 intentos el punto de acceso Wi-Fi sigue apagado, el nodo apaga la radio y duerme 300 s.

---

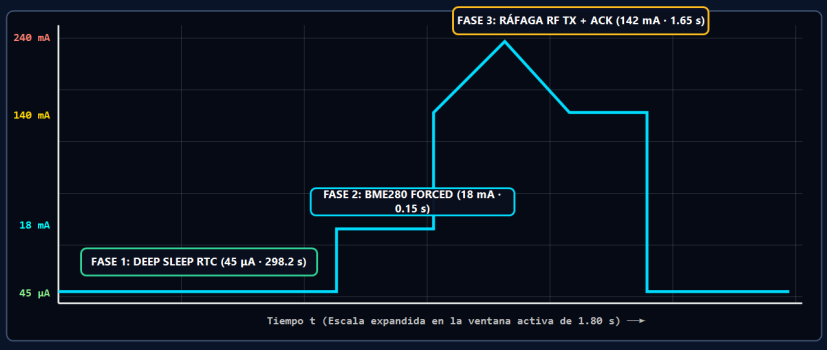
5. Plano 5: Perfil de Corriente $I(t)$ en `Deep Sleep` y Presupuesto de Enlace RF (`SF7..SF12`)

📐 Plano 5: Perfil Temporal de Corriente medido con INA219 y Presupuesto de Enlace RF

PLANO 5: PERFIL DE CORRIENTE I(t) EN SUEÑO PROFUNDO (DEEP SLEEP) Y PRESUPUESTO DE ENLACE RF (LINK BUDGET)

Caracterización Energética medida con INA219 (Ciclo T = 300 s) y Curvas de Sensibilidad LoRa (SF7..SF12) frente a Wi-Fi y BLE

A) PERFIL TEMPORAL DE CORRIENTE I(t) EN UN CICLO DE REPORTE (T_ciclo = 300 s = 5.0 MINUTOS)



BALANCE DE CARGA ELÉCTRICA Y AUTONOMÍA DE BATERÍA:

1. Corriente Media Ponderada en el Ciclo (T = 300 s):
   $I_{media} = (I_{on} \cdot t_{on} + I_{sleep} \cdot t_{sleep}) / T_{ciclo}$

- Caso A: Nodo Wi-Fi + MQTT (t_{on} = 1.80 s, I_{on} = 142 mA):
  $I_{media(Wi-Fi)} = (142 \times 1.80 + 0.045 \times 298.2) / 300 = 0.897$ mA
  $T_{vida(Wi-Fi)} = (3000$ mAh × 0.80) / 0.897 mA = 111.5 Días
- Caso B: Nodo LoRa SX1278 SF7 o BLE GATT (t_{on} = 0.25 s, I_{on} = 95 mA):
  $I_{media(LoRa)} = (95 \times 0.25 + 0.045 \times 299.75) / 300 = 0.124$ mA
  $T_{vida(LoRa)} = (3000$ mAh × 0.80) / 0.124 mA = 19,354 h ≈ 2.2 Años
- Caso C: Sin Deep Sleep (Radio Wi-Fi encendida 24/7 a 125 mA):
  $T_{vida(Continuo)}$ = 2400 mAh / 125 mA = 19.2 Horas (¡Inviable!)

B) PRESUPUESTO DE ENLACE DE RADIOFRECUENCIA (RF LINK BUDGET) Y FACTORES DE DISPERSIÓN LORA (SF7 A SF12)

ECUACIONES DE PROPAGACIÓN Y MARGEN DE DESVANECIMIENTO:

1. Potencia Recibida (Ecuación de Enlace en dBm):
   $P_{rx(dBm)} = P_{tx(dBm)} + G_{tx(dBi)} + G_{rx(dBi)} - FSPL(dB) - L_{misc(dB)}$

2. Pérdida en Espacio Libre de Friis (FSPL):
   $FSPL(dB) = 20 \cdot \log10(d_{km}) + 20 \cdot \log10(f_{MHz}) + 32.44$

3. Ventaja Espectral de Sub-GHz (433 MHz vs 2400 MHz a igual distancia):
   $\Delta FSPL = 20 \cdot \log10(2400 / 433) = +14.87$ dB menos atenuación en 433 MHz

4. Margen de Enlace (Link Margin / Fade Margin):
   $M_{enlace(dB)} = P_{rx(dBm)} - S_{min(dBm)} >= 15.0$ dB (Criterio Industrial)

- Ejemplo LoRa SF10 a d = 3.0 km (f=915 MHz, P_{tx}=+17 dBm, G=2 dBi):
  FSPL = 9.54 + 59.22 + 32.44 = 101.2 dB | P_{rx} = 17+4-101.2-12 = -92.2 dBm
  Como $S_{min(SF10)}$ = -132 dBm -> Margen M = -92.2 - (-132) = +39.8 dB (¡Sólido!)

MATRIZ COMPARATIVA DE TECNOLOGÍAS Y FACTORES LORA (BW = 125 kHz, Payload = 24 B):

TECNOLOGÍA / MODO	BANDA RF	VELOCIDAD	SENSIBILIDAD	TIEMPO AIRE	ALCANCE CAMPO
Wi-Fi 802.11n	2.4 GHz	72 Mbps	-75 dBm	1.5 ms	30 a 90 m
BLE 5.0 (GATT 1M)	2.4 GHz	1 Mbps	-95 dBm	0.4 ms	15 a 60 m
LoRa SF7 (2^7)	433/915M	5469 bps	-123 dBm	56.6 ms	1.5 a 2.5 km
LoRa SF8 (2^8)	433/915M	3125 bps	-126 dBm	102.9 ms	2.5 a 4.0 km
LoRa SF9 (2^9)	433/915M	1758 bps	-129 dBm	185.3 ms	4.0 a 6.0 km
LoRa SF10 (2^10)	433/915M	977 bps	-132 dBm	370.7 ms	6.0 a 8.5 km
LoRa SF11 (2^11)	433/915M	537 bps	-134.5 dBm	741.4 ms	8.5 a 11 km
LoRa SF12 (2^12)	433/915M	293 bps	-137 dBm	1482.8 ms	11.0 a 15 km