

Arduino para supervivencia: sensores y alarmas

Autor: EA4IPV

Fecha: 23/03/2026

Categoría: Electrónica y Energía

Etiquetas: Sin etiquetas

Arduino para supervivencia: sensores y alarmas

Las placas Arduino, con un consumo de apenas 20-50 mA a 5 V, permiten construir sistemas autónomos de vigilancia perimetral, monitorización ambiental y alarmas que funcionan durante semanas con una batería de 12 V y un panel solar pequeño. Este artículo cubre circuitos reales, código funcional y cálculos de consumo para mantener operativos estos sistemas fuera de la red eléctrica.

Plataformas recomendadas y consumo real

No todas las placas Arduino consumen lo mismo. Para supervivencia, el consumo en reposo (sleep) es el dato crítico, porque el sistema pasa la mayor parte del tiempo esperando un evento.

Placa

Voltaje

Consumo activo

Consumo en sleep

Precio aprox.

Arduino Uno R3

5 V

46 mA

35 mA (idle)

10-12 €

Arduino Pro Mini 3.3V

3.3 V

4.5 mA

0.057 mA (power-down)

3-5 €

ATtiny85 (chip solo)

3-5 V

5 mA

0.001 mA (power-down)

1-2 €

ESP32 (con WiFi)

3.3 V

240 mA (WiFi)
0.01 mA (deep sleep)
5-8 €

Para aplicaciones de supervivencia, el Arduino Pro Mini 3.3V es la mejor opción general: bajo consumo, compatible con sensores estándar y programable con el IDE de Arduino. Si necesitas comunicación inalámbrica, el ESP32 en deep sleep es imbatible.

Cálculo de autonomía: Con una batería de 12 V / 7 Ah (tipo alarma) y un regulador a 3.3 V (eficiencia 90%): energía disponible = $7 \text{ Ah} \times 12 \text{ V} \times 0.9 / 3.3 \text{ V} \approx 22.9 \text{ Ah}$ a 3.3 V. Un Pro Mini en sleep (0.057 mA) con activación cada 8 s consumiendo 5 mA durante 100 ms: consumo medio = $0.057 + (5 \times 0.1/8) \approx 0.12 \text{ mA}$. Autonomía = $22\,900 \text{ mAh} / 0.12 \text{ mA} \approx 190\,000 \text{ horas} \approx 21 \text{ años teóricos}$. En la práctica, con autodescarga de la batería, espera al menos 6-12 meses.

Sensores útiles para supervivencia

Los sensores que más valor aportan en un escenario de supervivencia son los que detectan amenazas o monitorizan recursos críticos.

PIR HC-SR501 (movimiento): Detecta cuerpos calientes (personas, animales) a 3-7 metros con ángulo de 120°. Consumo: 65 μA en reposo. Alimentación: 5-20 V. Salida digital directa. Ideal para vigilancia perimetral.

DHT22 (temperatura y humedad): Rango: -40 a 80 °C (± 0.5 °C), 0-100% HR (± 2 -5%). Consumo: 1.5 mA midiendo, 50 μA en standby. Lectura cada 2 s mínimo. Útil para monitorizar almacenes de comida.

HC-SR04 (ultrasonido/distancia): Mide distancia de 2 a 400 cm con precisión de 3 mm. Consumo: 15 mA activo. Permite detectar nivel de agua en depósitos o intrusiones en pasos estrechos.

MQ-2 / MQ-7 (gases): El MQ-2 detecta GLP, propano, metano e hidrógeno. El MQ-7 detecta monóxido de carbono. PROBLEMA: consumen 150-180 mA por el calentador interno. Solo activar para mediciones puntuales (necesitan 60 s de precalentamiento).

SW-420 (vibración): Sensor piezoeléctrico que detecta vibraciones y golpes. Consumo despreciable (Circuito de alarma perimetral completa)

Un sistema práctico de alarma perimetral con Arduino Pro Mini necesita: 2-4 sensores PIR, un buzzer piezoeléctrico (o sirena de 12 V mediante relé), un LED indicador y una batería de 12 V con regulador.

Esquema de conexiones: los PIR se conectan a pines digitales D2-D5 (con interrupciones en D2 y D3 para despertar del sleep). El buzzer al pin D9 (PWM para variar el tono). Un LED de estado al pin D13. El relé de sirena al pin D7.

El código usa la librería `avr/sleep.h` para poner el microcontrolador en modo `POWER_DOWN` y despertarlo mediante interrupciones externas en los pines D2/D3. Cuando un PIR detecta movimiento, genera un

pulso HIGH que dispara la interrupción, despierta al Arduino, activa la sirena durante 30 segundos y vuelve a dormir.

Componente

Consumo

Cantidad

Total

Arduino Pro Mini 3.3V (sleep)

0.057 mA

1

0.057 mA

PIR HC-SR501 (standby)

0.065 mA

4

0.26 mA

LED (apagado)

0 mA

1

0 mA

Regulador 3.3V quiescente

0.005 mA

1

0.005 mA

TOTAL en reposo

0.322 mA

Con una batería de 12 V / 7 Ah: autonomía en reposo = $7000 \text{ mAh} \times (12/3.3) \times 0.9 / 0.322 \approx 71\,000$ horas ≈ 8 años teóricos. En la práctica, contando activaciones de sirena y autodescarga: 3-6 meses sin recarga.

Con un panel solar de 5 W, funcionamiento indefinido.

Sobre los falsos positivos: Los sensores PIR son sensibles a cambios de temperatura rápidos (sol directo, viento caliente). En exteriores, oriéntalos hacia el norte y protégelos con un tubo de PVC que limite su ángulo de visión. Ajusta la sensibilidad con el potenciómetro del sensor al mínimo que aún detecte personas a 3 metros.

Alimentación solar para Arduino

Un sistema Arduino autónomo necesita: panel solar → controlador de carga → batería → regulador → Arduino.

Panel solar: Para el sistema de alarma descrito (0.32 mA reposo + activaciones), un panel de 1 W (6 V / 170 mA) es suficiente. Con 4 horas de sol efectivo al día genera $170 \text{ mA} \times 4 \text{ h} = 680 \text{ mAh}$, muy por encima del consumo diario de $0.32 \times 24 = 7.7 \text{ mAh}$.

Controlador de carga: Un módulo TP4056 (para Li-ion) o un CN3065 (para LiFePO4) regulan la carga y protegen la batería. Para baterías de plomo, un controlador PWM básico de 6/12 V es suficiente. Coste: 1-3 €.

Regulador de voltaje: Usa reguladores conmutados (buck), NO lineales (7805). Un AMS1117-3.3 tiene quiescent current de 5 mA, que es inaceptable. Un HT7333 tiene quiescent de 4 μA , ideal para ultra-bajo consumo. Eficiencia >90% frente al 50-60% de un regulador lineal.

⚠ Advertencia: Esta información es orientativa y educativa. En situaciones de emergencia real, consulte a profesionales cualificados siempre que sea posible. No ponga en riesgo su vida ni la de otros sin la formación adecuada.