MANUAL DE ESTUDIO · MECATRÓNICA AUTOMOTRIZ

Unidades de control electrónico (ECU) y AUTOSAR

Capítulo 02 · En el video: **minuto 15:47** · Duración 14:31

Objetivo. Seguir una señal desde el sensor hasta el actuador dentro de una ECU y verificar su software en tiempo real.

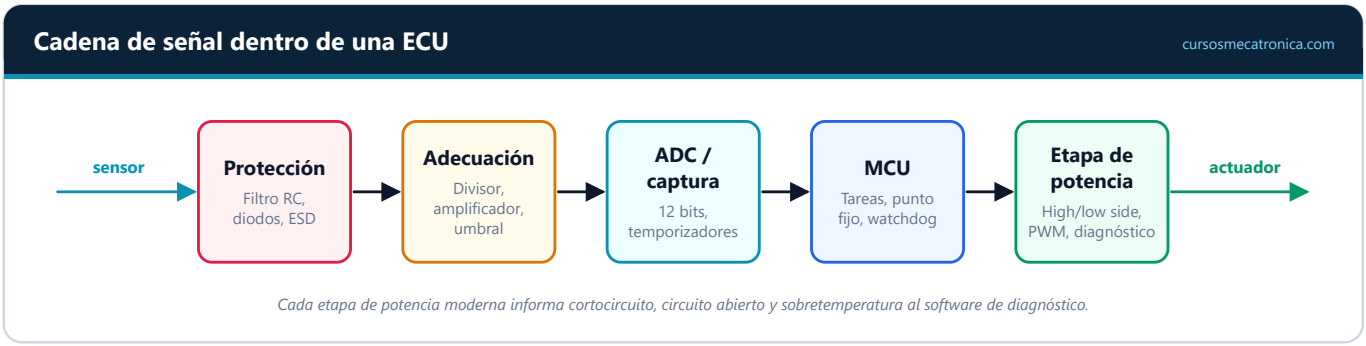
Ideas clave

Diseño constructivo

Una ECU soporta de −40 a 125 °C (más cerca del motor), vibraciones, humedad, sal y perturbaciones eléctricas. Por eso usa carcasas selladas, componentes calificados AEC-Q100/Q200 y protecciones en cada pin.

Entradas y conversión

Las entradas analógicas pasan por filtros y divisores hacia un ADC de 10 a 12 bits; las de frecuencia (régimen, velocidad) van a temporizadores de captura. Un sensor NTC con resistencia de pull-up es el caso típico.



Bloques de hardware de una unidad de control.

Etapas de potencia

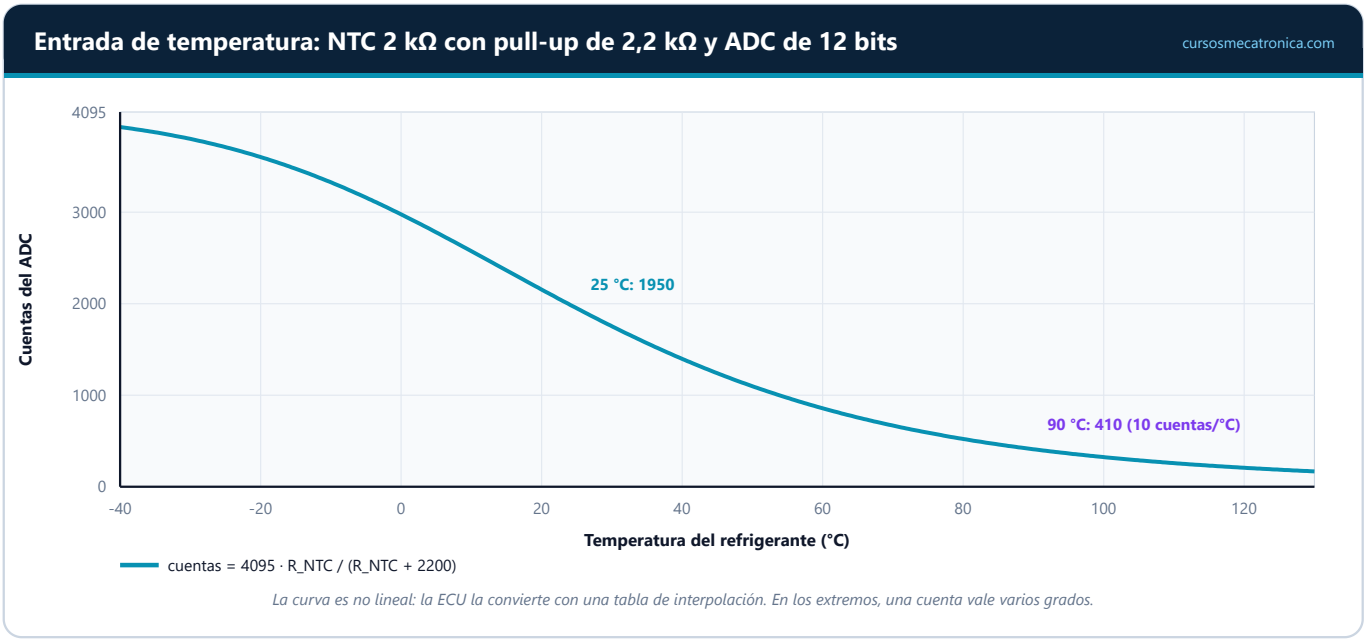
Interruptores de lado alto (al positivo) o bajo (a masa) con MOSFET inteligentes manejan lámparas, relés y válvulas; el PWM regula la potencia media. Cada etapa diagnostica cortocircuitos, circuitos abiertos y sobretemperatura; la forma de la corriente (firma) permite detectar si una válvula se movió.

Microcontrolador y memorias

Núcleos múltiples en paralelo (lockstep) para seguridad, flash para el programa y los mapas de calibración, RAM y EEPROM emulada para adaptaciones. El sincronismo angular con el cigüeñal dispara inyección y encendido en el grado exacto.

Punto fijo y tareas

Muchos cálculos usan enteros escalados (Q15, Q31) por velocidad y determinismo; hay que cuidar desbordes y saturar. El sistema operativo ejecuta tareas periódicas (1, 10, 100 ms) por prioridad; un watchdog de ventana reinicia la ECU si el software se cuelga o se acelera.



Salida de ecu_entradas_y_tareas.py.

AUTOSAR y validación

AUTOSAR Classic separa aplicación, entorno de ejecución (RTE) y software de base, para reutilizar componentes entre fabricantes; AUTOSAR Adaptive se usa en computadores centrales con POSIX y Ethernet. El software se calibra con ASAM MCD (XCP, A2L) y se prueba en bancos HIL que simulan el vehículo alrededor de la ECU real.

Fórmulas y relaciones

Relación	Expresión	Cómo leerla
Divisor NTC	$V = V_{ref} \cdot R_{NTC} / (R_{NTC} + R_{pull})$	NTC a masa.
Modelo β	$R(T) = R_{25} \cdot e^{\beta(1/T - 1/298{,}15)}$	T en kelvin.
ADC	$n = V / V_{ref} \cdot (2^N - 1)$	12 bits: 0 a 4095.
Producto Q15	$(a \cdot b) >> 15$	Con saturación a ±32 767.
Planificabilidad (RMS)	$U = \Sigma C_i / T_i \leq n(2^{1/n} - 1)$	Cota suficiente de Liu y Layland.

Ejemplo resuelto: resolución de la temperatura del motor

1. NTC de 2 kΩ a 25 °C (β = 3500) con pull-up de 2,2 kΩ a 5 V y ADC de 12 bits.
2. A 25 °C: 1950 cuentas; a 90 °C, R = 245 Ω y 410 cuentas.
3. Entre 89,5 y 90,5 °C cambian 10 cuentas: 0,1 °C por cuenta.
4. Tres tareas (0,2/1 ms, 2,5/10 ms, 20/100 ms) cargan la CPU al 65 %, bajo la cota de 78 %.

La medición es fina donde importa (zona de trabajo) y la carga de CPU es planificable. ecu_entradas_y_tareas.py simula además un watchdog de ventana que reinicia la ECU cuando una tarea se atrasa a 14,5 ms.

AUTOSAR Classic frente a Adaptive

	Classic	Adaptive
Hardware	Microcontrolador	Procesador de alto rendimiento
Sistema operativo	OSEK/AUTOSAR OS, estático	POSIX, dinámico
Comunicación	CAN, LIN, FlexRay	Ethernet orientado a servicios (SOME/IP)
Uso	Motor, frenos, carrocería	Computador central, ADAS, OTA

Código del capítulo

Los archivos completos vienen en el paquete ZIP del capítulo (carpeta `codigo/`).

ecu_entradas_y_tareas.py — NTC con ADC de 12 bits, producto Q15 con saturación, carga de CPU y watchdog de ventana.

```
"""
ecu_entradas_y_tareas.py — Capítulo 02: entrada NTC con ADC de 12 bits, aritmética Q15,
carga de CPU con planificación de prioridades fijas y watchdog de ventana.
"""
import math

VREF, R_PULL, R25, BETA = 5.0, 2200.0, 2000.0, 3500.0


def r_ntc(temp_c):
    return R25 * math.exp(BETA * (1 / (temp_c + 273.15) - 1 / 298.15))


def cuentas_adc(temp_c, bits=12):
    v = VREF * r_ntc(temp_c) / (r_ntc(temp_c) + R_PULL)    # NTC abajo, pull-up arriba
    return round(v / VREF * (2 ** bits - 1))


def q15(x):
    return max(-32768, min(32767, round(x * 32768)))


def q15_mul(a, b):
    return max(-32768, min(32767, (a * b) >> 15))


def carga_cpu(tareas):
    u = sum(c / t for c, t in tareas)
    n = len(tareas)
    return u, n * (2 ** (1 / n) - 1)


def watchdog(disparos_ms, ventana=(8.0, 12.0)):
    """Watchdog de ventana: cada disparo debe llegar entre 8 y 12 ms después del anterior."""
    for previo, actual in zip(disparos_ms, disparos_ms[1:]):
        dt = actual - previo
        if not ventana[0] <= dt <= ventana[1]:
            return f'RESET en t = {actual:.1f} ms (intervalo {dt:.1f} ms)'
    return 'sin reset'


if __name__ == '__main__':
    print('NTC 2 kΩ a 25 °C, B = 3500, pull-up 2,2 kΩ, ADC 12 bits')
    for t in (-40, 25, 90, 120):
```

```
        print(f'  {t:4d} °C -> R = {r_ntc(t):8.0f} Ω  cuentas = {cuentas_adc(t)}')
    sens = cuentas_adc(89.5) - cuentas_adc(90.5)
    print(f'  Resolución a 90 °C: {sens} cuentas por °C -> {1 / sens:.3f} °C por cuenta')

    a, b = q15(0.75), q15(-0.4)
    print(f'\nQ15: 0,75 × (−0,4) = {q15_mul(a, b) / 32768:.6f} (exacto −0,3)')
    print(f'Q15 saturado: 0,9 + 0,9 -> {min(32767, q15(0.9) + q15(0.9)) / 32768:.5f}')

    tareas = [(0.2, 1.0), (2.5, 10.0), (20.0, 100.0)]     # (ms de cómputo, periodo ms)
    u, cota = carga_cpu(tareas)
    print(f'\nCarga de CPU U = {100 * u:.1f} %; cota de Liu-Layland para 3 tareas = {100 * cota:.1f} %'
          f' -> {"planificable" if u <= cota else "revisar"}')
    print('Watchdog, disparos cada 10 ms:', watchdog([0, 10, 20, 30, 40]))
    print('Watchdog, una tarea se cuelga  :', watchdog([0, 10, 20, 34.5, 44.5]))
```

Errores frecuentes

- Multiplicar en Q15 sin saturar y obtener un cambio de signo por desborde.
- Dimensionar el watchdog solo con tiempo máximo: un software que corre "demasiado rápido" también es una falla.
- Suponer que la curva NTC es lineal.

Autoevaluación

1. ¿Qué hace un watchdog de ventana si se le dispara antes de tiempo?
2. ¿Qué capa de AUTOSAR comunica la aplicación con el software de base?
3. ¿Qué prueba un banco HIL?

Respuestas. 1. Reinicia la ECU, igual que si llega tarde. **2.** El RTE. **3.** La ECU real con el vehículo simulado a su alrededor.