حلول هندسية للمشاكل الست — V5-Pro Ultra

الإصدار: Solutions v2.0 | **التاريخ:** 29-04-2026

> هذا المستند يقدّم حلولاً ملموسة وقابلة للتنفيذ، لا توصيات عامة. كل حل مرتبط بـ: **قطعة محددة، إجراء PCB، سطر كود، أو رقم قياس**.

المشكلة 1 — الحساسية المفرطة للضوضاء (Analog Noise)

تحليل الجذر

الكسب الكلي 80,000× يحوّل أي 1µV ضوضاء على المدخل إلى 80mV على المخرج. مصادر الضوضاء الرئيسية:

- **رقمية:** ESP32 Wi-Fi/BLE (2.4GHz)، ساعات K210 (400MHz)، STM32H7 (480MHz).
- **طاقة:** Switching ripple على USB harmonics+، Ground bounce، 5V.
- **خارجية:** خطوط الكهرباء 50Hz، أبراج الإرسال.

الحل المركّب (5 طبقات دفاع)

1.1 عزل التغذية فعلياً

المرجع	القطعة	الإجراء
بديل لـ U16	TPS7A4700 (4µVrms)	استبدال AMS1117 لجزء الأنالوج بـ **LDO منخفض الضوضاء**
إضافة	Murata BLM18PG471 600Ω + 10µF + 100nF	إضافة (L+C+C) **π-Filter** قبل كل IC أنالوج
تعديل layout	جسر يدوي	فصل **3.3V_ANA+** عن **3.3V_DIG+** بـ Ferrite Bead + 0Ω
معتمد	NetTie	**Star-grounding واحد فقط** تحت U8 (ADS1256) — مذكور في v2

1.2 إخفاء ESP32 RF فيزيائياً

- ضع **علبة معدنية صغيرة (RF Shield Can)** فوق ESP32-S3 ملحومة بالأرضي. التكلفة ~$1.
- وجّه هوائي ESP32 إلى الجانب البعيد عن AFE (≥ 30mm).

- إذا أمكن: استخدم **Module مع هوائي خارجي u.FL** بدل المتكامل.

1.3 توقيت ذكي (Time-Multiplexing)

المبدأ: لا تجعل ESP32 يبث BLE وقت قراءة AFE.

Plain Text

```
// STM32 يدير الجدولة:
void afe_sample_burst(void) {
    esp32_set_silent(true);   // GPIO إلى ESP32 = توقف عن البث
    HAL_Delay(2);              // انتظر TX/RX يهدأ
    ads1256_read_burst(1024);  // 1024 عينة سريعة
    esp32_set_silent(false);   // أعد البث
    esp32_send_buffer();       // أرسل الدفعة
}
```

هذا يقلّل ضوضاء BLE من 5mV → 100mV على الخرج.

1.4 فلتر Notch رقمي 50/60Hz

في STM32H743 عبر CMSIS-DSP:

Plain Text

```
// Biquad Notch Filter @50Hz, fs=2500Hz
arm_biquad_casd_df1_inst_f32 notch;
float coeffs[5] = { 0.9921f, -1.9777f, 0.9921f, 1.9777f, -0.9842f };
float state[4] = {0};
arm_biquad_cascade_df1_init_f32(&notch, 1, coeffs, state);
arm_biquad_cascade_df1_f32(&notch, raw_buf, filt_buf, BLOCK_SIZE);
```

1.5 معيار قياس النجاح

- **قبل الحلول:** Noise Floor متوقع 30–80 nV/√Hz.
- **بعد الحلول:** يجب أن يصل **< 10 nV/√Hz** (هدف Datasheet لـ AD8429).
- **القياس الفعلي** على Perfboard أولاً (راجع AFE_Prototype_NoiseFloor_Guide).

المشكلة 2 — تعقيد البرمجيات وتزامن البيانات

تحليل الجذر

3 معالجات × 4 بروتوكولات = 12 احتمال فشل. النقاط الحرجة:

- ESP32 ↔ STM32 عبر UART قد يفقد بايتات إذا انشغل أحدهما.
- K210 ↔ STM32 عبر UART2 يحتاج تزامناً مع DMA.
- I^2C مشترك بين 5 قطع (RM3100, MPU9250, OLED, EEPROM, ADS1256, INA219).

الحل: معمارية Master/Slave واضحة + Priority Inversion Protection

2.1 توزيع المسؤوليات (Single Responsibility)

المعالج	يملك	لا يملك
STM32H743 (Master)	AFE، RM3100، التوقيت، MPU9250، EEPROM، INA219، SD، PGA281	لا BLE، لا كاميرا
ESP32-S3 (Bridge)	BLE 100 فقط، شفاف%	لا قراءة حساسات
K210 (Coprocessor)	الكاميرا، AI inference عند الطلب	لا تخزين

القاعدة الذهبية: ESP32 لا يفسّر بيانات. K210 لا ينطلق إلا بأمر صريح. STM32 هو القلب الوحيد.

2.2 RTOS مع أولويات مدروسة

استخدم **FreeRTOS على STM32H7** مع أولويات:

Plain Text

```
// أولوية أعلى = أهم
#define PRIO_AFE_ISR        7   // Hardware ISR، لا تحجبه أبداً
#define PRIO_RM3100_TASK    6   // 20Hz timing-critical
#define PRIO_GRADIO_CALC    5   // Differential math
#define PRIO_BLE_TX         4   // إرسال للهاتف
#define PRIO_SD_LOG         3   // Buffered write
#define PRIO_K210_MGR       2   // Camera on-demand
#define PRIO_OLED_UI        1   // واجهة محلية
#define PRIO_IDLE           0
```

2.3 فصل I^2C إلى Buses

بدل I^2C واحد لكل القطع:

Bus	السرعة	الأجهزة
I^2C1 (Fast 400kHz)	حساسات	RM3100×2, MPU9250

I²C2 (Standard 100kHz)	تحكم	OLED, EEPROM, INA219, PGA281 (إذا I²C)
SPI1 (10MHz)	High-speed	ADS1256
SPI2 (25MHz)	Storage	SD Card

هذا يمنع تعارض timing ويرفع reliability.

2.4 بروتوكول STM32↔ESP32 موثوق

استخدم نفس **إطار BLE_Spec_v1.0** على UART:

- DMA Circular Buffer 2KB.
- CRC-16 على كل إطار.
- ACK مطلوب لحزم التحكم فقط (0x2X).
- إذا فُقد إطار → SEQ skip → التطبيق يعرض إنذار.

2.5 Watchdogs على ثلاثة مستويات

```
// STM32: IWDG (Independent Watchdog)
HAL_IWDG_Init(&hiwdg);  // reset @ 4 ثوانٍ بدون نبض

// ESP32: Task Watchdog
esp_task_wdt_init(5, true);

// K210: Soft watchdog من STM32
// إذا K210 لم يرد على ping 2 خلال s → أعد تشغيله عبر EN GPIO
```

2.6 معيار النجاح

- **0% فقد إطارات** خلال جلسة 10 دقائق على معدل 20Hz.
- **Latency < 50ms** من حساس إلى الهاتف.
- **CPU usage < 60%** على STM32 (هامش للتوسعة).

المشكلة 3 — استهلاك الطاقة والحرارة (Thermal Drift)

تحليل الجذر

RM3100 له **Thermal Coefficient = 200ppm/°C**. إذا سخنت اللوحة من 25°C إلى 55°C:

- الانحراف = 30°C × 200ppm = 0.6% = **300nT خطأ ثابت** (لكامل النطاق 50µT).

الحل المركّب

3.1 استبدال LDOs الحرارية بـ Buck فعّال

القديم	الجديد	الميزة
AMS1117-3.3 (LDO، يبدد 1.5W)	**TPS62130A** (Buck 90% efficiency، يبدد 0.15W)	حرارة أقل 10×
RT9013 1.8V (LDO 0.3 صغير، W)	**TPS62740** (Buck DCS-Control، 0.03W)	استهلاك ÷ 10

3.2 فصل RM3100 حرارياً عن المعالجات

- ضع RM3100 على **PCB منفصلة صغيرة** (Daughterboard 15×15mm) موصولة بكابل FFC قصير.
- أو: على PCB الرئيسية، اعزله بـ **Slot في FR-4** (شق ميكانيكي عرض 1mm) يفصل المنطقة الحرارية.

3.3 Compensation برمجي للحرارة

RM3100 لديه Temperature Register:

Plain Text

```
// قراءة درجة حرارة RM3100 مدمجة
int16_t temp_raw = rm3100_read_temp();
float temp_c = (temp_raw / 16.0f) + 25.0f;

// Compensation linear للحقل
const float TC_X = 200e-6f;  // ppm/°C من Datasheet
float mag_x_corrected = mag_x_raw * (1.0f - TC_X * (temp_c - 25.0f));
```

3.4 Sleep أذكى (Power Modes)

الحالة	الاستهلاك	المتى
Active Scan	350mA	المسح الفعلي
Idle Hold	80mA	عرض القراءة الأخيرة

Deep Sleep	0.5mA	بعد 5 دقائق خمول

Plain Text

```
void enter_deep_sleep(void) {
    HAL_PWR_DisableBkUpAccess();
    rm3100_set_standby();
    ads1256_set_powerdown();
    k210_disable();
    esp32_signal_sleep();
    HAL_PWR_EnterSTANDBYMode();
}
```

3.5 Thermal Imaging Test

قبل الإطلاق: **صوّر اللوحة بكاميرا حرارية FLIR** بعد ساعة تشغيل. أي بقعة > 50°C = إعادة تصميم.

3.6 معيار النجاح

- **Thermal drift < 50nT** بعد ساعة تشغيل في غرفة 25°C.
  - **عمر بطارية > 8 ساعات** على Active Scan.
  - **زمن استيقاظ من Deep Sleep < 200ms**.

المشكلة 4 — صعوبة المعايرة (Gradiometer Mismatch)

تحليل الجذر

- إزاحة ميكانيكية 1mm بين الحساسين على بعد 30cm = 0.3% زاوية = ~150nT خطأ في الحقل الأرضي.
  - اختلاف Gain البرمجي بين الحساسين 0.5% = 250nT.
- Hard-Iron من الـ PCB 1000–5000 = (نحاس، فيرومنغنيتي) نفسها nT.

الحل: معايرة 3 مراحل (مصنع → ميدان → مستمر)

4.1 معايرة المصنع (Factory Calibration)

في طاولة عمل مع **Helmholtz Coil 3-axis**:

Plain Text

```
1. Zero-Field Test:    قياس bias داخل علبة μ-metal لكل حساس
2. Reference Field:    50 حقنμT بدقة، حساب Gain لكل محور
3. Cross-Coupling:     بناء مصفوفة تصحيح 3×3، Y على X قياس تأثير
4. 24 خزّن النتائج في LC256 EEPROM
```

خوارزمية:

Plain Text

```
typedef struct {
    int32_t bias[3];           // hard iron offset (nT)
    float scale[3][3];         // soft iron + gain matrix
    float temp_coef[3];        // ppm/°C per axis
    uint16_t serial;
    uint16_t crc;
} factory_cal_t;
```

4.2 معايرة الميدان (Field Calibration Wizard)

في التطبيق Android: **معالج 60 ثانية**:

Plain Text

```
1. ضع الجهاز بعيداً عن المعادن (3 متر).
2. أدر الجهاز في 8 شكل (Figure-8) 30 لمدةs.
3. ثبّته 10 s (شمال/جنوب/شرق/غرب) في كل اتجاه.
4. التطبيق يحسب Hard-Iron + Soft-Iron التراكمي.
5. يحفظ في EEPROM ويرسل تأكيداً.
```

الخوارزمية: **Magneto Calibration Ellipsoid Fit** (مفتوحة المصدر).

4.3 معايرة مستمرة (Continuous Auto-Cal)

أثناء العمل العادي:

Plain Text

```
// كل دقيقة، إذا الجهاز ساكن (يؤكد IMU):
if (mpu_motion_detected() == false) {
    update_running_avg(&hard_iron_estimate);
    if (delta > THRESHOLD) {
        save_to_eeprom();
        notify_user("معايرة محسّنة");
    }
}
```

4.4 Gradiometer Matching الإلكتروني

بدل الاعتماد على الميكانيك فقط:

```
Plain Text

// "Software Trim" بعد التركيب
float k_match = config.sensor2_correction;  // من factory cal
mag2_x_trimmed = mag2_x_raw * k_match;
gradient_x = mag1_x - mag2_x_trimmed;
```

4.5 تثبيت ميكانيكي

- استخدم **Carbon Fiber Tube** (لا حديد، لا ألمنيوم) بطول الـ baseline.
- ثبّت RM3100 بـ **براغي بلاستيكية Nylon**، لا تلصقها بالغراء العادي.
- أضف **علامات تحاذي بالليزر** على القاعدة لتأكيد عدم الانحراف.

4.6 معيار النجاح

- **Common-Mode Rejection > 60dB** في الحقل الأرضي.
- **Drift < 100nT/ساعة** بعد المعايرة.
- **زمن المعايرة الميدانية ⩽ 90 ثانية.**

المشكلة 5 — قيود التصميم الميكانيكي والـ PCB

تحليل الجذر

كارثي EMI = فقط 5cm يبعد TX coil قطعة سلبية مكدّسة، مع 20 IC + ~3 AFE = لـ 30×30mm.

الحل: توسعة + فصل + تخطيط طبقات احترافي (تم اعتماده في Hardware_Finalization_v2)

5.1 الأبعاد الجديدة المعتمدة

- اللوحة الرئيسية: **70mm 4-Layer × 50**.
- منطقة AFE: 50 × 25mm (ضعف الأصلي).
- منطقة Power: 50 × 20mm.
- منطقة Digital (STM32+ESP32+K210): 50 × 25mm.

5.2 ترتيب الطبقات الذهبي

Layer	المحتوى	الوظيفة
L1 (Top)	Components + Critical signals	معظم الإشارة قصيرة
L2 (Inner1)	**AGND (left) + DGND (right) Solid Pours**	عودة الأرضي بأقصر مسار
L3 (Inner2)	Power islands (+12V, +5V, +3.3V_A, +3.3V_D)	منع الـ IR drop
L4 (Bottom)	Secondary signals + GND fill	عزل إضافي

5.3 قواعد عزل EMI

Plain Text

```
1. TX Coil drive 35 ≤ يبعدmm عن AFE (مناطق منفصلة على PCB).
2. Mosfet IRF740 له Local Star-Ground واحدة بنقطة يلتقي بالرئيسي، منفصل.
3. حول AD8429 و AD797: Guard Ring (مسار GND مغلق على نفس الطبقة).
4. ساعات STM32H7 (25MHz) بـ محاطة GND vias 1 كلmm.
5. Differential pairs (USB, SPI ADS1256) محسوبة Impedance 90Ω.
```

5.4 RF Shielding Cans (مذكور في حل #1)

3 علب صغيرة:

- فوق AFE (15×30mm).
- فوق ESP32 (15×15mm).
- فوق K210 (20×20mm).

تكلفة الثلاث ~$3 من LCSC (مثلاً Wurth 36103305).

5.5 Stack-up مع JLCPCB

اطلب (FR-4 Standard 4-layer 1.6mm) **JLC04161H-7628** مع **Impedance Control**.

5.6 معيار النجاح

- **DRC في KiCad: 0 errors, 0 warnings**.
- **3D Preview من JLC 100 مطابق%**.
- **EMI scan بمسبار مغناطيسي**: لا قمم < -60 dBm في نطاق 100 kHz-100MHz.

المشكلة 6 — مشاكل Toolchain (KiCad ↔ Quilter/Flux.ai)

تحليل الجذر

- Quilter لا يدعم Edge.Cuts بعض إصدارات KiCad.
- Flux.ai يقصّ بعض القيود على Layer-stack.
- IPC-2581 export في KiCad 7 ما زال beta لبعض السمات.

الحل: اعتمد KiCad فقط للنماذج، Auto-router فقط للأقسام البسيطة

6.1 منهجية الإنتاج

المرحلة	الأداة	لماذا
Schematic	**KiCad 7.0 Eeschema**	مفتوح، مستقر
Component Library	**KiCad + SnapEDA + UltraLibrarian**	مكتبة موثقة
PCB Layout عام	**KiCad PCBnew**	تحكم كامل
Auto-routing عام	**يدوي 100%**	لا تستخدم Quilter للوحات Mixed-Signal
Auto-routing Digital فقط	Freerouting (في KiCad)	كافٍ للـ DGND area
3D Preview	KiCad + Step Export	تحقق ميكانيكي
Gerber + Drill	KiCad → ZIP → JLCPCB	مباشر
BOM + CPL	KiCad Plugin **JLC-PCB-Tools**	تنسيق صحيح للـ JLC

6.2 لماذا تجنّب Quilter/Flux.ai في v1؟

- AI auto-routers لا يفهمون فلسفة Star-grounding ولا منطقة AGND/DGND.
  - لوحات Mixed-Signal تحتاج عقلاً بشرياً يقرر كل trace حرج.
  - اقتصد وقتك: راجع **YouTube — Robert Feranec** لتعلم layout يدوي خلال 10 ساعات.

6.3 Plugin JLC-PCB-Tools (إلزامي)

```
Bash
```

```
# في KiCad 7:
Tools → Plugin → KiCad PCM
ثم بحث عن: JLCPCB
ثبّت: "JLC PCB Plugin"
```

يولّد **CPL + BOM** بصيغة JLCPCB المباشرة (3 ملفات بنقرة واحدة).

6.4 Checklist قبل إرسال لـ JLC

- [ ] DRC = Pass
- [ ] ERC = Pass
- [ ] 3D View = صحيح
- [ ] BOM يحتوي LCSC في عمود مخصص
- [ ] CPL يحتوي إحداثيات صحيحة (JLC 0,0 = bottom-left)
- [ ] Edge.Cuts مغلق بالكامل (closed loop)
- [ ] لا Footprint بدون 3D Model
- [ ] Solder Mask Expansion = 0.05mm (default)
- [ ] Silk 1 أكبر من 0.15mm، Pads أبعد من 0.15mm

6.5 معيار النجاح

- **Gerber-Verify (mfg.com viewer)** بدون أخطاء.
- **JLC review pass من أول محاولة** (لا "Engineering query").
- **Yield > 90%** (5/5 لوحات سليمة) على الدفعة الأولى.

ملخص الحلول في جدول واحد

المشكلة	أهم 3 حلول عملية	تكلفة إضافية	الجهد
1. الضوضاء	TPS7A4700 + RF Shield + Time-multiplex	$5	3 أيام
2. التعقيد البرمجي	FreeRTOS + 4 buses + 3-level watchdog	$0	7 أيام
3. الحرارة	TPS62130 + Daughterboard	$4	4 أيام

	RM3100 + Sleep modes		
4. المعايرة	3-stage cal + Carbon Fiber + Magneto Wizard	$8	5 أيام
5. PCB/EMI	50×70mm 4L + 3 RF Cans + Layout pro	$6	7 أيام
6. Toolchain	KiCad فقط + JLC Plugin + Manual route	$0	يومان
المجموع		**~$23**	**~28 يوم**

القرار التنفيذي

ليست عيوباً قاتلة، بل تحديات هندسية معروفة لكل مشروع Mixed-Signal جاد.

كل المشاكل الست لها حلول مُجرّبة في صناعة الإلكترونيات الطبية (ECG، EEG) التي تواجه نفس تحديات التكبير العالي + التعقيد + الحرارة + المعايرة.

التوصية: نفّذ الحلول الستة بالترتيب أعلاه قبل إرسال الـ Gerber النهائي. التكلفة الإضافية $23 لكل وحدة، لكنها تنقذ المشروع من فشل ميداني محتمل.

> **القاعدة:** "حلّ المشاكل على الورق قبل النحاس، وعلى النحاس قبل الميدان، وفي الميدان قبل العميل."