**项目一：基于无人机的智能农田监测系统**

**项目概述**

利用无人机搭载多光谱摄像头和嵌入式系统，实现对农田的实时监测和数据采集。通过嵌入式系统处理传感器数据，结合YOLO机器视觉技术，识别作物健康状况、病虫害等，并将数据上传至云端进行分析。

**功能亮点**

1. **多光谱数据采集**：无人机搭载多光谱摄像头，采集农田的多光谱图像。
2. **实时数据分析**：嵌入式系统处理图像数据，结合YOLO算法识别作物健康状况。
3. **云端数据上传**：通过Wi-Fi或LoRa将数据上传至云端，供农户或农业部门分析。

**学习路径**

1. **基础知识**：
   * 学习C语言和Python编程基础17。
   * 了解嵌入式系统的基本概念和硬件组成8。
2. **硬件选型**：
   * 选择STM32或Raspberry Pi作为主控板17。
   * 选用多光谱摄像头和GPS模块。
3. **开发工具**：
   * 学习使用Keil或PlatformIO进行嵌入式开发11。
   * 掌握FreeRTOS进行任务调度111。
4. **项目实践**：
   * 实现无人机的基本飞行控制。
   * 集成多光谱摄像头和YOLO算法，完成图像采集与处理。
   * 通过Wi-Fi或LoRa实现数据上传。

**预算估算**

* 无人机框架：约1000元
* STM32开发板：约200元
* 多光谱摄像头：约1500元
* 其他传感器和模块：约500元
* 总预算：约3200元

**项目二：无人机辅助的智能物流配送系统**

**项目概述**

设计一款基于无人机的物流配送系统，能够自动规划路径、避障并完成货物投递。通过嵌入式系统实现无人机的飞行控制、路径规划和避障功能，结合YOLO算法实现目标识别和精准投递。

**功能亮点**

1. **路径规划与避障**：利用嵌入式系统实现无人机的路径规划和避障功能。
2. **目标识别与投递**：通过YOLO算法识别投递目标，实现精准投递。
3. **远程监控**：通过4G模块实现远程监控和控制。

**学习路径**

1. **基础知识**：
   * 学习C语言和Python编程基础17。
   * 了解嵌入式系统的基本概念和硬件组成8。
2. **硬件选型**：
   * 选择Raspberry Pi或NVIDIA Jetson Nano作为主控板711。  
     选择 Raspberry Pi 或 NVIDIA Jetson Nano 作为主控板 711。
   * 选用超声波传感器、红外传感器和摄像头。
3. **开发工具**：
   * 学习使用ROS（机器人操作系统）进行路径规划和避障开发11。
   * 掌握OpenCV和YOLO算法进行目标识别。  掌握 OpenCV 和 YOLO 算法进行
4. **项目实践**：
   * 实现无人机的基本飞行控制。
   * 集成传感器和摄像头，完成路径规划和避障功能。
   * 通过YOLO算法实现目标识别和精准投递。

**预算估算**

* 无人机框架：约1000元
* Raspberry Pi或Jetson Nano：约800元  
  树莓派或 Jetson Nano：约 800 元
* 摄像头和传感器：约500元
* 4G模块：约300元  4G 模块：约 300 元
* 总预算：约2600元

**项目三：无人机辅助的环境监测与灾害预警系统**

**项目概述**

设计一款基于无人机的环境监测系统，能够实时采集空气质量、温湿度等环境数据，并结合嵌入式系统进行数据分析。通过YOLO算法识别灾害隐患（如山体滑坡、火灾等），并发出预警信号。

**功能亮点**

1. **环境数据采集**：无人机搭载多种传感器，实时采集环境数据。
2. **灾害隐患识别**：通过YOLO算法识别灾害隐患。
3. **预警系统**：通过嵌入式系统分析数据，发出预警信号。

**学习路径**

1. **基础知识**：
   * 学习C语言和Python编程基础17。
   * 了解嵌入式系统的基本概念和硬件组成8。
2. **硬件选型**：
   * 选择STM32或ESP32作为主控板17。
   * 选用温湿度传感器、空气质量传感器和摄像头。
3. **开发工具**：
   * 学习使用Arduino IDE或PlatformIO进行嵌入式开发711。
   * 掌握FreeRTOS进行任务调度111。
4. **项目实践**：
   * 实现无人机的基本飞行控制。
   * 集成传感器和摄像头，完成环境数据采集和灾害隐患识别。
   * 通过Wi-Fi或LoRa实现预警信号发送。

**预算估算**

* 无人机框架：约1000元
* STM32或ESP32开发板：约200元
* 传感器和摄像头：约600元
* 总预算：约1800元

## ****项目四：基于手势识别的无人机编队控制系统（交互创新类）****

### ****项目核心价值****

通过嵌入式系统+手势识别算法，实现单人控制多台无人机的编队飞行，降低传统遥控器操作门槛，适合展会表演、教育演示等场景。

### ****技术实现细节****

#### ****1. 硬件配置****

* **主控板**：ESP32-CAM（双核处理器+摄像头，¥80）\*3台（一主二从）
* **无人机框架**：空心杯微型机架（载重200g以内，¥150/台）\*3台
* **传感器**：MPU6050陀螺仪（¥15/个）、VL53L0X激光测距（¥25/个，用于编队间距控制）
* **通信模块**：ESP32内置WiFi/蓝牙，主从机通过MQTT协议通信

#### ****2. 核心功能实现步骤****

* **阶段一：单机基础控制**
  + 学习Arduino IDE开发环境，用PlatformIO插件管理多设备代码
  + 通过PWM信号控制空心杯电机，实现起飞/降落/悬停
  + 用PlatformIO的Multi-Project Workspace功能同步调试三台设备
* **阶段二：手势识别交互**
  + 使用ESP32-CAM采集手势图像，通过TensorFlow Lite Micro部署简化版YOLOv5模型（需预训练手势数据集）
  + 定义5种基础手势：✋（悬停）、👆（前进）、🤟（编队解散）、👐（圆形环绕）、👍（降落）
* **阶段三：编队协同算法**
  + 从机通过激光测距实时调整与主机距离
  + 采用PID算法动态修正飞行姿态，代码参考GitHub开源项目《DroneSwarm》

#### ****3. 成本优化方案****

* **摄像头复用**：仅主机搭载ESP32-CAM，从机通过接收指令实现跟随
* **3D打印部件**：设计并打印摄像头保护罩、传感器支架（耗材约¥50）
* **总预算**：3台无人机硬件（¥1503） + 主控及传感器（¥80+¥153+¥25\*3） + 杂项 ≈ **¥800**

## ****项目五：自主充电的太阳能水质监测无人机（环保能源类）****

### ****项目核心价值****

解决传统水质监测无人机续航短的问题，通过嵌入式系统管理太阳能充电+水质传感器，实现湖泊/河流的长期自动监测。

### ****技术实现细节****

#### ****1. 硬件配置****

* **主控板**：STM32F407（带浮点运算单元，¥120）
* **无人机框架**：六轴防水机架（¥600，含防腐蚀涂层）
* **能源模块**：柔性太阳能电池板（20W，¥200）+ TP4056充电管理芯片（¥8）
* **传感器**：SEN0244水质检测模组（pH/电导率/温度，¥380）、超声波液位传感器（¥65）

#### ****2. 关键技术创新点****

* **低功耗设计**：
  + 使用STM32的Stop Mode模式，休眠时功耗降至1μA  使用 STM32 的停止模式
  + 通过Embedded Wizard设计GUI界面，实时显示剩余电量/水质数据
* **自主充电逻辑**：

c

复制

// 伪代码示例：太阳能充电状态机

if (battery\_voltage < 3.7V && light\_intensity > 20000lux) {

enter\_charging\_mode();

deploy\_landing\_gear(); // 展开支架接触水面形成接地回路

}

* **防水处理**：
  + 所有电路板涂覆三防漆（¥50/瓶）
  + 使用磁吸式防水接头（¥15/个）连接传感器

#### ****3. 学习路径****

1. **硬件层**：学习Altium Designer绘制简单的PCB（如传感器接口板）
2. **驱动层**：掌握STM32CubeMX配置ADC采集水质数据
3. **算法层**：研究开源的Pathfinder算法实现自动返航充电
4. **实测阶段**：先用塑料水箱模拟测试，再过渡到真实水域

#### ****4. 总预算****：无人机硬件¥600 + 主控及传感器¥573 + 能源模块¥208 ≈ ****¥1381****

## ****项目六：基于UWB定位的室内救援无人机（应急救援类）****

### ****项目核心价值****

在GPS失效的室内环境，通过UWB高精度定位+热成像技术，实现灾害现场的被困人员搜索与物资投送。

### ****技术实现细节****

#### ****1. 系统架构****

plaintext  纯文本输入

复制

[UWB Anchor]×(4个)

│

├─[UWB Tag]→无人机定位（精度10cm）

│

├─[STM32H743]→实时路径规划

│

└─[FLIR Lepton 3.5热成像模组]→人员探测

#### ****2. 关键技术实现****

* **精准定位**：
  + 使用Decawave DWM1000模块（¥280/个）搭建UWB基站
  + 参考GitHub项目《Pozyx\_Indoor\_Positioning》实现TDOA算法  
    参考 GitHub 项目《Pozyx\_Indoor\_Positioning》实现 TDOA 算法
* **热成像处理**：
  + 在Raspberry Pi 4（¥500）上运行Python脚本，将热成像数据转换为温度矩阵
  + 设置阈值报警：检测到35-40℃区域标记为潜在生命体
* **避障强化**：
  + 毫米波雷达（A111，¥320）替代传统超声波，检测距离达10米
  + 在FreeRTOS中创建高优先级避障任务

#### ****3. 开发里程碑****

* **第1-2周**：学习UWB原理，搭建4基站定位系统（购买现成开发套件加速进度）
* **第3-4周**：实现无人机基础飞行+UWB定位数据融合
* **第5-6周**：集成热成像算法，设计3D打印的云台支架
* **第7周**：模拟火灾烟雾环境测试（可使用干冰制造低可见度场景）

#### ****4. 预算分配****

* 定位系统：DWM1000模块\*4（¥1120）
* 主无人机：DJI F450机架+无刷电机套装（¥800）
* 热成像模组：FLIR Lepton 3.5（¥1500）
* 总预算：约\*\*¥3420\*\*（可通过租借部分设备降低成本）

## ****小白团队学习策略****

1. **知识分级获取**：
   * 优先掌握：C语言指针操作、RTOS任务调度原理、PWM信号控制
   * 延后学习：SLAM算法、神经网络模型压缩（初期直接调用预训练模型）
2. **开发环境建议**：
   * 在Visual Studio Code中整合PlatformIO + Git，实现代码版本管理  在 Visual Studio Code 中集成
   * 使用Wokwi在线仿真平台（免费）测试基础电路逻辑
3. **风险规避方案**：
   * 购买现成的飞控开发板（如Betaflight F4）替代从头开发飞行控制
   * 在初期原型阶段使用泡沫板+胶枪快速验证机械结构