
[附件]

工作/项目 详细介绍 (2016版)

任军 高级嵌入式软件工程师

185 7164 0900

819280802@qq.com

<https://github.com/cedar-renjun>

工作经历

2015-01 至	2016-01	深圳市凡道科技有限公司	CTO
2014-05 至	2015-01	深圳市大疆创新科技有限公司－Robomaster机器人部门	嵌入式研发
2012-01 至	2014-05	英蓓特科技有限公司－CooCox嵌入式开发工具线	嵌入式研发

在校经历

2012-01 至	2012-06	基于STM32的触摸屏MP3设计	项目负责人
2011-07 至	2011-09	中国机器人大赛暨RoboCup中国公开赛仿人组标准平台－国家二等奖	技术负责人
2010-10 至	2011-04	基于mega128的超声波路况检测系统	组长+技术负责人
2010-09 至	2010-11	广茂达杯国际灭火机器人组－国家三等奖	组长+代码编写
2010-06 至	2010-08	基于MEMS三轴加速度传感器的智能小车	组长+技术负责人

深圳凡道科技有限公司

CTO

起止时间：2015-01 至 2015-12

软件环境：windows7, IAR, MDK, Eclipse, C/C++, Mindmap, teambition

硬件环境：CC254x, CC253x, NRF51822, MT2502

任务职能：

1. 公司内部组织管理
2. 公司融资事务
3. 设计BLE产品技术架构
4. 编写BLE相关产品核心代码
5. 设计防丢手表技术架构
6. 编写相关产品手册

大疆创新 (DJI) 科技有限公司—ROBOMASTERS机器人部门

嵌入式研发工程师

起止时间：2014-05 至 2014-12

软件环境：windows7, Alitum Designer, IAR, MDK, VS2008, 汇编, C/C++, QT

硬件环境：STM32 F1/F4, 战车底盘

任务职能：

1. 编写战车联网通讯系统

利用TI CC1101/1120 Sub-1G芯片，从0编写高性能跳频自适应，多机组网通讯系统，从而保证整个裁判系统正常运行

2. 编写战车底盘裁判系统

采集车体分布式的加速度传感器数据，得到战车受攻击位置和受攻击力度，从而计算相应的失血量；通过RFID接口，获取底盘加血系统数据，从而让系统恢复血量；根据裁判系统服务器中心的控制指令，采取相应的血量动作。在车体尾部实时显示当前血量信息

3. 编写战车固件升级系统

通过USB和CAN接口，编写固件升级系统，完成战车内所有MCU模块的一键自动升级

4. 制定战车通讯协议和设计软件整体架构

英蓓特科技有限公司--COOCOX开发工具线

嵌入式软件研发工程师

起止时间：2012-01 至 2014-05

软件环境：windows7, IAR, MDK, ColIDE (Gcc) , VS2008, 汇编, C/C++

硬件环境：

1. ST/NXP/Freescale/TI/Nuvoton Cortex-M0~4系列开发板
2. 各种开源硬件商提供的模块 (Seeed Studio/DFRobot/Arduino/Mbed)

任务职能：

1. 编写Cortex-M系列Cox库，并改进Cox接口规范

各个半导体厂商在推出Cortex-M库时，为了加速应用者开发速度，都会推出一套对应的芯片支持库，ARM仅仅对内核部分做了统一化处理（CMSIS接口），但外设（如UART/PWM/DMA等）接口尚无统一标准，这使得基于Cortex-M芯片开发的软件，没有良好的移植性。所以，CooCox在标准厂商库的基础上，提炼出一套外设HAL接口标准，这就是Cox驱动接口层。基于Cox接口编写的软件，可以平滑的在各个Cortex-M MCU平台之间切换

我刚进公司时，负责编写基于STM32芯片的Cox代码（部分功能模块）；前期的Cox库注释标准和接口不清晰，我独立整理了Doxygen注释规范，并独自完成一套NXP 17xx M3芯片Cox库，同时更新了Cox测试框架。这些工作，有效的提高的Cox易读性，易用性

参考链接：

- | | |
|---------------------|---|
| 1. Cox的介绍资料 | http://www.coocox.org/cox.html |
| 2. Cox的github地址 | https://github.com/coocox/cox |
| 3. LPC17xx的github地址 | https://github.com/cedar-renjun/Cox_LPC17xx |

2. 整理并编写市面上的各种硬件驱动

CooCox拥有完整的嵌入式开发工具链，包括新一代网络化的ColIDE，组件化的驱动平台；在CooCox这个驱动平台上，任何人都可以上传自己的代码（基础库，中间件，示例性代码），其它人可以免费使用。CooCox试图打造一个嵌入式开发者共享交流平台，我和组件小组其它3个人，负责收集市面上的硬件驱动模块/中间件，并编写基于Cox的驱动代码。开始时，驱动不好移植，我着手制定了一个驱动接口规范来解决这个问题；同时调研Mbed接口驱动支持架构，准备让ColIDE支持Mbed接口标准（规划ing，待做）

参考资料：

- 1. CooCox驱动列表 <http://www.coocox.com/driver.html>
- 2. 个人github仓库地址 <https://github.com/cedar-renjun?tab=repositories>

3. 配合和协调ColIDE

组件这里配合Java组，为ColIDE增加新功能和制定规范，具体包括：

3.1 为ColIDE增加C++支持

ColIDE 1.7.5之前的版本不支持C++，为此Coocox的很多国外用户不断抱怨，后来我负责调研gcc支持基本原理，编写GCC链接脚本，配合上层，为ColIDE增加C++支持（1.7.6版本中已支持C++）

3.2 编写ColIDE文档规范

现在ColIDE改变进行中，新的UI表现层弃用原有Java引擎，改用新的Web内核，所有界面采用HTML5来编写，同时引入github的支持，增加公有仓库和私有仓库功能；我负责组件（指所有模块化的嵌入式代码）的文档规范和ColIDE交互行为需求，主要包括：Markdown和Doxygen交互接口规范，组件文档组织规范，修订代码编写规范

4. 参与制定市场推广策略和产品战略规划

做的不是具体工作，主要是战略层的考虑，制定决策（涉及公司机密，细节不便透露），可以细分为两大部分：

4.1 CooCox产品战略

分析CooCox竞争对手的优缺点，进行SWTO分析；

时刻注意竞争对手的动作，并考虑可能对我们造成的威胁，适时制定决策加以应对

4.2 产品推广策略

细分顾客目标群，并分析CooCox产品顾客行为特征，针对不同用户制定不同的推广策略；制定策略吸引开发者来参与开源项目

毕业设计：基于STM32的触摸屏MP3设计

项目时间： 2012-01 至 2012-06

软件环境： windows7， IAR

硬件环境： STM32 103VET6， VS1003

开发工具： IAR， JlinkV8

项目描述：

=====设计目标=====

设计一款电阻触摸屏的MP3播放器，该MP3可以自动读取SD卡上的MP3，wav音频文件，搜索完存储卡后，自动提供播放列表供使用者查阅，使用者可以选中某一首歌曲，然后点击播放按钮开始播放歌曲，在播放过程中，可以停止播放歌曲，可以上一曲，下一曲播放，同时该播放器提供音量增减功能

=====技术分析=====

1. 本项目的核心芯片为STM32103VT6，它采用ARM最新设计的cortex-m3内核，该软核指令执行效率很高，同时内嵌NVIC（嵌套向量中断控制器），SysTick，调试组件，极大的方便了使用者的移植工作。同时STM32内部模块丰富，功能强大，且ST公司提供标准库函数，方便了移植和用户使用
2. 采用ucos-ii操作系统，操作系统的引入，多任务并发执行，提高了cpu的利用效率，简化编程
3. 由于项目中采用TFT液晶屏，加上触摸屏，直接操作TFT非常复杂，所以引入UCGUI，便于用户使用，同时可以创建美观大方的用户界面
4. 由于MP3的体积比较大，所以需要外部存储器，采用SD卡比较方便，需要在SD上建立文件系统，采用嵌入式文件系统FatFS，该文件系统结构精简，便于移植，非常适合于嵌入式系统
5. 采用VS1003解码芯片用解码mp3文件，处理器不需要软解码mp3，只需要将mp3数据通过SPI接口发送给VS1003模块即可

项目职责：

本作品是毕业设计作品，本人作为整个项目负责人

中国机器人大赛暨ROBOCUP中国公开赛：仿人组标准平台

项目时间： 2011-07至2011-09

软件环境： windows7，C语言，汇编语言

硬件环境： 机器人大赛指定开发套件（高级版）

开发工具： IAR

项目描述：

=====竞赛要求=====

利用指定的机器人套件，设计一款人形的机器人，该机器人具有明显的人体结构，如头，手臂，腰部，且能自主上台，并在台上自由行进，不掉下台子。比赛场地大小为长、宽分别为是2400 mm，高150mm的正方形矮台，台上表面即为擂台场地。底色从外侧四角到中心分别为纯黑到纯白渐变的灰度。场地的两个角落设有坡道，机器人从出发区启动后，沿着该坡道走上擂台。在指定的擂台上双方机器人和5个中国象棋棋子。双方机器人模拟中国传统擂台搏击的规则，互相击打或者推挤。如果一方机器人整体离开擂台区域或者不能再继续行动或被对方机器人打倒，则另一方获胜。

=====技术分析=====

1. 硬件平台采用博创机器人高级套件，核心控制板为AVR mega128，机器人形状为人型
2. 手臂和腰部采用舵机组成，核心控制板通过UART半双工串口控制总线式舵机，通过预设定的动作来产生击打动作
3. 运动控制采用直流减速电机，驱动卡为博创的控制卡，采用半双工总线控制
4. 地面场地检测地面场地是从中心往外灰度递减的，故采用灰度传感器来检测地面灰度值，从而确定机器人在场地中的位置
5. 场地边缘的识别采用固定距离值的红外传感器来检测边缘，该传感器为数字接口，当检测到边缘时输出高电平，该接口直接连接至mcu的中断线，从而紧急后退
6. 敌人的识别敌人识别采用红外距离传感器，其接口采用模拟接口，障碍物与机器人本体的距离与输出的模拟量成正比，通过ADC模块，通过软件插值法可确定实际距离值

项目职责：

本人作为组长，负责人员的协调和项目代码的编写工作

1. 作为组长，负责培训新人和协调整个团队 负责培训队里的新人学习C语言和相关机器人模块，负责分配整个队伍的任务安排
2. 负责整体代码的编写工作，该项目采用ucos-ii实时多任务环境，负责ucos-ii的操作系统的移植和应用代码的编写，以及底层驱动的部分，如R/C舵机控制，ADC，GPIO等，负责上层比赛策略的实现

郑州大学创新实验项目：基于MEGA128的超声波路况检测系统

项目时间： 2010-10至2011-04

软件环境： windows7， C语言， VB编程语言

硬件环境： mega128开发板， 超声波模块

开发工具： Winavr， VS2008

项目描述：

=====设计要求=====

设计一个系统,该系统能实时检测道路车况信息(如车流量,车速等)并将采集到的数据通过无线模块发送到数据中心,数据中心将采集到的数据图形化显示出来,以便供高层决策者分析

=====技术分析=====

1. 整体构思

本项目为郑州大学创新实验项目,重点在于让学生通过该项目,培养工程意识,提高动手能力.且若做成实际使用的系统,需要在道路上实际安装调试, 但此时存在安装调试不便,且不安全,所以经过讨论,整个实验项目采用模拟方式进行, 按比例制作出实物模型,通过测试实物来达到预期目的

2. 芯片选择

可选芯片有51和AVR, 因AVR系列自带定时器具有输入捕捉功能, 便于测量超声波模块的输入脉冲,且内部定时器资源丰富, 主频比较高,相比而言,若采用51系列处理器,性价比不高,且编程复杂难调试,故最后选择AVR 系列的mega128

3. 测量模块

本项目要求同时测量道路车速和车辆数目等信息, 故至少需要两个超声波探头, 且为了避免探头相互干扰, 模块的安装间距需要符合要求. 同时为了不影响车辆的正常行驶, 探头需要安装在道路的正上方.

4. 数据中心的模拟

编写个上位机来模拟数据中心,该上位机采用VB编写,通过comm控件来获取串口信息,同时将采集到的数据用图形化的方式表现在屏幕上

项目职责：

本人在项目中负责下位机代码及论文的撰写工作

1. 下位机部分

本系统采用成品超声波模块，该模块接口为VCC，GND，ECHO，TRIG。使用时先在TRIG端口施加一个高电平触发脉冲（大于10us），然后ECHO会输出一个高电平脉冲，该脉冲宽度正比于障碍物到探头的距离，使用mega128定时器的输入捕捉模式可以测量该脉冲宽度。设定一个距离值H_Lim，当检测到距离值小于H_Lim时，可以断定检测到车辆当前面的探头检测到车辆后，车辆计数器增1，同时记录此时的时间值，当后面的车辆检测到车辆后，记录此时的时间值，根据这两次的时间差和探头的间距值，可以测算车辆的通过速度。得到车辆速度和车辆数目这些路况信息后，通过mega128的串口发送到模拟数据中心

2. 论文撰写

根据本项目的情况,撰写论文《基于超声波传感器的车流量检测系统》，并发表在核心期刊《实验室研究与创新》（2011-8,72-74页）

广茂达杯国际灭火机器人比赛

项目时间： 2010-09至2010-11

软件环境： windows7， C语言

硬件环境： 广茂达专用灭火机器人套件

开发工具： MF09(灭火机器人专用开发工具)

项目描述：

=====竞赛要求=====

在模拟房间中,放置一个蜡烛,设计个灭火机器人从出发点开始,自主搜寻每个房间并检测是否起火(用点燃的蜡烛模拟),机器人找到灭火源后进入房间,并驶向火源,走到据火源一定距离时,停止行进,开始灭火,扑灭火源后,自动返回出发点并停止在出发点处

=====技术分析=====

1. 机器人硬件平台采用比赛方指定套件,核心处理器为300M的高速DSP
2. 火源的识别采用热红外传感器,该传感器输出量为模拟值,通过DSP的ADC接口转换为数字量,在软件中设定火源检测阈值,大于阈值则认为发现火源
3. 采用夏普红外距离传感器检测墙壁和房间入口,该传感器为模拟接口,距离值与输出模拟量近似成线性关系,通过检测模拟量,即可获取机器人与墙壁距离信息
4. 出发点和房间门口均为白色,其它地面为黑色,故采用灰度传感器来检测黑白地面
5. 为便于搜寻房间和扑灭火焰,故设计成车型机器人,动力装置采用直流减速电机,无速度反馈装置,驱动方式为PWM控制
6. 因场地限制无法采用灭火器和来扑灭火焰,故采用高速电风扇吹风方式扑灭火源

项目职责：

本次比赛为组队参加，三人一组，我作为队长负责整体安排和项目代码编写

具体所做工作如下：

1. 作为队长负责整体安排，负责整体项目进度，分阶段分配队员任务
2. 负责机器人代码编写工作，由于比赛方提供机器人套件底层驱动接口,所以我们只需要编写机器人搜索房间策略和规划路径，策略由我们组三个人共同讨论制定,我负责编码实现,并根据测试情况及时修改代码

美新杯国际物联网大赛：基于MEMS（微机电系统）三轴加速度传感器的智能车

项目时间： 2010-06至2010-08

软件环境： windows7， C语言

硬件环境： AT89S52， 无线模块NRF24L01， 液晶模块， 直流电机驱动器， 超声波模块， 三轴加速度传感器

开发工具： Keil

项目描述：

=====设计目标=====

设计一个智能小车，本车具备无人操作和人工控制两种模式

自动控制模式：智能小车自主漫游，主动躲避障碍物

人工控制模式：操作者利用装有三轴加速度传感器的遥控器，通过感知遥控器和地面的倾角来控制小车的运动

同时该车具备液晶显示屏，可以实时的显示一些控制信息：如是否有障碍物，车速，当前控制模式等

=====技术分析=====

1. 车体和遥控器采用AT89S52单片机（因为当时大二，只会这一种单片机）
2. 考虑到需要远程控制,故采用nordic公司NRF24L01模块，该芯片采用SPI接口和单片机通信
3. AT89S52不具备接口,需GPIO模拟接口与三轴加速度传感器(MXC6202xG)自带接口通信
4. 采用飞思卡尔mc33886电机芯片来控制直流电机，采用PWM模式控制，频率小于10KHz
5. 因车体部分含有液晶，超声波，直流电机驱动卡，无线等模块，单MCU控制方案会导致该MCU负担过重，所以采用双MCU控制方案，一个MCU负责和NRF24L01交互，负责液晶显示，另外一个负责控制RC舵机，直流减速电机，超声波模块

项目职责：

作为队长，负责整体项目进度安排和代码编写工作

1. 根据组员的自身特长，合理分配任务，如：某位组员的口才比较好，让他负责作品的宣传工作，某位组员熟悉底层，就让他参与硬件驱动编写
2. 编写部分底层的硬件驱动程序（三轴加速度传感器，直流减速电机驱动器，NRF24L01无线模块，R/C舵机驱动）
3. 负责设计底层通信协议：车体上两个MCU通过UART通信，负责制定数据传输协议
4. 负责上层应用软件的开发。调用LCD驱动函数，在车载LCD液晶屏实时显示车辆状态信息；根据采集到的三轴加速度数据，来判断遥控器相对于地面的倾角信息，然后控制车体运动