**学习和工作笔记本**

**c++面向对象编程笔记**本

class box

{

public:

private:

protected:

}

public:公有，外部可以随意访问

private:私有，外部不可以访问，甚至不可以查看，如果没有指定，那么默认为私有。我们一般会在公有

区域定义函数，方便外部调用，私有区域定义变量数据。设置私有不可以直接调用，通过公有函数调用。

proteced:保护，保护成员再派生类（子类）中是可以访问的。

class box

{

public:

double leng;

double bread;

double height;

double getvolume(void);

void set\_leng(double len);

}

class small:box 子类/继承/派生类

{

}

范围解析运算符：：

double box::getvolume(void)

构造函数：它会在每次创建类的新对象时执行，可以设置初始值。

析构函数：它会在每次删除所创建大的对象时执行。

多继承：公有继承，保护继承，私有继承。

重载函数：函数名相同，形参不同。

运算符重载

数据封装：是一种把数据和操作数据的函数捆绑在一起的机制，数据抽象是一种仅向用户暴露接口而把具体的实现细节隐藏起来的机制。

c++接口抽象:基类只作为一个接口使用，具体实现方法由派生类重新定义。

class box

{

Public:

Virtual double getvolume()=0;

Private:

Int leng;

Int bread;

Int heoght;

}

This指针：每一个对象都能通过this指针来访问自己的地址。This指针是所有成员函数的隐含参数。因此，在成员函数内部，他可以用来指向调用对象。

C++类中的静态成员：  
c++指向类的指针：

**Modbud ascII/rtu/tcp的区别**

AscII：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 开始 | 地址 | 功能码 | 数据数量 | 数据 | Lrc | 结束 |
| : |  |  |  |  |  | 0x0d 0x0a |

发送的数据会分解为ascII字符，4为二进制为1个字符，发送的数据量相对于rtu翻一倍。

Tcp：

无校验码

Rtu：

最常用，使用的是crc校验，地位字节在前，高位字节在后

PID控制

卡尔曼滤波

模糊控制

**程控仪器**

在工控测试系统中，经常需要使用到各种程控仪器，这些程控仪器通常具有GPIB、LNA、USB等硬件接口。计算机通过这些接口能够与其通信，从而实现自动测量、数据采集、数据分析和数据处理等操作。

Scpi：scpi是可编程仪器标准命令，standard commands for programmable instruments,scpi规定了在程控器到仪器和仪器到程控器之间的信息交换层消息的构造和内容，因而使得在垂直层面上，同一类型的仪器命令集相似；在水平层面上，不同类型仪器之间，同种功能的scpi命令也一致。Scpi命令分为仪器公用命令和仪器特定控制命令两部分。公用命令用于控制仪器的某些基本功能操作，其语法和语义遵循ieee488.2规定。仪器特定控制命令是指每个程控仪器完成自生命令操作的特定命令

Visa:visa是虚拟仪器软件架构的简称。Visa提供了用于仪器编程的标准io函数库，称为visa库，visa库提供了统一的设备资源管理、操作和使用机制，他独立于硬件设备，接口，操作系统和编程语言，具有与硬件结构无关的特点。它将用于每个仪器硬件总线类型的更低层次的驱动程序抽象化，并提供一个单一的api实现与仪器的通信，而不必考虑具体的总线接口。

**设备问题锁定和检修方法**

1.通过故障现象目测大概是某部分的问题，通过程序仿真来锁定问题点，确定硬件或是软件问题。

2.如果能确定是软件问题则对软件做出修改，否则检查硬件。

3.对于硬件的检修，先目测有无明细烧毁或短路的地方，或用万用表检查电源有无短路，有无器件击穿，内部烧断，是否和原理图一致等问题，

4.检修之后用万用表检查电压是否正常，通讯是否有数据。

5.分块解决，上一级和下一级的连接要断开，单独测试有问题的单元。

对于一张新原理图，应当认识上面的所有元器件，查询器件参数，整理连接信号。然后再去编写软件或者熟悉已有的软件程序。

新装配的设备通电以前要进行短路检查。

半导体材料：  
第一代半导体材料，硅

第二代半导体材料，砷化镓

第三代半导体材料，氮化镓，主要用于固态光源，电力电子，微波射频器件。

对于固态一词的理解，在微电子技术中，是半导体材料存贮数据，控制开关，

对于工作的态度，对事不对人。

对事情一丝不苟，按照规矩来。

人是社会性的动物，不受环境影响的人指示很小很小的一部份，你我皆平凡之躯，还是到更好的环境当中去接收更好的熏陶吧。

RS232:txd,rxd,gnd,不支持多点通讯，负逻辑

RS485:差分信号，同名端，支持多点通讯

IOS：

核心操作系统层

核心服务层

媒体层

触控界面层

Android:

Linux内核

系统运行库

应用程序框架

应用程序

内核，hal用c/c++

框架，应用用Java

Ucos ii:

休眠态

就绪态

运行态

等待态

被中断态

OS\_ENTEY\_CRITICAL

OS\_EXIT\_CRITICAL

OSTaskCreate()

OSTaskDel()

OSTaskSuspned()

OSTimeDly()

点阵法

轮廓法

在线仿真器，片上调试技术，指令集模拟器，驻留监控软件调试

GCC:预处理，编译，汇编，连接

指令集：

发送设备必须把要传送的数据分成小块，称为帧

LDR R1,[R2,R3]!

R2+R3写入到R2中，将 R2+R3的数据存入R1

BL 跳转指令

MOV PC,LR

AND 与

ORR 或

看清楚题，不要多写字！！！！！！！

Java

实时系统中，从低优先级切换到高优先级所用的时间：任务抢占时间，接收到可屏蔽中断到进入中断服务程序所需最长的时间：中断延迟时间

OSIntExit(); 中断退出函数。

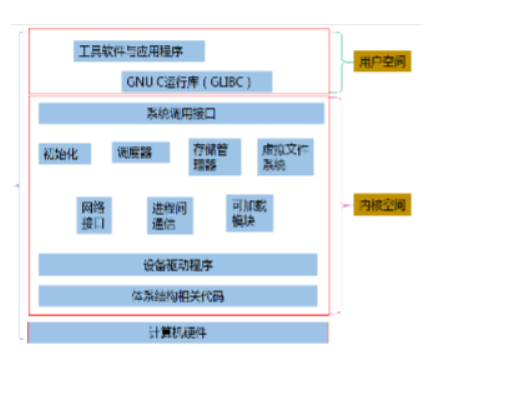
OSIntNesting 中断层数

255层

Linux 内核

以太网控制芯片，以太网驱动芯片，以太网接口，

通过JTAG接口可以访问CPU内部寄存器，以及挂在中线上的设备，如：ROM，RAM，Soc内置模块的寄存器。



数码相机中所用于存储拍摄照片大多都是flash存储器

传输速率可达到11，54，108Mbps

复位RESET

未定义指令UND

软件中断SWI

指令预取中止PABT

数据访问中止DABT

外部中断请求IRQ

快速中断请求FIQ

CPSID I 开

CPSIE I 关

IIC 遵循 低电平优先原则

UART 发送器，接收器，控制单元，波特率发生器

嵌入式系统与计算机相同之处，**引导加载程序**，外设驱动程序，操作系统，文件系统，网络协议，图形用户界面，数据库，各种应用程序。

实时系统时间：

截止时间，任务预设时间

可预测性成为实时系统的一项重要性能要求，它是指RTOS能够对外部事件响应时间和实时任务的执行时间进行判断。

强实时，普通实时，弱实时

需求分析与规格说明，系统设计，构件设计，系统集成与测试

实时系统的任务时限有两种类型，分别是**截止时间**和**任务执行预设时间**。

ADS1.2,工程项目中需要包含的源文件，库文件，头文件以及其他的输入文件

GCC的4个相关阶段组成：预处理，编译，汇编，连接。

创建信号量的函数：OSSemCreate（）；

Ucos的5种任务状态：运行态，就绪态，休眠态，等待态，被中断态

Ios：核心操作系统层，核心服务层，媒体层，触控界面层。

评价实时系统实时性的最重要指标是响应时间，即系统从事件请求开始到任务完成的时间间隔。与任务切换和中断延迟时间有关的参数是衡量实时操作系统性能的最重要指标。

PWM TCNTBn：频率

TCMPBn：占空比

USB3.0能提供最大900ma，其余500ma

首先发送的是 发送设备的MAC地址，最后传输的数据是校验信息

字符：轮廓法，点阵法。

WIC 唤醒中断控制器

LSL 左

LSR 右

MOV R1,R0,LSL#3

小端模式，地位在低地址当中。

NOR NAND

UEXTCLK

PCLK

波特率寄存器=时钟/（波特率\*16）-1

OSSchedLock()

MOV R1,R0,LSR#3

控制器局域网，采用差分传输方式，

使用GCC方式开发

IDE

Gcc-Arm嵌入式工具链

OpenOCD

CAN总线：

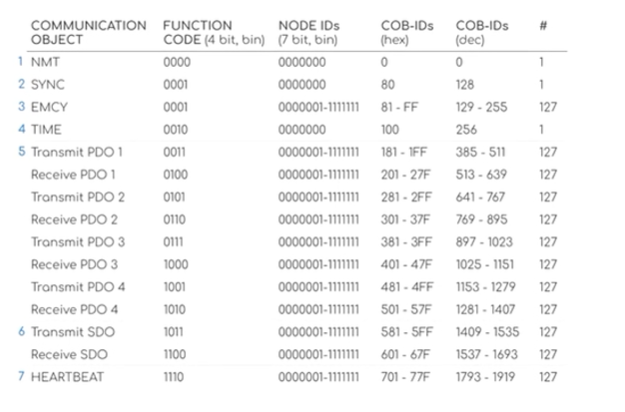
鲁棒性：robust，控制科学中的一个名词，被称为强健性/抗干扰性，在异常和危险情况下系统生存的能力。

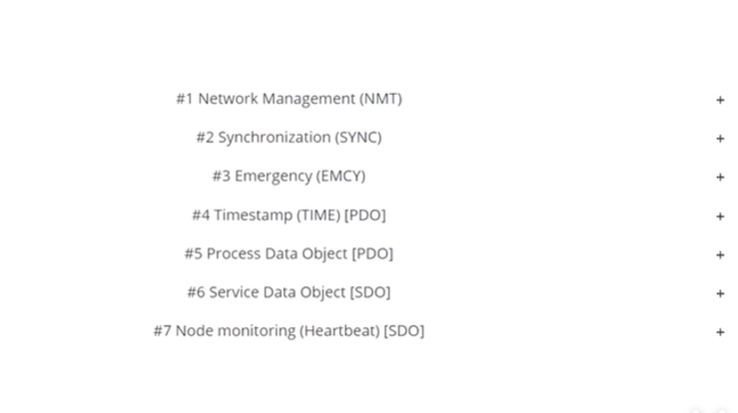
五种帧：数据帧，远程帧，错误帧，过载帧，帧间隔

数据帧：帧起始，仲裁段，控制段，数据段，CRC段，ACK段，帧结束

Canopen：

1. 主机/从机，0-12
2. 服务器/客户端
3. 消费/生产者







关于使用clion搭建STM32开发环境的步骤

Vscode开发stm32配置如下

{

    "configurations": [

        {

            "name": "Win32",

            "includePath": [

                "${workspaceFolder}/Core/Inc",

                "${workspaceFolder}/Drivers/STM32H7xx\_HAL\_Driver/Inc",

                "${workspaceFolder}/Drivers/STM32H7xx\_HAL\_Driver/Inc/Legacy",

                "${workspaceFolder}/Drivers/CMSIS/Device/ST/STM32H7xx/Include",

                "${workspaceFolder}/Drivers/CMSIS/Include"

            ],

            "defines": [

                "USE\_HAL\_DRIVER",

                "STM32H743xx"

            ],

            "windowsSdkVersion": "10.0.17763.0",

            "compilerPath": "C:/Program Files (x86)/GNU Arm Embedded Toolchain/10 2021.10/bin/arm-none-eabi-gcc.exe",

            "cStandard": "c17",

            "cppStandard": "c++17",

            "intelliSenseMode": "gcc-x64"

        }

    ],

    "version": 4

Launch.json

{

    // 使用 IntelliSense 了解相关属性。

    // 悬停以查看现有属性的描述。

    // 欲了解更多信息，请访问: https://go.microsoft.com/fwlink/?linkid=830387

    "version": "0.2.0",

    "configurations": [

        {

            "name": "Cortex Debug",

            "cwd": "${workspaceRoot}",

            "executable": "${workspaceRoot}/build/72.5-15kw.elf",

            "request": "launch",

            "type": "cortex-debug",

//            "device":"STM32F429VE",        //使用J-link GDB Server时必须；其他GBD Server时可选（有可能帮助自动选择SVD文件）。支持的设备见 https://www.segger.com/downloads/supported-devices.php

//            "svdFile": "./STM32F429.svd",  //svd文件，有这个文件才能查看寄存器的值，每个单片机都不同。可以在以下地址找到 https://github.com/posborne/cmsis-svd

            "servertype": "openocd",       //使用的GDB Server

            "configFiles": [

                "stm32h743.cfg"

            ],

            "preLaunchTask": "build"

        }

    ]

}

Tasks.json

{

    // See https://go.microsoft.com/fwlink/?LinkId=733558

    // for the documentation about the tasks.json format

        "version": "2.0.0",

        "tasks": [

            {

                "label": "build",

                "type": "shell",

                "command": "make",

                "args": [

                    "-j4"

                ],

            },

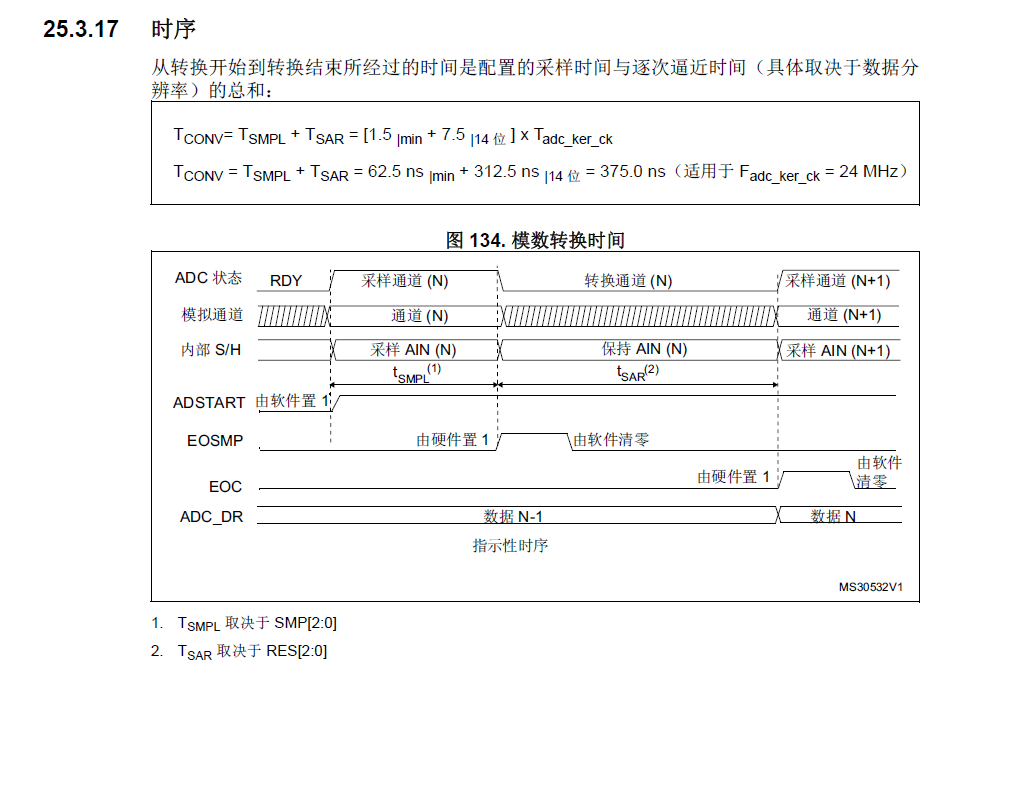
        ]

}

STM32时钟树

其中FCLK,HCLK,PCLK都称为系统时钟,但区别如下,   
FCLK,提供给CPU内核的时钟信号,CPU的主频就是指这个信号;   
HCLK,提供给高速总线AHB的时钟信号; 

PCLK,提供给低速总线APB的时钟信号;



git init 初始化

git add . 添加所有文件

git add 文件夹 添加某个文件夹

git commit –m “1.0” 提交

git status 查看状态

git log 查看log

git reset --hard HEAD^ 退回上一个版本

git reflog 查看操作记录，用于返回最新的版本

创建分支（重点）

git switch -c dev dev是分支名称

git branch 查看分支

git switch dev 切换分支(更推荐)

git branch –d dev 删除dev分支

git checkout master 切换分支

git marge dev19 将当前分支合并分支到dev19

同步到github

1. git init
2. git add .
3. git commit –m “xxxx”
4. 配置ssh密钥
5. 生成密钥 **ssh-keygen -t rsa -C "1247748764@qq.com"**
6. 查看生成的密钥 路径“C:\Users\用户名\.ssh” id\_rsa.pub
7. 添加ssh密钥到github
8. 测试 ssh –T git@github.com
9. git remote add origin git@github.com:michaelliao/learngit.git
10. git remote rm origin 删除远程仓库地址
11. git push -u origin master / git push
12. 从github克隆仓库 git clone git@github.com:michaelliao/gitskills.git

git remote remove origin

git config --list

本地与远程不一致 git pull --rebase origin master

git push 推

git pull 拉

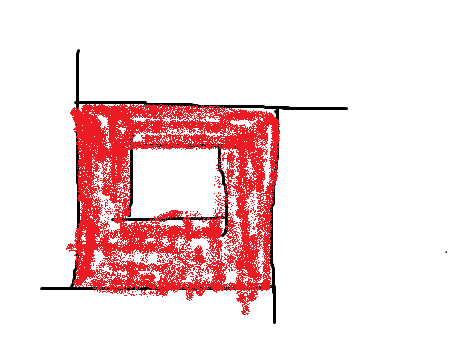
git remote rm origin

我的公司（机会留给有准备的人）

成都樱桃科技有限公司

做一家百年科技技术企业

以科技驱动世界



公司电子邮件

产品溯源管理软件/仓库管理软件/项目管理软件

人员项目分配软件

设计文件-生产产品-资料归档

财务/税务/福利上的问题

租一个公寓当公司办公地点

嵌入式软件/PCB/机械结构

用github来让大家的资料同步

产品的料号，一个产品需要多少个料构成，它的软件和硬件资料，版本迭代。它的维修记录，发货时间等。

公司章程和制度

一些规范化的文档