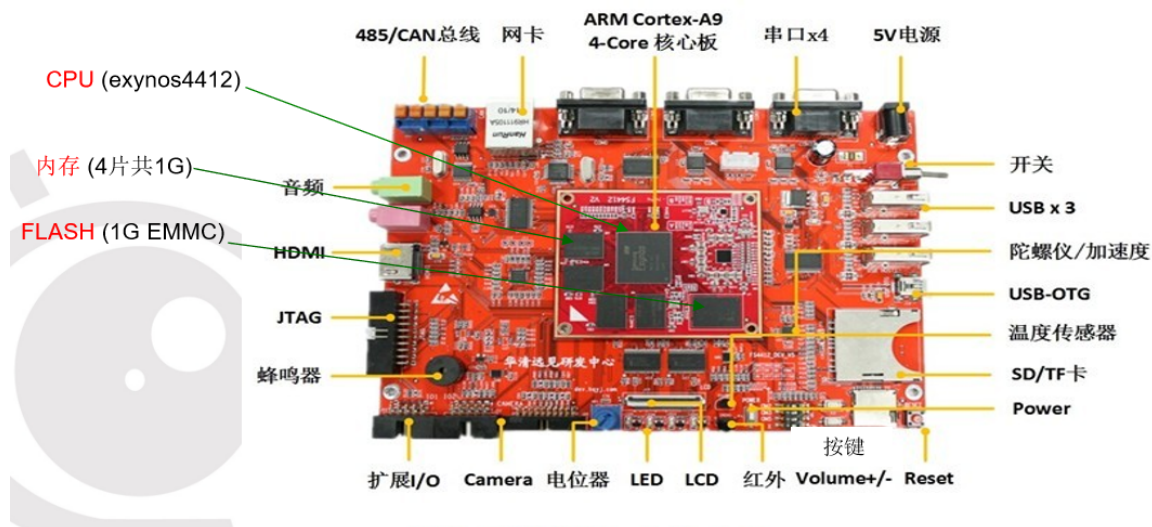


一、开发板介绍



二、环境搭建

- 1.安装交叉编译器
- 2.安装串口驱动 (ch340) ,最好用驱动精灵来安装
- 3.开发板测试，下载编译好的裸机程序，测试开发板是否ok

三、裸机开发流程

目标：点亮开发板上的LED3

//步骤1：准备好开发板，各种资料（电路原理图 + 芯片手册）

//步骤2：看电路原理图(找到对应于芯片的管脚)，先看底板，再看核心板

对应的管脚为：GPX1_0

点亮LED3:GPX1_0管脚拉高

熄灭LED3:GPX1_0管脚拉低

//步骤3：看芯片手册（找寄存器）

//最困难的部分

GPX1CON (配置寄存器) 0x11000c20 [3:0] = 0xf; //输出模式
GPX1DAT (数据寄存器) 0x11000c24 [0] = 1; //输出高电平

//步骤4：编写代码，下载到开发板测试

汇编代码 or C代码

/* 汇编实现点灯 */

LDR R0,=0x11000c20

LDR R1,[R0]

BIC R1,#0xF

ORR R1,#0x1

STR R1,[R0]

```

LDR R0,=0x11000c24
LDR R1,[R0]
ORR R1,#0x1
STR R1,[R0]

/* c代码实现点灯 */
#define GPX1CON *(volatile unsigned int *)0x11000c20
#define GPX1DAT *(volatile unsigned int *)0x11000c24

GPX1CON = GPX1CON&(~0xf);
GPX1CON = GPX1CON|(0x1);

GPX1DAT = GPX1DAT|(0x1);

```

课堂练习

1. 尝试分析 点亮LED4，的步骤

四、串口编程

1. 目标：编程实现串口2发送 'T'，波特率115200，数据位8位，校验位无，停止位1

2. 查看电路原理图(找管脚)

串口2->con7->

//GPA1_0

//GPA1_1

3. 查看芯片手册(找寄存器)

//第6章节，管脚模式配置

GPA1CON 0x11400020

[3:0] = 0x2 // UART_2_RXD

[7:4] = 0x2 // UART_2_TXD

[7:0] = 0x22;

//第28章节，串口2相关寄存器

ULCON2 0x13820000

[1:0] = 0x3;

[2] = 0;

[5:3] = 0;

[6] = 0;

UCON2 0x13820004

[1:0] = 0x1;

[3:2] = 0x1;

UTRSTAT2 0x13820010

[0] = //接收状态 0: empty 1: 收到数据

[1] = //发送状态 0: 发送中 1: 发送完成

UTXH2 0x13820020

[7:0] = 'T'; //发送寄存器

```

URXH2      0x13820024
           [7:0] = //接收到的数据

UBRDIV2     0x13820028
           [15:0] = Baud-rate division value (UBRDIVn) = 53

UFRACVAL2   0x1382002C
           [3:0] = Determines the fractional part of Baud-rate divisor.
           = UFRACVALn = 4

/* 波特率计算公式 */
SCLK_UART: 串口时钟, 100Mhz
bps: 设定的波特率 115200

DIV_VAL = (SCLK_UART/(bps * 16)) - 1
         = 100000000/(115200 * 16) - 1
         = 53.25

UBRDIVn = 53 (integer part of DIV_VAL)

UFRACVALn/16 = 0.25
Therefore, UFRACVALn = 0.25*16 = 4

```

4. 编写c程序，并下载到开发板测试。

```

void uart_init(void); //串口初始化
void uart_putchar(char ch); //发送字符
char uart_getchar(void); //获取字符
void send_string(char *str); //发送字符串

```

练习

1. 熟悉串口编程基本流程

2. 尝试编写串口相关函数

```

void uart_init(void); //串口初始化
void uart_putchar(char ch); //发送字符
char uart_getchar(void); //获取字符
void send_string(char *str); //发送字符串

```

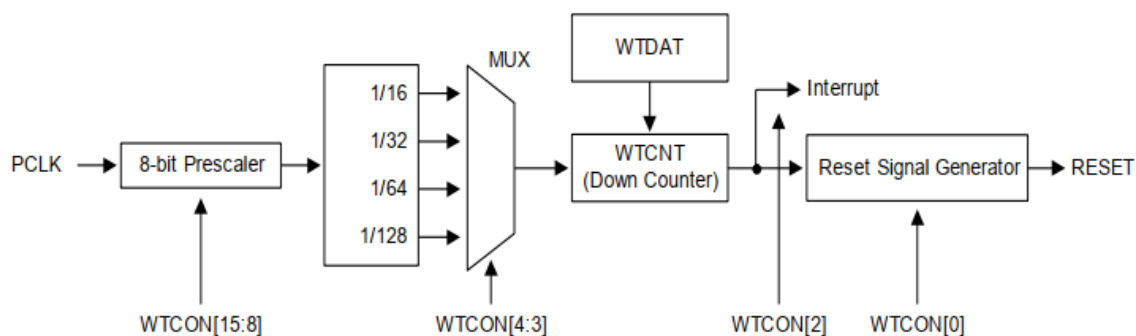
C语言基础题复习

1. 谈谈你的宏的理解

2. 编写mystrcmp

五、WTCON

工作原理：在系统运行以后也就启动了看门狗的计数器，看门狗就开始自动计数，如果到了一定的时间还不去看门狗，那么看门狗计数器就会溢出从而引起看门狗中断，造成系统复位



//目的：设置看门狗定时器，5s复位，（如果在5s内没有喂狗，自动复位）

步骤1：看芯片手册，找看门狗相关寄存器

WTCON 0x10060000 (控制寄存器)

[0] = 1; //使能看门狗复位

[2] = 0; //非中断

[4:3] = 3 //二分频

[5] = 1; //使能

[15:8] = 255; //预分频

WTCNT 0x10060008 (count register)

[15:0] = //计数值

看门狗周期timer clock cycle

= $1 / ((PCLK / (Prescaler\ value + 1)) / Division_factor))$

= $1 / (100000000 / 256 / 128)$

= 0.00032768

复位时间

= 周期 * 计数值

= $0.00032768 * 15259$

= 5s

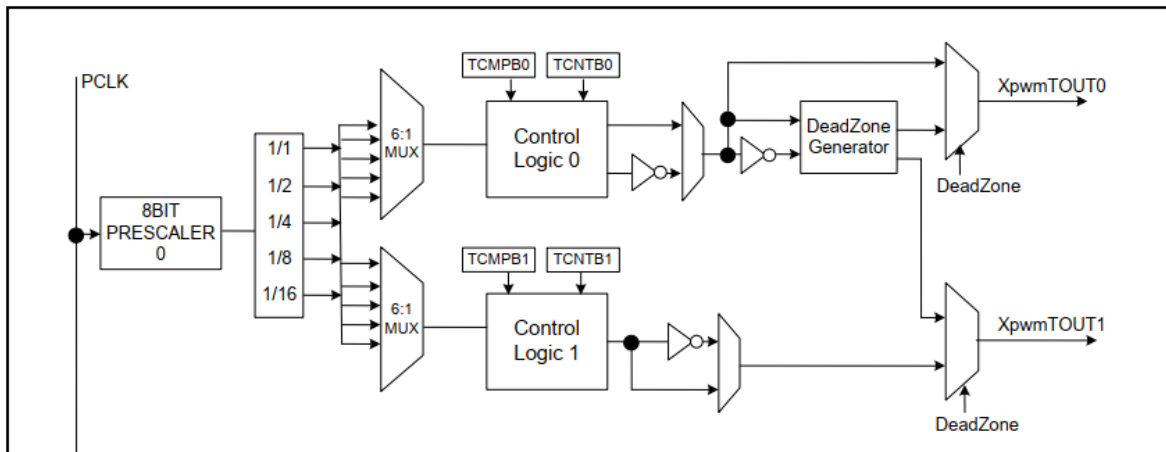
步骤2：编程实现，下载到开发板测试

六、PWM(Pulse width modulation 脉冲宽度调制)

它是利用微处理器的数字输出来对模拟电路进行控制的一种非常有效的技术，广泛应用于测量，通信，功率控制与变换等许多领域。

常见的应用：实现电机调速，呼吸灯，蜂鸣器。

Figure 24-2 illustrates the clock generation scheme for individual PWM channels.



//目的：蜂鸣器发出440hz (1a)

步骤1：看电路原理图

GPD0_0

步骤2：看芯片手册

看第6章节：

GPD0CON 0x114000A0

[3:0] = 0x2 ; //作为TOUT_0

看第24章节：

TCFG0 0x139D0000

[7:0] = 255; //预分频

TCFG1 0x139D0004

[3:0] = 0x2; //二频 4

TCON 0x139D0008

[0] = 1; //开启定时器0

[1] = 1; //手动更新

[3] = 1; //自动加载

TCNTB0 0x139D000C

[31:0] = 222 ; //计数值 (周期)

TCMPB0 0x139D0010

[31:0] = 111 ; //比较寄存 (占空比)

Timer Input Clock Frequency = $PCLK / (\{prescaler\ value + 1\} / \{divider\ value\})$

$100000000 / 256 / 4 = 97656.25\text{hz}$

Toutf = $97656.25 / 222$

= 440 hz

步骤3：编程实现，下载到开发板进行测试

练习

1. 梳理一下，uart，wdt，pwm编程流程

2. 复习c语言，指针，结构体

