物理层头文件：

//#ifndef \_PHY\_H

//#define \_PHY\_H

#include <msp430x54XA.h>

#include "RFlib.h"

//系统子系统时钟频率

#define PHY\_SMCLK\_4MHZ 1

#define PHY\_SMCLK\_4200KHZ 2

#define PHY\_SMCLK\_8MHZ 3

//SMCLK驱动

#define PHY\_SMCLK\_DCO 1

#define PHY\_SMCLK\_XT2 2

//#define PHY\_FLLN 122 //DCO倍频 4MHz

#define PHY\_FLLN 127 //DCO倍频 4.2MHz

#define PHY\_ACLK 32768 //辅助时钟

//#define PHY\_MCLK 4030464 // MCLK=((PHY\_FLLN+1)\*32768)=4030464

//#define PHY\_MCLK\_1MS 3936 // ((PHY\_FLLN+1)\*32768)/3936 = 1024

#define PHY\_MCLK 4194304 // MCLK=((PHY\_FLLN+1)\*32768)=4194304

#define PHY\_MCLK\_1MS 4096 // ((PHY\_FLLN+1)\*32768)/4096 = 1024

#define PHY\_ACLK\_1MS 32 // 32768/32 = 1024

#define EN\_XT2CLK 1 //开启XT2时钟

#define DIS\_XT2CLK 0//关闭XT2时钟

//XT2时钟开启标志

extern unsigned char g\_uchPHYXt2Flag;

//SMCLK驱动时钟选择

extern unsigned char g\_uchPHYSMclkSel;

//SMCLK时钟频率设定

extern unsigned char g\_uchPHYSMclkSet;

//定时器变量

extern unsigned int g\_uPHYTimer0A0Count; //用于三个定时器定时次数的设定

extern unsigned int g\_uPHYTimer1A0Count;

extern unsigned int g\_uPHYTimerBCount;

//定时器中断标志

extern unsigned char g\_chPHYTimerBFlag ;

extern unsigned int g\_uPHYTimerBAppCount; //应用层计数 中断次数

extern unsigned int g\_uPHYTimerBDLLCount ; //链路层计数 中断次数

extern unsigned int g\_uPHYTimerB\_TimeElapse; //注册设备自上次注册到现在的逝去时间

//IO中断标志

extern unsigned char g\_uchPHYIntP14;//端口P1.4中断标志:0 ：无中断；1：有中断

//比较器TA0参数

extern unsigned int TA0StartTime;

extern unsigned int TA0EndTime;

extern unsigned char TA0Overflow;

extern unsigned char TA0Flag;

extern unsigned long TA0TimeCount;

extern unsigned char g\_uchPHYWDFlag; //防止程序跑飞而设定的看门狗打开标记，0 ：关闭；1：打开

extern unsigned int g\_uPHYWDTimeHead; //防止程序跑飞而设定的看门狗时间头

extern unsigned int g\_uPHYWDTimeTail; //防止程序跑飞而设定的看门狗时间尾

#define PHY\_SLEEP\_QUIT 0 //无休眠

#define PHY\_SLEEP\_MODE3 1 //低功耗模式3 休眠

#define PHY\_SLEEP\_FULL 2 //全休眠

extern unsigned char g\_uchPHYSleepMode; //休眠模式 0表示未休眠

extern unsigned int g\_uPHYSleppTicks ; //修眠的时钟次数

//定义采用哪个UART

#define UARTA0 1

#define UARTA1 2

//#define UARTA2 3

#define UARTA3 4

extern unsigned char g\_uchPHYUART0Open ;//UART0打开标记0:开，1：关

extern unsigned char g\_uchPHYUART1Open ;//UART1打开标记0:开，1：关

extern unsigned char g\_uchPHYUART2Open ;//UART2打开标记0:开，1：关

extern unsigned char g\_uchPHYUART3Open ;//UART3打开标记0:开，1：关

extern unsigned char g\_uchPHYPAType; //物理层全局变量，设备功放类型（A型或B型）A型：0x0a; B型：0x0b

extern unsigned int g\_uPHYBattPower; //电池电量

extern unsigned int g\_uPHYChargePower; //充电电压

extern unsigned int g\_uPHYBoardPower; //板极电压

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* 定义FLASH存储器B段地址 \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//#define FLASH\_ADDRESS 0x01000

#define FLASH\_ADDRESS 0x01800

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Baud rate settings \*/

/\* 串口通讯波特率设置定义 \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define Baud\_2400 0x00

#define Baud\_4800 0x01

#define Baud\_9600 0x02

#define Baud\_19200 0x03

//#define Baud\_38400 0x04

#define Baud\_57600 0x05

#define Baud\_115200 0x07

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

利用主系统时钟MCLK延时1/1024s ，约等于1ms

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

extern void PHYMCLKSoftDelay1ms(void);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

利用主系统时钟延时1/2048s ,约等于0.5ms

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

led green control

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_LEDG\_PXSEL P2SEL

#define PHY\_LEDG\_PXDIR P2DIR

#define PHY\_LEDG\_PXOUT P2OUT

#define PHY\_LEDG\_PXPIN BIT1

extern void LedGreenOn(void); //绿灯打开函数

extern void LedGreenOff(void); //绿灯关闭函数

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

led red control

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_LEDR\_PXSEL P2SEL

#define PHY\_LEDR\_PXDIR P2DIR

#define PHY\_LEDR\_PXOUT P2OUT

#define PHY\_LEDR\_PXPIN BIT2

extern void LedRedOn(void); //红灯打开函数

extern void LedRedOff(void); //红灯关闭函数

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

beep control

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_BEEP\_PXSEL P2SEL

#define PHY\_BEEP\_PXDIR P2DIR

#define PHY\_BEEP\_PXOUT P2OUT

#define PHY\_BEEP\_PXPIN BIT3

extern void BeepOn(void); //蜂鸣器打开函数

extern void BeepOff(void); //蜂鸣器关闭函数

extern void BeepSound(void); //蜂鸣器打开一段时间函数

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC2 AAT3221 Switch Control P10.1

UA1/CAM接口电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_POWER2\_PXSEL P10SEL

#define PHY\_POWER2\_PXIDR P10DIR

#define PHY\_POWER2\_PXOUT P10OUT

#define PHY\_POWER2\_PXPIN BIT1

extern void Power2\_3221\_On(void); //打开IC2第二路电源

extern void Power2\_3221\_Off(void);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC3 AAT3221 Switch Control P10.2

UB0/HT 电源接口

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_POWER3\_PXSEL P10SEL

#define PHY\_POWER3\_PXIDR P10DIR

#define PHY\_POWER3\_PXOUT P10OUT

#define PHY\_POWER3\_PXPIN BIT2

extern void Power3\_3221\_On(void); //打开IC3第三路电源

extern void Power3\_3221\_Off(void);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC6 AA4280 Switch Control P10.3

UA3/COM3电源接口

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_POWER4\_PXSEL P10SEL

#define PHY\_POWER4\_PXIDR P10DIR

#define PHY\_POWER4\_PXOUT P10OUT

#define PHY\_POWER4\_PXPIN BIT3

extern void Power4\_4280\_On(void); //打开IC6第四路电源

extern void Power4\_4280\_Off(void);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UA0/COM0 TTL/232控制管脚

MAX3221开启与关闭控制 P5.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_232A0\_PXSEL P5SEL

#define PHY\_232A0\_PXDIR P5DIR

#define PHY\_232A0\_PXOUT P5OUT

#define PHY\_232A0\_PXPIN BIT4

extern void EnableRs232A0(void); //打开UA0 TTL/RS232转换芯片

extern void DisableRs232A0(void);// 关闭UA0 TTL/RS232转换芯片

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UA3/COM3 TTL/232控制管脚

MAX3221开启与关闭控制 P11.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define PHY\_232A3\_PXSEL P11SEL

#define PHY\_232A3\_PXDIR P11DIR

#define PHY\_232A3\_PXOUT P11OUT

#define PHY\_232A3\_PXPIN BIT1

extern void EnableRs232A3(void); //打开UA3 TTL/RS232转换芯片

extern void DisableRs232A3(void);// 关闭UA3 TTL/RS232转换芯片

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

打开/关闭UARTA0

用到的管脚：P3.4\_TX P3.5\_RX

函数参数：char bps

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

extern void OpenUartA0(char bps); //打开UART\_A0

extern void CloseUartA0(char bps); //关闭UART\_A0

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

打开/关闭UARTA3

用到的管脚：P10.4\_TX P10.5\_RX

函数参数：char bps

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

extern void OpenUartA3(char bps); //打开UART\_A0

extern void CloseUartA3(char bps); //关闭UART\_A0

//#endif

物理层源文件：

#include "PHY.h"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

利用主系统时钟MCLK延时1/1024s ，约等于1ms //测试代码，后面需要修改

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void PHYMCLKSoftDelay1ms()

{

for(int i=0;i<=10;i++)

{

for(int j=0;j<10;j++)

{

;

}

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Led Green On

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void LedGreenOn(void)

{

PHY\_LEDG\_PXSEL&=~ PHY\_LEDG\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_LEDG\_PXPIN; //OUTPUT

PHY\_LEDG\_PXOUT|=PHY\_LEDG\_PXPIN; //output high

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Led Green Off

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void LedGreenOff(void)

{

PHY\_LEDG\_PXSEL&=~ PHY\_LEDG\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_LEDG\_PXPIN; //output

PHY\_LEDG\_PXOUT&=~PHY\_LEDG\_PXPIN; //output low

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Led Red On

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void LedRedOn(void)

{

PHY\_LEDR\_PXSEL&=~PHY\_LEDR\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_LEDR\_PXPIN; //output direction

PHY\_LEDG\_PXOUT|=PHY\_LEDR\_PXPIN; //output high

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Led Red Off

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void LedRedOff(void)

{

PHY\_LEDR\_PXSEL&=~PHY\_LEDR\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_LEDR\_PXPIN; //output direction

PHY\_LEDG\_PXOUT&=~PHY\_LEDR\_PXPIN; //output low

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEEP On

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void BeepOn(void)

{

PHY\_BEEP\_PXSEL&=~PHY\_BEEP\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_BEEP\_PXPIN;// output direction

PHY\_LEDG\_PXOUT|=PHY\_BEEP\_PXPIN; //output low

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEEP Off

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void BeepOff(void)

{

PHY\_BEEP\_PXSEL&=~PHY\_BEEP\_PXPIN; //general IO

PHY\_LEDG\_PXDIR|=PHY\_BEEP\_PXPIN; //output direction

PHY\_LEDG\_PXOUT&=~PHY\_BEEP\_PXPIN; //output low

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEEP Sound

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void BeepSound()

{

BeepOn();

for(int m=0;m<100;m++)

{

for(int n=0;n<100;n++)

{

;

}

}

BeepOff();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC2 AAT3221 Switch Control P10.1

UA1/CAM接口电源 打开AAT3221第二路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power2\_3221\_On(void)

{

PHY\_POWER2\_PXSEL&=~PHY\_POWER2\_PXPIN; //general IO

PHY\_POWER2\_PXIDR|=PHY\_POWER2\_PXPIN; //output direction

PHY\_POWER2\_PXOUT|=PHY\_POWER2\_PXPIN; //output high for open the AAT3221

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC2 AAT3221 Switch Control P10.1

UA1/CAM接口电源 关闭AAT3221第二路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power2\_3221\_Off(void)

{

PHY\_POWER2\_PXSEL&=~PHY\_POWER2\_PXPIN; //general IO

PHY\_POWER2\_PXIDR|=PHY\_POWER2\_PXPIN; //output direction

PHY\_POWER2\_PXOUT&=~PHY\_POWER2\_PXPIN; //output LOW for close the AAT3221

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC3 AAT3221 Switch Control P10.2

UB0/HT 电源接口 打开AAT3221第三路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power3\_3221\_On(void)

{

PHY\_POWER3\_PXSEL&=~PHY\_POWER3\_PXPIN; //general IO

PHY\_POWER3\_PXIDR|=PHY\_POWER3\_PXPIN; //output direction

PHY\_POWER3\_PXOUT|=PHY\_POWER3\_PXPIN; //output high for open AAT3221

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC3 AAT3221 Switch Control P10.2

UB0/HT 电源接口 关闭AAT3221第三路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power3\_3221\_Off(void)

{

PHY\_POWER3\_PXSEL&=~PHY\_POWER3\_PXPIN; //general IO

PHY\_POWER3\_PXIDR|=PHY\_POWER3\_PXPIN; //output direction

PHY\_POWER3\_PXOUT&=~PHY\_POWER3\_PXPIN; //output low for close AAT3221

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC6 AA4280 Switch Control P10.3

UA3/COM3电源接口 打开AAT4280第四路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power4\_4280\_On(void) //打开IC6第四路电源

{

PHY\_POWER4\_PXSEL&=~PHY\_POWER4\_PXPIN; //general IO

PHY\_POWER4\_PXIDR|=PHY\_POWER4\_PXPIN; //output direction

PHY\_POWER4\_PXOUT|=PHY\_POWER4\_PXPIN; //output low for open AAT4280

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IC6 AA4280 Switch Control P10.3

UA3/COM3电源接口 关闭AAT4280第四路电源

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Power4\_4280\_Off(void)

{

PHY\_POWER4\_PXSEL&=~PHY\_POWER4\_PXPIN; //general IO

PHY\_POWER4\_PXIDR|=PHY\_POWER4\_PXPIN; //output direction

PHY\_POWER4\_PXOUT&=~PHY\_POWER4\_PXPIN; //output low for close AAT4280

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UA0/COM0 TTL/232控制管脚

MAX3221开启与关闭控制 P5.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void EnableRs232A0(void) //打开UA0 TTL/RS232转换芯片

{

PHY\_232A0\_PXSEL&=~PHY\_232A0\_PXPIN; //general IO

PHY\_232A0\_PXDIR|=PHY\_232A0\_PXPIN; //output direction

PHY\_232A0\_PXOUT|=PHY\_232A0\_PXPIN; //output high for open MAX3221

}

void DisableRs232A0(void)// 关闭UA0 TTL/RS232转换芯片

{

PHY\_232A0\_PXSEL&=~PHY\_232A0\_PXPIN; //general IO

PHY\_232A0\_PXDIR|=PHY\_232A0\_PXPIN; //output direction

PHY\_232A0\_PXOUT&=~PHY\_232A0\_PXPIN; //output low for close MAX3221

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UA3/COM3 TTL/232控制管脚

MAX3221开启与关闭控制 P11.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void EnableRs232A3(void) //打开UA3 TTL/RS232转换芯片

{

PHY\_232A3\_PXSEL&=~PHY\_232A3\_PXPIN; //general IO

PHY\_232A3\_PXDIR|=PHY\_232A3\_PXPIN; //output direction

PHY\_232A3\_PXOUT|=PHY\_232A3\_PXPIN; //output high for open MAX3221

}

void DisableRs232A3(void)// 关闭UA3 TTL/RS232转换芯片

{

PHY\_232A3\_PXSEL&=~PHY\_232A3\_PXPIN; //general IO

PHY\_232A3\_PXDIR|=PHY\_232A3\_PXPIN; //output direction

PHY\_232A3\_PXOUT&=~PHY\_232A3\_PXPIN; //output low for close MAX3221

}