|  |
| --- |
| /\*-----------------------------------------------  名称：程序主控文件  编写：夏杰  日期：2017.4.1  内容：程序入口，及主要控制部分  ------------------------------------------------\*/  #include "sc.h"  /\*-----------------------------------------------  TODO：主函数  参数：  编写：夏杰  日期：2017.4.1  ------------------------------------------------\*/  void main()  {  // 系统初始化  sc\_init();  DelayMs(500);  // 液晶布局  led\_wirte\_str(20, 0, "xzit-Xia Jie");  while(1)  {  // 停车  if(rec\_data == 0)  {  motor\_left\_stop();  motor\_right\_stop();  steer\_middle();  }  // 前进  else if(rec\_data == 1)  {  motor\_left\_forward(100);  motor\_right\_forward(100);  steer\_middle();  }  // 倒车  else if(rec\_data == 2)  {  motor\_left\_back(100);  motor\_right\_back(100);  steer\_middle();  }  // 左转  else if(rec\_data == 3)  {  motor\_left\_forward(100);  motor\_right\_forward(100);  steer\_left();  }  // 右转  else if(rec\_data == 4)  {  motor\_left\_forward(100);  motor\_right\_forward(100);  steer\_right();  }  // 液晶刷新  if(led\_refresh\_flag == 1)  {  led\_refresh\_flag = 0;  led\_write\_num(10, 2, rec\_data);  }  }  }  /\*-----------------------------------------------  名称：蓝牙调用函数库  编写：夏杰  日期：2017.4.24  内容：蓝牙模块相关控制函数  ------------------------------------------------\*/  #include "sc.h"  // 定义接收的数据  uchar rec\_data = 0;  /\*-----------------------------------------------  TODO：蓝牙模块初始化  参数：  编写：夏杰  日期：2017.4.24  ------------------------------------------------\*/  void blueteeth\_init()  {  // SCON = 0x50; //8位数据,可变波特率  // TMOD |= 0x20; //设定定时器1为8位自动重装方式  // TL1 = 0xFD; //设定定时初值  // TH1 = 0xFD; //设定定时器重装值  // ET1 = 0; //禁止定时器1中断  // TR1 = 1; //启动定时器1  // ES = 1;  SCON = 0x50; // SCON: 模式 1, 8-bit UART, 使能接收  TMOD |= 0x20; // TMOD: timer 1, mode 2, 8-bit 重装  TH1 = 0xFD; // TH1: 重装值 9600 波特率 晶振 11.0592MHz  TR1 = 0xFD; // TR1: timer 1 打开  ET1 = 0; //打开总中断  ES = 1; //打开串口中断  }  /\*-----------------------------------------------  TODO：串口发送数据到蓝牙模块  参数：dat 要发送的数据  编写：夏杰  日期：2017.4.24  ------------------------------------------------\*/  void blueteeth\_send\_byte(uchar dat)  {  SBUF = dat;  while(!TI);  TI = 0;  }  /\*-----------------------------------------------  TODO：蓝牙发送字符串  参数：\*s 字符串首地址  编写：夏杰  日期：2017.4.24  ------------------------------------------------\*/  void blueteeth\_send\_str(uchar \*s)  {  while(\*s != '\0') // \0 表示字符串结束标志，通过检测是否字符串末尾  {  blueteeth\_send\_byte(\*s);  s++;  }  }  /\*-----------------------------------------------  TODO：串口中断程序  参数：null  编写：夏杰  日期：2017.4.24  ------------------------------------------------\*/  void UART\_SER (void) interrupt 4 //串行中断服务程序  {  if( RI ) //判断是接收中断产生  {  RI = 0; //标志位清零  rec\_data = SBUF - 48; //读入缓冲区的值  }  }  /\*-----------------------------------------------  名称：延迟函数库  编写：夏杰  日期：2017.4.8  内容：通过变量自减，以实现延时  ------------------------------------------------\*/  #include "delay.h"  #include <intrins.h>  /\*-----------------------------------------------  TODO：delay for 2\*t us  参数：t delay for 2\*t us  编写：夏杰  日期：2017.4.8  ------------------------------------------------\*/  void DelayUs2x(unsigned int t)  {  // 自减循环  while(--t);  }  /\*-----------------------------------------------  TODO：delay for t mS  参数：t delay for t mS  编写：夏杰  日期：2017.4.8  ------------------------------------------------\*/  void DelayMs(unsigned int t)  {  unsigned int i = 0, j = 0;  for(i = t;i > 0;i--)  for(j = 110;j > 0;j--);  // 调整延迟  DelayUs2x(0);  }  /\*-----------------------------------------------  名称：OLED液晶函数库  编写：夏杰  日期：2017.4.22  内容：OLED液晶相关函数、数据  ------------------------------------------------\*/  #include "sc.h"  #include <intrins.h>  // 变量  uchar led\_refresh\_flag = 0;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_WrCmd  功能描述 : 向OLED写命令  输入参数 : UCHAR8 ucCmd  输出参数 : NONE  返 回 值 : NONE  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_WrCmd(uchar ucCmd)  {  uchar i = 8;  LED\_DC = 0;  LED\_SCL = 0;  while (i--)  {  if (ucCmd & 0x80)  {  LED\_SDA = 1;  }  else  {  LED\_SDA = 0;  }  LED\_SCL = 1;  \_nop\_();  LED\_SCL = 0;  ucCmd <<= 1;  }  return;  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_WrDat  功能描述 : 向OLED写数据  输入参数 : UCHAR8 ucData  输出参数 : NONE  返 回 值 : NONE  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_WrDat(uchar ucData)  {  uchar i = 8;  LED\_DC = 1;  LED\_SCL = 0;    while (i--)  {  if (ucData & 0x80)  {  LED\_SDA = 1;  }  else  {  LED\_SDA = 0;  }    LED\_SCL = 1;  \_nop\_();  LED\_SCL = 0;  ucData <<= 1;  }  return;  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_SetPos  功能描述 : 设置坐标  输入参数 : UCHAR8 ucIdxX  UCHAR8 ucIdxY  输出参数 : NONE  返 回 值 : NONE  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_SetPos(uchar ucIdxX, uchar ucIdxY)  {  LED\_WrCmd(0xb0 + ucIdxY);  LED\_WrCmd(((ucIdxX & 0xf0) >> 4) | 0x10);  LED\_WrCmd((ucIdxX & 0x0f) | 0x00);  }  void SetDisplayOnOff(uchar ucData)  {  LED\_WrCmd(0xAE|ucData); // Set Display On/Off  // Default => 0xAE  // 0xAE (0x00) => Display Off  // 0xAF (0x01) => Display On  }  void SetDisplayClock(uchar ucData)  {  LED\_WrCmd(0xD5); // Set Display Clock Divide Ratio / Oscillator Frequency  LED\_WrCmd(ucData); // Default => 0x80  // D[3:0] => Display Clock Divider  // D[7:4] => Oscillator Frequency  }  void SetMultiplexRatio(uchar ucData)  {  LED\_WrCmd(0xA8); // Set Multiplex Ratio  LED\_WrCmd(ucData); // Default => 0x3F (1/64 Duty)  }  void SetDisplayOffset(uchar ucData)  {  LED\_WrCmd(0xD3); // Set Display Offset  LED\_WrCmd(ucData); // Default => 0x00  }  void SetStartLine(uchar ucData)  {  LED\_WrCmd(0x40|ucData); // Set Display Start Line  // Default => 0x40 (0x00)  }  void SetAddressingMode(uchar ucData)  {  LED\_WrCmd(0x20); // Set Memory Addressing Mode  LED\_WrCmd(ucData); // Default => 0x02  // 0x00 => Horizontal Addressing Mode  // 0x01 => Vertical Addressing Mode  // 0x02 => Page Addressing Mode  }  void SetChargePump(uchar ucData)  {  LED\_WrCmd(0x8D); // Set Charge Pump  LED\_WrCmd(0x10|ucData); // Default => 0x10  // 0x10 (0x00) => Disable Charge Pump  // 0x14 (0x04) => Enable Charge Pump  }  void SetSegmentRemap(uchar ucData)  {  LED\_WrCmd(0xA0|ucData); // Set Segment Re-Map  // Default => 0xA0  // 0xA0 (0x00) => Column Address 0 Mapped to SEG0  // 0xA1 (0x01) => Column Address 0 Mapped to SEG127  }  void SetCommonRemap(uchar ucData)  {  LED\_WrCmd(0xC0|ucData); // Set COM Output Scan Direction  // Default => 0xC0  // 0xC0 (0x00) => Scan from COM0 to 63  // 0xC8 (0x08) => Scan from COM63 to 0  }  void SetCommonConfig(uchar ucData)  {  LED\_WrCmd(0xDA); // Set COM Pins Hardware Configuration  LED\_WrCmd(0x02|ucData); // Default => 0x12 (0x10)  // Alternative COM Pin Configuration  // Disable COM Left/Right Re-Map  }  void SetContrastControl(uchar ucData)  {  LED\_WrCmd(0x81); // Set Contrast Control  LED\_WrCmd(ucData); // Default => 0x7F  }  void SetPrechargePeriod(uchar ucData)  {  LED\_WrCmd(0xD9); // Set Pre-Charge Period  LED\_WrCmd(ucData); // Default => 0x22 (2 Display Clocks [Phase 2] / 2 Display Clocks [Phase 1])  // D[3:0] => Phase 1 Period in 1~15 Display Clocks  // D[7:4] => Phase 2 Period in 1~15 Display Clocks  }  void SetVCOMH(uchar ucData)  {  LED\_WrCmd(0xDB); // Set VCOMH Deselect Level  LED\_WrCmd(ucData); // Default => 0x20 (0.77\*VCC)  }  void SetEntireDisplay(uchar ucData)  {  LED\_WrCmd(0xA4|ucData); // Set Entire Display On / Off  // Default => 0xA4  // 0xA4 (0x00) => Normal Display  // 0xA5 (0x01) => Entire Display On  }  void SetInverseDisplay(uchar ucData)  {  LED\_WrCmd(0xA6|ucData); // Set Inverse Display On/Off  // Default => 0xA6  // 0xA6 (0x00) => Normal Display  // 0xA7 (0x01) => Inverse Display On  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_Fill  功能描述 : 对全屏写入同一个字符函数  如 LED\_Fill(0x01); 对于某一位0为不亮 1为亮  ------------------------------------------------------  共128列  1 1 1  共 0 0 0  8 0 0 0  组 0 ……0 ………… 0  字 0 0 0  符 0 0 0  即 0 0 0  8 0 0 0  页 1 1 1  64 0 0 0  行 . . .  . . ………… .  . . .  . . .  0 0 0  ------------------------------------------------------  输入参数 : UCHAR8 ucData  输出参数 : NONE  返 回 值 : NONE  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_Fill(uchar ucData)  {  uchar ucPage, ucColumn;    for(ucPage = 0; ucPage < 8; ucPage++)  {  LED\_WrCmd(0xb0 + ucPage); //0xb0+0~7表示页0~7  LED\_WrCmd(0x00); //0x00+0~16表示将128列分成16组其地址在某组中的第几列  LED\_WrCmd(0x10); //0x10+0~16表示将128列分成16组其地址所在第几组  for(ucColumn = 0; ucColumn < 128; ucColumn++)  {  LED\_WrDat(ucData);  }  }  }  /\*-----------------------------------------------  TODO：OLED初始化  参数：  编写：夏杰  日期：2017.4.22  ------------------------------------------------\*/  void LED\_init()  {  uchar i;  LED\_SCL = 1;  LED\_RST = 0;    for(i = 0; i < 100; i++)  {  \_nop\_(); //从上电到下面开始初始化要有足够的时间，即等待RC复位完毕  }    LED\_RST = 1;  SetDisplayOnOff(0x00); // Display Off (0x00/0x01)  SetDisplayClock(0x80); // Set Clock as 100 Frames/Sec  SetMultiplexRatio(0x3F); // 1/64 Duty (0x0F~0x3F)  SetDisplayOffset(0x00); // Shift Mapping RAM Counter (0x00~0x3F)  SetStartLine(0x00); // Set Mapping RAM Display Start Line (0x00~0x3F)  SetChargePump(0x04); // Enable Embedded DC/DC Converter (0x00/0x04)  SetAddressingMode(0x02); // Set Page Addressing Mode (0x00/0x01/0x02)  SetSegmentRemap(0x01); // Set SEG/Column Mapping 0x00左右反置 0x01正常  SetCommonRemap(0x08); // Set COM/Row Scan Direction 0x00上下反置 0x08正常  SetCommonConfig(0x10); // Set Sequential Configuration (0x00/0x10)  SetContrastControl(0xCF); // Set SEG Output Current  SetPrechargePeriod(0xF1); // Set Pre-Charge as 15 Clocks & Discharge as 1 Clock  SetVCOMH(0x40); // Set VCOM Deselect Level  SetEntireDisplay(0x00); // Disable Entire Display On (0x00/0x01)  SetInverseDisplay(0x00); // Disable Inverse Display On (0x00/0x01)  SetDisplayOnOff(0x01); // Display On (0x00/0x01)  LED\_Fill(0x00); // 初始清屏  LED\_SetPos(0,0);  return;  }  const uchar code F6x8[][6] =  {  { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }, //sp0  { 0x00, 0x00, 0x00, 0x2f, 0x00, 0x00 }, // !1  { 0x00, 0x00, 0x07, 0x00, 0x07, 0x00 }, // "2  { 0x00, 0x14, 0x7f, 0x14, 0x7f, 0x14 }, // #3  { 0x00, 0x24, 0x2a, 0x7f, 0x2a, 0x12 }, // $4  { 0x00, 0x62, 0x64, 0x08, 0x13, 0x23 }, // %5  { 0x00, 0x36, 0x49, 0x55, 0x22, 0x50 }, // &6  { 0x00, 0x00, 0x05, 0x03, 0x00, 0x00 }, // '7  { 0x00, 0x00, 0x1c, 0x22, 0x41, 0x00 }, // (8  { 0x00, 0x00, 0x41, 0x22, 0x1c, 0x00 }, // )9  { 0x00, 0x14, 0x08, 0x3E, 0x08, 0x14 }, // \*10  { 0x00, 0x08, 0x08, 0x3E, 0x08, 0x08 }, // +11  { 0x00, 0x00, 0x00, 0xA0, 0x60, 0x00 }, // ,12  { 0x00, 0x08, 0x08, 0x08, 0x08, 0x08 }, // -13  { 0x00, 0x00, 0x60, 0x60, 0x00, 0x00 }, // .14  { 0x00, 0x20, 0x10, 0x08, 0x04, 0x02 }, // /15  { 0x00, 0x3E, 0x51, 0x49, 0x45, 0x3E }, // 016  { 0x00, 0x00, 0x42, 0x7F, 0x40, 0x00 }, // 117  { 0x00, 0x42, 0x61, 0x51, 0x49, 0x46 }, // 218  { 0x00, 0x21, 0x41, 0x45, 0x4B, 0x31 }, // 319  { 0x00, 0x18, 0x14, 0x12, 0x7F, 0x10 }, // 420  { 0x00, 0x27, 0x45, 0x45, 0x45, 0x39 }, // 521  { 0x00, 0x3C, 0x4A, 0x49, 0x49, 0x30 }, // 622  { 0x00, 0x01, 0x71, 0x09, 0x05, 0x03 }, // 723  { 0x00, 0x36, 0x49, 0x49, 0x49, 0x36 }, // 824  { 0x00, 0x06, 0x49, 0x49, 0x29, 0x1E }, // 925  { 0x00, 0x00, 0x36, 0x36, 0x00, 0x00 }, // :26  { 0x00, 0x00, 0x56, 0x36, 0x00, 0x00 }, // ;27  { 0x00, 0x08, 0x14, 0x22, 0x41, 0x00 }, // <28  { 0x00, 0x14, 0x14, 0x14, 0x14, 0x14 }, // =29  { 0x00, 0x00, 0x41, 0x22, 0x14, 0x08 }, // >30  { 0x00, 0x02, 0x01, 0x51, 0x09, 0x06 }, // ?31  { 0x00, 0x32, 0x49, 0x59, 0x51, 0x3E }, // @32  { 0x00, 0x7C, 0x12, 0x11, 0x12, 0x7C }, // A33  { 0x00, 0x7F, 0x49, 0x49, 0x49, 0x36 }, // B34  { 0x00, 0x3E, 0x41, 0x41, 0x41, 0x22 }, // C35  { 0x00, 0x7F, 0x41, 0x41, 0x22, 0x1C }, // D36  { 0x00, 0x7F, 0x49, 0x49, 0x49, 0x41 }, // E37  { 0x00, 0x7F, 0x09, 0x09, 0x09, 0x01 }, // F38  { 0x00, 0x3E, 0x41, 0x49, 0x49, 0x7A }, // G39  { 0x00, 0x7F, 0x08, 0x08, 0x08, 0x7F }, // H40  { 0x00, 0x00, 0x41, 0x7F, 0x41, 0x00 }, // I41  { 0x00, 0x20, 0x40, 0x41, 0x3F, 0x01 }, // J42  { 0x00, 0x7F, 0x08, 0x14, 0x22, 0x41 }, // K43  { 0x00, 0x7F, 0x40, 0x40, 0x40, 0x40 }, // L44  { 0x00, 0x7F, 0x02, 0x0C, 0x02, 0x7F }, // M45  { 0x00, 0x7F, 0x04, 0x08, 0x10, 0x7F }, // N46  { 0x00, 0x3E, 0x41, 0x41, 0x41, 0x3E }, // O47  { 0x00, 0x7F, 0x09, 0x09, 0x09, 0x06 }, // P48  { 0x00, 0x3E, 0x41, 0x51, 0x21, 0x5E }, // Q49  { 0x00, 0x7F, 0x09, 0x19, 0x29, 0x46 }, // R50  { 0x00, 0x46, 0x49, 0x49, 0x49, 0x31 }, // S51  { 0x00, 0x01, 0x01, 0x7F, 0x01, 0x01 }, // T52  { 0x00, 0x3F, 0x40, 0x40, 0x40, 0x3F }, // U53  { 0x00, 0x1F, 0x20, 0x40, 0x20, 0x1F }, // V54  { 0x00, 0x3F, 0x40, 0x38, 0x40, 0x3F }, // W55  { 0x00, 0x63, 0x14, 0x08, 0x14, 0x63 }, // X56  { 0x00, 0x07, 0x08, 0x70, 0x08, 0x07 }, // Y57  { 0x00, 0x61, 0x51, 0x49, 0x45, 0x43 }, // Z58  { 0x00, 0x00, 0x7F, 0x41, 0x41, 0x00 }, // [59  { 0x00, 0x02, 0x04, 0x08, 0x10, 0x20 }, // \60  { 0x00, 0x00, 0x41, 0x41, 0x7F, 0x00 }, // ]61  { 0x00, 0x04, 0x02, 0x01, 0x02, 0x04 }, // ^62  { 0x00, 0x40, 0x40, 0x40, 0x40, 0x40 }, // \_63  { 0x00, 0x00, 0x01, 0x02, 0x04, 0x00 }, // '64  { 0x00, 0x20, 0x54, 0x54, 0x54, 0x78 }, // a65  { 0x00, 0x7F, 0x48, 0x44, 0x44, 0x38 }, // b66  { 0x00, 0x38, 0x44, 0x44, 0x44, 0x20 }, // c67  { 0x00, 0x38, 0x44, 0x44, 0x48, 0x7F }, // d68  { 0x00, 0x38, 0x54, 0x54, 0x54, 0x18 }, // e69  { 0x00, 0x08, 0x7E, 0x09, 0x01, 0x02 }, // f70  { 0x00, 0x18, 0xA4, 0xA4, 0xA4, 0x7C }, // g71  { 0x00, 0x7F, 0x08, 0x04, 0x04, 0x78 }, // h72  { 0x00, 0x00, 0x44, 0x7D, 0x40, 0x00 }, // i73  { 0x00, 0x40, 0x80, 0x84, 0x7D, 0x00 }, // j74  { 0x00, 0x7F, 0x10, 0x28, 0x44, 0x00 }, // k75  { 0x00, 0x00, 0x41, 0x7F, 0x40, 0x00 }, // l76  { 0x00, 0x7C, 0x04, 0x18, 0x04, 0x78 }, // m77  { 0x00, 0x7C, 0x08, 0x04, 0x04, 0x78 }, // n78  { 0x00, 0x38, 0x44, 0x44, 0x44, 0x38 }, // o79  { 0x00, 0xFC, 0x24, 0x24, 0x24, 0x18 }, // p80  { 0x00, 0x18, 0x24, 0x24, 0x18, 0xFC }, // q81  { 0x00, 0x7C, 0x08, 0x04, 0x04, 0x08 }, // r82  { 0x00, 0x48, 0x54, 0x54, 0x54, 0x20 }, // s83  { 0x00, 0x04, 0x3F, 0x44, 0x40, 0x20 }, // t84  { 0x00, 0x3C, 0x40, 0x40, 0x20, 0x7C }, // u85  { 0x00, 0x1C, 0x20, 0x40, 0x20, 0x1C }, // v86  { 0x00, 0x3C, 0x40, 0x30, 0x40, 0x3C }, // w87  { 0x00, 0x44, 0x28, 0x10, 0x28, 0x44 }, // x88  { 0x00, 0x1C, 0xA0, 0xA0, 0xA0, 0x7C }, // y89  { 0x00, 0x44, 0x64, 0x54, 0x4C, 0x44 }, // z90  { 0x14, 0x14, 0x14, 0x14, 0x14, 0x14 } // horiz lines91  };  /\*======================================================  128X64OLED液晶底层驱动[8X16]字体库  设计者: Andy  描 述: [8X16]西文字符的字模数据 (纵向取模,字节倒序)  !"#$%&'()\*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^\_`abcdefghijklmnopqrstuvwxyz{|}~  ======================================================\*/  const uchar code F8X16[]=  {  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,// 0  0x00,0x00,0x00,0xF8,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x33,0x30,0x00,0x00,0x00,//!1  0x00,0x10,0x0C,0x06,0x10,0x0C,0x06,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//"2  0x40,0xC0,0x78,0x40,0xC0,0x78,0x40,0x00,0x04,0x3F,0x04,0x04,0x3F,0x04,0x04,0x00,//#3  0x00,0x70,0x88,0xFC,0x08,0x30,0x00,0x00,0x00,0x18,0x20,0xFF,0x21,0x1E,0x00,0x00,//$4  0xF0,0x08,0xF0,0x00,0xE0,0x18,0x00,0x00,0x00,0x21,0x1C,0x03,0x1E,0x21,0x1E,0x00,//%5  0x00,0xF0,0x08,0x88,0x70,0x00,0x00,0x00,0x1E,0x21,0x23,0x24,0x19,0x27,0x21,0x10,//&6  0x10,0x16,0x0E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//'7  0x00,0x00,0x00,0xE0,0x18,0x04,0x02,0x00,0x00,0x00,0x00,0x07,0x18,0x20,0x40,0x00,//(8  0x00,0x02,0x04,0x18,0xE0,0x00,0x00,0x00,0x00,0x40,0x20,0x18,0x07,0x00,0x00,0x00,//)9  0x40,0x40,0x80,0xF0,0x80,0x40,0x40,0x00,0x02,0x02,0x01,0x0F,0x01,0x02,0x02,0x00,//\*10  0x00,0x00,0x00,0xF0,0x00,0x00,0x00,0x00,0x01,0x01,0x01,0x1F,0x01,0x01,0x01,0x00,//+11  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0xB0,0x70,0x00,0x00,0x00,0x00,0x00,//,12  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x01,0x01,0x01,0x01,0x01,0x01,//-13  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x30,0x00,0x00,0x00,0x00,0x00,//.14  0x00,0x00,0x00,0x00,0x80,0x60,0x18,0x04,0x00,0x60,0x18,0x06,0x01,0x00,0x00,0x00,///15  0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,0x00,0x0F,0x10,0x20,0x20,0x10,0x0F,0x00,//016  0x00,0x10,0x10,0xF8,0x00,0x00,0x00,0x00,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//117  0x00,0x70,0x08,0x08,0x08,0x88,0x70,0x00,0x00,0x30,0x28,0x24,0x22,0x21,0x30,0x00,//218  0x00,0x30,0x08,0x88,0x88,0x48,0x30,0x00,0x00,0x18,0x20,0x20,0x20,0x11,0x0E,0x00,//319  0x00,0x00,0xC0,0x20,0x10,0xF8,0x00,0x00,0x00,0x07,0x04,0x24,0x24,0x3F,0x24,0x00,//420  0x00,0xF8,0x08,0x88,0x88,0x08,0x08,0x00,0x00,0x19,0x21,0x20,0x20,0x11,0x0E,0x00,//521  0x00,0xE0,0x10,0x88,0x88,0x18,0x00,0x00,0x00,0x0F,0x11,0x20,0x20,0x11,0x0E,0x00,//622  0x00,0x38,0x08,0x08,0xC8,0x38,0x08,0x00,0x00,0x00,0x00,0x3F,0x00,0x00,0x00,0x00,//723  0x00,0x70,0x88,0x08,0x08,0x88,0x70,0x00,0x00,0x1C,0x22,0x21,0x21,0x22,0x1C,0x00,//824  0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,0x00,0x00,0x31,0x22,0x22,0x11,0x0F,0x00,//925  0x00,0x00,0x00,0xC0,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x30,0x00,0x00,0x00,//:26  0x00,0x00,0x00,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x60,0x00,0x00,0x00,0x00,//;27  0x00,0x00,0x80,0x40,0x20,0x10,0x08,0x00,0x00,0x01,0x02,0x04,0x08,0x10,0x20,0x00,//<28  0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x00,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x00,//=29  0x00,0x08,0x10,0x20,0x40,0x80,0x00,0x00,0x00,0x20,0x10,0x08,0x04,0x02,0x01,0x00,//>30  0x00,0x70,0x48,0x08,0x08,0x08,0xF0,0x00,0x00,0x00,0x00,0x30,0x36,0x01,0x00,0x00,//?31  0xC0,0x30,0xC8,0x28,0xE8,0x10,0xE0,0x00,0x07,0x18,0x27,0x24,0x23,0x14,0x0B,0x00,//@32  0x00,0x00,0xC0,0x38,0xE0,0x00,0x00,0x00,0x20,0x3C,0x23,0x02,0x02,0x27,0x38,0x20,//A33  0x08,0xF8,0x88,0x88,0x88,0x70,0x00,0x00,0x20,0x3F,0x20,0x20,0x20,0x11,0x0E,0x00,//B34  0xC0,0x30,0x08,0x08,0x08,0x08,0x38,0x00,0x07,0x18,0x20,0x20,0x20,0x10,0x08,0x00,//C35  0x08,0xF8,0x08,0x08,0x08,0x10,0xE0,0x00,0x20,0x3F,0x20,0x20,0x20,0x10,0x0F,0x00,//D36  0x08,0xF8,0x88,0x88,0xE8,0x08,0x10,0x00,0x20,0x3F,0x20,0x20,0x23,0x20,0x18,0x00,//E37  0x08,0xF8,0x88,0x88,0xE8,0x08,0x10,0x00,0x20,0x3F,0x20,0x00,0x03,0x00,0x00,0x00,//F38  0xC0,0x30,0x08,0x08,0x08,0x38,0x00,0x00,0x07,0x18,0x20,0x20,0x22,0x1E,0x02,0x00,//G39  0x08,0xF8,0x08,0x00,0x00,0x08,0xF8,0x08,0x20,0x3F,0x21,0x01,0x01,0x21,0x3F,0x20,//H40  0x00,0x08,0x08,0xF8,0x08,0x08,0x00,0x00,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//I41  0x00,0x00,0x08,0x08,0xF8,0x08,0x08,0x00,0xC0,0x80,0x80,0x80,0x7F,0x00,0x00,0x00,//J42  0x08,0xF8,0x88,0xC0,0x28,0x18,0x08,0x00,0x20,0x3F,0x20,0x01,0x26,0x38,0x20,0x00,//K43  0x08,0xF8,0x08,0x00,0x00,0x00,0x00,0x00,0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x00,//L44  0x08,0xF8,0xF8,0x00,0xF8,0xF8,0x08,0x00,0x20,0x3F,0x00,0x3F,0x00,0x3F,0x20,0x00,//M45  0x08,0xF8,0x30,0xC0,0x00,0x08,0xF8,0x08,0x20,0x3F,0x20,0x00,0x07,0x18,0x3F,0x00,//N46  0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,0x0F,0x10,0x20,0x20,0x20,0x10,0x0F,0x00,//O47  0x08,0xF8,0x08,0x08,0x08,0x08,0xF0,0x00,0x20,0x3F,0x21,0x01,0x01,0x01,0x00,0x00,//P48  0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,0x0F,0x18,0x24,0x24,0x38,0x50,0x4F,0x00,//Q49  0x08,0xF8,0x88,0x88,0x88,0x88,0x70,0x00,0x20,0x3F,0x20,0x00,0x03,0x0C,0x30,0x20,//R50  0x00,0x70,0x88,0x08,0x08,0x08,0x38,0x00,0x00,0x38,0x20,0x21,0x21,0x22,0x1C,0x00,//S51  0x18,0x08,0x08,0xF8,0x08,0x08,0x18,0x00,0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,//T52  0x08,0xF8,0x08,0x00,0x00,0x08,0xF8,0x08,0x00,0x1F,0x20,0x20,0x20,0x20,0x1F,0x00,//U53  0x08,0x78,0x88,0x00,0x00,0xC8,0x38,0x08,0x00,0x00,0x07,0x38,0x0E,0x01,0x00,0x00,//V54  0xF8,0x08,0x00,0xF8,0x00,0x08,0xF8,0x00,0x03,0x3C,0x07,0x00,0x07,0x3C,0x03,0x00,//W55  0x08,0x18,0x68,0x80,0x80,0x68,0x18,0x08,0x20,0x30,0x2C,0x03,0x03,0x2C,0x30,0x20,//X56  0x08,0x38,0xC8,0x00,0xC8,0x38,0x08,0x00,0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,//Y57  0x10,0x08,0x08,0x08,0xC8,0x38,0x08,0x00,0x20,0x38,0x26,0x21,0x20,0x20,0x18,0x00,//Z58  0x00,0x00,0x00,0xFE,0x02,0x02,0x02,0x00,0x00,0x00,0x00,0x7F,0x40,0x40,0x40,0x00,//[59  0x00,0x0C,0x30,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x06,0x38,0xC0,0x00,//\60  0x00,0x02,0x02,0x02,0xFE,0x00,0x00,0x00,0x00,0x40,0x40,0x40,0x7F,0x00,0x00,0x00,//]61  0x00,0x00,0x04,0x02,0x02,0x02,0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//^62  0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,//\_63  0x00,0x02,0x02,0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//`64  0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,0x00,0x19,0x24,0x22,0x22,0x22,0x3F,0x20,//a65  0x08,0xF8,0x00,0x80,0x80,0x00,0x00,0x00,0x00,0x3F,0x11,0x20,0x20,0x11,0x0E,0x00,//b66  0x00,0x00,0x00,0x80,0x80,0x80,0x00,0x00,0x00,0x0E,0x11,0x20,0x20,0x20,0x11,0x00,//c67  0x00,0x00,0x00,0x80,0x80,0x88,0xF8,0x00,0x00,0x0E,0x11,0x20,0x20,0x10,0x3F,0x20,//d68  0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,0x00,0x1F,0x22,0x22,0x22,0x22,0x13,0x00,//e69  0x00,0x80,0x80,0xF0,0x88,0x88,0x88,0x18,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//f70  0x00,0x00,0x80,0x80,0x80,0x80,0x80,0x00,0x00,0x6B,0x94,0x94,0x94,0x93,0x60,0x00,//g71  0x08,0xF8,0x00,0x80,0x80,0x80,0x00,0x00,0x20,0x3F,0x21,0x00,0x00,0x20,0x3F,0x20,//h72  0x00,0x80,0x98,0x98,0x00,0x00,0x00,0x00,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//i73  0x00,0x00,0x00,0x80,0x98,0x98,0x00,0x00,0x00,0xC0,0x80,0x80,0x80,0x7F,0x00,0x00,//j74  0x08,0xF8,0x00,0x00,0x80,0x80,0x80,0x00,0x20,0x3F,0x24,0x02,0x2D,0x30,0x20,0x00,//k75  0x00,0x08,0x08,0xF8,0x00,0x00,0x00,0x00,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//l76  0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x00,0x20,0x3F,0x20,0x00,0x3F,0x20,0x00,0x3F,//m77  0x80,0x80,0x00,0x80,0x80,0x80,0x00,0x00,0x20,0x3F,0x21,0x00,0x00,0x20,0x3F,0x20,//n78  0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,0x00,0x1F,0x20,0x20,0x20,0x20,0x1F,0x00,//o79  0x80,0x80,0x00,0x80,0x80,0x00,0x00,0x00,0x80,0xFF,0xA1,0x20,0x20,0x11,0x0E,0x00,//p80  0x00,0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,0x0E,0x11,0x20,0x20,0xA0,0xFF,0x80,//q81  0x80,0x80,0x80,0x00,0x80,0x80,0x80,0x00,0x20,0x20,0x3F,0x21,0x20,0x00,0x01,0x00,//r82  0x00,0x00,0x80,0x80,0x80,0x80,0x80,0x00,0x00,0x33,0x24,0x24,0x24,0x24,0x19,0x00,//s83  0x00,0x80,0x80,0xE0,0x80,0x80,0x00,0x00,0x00,0x00,0x00,0x1F,0x20,0x20,0x00,0x00,//t84  0x80,0x80,0x00,0x00,0x00,0x80,0x80,0x00,0x00,0x1F,0x20,0x20,0x20,0x10,0x3F,0x20,//u85  0x80,0x80,0x80,0x00,0x00,0x80,0x80,0x80,0x00,0x01,0x0E,0x30,0x08,0x06,0x01,0x00,//v86  0x80,0x80,0x00,0x80,0x00,0x80,0x80,0x80,0x0F,0x30,0x0C,0x03,0x0C,0x30,0x0F,0x00,//w87  0x00,0x80,0x80,0x00,0x80,0x80,0x80,0x00,0x00,0x20,0x31,0x2E,0x0E,0x31,0x20,0x00,//x88  0x80,0x80,0x80,0x00,0x00,0x80,0x80,0x80,0x80,0x81,0x8E,0x70,0x18,0x06,0x01,0x00,//y89  0x00,0x80,0x80,0x80,0x80,0x80,0x80,0x00,0x00,0x21,0x30,0x2C,0x22,0x21,0x30,0x00,//z90  0x00,0x00,0x00,0x00,0x80,0x7C,0x02,0x02,0x00,0x00,0x00,0x00,0x00,0x3F,0x40,0x40,//{91  0x00,0x00,0x00,0x00,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0x00,0x00,0x00,//|92  0x00,0x02,0x02,0x7C,0x80,0x00,0x00,0x00,0x00,0x40,0x40,0x3F,0x00,0x00,0x00,0x00,//}93  0x00,0x06,0x01,0x01,0x02,0x02,0x04,0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//~94  };  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_P8x16Str  功能描述 : 写入一组8x16标准ASCII字符串  输入参数 : uchar ucIdxX 为显示的横坐标0~120  uchar ucIdxY 为页范围0～3  uchar ucDataStr[] 要显示的字符串  输出参数 : uchar  返 回 值 : uchar  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_P8x16Str(uchar ucIdxX, uchar ucIdxY, uchar ucDataStr[])  {  uchar i, j, ucDataTmp;  ucIdxY <<= 1;    for (j = 0; ucDataStr[j] != '\0'; j++)  {  ucDataTmp = ucDataStr[j] - 32;  if(ucIdxX > 120)  {  ucIdxX = 0;  ucIdxY += 2;  }  LED\_SetPos(ucIdxX, ucIdxY);    for(i = 0; i < 8; i++)  {  LED\_WrDat(F8X16[(ucDataTmp << 4) + i]);  }    LED\_SetPos(ucIdxX, ucIdxY + 1);    for(i = 0; i < 8; i++)  {  LED\_WrDat(F8X16[(ucDataTmp << 4) + i + 8]);  }  ucIdxX += 8;    }  return;  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_P6x8Char  功能描述 : 显示一个6x8标准ASCII字符  输入参数 : UCHAR8 ucIdxX 显示的横坐标0~122  UCHAR8 ucIdxY 页范围0～7  UCHAR8 ucData 显示的字符  输出参数 : NONE  返 回 值 : NONE  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_P6x8Char(uchar ucIdxX, uchar ucIdxY, uchar ucData)  {  uchar i, ucDataTmp;    ucDataTmp = ucData-32;  if(ucIdxX > 122)  {  ucIdxX = 0;  ucIdxY++;  }    LED\_SetPos(ucIdxX, ucIdxY);    for(i = 0; i < 6; i++)  {  LED\_WrDat(F6x8[ucDataTmp][i]);  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_PrintChar  功能描述 : 将一个Char型数转换成3位数进行显示  输入参数 : UCHAR8 ucIdxX ucIdxX的范围为0～122  UCHAR8 ucIdxY ucIdxY的范围为0～7  CHAR8 cData cData为需要转化显示的数值 -128~127  输出参数 : none  返 回 值 : none  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void LED\_PrintChar(uchar ucIdxX, uchar ucIdxY, char cData)  {  uchar i, j, k, usData;  if(cData < 0)  {  LED\_P6x8Char(ucIdxX, ucIdxY, '-');  usData = (ushort)(-cData);  }  else  {  LED\_P6x8Char(ucIdxX, ucIdxY, '+');  usData = (ushort)cData;  }  i = usData / 100;  j = (usData % 100) / 10;  k = usData % 10;    LED\_P6x8Char(ucIdxX+6, ucIdxY, i+48);  LED\_P6x8Char(ucIdxX+12, ucIdxY, j+48);  LED\_P6x8Char(ucIdxX+18, ucIdxY, k+48);  return;  }  /\*-----------------------------------------------  TODO：向OLED写英文字符串  参数：ucIdxX 起始行位置  ucIdxY 其实列位置  ucDataStr 字符串头指针  编写：夏杰  日期：2017.4.22  ------------------------------------------------\*/  void led\_wirte\_str(uchar ucIdxX, uchar ucIdxY, uchar ucDataStr[])  {    LED\_P8x16Str(ucIdxX, ucIdxY, ucDataStr);  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  函 数 名 : LED\_PrintChar  功能描述 : 将一个Char型数转换成3位数进行显示  输入参数 : UCHAR8 ucIdxX ucIdxX的范围为0～122  UCHAR8 ucIdxY ucIdxY的范围为0～7  CHAR8 cData cData为需要转化显示的数值 -128~127  输出参数 : none  返 回 值 : none  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void led\_write\_num(uchar ucIdxX, uchar ucIdxY, char cData)  {  // 写入数字  LED\_PrintChar(ucIdxX, ucIdxY, cData);  }  /\*-----------------------------------------------  名称：电机驱动函数库  编写：夏杰  日期：2017.4.13  内容：  ------------------------------------------------\*/  #include "sc.h"  // 电机PWM波计数器  uint motor\_pwm\_counter = 0, motor\_pwm\_t = 100;  /\*-----------------------------------------------  TODO：电机驱动初始化  参数：null  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_init()  {  // 初始化电机停止  motor\_left\_stop();  motor\_right\_stop();  }  /\*-----------------------------------------------  TODO：电机驱动正反转、占空比控制  参数：  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_control(MOTOR\_DIR motor\_left\_dir, uint motor\_left\_duty, MOTOR\_DIR motor\_right\_dir, uint motor\_right\_duty)  {  // 电机PWM波周期  if(motor\_pwm\_counter >= motor\_pwm\_t)  {  motor\_pwm\_counter = 0;  }    // 左电机驱动  switch(motor\_left\_dir)  {  case motor\_left\_dir\_forward:  motor\_left\_forward(motor\_left\_duty);  break;  case motor\_left\_dir\_back:  motor\_left\_back(motor\_left\_duty);  break;  case motor\_left\_dir\_stop:  motor\_left\_stop();  break;  }  // 右电机驱动  switch(motor\_right\_dir)  {  // 右电机正转  case motor\_right\_dir\_forward:  motor\_right\_forward(motor\_right\_duty);  break;  case motor\_right\_dir\_back:  motor\_right\_back(motor\_right\_duty);  break;  case motor\_right\_dir\_stop:  motor\_right\_stop();  break;  }  }  /\*-----------------------------------------------  TODO：左电机正转  参数：@motor\_duty 电机PWM占空比  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_left\_forward(uint motor\_left\_duty)  {  // 电平驱动左电机正转  motor\_left\_in = MOTOR\_ON;  motor\_left\_out = MOTOR\_OFF;    // PWM波驱动  if(motor\_pwm\_counter <= motor\_left\_duty)  {  motor\_left\_in = MOTOR\_ON;  }  else  {  motor\_left\_in = MOTOR\_OFF;  }  motor\_left\_out = MOTOR\_OFF;  }  /\*-----------------------------------------------  TODO：左电机反转  参数：@motor\_duty 电机PWM占空比  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_left\_back(uint motor\_left\_duty)  {  // 电平驱动  motor\_left\_in = MOTOR\_OFF;  motor\_left\_out = MOTOR\_ON;  // PWM波驱动  if(motor\_pwm\_counter <= motor\_left\_duty)  {  motor\_left\_out = MOTOR\_ON;  }  else  {  motor\_left\_out = MOTOR\_OFF;  }  motor\_left\_in = MOTOR\_OFF;  }  /\*-----------------------------------------------  TODO：左电机停止  参数：null  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_left\_stop()  {  // 电平驱动  motor\_left\_in = MOTOR\_OFF;  motor\_left\_out = MOTOR\_OFF;  }  /\*-----------------------------------------------  TODO：右电机正转  参数：@motor\_duty 电机PWM占空比  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_right\_forward(uint motor\_right\_duty)  {  // 电平驱动  motor\_right\_in = MOTOR\_ON;  motor\_right\_out = MOTOR\_OFF;  // PWM波驱动  if(motor\_pwm\_counter <= motor\_right\_duty)  {  motor\_right\_in = MOTOR\_ON;  }  else  {  motor\_right\_in = MOTOR\_OFF;  }  motor\_right\_out = MOTOR\_OFF;  }  /\*-----------------------------------------------  TODO：右电机反转  参数：@motor\_duty 电机PWM占空比  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_right\_back(uint motor\_right\_duty)  {  // 电平驱动  motor\_right\_in = MOTOR\_OFF;  motor\_right\_out = MOTOR\_ON;  // PWM波驱动  if(motor\_pwm\_counter <= motor\_right\_duty)  {  motor\_right\_out = MOTOR\_ON;  }  else  {  motor\_right\_out = MOTOR\_OFF;  }  motor\_right\_in = MOTOR\_OFF;  }  /\*-----------------------------------------------  TODO：右电机停止  参数：null  编写：夏杰  日期：2017.4.13  ------------------------------------------------\*/  void motor\_right\_stop()  {  // 电平驱动  motor\_right\_in = MOTOR\_OFF;  motor\_right\_out = MOTOR\_OFF;  }  /\*-----------------------------------------------  名称：系统初始化函数库  编写：夏杰  日期：2017.4.11  内容：按时序调用各个模块的初始化程序  ------------------------------------------------\*/  #include "sc.h"  /\*-----------------------------------------------  TODO：初始化sc系统  参数：null  编写：夏杰  日期：2017.4.11  ------------------------------------------------\*/  void sc\_init()  {  // 舵机初始化  steer\_init();  // 电机驱动初始化  motor\_init();  // OLED初始化  LED\_init();  // 蓝牙通信初始化  blueteeth\_init();  // 定时器初始化  timer\_init();  }  /\*-----------------------------------------------  名称：舵机转向驱动函数库  编写：夏杰  日期：2017.4.12  内容：舵机控制相关函数  ------------------------------------------------\*/  #include "sc.h"  // 舵机PWM波周期、计数器  uchar steer\_pwm\_t = 200, steer\_pwm\_counter = 0;  // 舵机PWM占空比,范围:5/200~25/200  uchar steer\_left\_duty = 13, steer\_right\_duty = 17, steer\_middle\_duty = 15, steer\_duty = 0;  /\*-----------------------------------------------  TODO：舵机初始化  参数：null  编写：夏杰  日期：2017.4.12  ------------------------------------------------\*/  void steer\_init()  {  // 舵机计数器置位  steer\_pwm\_counter = 0;  // 舵机初始位置置中  steer\_duty = steer\_middle\_duty;  }  /\*-----------------------------------------------  TODO：舵机转向控制  参数：  编写：夏杰  日期：2017.4.12  ------------------------------------------------\*/  void steer\_control(STEER\_DIR steer\_dir)  {  // 舵机周期控制  if(steer\_pwm\_counter == steer\_pwm\_t)  {  steer\_pwm\_counter = 0;  }  // 舵机转向判断  switch(steer\_dir)  {  case steer\_dir\_left:  // 舵机左转  steer\_left();  break;  case steer\_dir\_middle:  // 舵机右转  steer\_right();  break;  case steer\_dir\_right:  // 舵机中置  steer\_middle();  break;  }  // 高电平占空  if(steer\_pwm\_counter <= steer\_duty) //PWM周期  steer\_out = 1;  else  steer\_out = 0;    }  /\*-----------------------------------------------  TODO：舵机左转向PWM波占空比  参数：null  编写：夏杰  日期：2017.4.12  ------------------------------------------------\*/  void steer\_left()  {  steer\_duty = steer\_left\_duty;  }  /\*-----------------------------------------------  TODO：舵机右转向PWM波占空比  参数：null  编写：夏杰  日期：2017.4.12  ------------------------------------------------\*/  void steer\_right()  {  steer\_duty = steer\_right\_duty;  }  /\*-----------------------------------------------  TODO：舵机正向PWM波占空比  参数：null  编写：夏杰  日期：2017.4.12  ------------------------------------------------\*/  void steer\_middle()  {  steer\_duty = steer\_middle\_duty;  }  /\*-----------------------------------------------  名称：定时器库函数  编写：夏杰  日期：2017.4.10  内容：定时器初始化、中断函数  ------------------------------------------------\*/  #include "sc.h"  // 100us, 1ms, 1s 定时计数器  uchar timer\_100us\_counter = 0, timer\_ms\_counter = 0, timer\_s\_counter = 0;  /\*-----------------------------------------------  TODO：定时器初始化  参数：  编写：夏杰  日期：2017.4.10  ------------------------------------------------\*/  void timer\_init()  {  // 定时器0,1初始化  TMOD |= 0x01;  // 定时器0定时初值设定  TH0 = (TIMER\_FULL\_T - TIMER0\_T) / 256; //定时100us,,即0.1ms  TL0 = (TIMER\_FULL\_T - TIMER0\_T) % 256;  // 定时器1定时初值设定  // TH1 = (TIMER\_FULL\_T - TIMER1\_T) / 256; //定时200us,,即0.2ms  // TL1 = (TIMER\_FULL\_T - TIMER1\_T) % 256;  // 使能总中断  EA = 1;  // 使能内部中断0,1  ET0 = 1;  // ET1 = 1;    // 启动定时器0,1(start timer0,timer1)  TR0 = 1;  // TR1 = 0;  }  /\*-----------------------------------------------  TODO：定时器0中断函数  参数：  编写：夏杰  日期：2017.4.10  ------------------------------------------------\*/  void timer0() interrupt 1 using 1  {  // pause timer0  TR0 = 0;  // 定时器重置  TH0 = (TIMER\_FULL\_T - TIMER0\_T) / 256; //定时100us,,即0.1ms  TL0 = (TIMER\_FULL\_T - TIMER0\_T) % 256;  // 定时器计数器累加, timer\_100us\_counter  timer\_100us\_counter++;  // 1ms定时  if(timer\_100us\_counter >= 10)  {  timer\_100us\_counter = 0;  timer\_ms\_counter++;  // 200ms 定时  if((timer\_ms\_counter+1) % 200 == 0)  {  timer\_ms\_counter = 0;  // 液晶刷新标识位  led\_refresh\_flag = 1;  timer\_s\_counter++;  if(timer\_s\_counter >= 250)  {  timer\_s\_counter = 0;  }  }  // 定时 1s  }  /\*---------------- 舵机控制 -----------------\*/  // 舵机计数器累加  steer\_pwm\_counter++;  // 舵机PWM波周期控制  if(steer\_pwm\_counter == steer\_pwm\_t)  {  steer\_pwm\_counter = 0;  }  // 高电平占空  if(steer\_pwm\_counter <= steer\_duty) //PWM周期  steer\_out = 1;  else  steer\_out = 0;    // start timer0  TR0 = 1;  }  //void timer1() interrupt 3 using 2  //{  // // pause timer1  //// TR1 = 0;  //  // // 定时器1定时值重置  // TH1 = (TIMER\_FULL\_T - TIMER1\_T) / 256; //定时100us,,即0.1ms  // TL1 = (TIMER\_FULL\_T - TIMER1\_T) % 256;  //  // // 电机PWM波计数器  // motor\_pwm\_counter++;  // // 电机控制  // motor\_control(motor\_left\_dir\_back, 10, motor\_right\_dir\_forward, 10);  //  // // start timer1  //// TR1 = 1;  //}  package com.xzit.sc;  import android.app.Activity;  import android.bluetooth.BluetoothAdapter;  import android.bluetooth.BluetoothDevice;  import android.content.Intent;  import android.content.pm.ActivityInfo;  import android.hardware.Sensor;  import android.hardware.SensorEvent;  import android.hardware.SensorEventListener;  import android.hardware.SensorManager;  import android.os.Bundle;  import android.os.Handler;  import android.os.Message;  import android.util.Log;  import android.view.Menu;  import android.view.MenuItem;  import android.view.MotionEvent;  import android.view.View;  import android.view.View.OnClickListener;  import android.view.Window;  import android.widget.Button;  import android.widget.TextView;  import android.widget.Toast;  public class SmartCarActivity extends Activity implements OnClickListener {    // Debugging  private static final String TAG = "SmartCarActivity";  private static final String TAG\_LINE = " + + + + ";  private static final boolean D = true;    // À¶ÑÀ·þÎñ´¦ÀíÆ÷·¢ËÍÏûÏ¢µÄÖÖÀà (Message types sent from BTService Handler)  public static final int MESSAGE\_STATE\_CHANGE = 1;  public static final int MESSAGE\_READ = 2;  public static final int MESSAGE\_WRITE = 3;  public static final int MESSAGE\_DEVICE\_NAME = 4;  public static final int MESSAGE\_TOAST = 5;    public static final String DEVICE\_NAME = "device\_name";  public static final String TOAST = "toast";  public static final String mscan = null;    // ÒâÍ¼ÇëÇó´úÂë(Intent request codes)  private static final int REQUEST\_CONNECT\_DEVICE\_SECURE = 1;  private static final int REQUEST\_ENABLE\_BT = 2;  private static final int CLOSE\_CONNECT = 3;    // °´Å¥±êÌâ  private TextView deviceScanText;  private TextView deviceSwitchText;    // ´«¸ÐÆ÷¶ÔÏó  private SensorManager sensorManager;  private Sensor sensor;  private SensorEventListener sensorEventListener;    // À¶ÑÀÊÊÅäÆ÷  private BluetoothAdapter bluetoothAdapter;  // À¶ÑÀ·þÎñ¶ÔÏó  private BluetoothService bluetoothService;  // ÒÑÁ¬½ÓÀ¶ÑÀÉè±¸µÄÃû³Æ  private String deviceConnectedName;    // °´Å¥Ò£¿ØÃüÁîÐ­Òé  String FORWARD = "1";  String BACK = "2";  String STOP = "0";  String LEFT = "3";  String RIGHT = "4";    // ÖØÁ¦¸ÐÓ¦Ò£¿ØÃüÁîÐ­Òé  TextView xViewA = null;  TextView yViewA = null;  TextView zViewA = null;  private float X = 0;  private float Y = 0;  private float Z = 0;        /\*\*  \* @Title: onCreate  \* @Description: TODO Èí¼þÆô¶¯Ê±³õÊ¼»¯  \* @param @param savedInstanceState  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-26  \*/  @Override  public void onCreate(Bundle savedInstanceState) {  // Æô¶¯ÈÕÖ¾Êä³ö  super.onCreate(savedInstanceState);  if(D) Log.e(TAG, TAG\_LINE + "ON CREATE" + TAG\_LINE);    // ÉèÖÃ²¼¾Ö  requestWindowFeature(Window.FEATURE\_NO\_TITLE);  setContentView(R.layout.main);    // °´Å¥²¼¾Ö  findViewById(R.id.dir1).setVisibility(View.INVISIBLE);  findViewById(R.id.dir3).setVisibility(View.INVISIBLE);  findViewById(R.id.dir7).setVisibility(View.INVISIBLE);  findViewById(R.id.dir9).setVisibility(View.INVISIBLE);    // ºáÆÁÏÔÊ¾  setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_LANDSCAPE);      // »ñÈ¡°´Å¥±êÌâ¶ÔÏó  // À¶ÑÀÉ¨Ãè°´Å¥  deviceScanText = (TextView) findViewById(R.id.device\_scan);  // À¶ÑÀ¿ª¹Ø°´Å¥(Î´´ò¿ªÀ¶ÑÀÊ±ÏÔÊ¾´ò¿ª£¬´ò¿ªÀ¶ÑÀºóÏÔÊ¾¹Ø±Õ)  deviceSwitchText = (TextView) findViewById(R.id.device\_switch);    // Îª°´Å¥ÉèÖÃ µã»÷¼àÌýÆ÷  // É¨Ãè°´Å¥  View deviceScanButton = this.findViewById(R.id.device\_scan);  deviceScanButton.setOnClickListener(this);  // ¶Ï¿ªÁ¬½Ó°´Å¥  View deviceSwitchButton = this.findViewById(R.id.device\_switch);  deviceSwitchButton.setOnClickListener(this);  // ¿ªÆôÖØÁ¦¸ÐÓ¦°´Å¥  View gravityOpenButton = this.findViewById(R.id.gravity\_open);  gravityOpenButton.setOnClickListener(this);  // ¹Ø±ÕÖØÁ¦¸ÐÓ¦°´Å¥  View gravityCloseButton = this.findViewById(R.id.gravity\_close);  gravityCloseButton.setOnClickListener(this);  // ÍË³öÈí¼þ  View exitButton = this.findViewById(R.id.exit\_button);  exitButton.setOnClickListener(this);      // »ñÈ¡´«¸ÐÆ÷¹ÜÀí¶ÔÏó(SensorManager)  sensorManager = (SensorManager) getSystemService(SENSOR\_SERVICE);  // »ñÈ¡¼ÓËÙ¶È´«¸ÐÆ÷¶ÔÏó  sensor = sensorManager.getDefaultSensor(Sensor.TYPE\_ACCELEROMETER);      // »ñÈ¡À¶ÑÀÊÊÅäÆ÷¶ÔÏó  bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();  // Éè±¸²»Ö§³ÖÀ¶ÑÀ(Device do not support bluetooth)  if(bluetoothAdapter == null){  Toast.makeText(this, "Éè±¸²»Ö§³ÖÀ¶ÑÀ", Toast.LENGTH\_LONG).show();  finish();  return;  }  //À¶ÑÀ¿ª¹Ø  deviceSwitchText.setOnClickListener(new OnClickListener() {  // µã»÷ÊÂ¼þ  public void onClick(View v) {  // À¶ÑÀÉè±¸´¦ÓÚ´ò¿ª×´Ì¬Ê±  if(bluetoothAdapter.isEnabled()){  // ¹Ø±ÕÀ¶ÑÀÉè±¸  bluetoothAdapter.disable();  // ÐÞ¸Ä°´Å¥±êÌâ  deviceSwitchText.setText("´ò¿ª");  } else {  // ´ò¿ªÀ¶ÑÀÉè±¸  bluetoothAdapter.enable();  // ÐÞ¸Ä°´Å¥±êÌâ  deviceSwitchText.setText("¹Ø±Õ");  }  }  });    }    @Override  protected void onStart() {  super.onStart();    // Èí¼þ¿ªÊ¼ÔËÐÐÈÕÖ¾Êä³ö  if(D) Log.e(TAG, TAG\_LINE + "on start" + TAG\_LINE);    // ÇëÇóÆôÓÃÀ¶ÑÀ  if(!bluetoothAdapter.isEnabled()){  // À¶ÑÀÊÊÅäÆ÷¿ªÆôÀ¶ÑÀ ÒâÍ¼  Intent enableIntent = new Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE);  startActivityForResult(enableIntent, REQUEST\_ENABLE\_BT);  } else {  deviceSwitchText.setText("¹Ø±Õ");  // ¿ØÖÆÐ¡³µ  if (bluetoothService == null) {  directionControl();  }  }  }      /\*\*  \* @Title: onClick  \* @Description: TODO Ö÷½çÃæ°´Å¥µã»÷ÊÂ¼þ  \* @param @param v  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  @Override  public void onClick(View v) {    // ÅÐ¶Ï°´Å¥µã»÷ÊÂ¼þ  switch (v.getId()) {    // É¨ÃèÉè±¸  case R.id.device\_scan:  if (bluetoothService != null && bluetoothService.getState() != BluetoothService.STATE\_CONNECTED){  Intent serverIntent = new Intent(this, Scan.class);  startActivityForResult(serverIntent, REQUEST\_CONNECT\_DEVICE\_SECURE);  } else {  deviceScanText.setText("Á¬½Ó");  bluetoothService.close();  }  break;    // ¿ªÆôÖØÁ¦¸ÐÓ¦  case R.id.gravity\_open:  Toast.makeText(this, "¿ªÆôÖØÁ¦¸ÐÓ¦", Toast.LENGTH\_SHORT).show();  Gravity();  sensorManager.registerListener(sensorEventListener, sensor, SensorManager.SENSOR\_DELAY\_NORMAL);  break;    // ¹Ø±ÕÖØÁ¦¸ÐÓ¦  case R.id.gravity\_close:  Toast.makeText(this, "¹Ø±ÕÖØÁ¦¸ÐÓ¦", Toast.LENGTH\_SHORT).show();  sensorManager.unregisterListener(sensorEventListener);  break;    // ÍË³öÈí¼þ  case R.id.exit\_button:  bluetoothService.close();  finish();  break;  }    }    /\*\*  \* @Title: sendCommand  \* @Description: TODO Í¨¹ýÀ¶ÑÀ·¢ËÍÃüÁî  \* @param @param command ´ý·¢ËÍµÄÃüÁî  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  private void sendCommand(String command) {  // À¶ÑÀ´¦ÓÚÁ¬½Ó×´Ì¬Ôò·¢ËÍÊý¾Ý  if (bluetoothService.getState() != BluetoothService.STATE\_CONNECTED) {  return ;  }  // Ð£ÑéÃüÁî  if(command != null && command.length() > 0){  // »ñÈ¡ÃüÁî×Ö·ûÁ÷  byte[] commandBytes = command.getBytes();  // ·¢ËÍÊý¾Ý  bluetoothService.write(commandBytes);  }  }    /\*\*  \* @Title: addCommandToButton  \* @Description: TODO ¸ø°´Å¥Ìí¼Ó´¥ÃþÊÂ¼þ  \* @param @param buttonId °´Å¥ID  \* @param @param command ´ýÌí¼ÓÃüÁî  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  private void addCommandToButton(int buttonId, final String command) {  // Í¨¹ýId»ñÈ¡°´Å¥  Button button = (Button)findViewById(buttonId);  // ¸ø°´Å¥Ìí¼Ó´¥ÃþÊÂ¼þ  button.setOnTouchListener(new Button.OnTouchListener()  {  @Override  public boolean onTouch(View v, MotionEvent event) {  int action = event.getAction();  switch(action)  {  case MotionEvent.ACTION\_DOWN:  sendCommand(command);  break;  case MotionEvent.ACTION\_UP:  sendCommand(STOP);  break;  }  return false;  }  });  }    /\*\*  \* @Title: directionControl  \* @Description: TODO ÖÇÄÜ³µ·½Ïò¿ØÖÆ  \* @param  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  public void directionControl() {  // Debugging  if(D) Log.e(TAG, " + + + direction control + + + ");    // Í£³µ°´Å¥Ìí¼Óµã»÷ÊÂ¼þ  Button dirStopBtn = (Button) findViewById(R.id.dir5);  dirStopBtn.setOnClickListener(new Button.OnClickListener(){  @Override  public void onClick(View v) {  sendCommand(STOP);  }  });    // ¸ø·½Ïò°´Å¥Ìí¼ÓÊÂ¼þ  addCommandToButton(R.id.dir2, FORWARD);  addCommandToButton(R.id.dir4, LEFT);  addCommandToButton(R.id.dir6, RIGHT);  addCommandToButton(R.id.dir8, BACK);    // ³õÊ¼»¯À¶ÑÀ·þÎñÀà£¬½øÐÐÀ¶ÑÀÁ¬½Ó(Initialize the BTService to perform bluetooth connections)  bluetoothService = new BluetoothService(this, bluetoothHandler);  }    /\*\*  \* @Title: Gravity  \* @Description: TODO ¶ÁÈ¡ÖØÁ¦´«¸ÐÆ÷µÄÖµ£¬²¢¸ù¾Ý´«¸ÐÆ÷µÄÖµ¿ØÖÆ³µµÄ·½Ïò  \* @param  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-28  \*/  private void Gravity(){  setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_LANDSCAPE);  sensorEventListener = new SensorEventListener() {  // ¶ÁÈ¡´«¸ÐÆ÷µÄÖµ  @Override  public void onSensorChanged(SensorEvent event) {  // È¡Öµ  X = event.values[SensorManager.DATA\_X];  Y = event.values[SensorManager.DATA\_Y];  Z = event.values[SensorManager.DATA\_Z];  // ÖØÁ¦¸ÐÓ¦¿ØÖÆ  gravityControl();  }    @Override  public void onAccuracyChanged(Sensor sensor, int accuracy) {  // TODO Auto-generated method stub    }  };  }    /\*\*  \* @Title: gravityControl  \* @Description: TODO ¸ù¾ÝÍÓÂÝÒÇµÄÖµ¿ØÖÆÖÇÄÜ³µ·½Ïò  \* @param  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-28  \*/  public void gravityControl(){  // ÊÖ»úÕýÃæ³¯ÉÏ  if(Z >= 0) {  // y·½ÏòÇã½Ç  if(Y <= -2.0 && Y > -9.0) {  sendCommand(LEFT);  } else if(Y > 2.0 && Y < 9.0) {  sendCommand(RIGHT);  } else if(X < -2.0 && X > -9.0) {  sendCommand(FORWARD);  } else if(X > 2.0 && X < 9.0) {  sendCommand(BACK);  } else {  sendCommand(STOP);  }  }  }      private final Handler bluetoothHandler = new Handler(){  @Override  public void handleMessage (Message msg) {  switch (msg.what) {  case MESSAGE\_STATE\_CHANGE:  if(D) Log.i(TAG, "MESSAGE\_STATE\_CHANGE: " + msg.arg1);  switch (msg.arg1) {  case BluetoothService.STATE\_CONNECTED:  Toast.makeText(getApplicationContext(), R.string.title\_connecting, Toast.LENGTH\_SHORT).show();  deviceScanText.setText("¶Ï¿ª");  break;  case BluetoothService.STATE\_CONNECTING:  break;  case BluetoothService.STATE\_LISTEN:  case BluetoothService.STATE\_NONE:  deviceScanText.setText("Á¬½Ó");  break;  case BluetoothService.STATE\_CLOSE:  Toast.makeText(getApplicationContext(), R.string.title\_disconnected, Toast.LENGTH\_SHORT).show();  deviceScanText.setText("Á¬½Ó");  break;  }  break;    case MESSAGE\_DEVICE\_NAME:  // save the connected device's name  deviceConnectedName = msg.getData().getString(DEVICE\_NAME);  Toast.makeText(getApplicationContext(), "Connected to " + deviceConnectedName, Toast.LENGTH\_SHORT).show();  break;  case MESSAGE\_TOAST:  Toast.makeText(getApplicationContext(), msg.getData().getString(TOAST), Toast.LENGTH\_SHORT).show();  break;    }  }  };    /\*\*  \* @Title: onResume  \* @Description: TODO »Øµ½Ö÷½çÃæÊ±¼àÌý´«¸ÐÆ÷  \* @param  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  @Override  protected void onResume() {  super.onResume();  sensorManager.registerListener(sensorEventListener, sensor, SensorManager.SENSOR\_DELAY\_NORMAL);  }    /\*\*  \* @Title: onPause  \* @Description: TODO Àë¿ªÖ÷½çÃæÊ±È¡Ïû¼àÌý´«¸ÐÆ÷  \* @param  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  @Override  public synchronized void onPause() {  super.onPause();  if(D) Log.e(TAG, "- ON PAUSE -");  sensorManager.unregisterListener(sensorEventListener);  }    @Override  public void onStop() {  super.onStop();  if(D) Log.e(TAG, "-- ON STOP --");  }  @Override  public void onDestroy() {  super.onDestroy();  if(D) Log.e(TAG, "--- ON DESTROY ---");  }    @Override  public boolean onCreateOptionsMenu(Menu menu)  {  // Inflate the menu; this adds items to the action bar if it is present.  getMenuInflater().inflate(R.menu.menu, menu);  return true;  }    @Override  public boolean onOptionsItemSelected(MenuItem item) {  // Intent serverIntent = null;  switch (item.getItemId()) {  case R.id.discoverable:  // Ensure this device is discoverable by others  ensureDiscoverable();  return true;  }  return false;    }    /\*\*  \* @Title: ensureDiscoverable  \* @Description: TODO È·±£Éè±¸ÄÜ±»ËÑË÷  \* @param  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  private void ensureDiscoverable()  {  // Debugging  if(D) Log.d(TAG, "ensure discoverable");  //  if (bluetoothAdapter.getScanMode() != BluetoothAdapter.SCAN\_MODE\_CONNECTABLE\_DISCOVERABLE)  {  Intent discoverableIntent = new Intent(BluetoothAdapter.ACTION\_REQUEST\_DISCOVERABLE);  discoverableIntent.putExtra(BluetoothAdapter.EXTRA\_DISCOVERABLE\_DURATION, 300);  startActivity(discoverableIntent);  }  }    /\*\*  \* @Title: onActivityResult  \* @Description: TODO  \* @param @param requestCode  \* @param @param resultCode  \* @param @param data  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  public void onActivityResult(int requestCode, int resultCode, Intent data) {  // Debugging  if(D) Log.d(TAG, "onActivityResult " + resultCode);  //  switch (requestCode) {  case REQUEST\_CONNECT\_DEVICE\_SECURE:  // When DeviceListActivity returns with a device to connect  if (resultCode == Activity.RESULT\_OK) {  connectDevice(data, true);  }  break;  case CLOSE\_CONNECT:  connectDevice(data, false);    case REQUEST\_ENABLE\_BT:  // When the request to enable Bluetooth returns  if (resultCode == Activity.RESULT\_OK) {  directionControl();  }  else {  // User did not enable Bluetooth or an error occurred  Log.d(TAG, "Bluetooth not enabled");  Toast.makeText(this, R.string.bt\_not\_enabled, Toast.LENGTH\_SHORT).show();  finish();  }  }  }    /\*\*  \* @Title: connectDevice  \* @Description: TODO Á¬½ÓÉè±¸  \* @param @param data  \* @param @param secure  \* @return void  \* @throws  \* @author: ÏÄ½Ü 1272570701@qq.com  \* @date: 2017-4-27  \*/  private void connectDevice(Intent data, boolean secure) {  // Get the device MAC address »ñÈ¡Éè±¸µÄMACµØÖ·  String address = data.getExtras().getString(Scan.EXTRA\_DEVICE\_ADDRESS);  // Get the BluetoothDevice object BluetoothDevice¶ÔÏó  BluetoothDevice device = bluetoothAdapter.getRemoteDevice(address);    // Attempt to connect to the device ³¢ÊÔÁ¬½Óµ½Éè±¸  bluetoothService.connect(device, secure);    }  }  package com.xzit.sc;  import java.io.IOException;  import java.io.InputStream;  import java.io.OutputStream;  import java.util.UUID;  import android.bluetooth.BluetoothAdapter;  import android.bluetooth.BluetoothDevice;  import android.bluetooth.BluetoothServerSocket;  import android.bluetooth.BluetoothSocket;  import android.content.Context;  import android.os.Bundle;  import android.os.Handler;  import android.os.Message;  import android.util.Log;  /\*\*  \* @ClassName: BluetoothService  \* @Description: TODO À¶ÑÀ·þÎñ  \* @author ÏÄ½Ü 1272570701@qq.com  \* @date 2017-4-27 ÏÂÎç07:35:29  \*/  public class BluetoothService {    // Debugging µ÷ÊÔ  private static final String TAG = "BluetoothService";  private static final boolean D = true;    // private static final String NAME\_SECURE = "BTSecure";  // UUID for this application UUIDÎªÕâ¸öÓ¦ÓÃ³ÌÐò  private static final UUID MY\_UUID =  UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");  // Member fields ³ÉÔ±×Ö¶Î  private final BluetoothAdapter mBluetoothAdapter;  private final Handler mHandler;  private ConnectThread mConnectThread;  private ConnectedThread mConnectedThread;  private AcceptThread mSecureAcceptThread;  private int mState;    // Constants that indicate the current connection state  // ³£Á¿,±íÃ÷µ±Ç°µÄÁ¬½Ó×´Ì¬  public static final int STATE\_NONE = 0; // we're doing nothing  public static final int STATE\_LISTEN = 1; // now listening for incoming connections ÏÖÔÚ¼àÌý´«ÈëµÄÁ¬½Ó  public static final int STATE\_CONNECTING = 2; // now initiating an outgoing connection ÏÖÔÚÆô¶¯Ò»¸öÍâÏòÁ¬½Ó  public static final int STATE\_CONNECTED = 3; // now connected to a remote device ÏÖÔÚÁ¬½Óµ½Ò»¸öÔ¶³ÌÉè±¸  public static final int STATE\_CLOSE = 4; // ¶Ï¿ªÁ¬½Ó  /\*\*  \* Constructor. Prepares a new BluetoothChat session.  \* @param context The UI Activity Context  \* @param handler A Handler to send messages back to the UI Activity  \*/  public BluetoothService(Context context, Handler handler) {  mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();  mState = STATE\_NONE;  mHandler = handler;  }      /\*\*  \* Set the current state of the chat connection  \* @param state An integer defining the current connection state  \*/  private synchronized void setState(int state) {  if (D) Log.d(TAG, "setState() " + mState + " -> " + state);  mState = state;  // Give the new state to the Handler so the UI Activity can update  // ¸øÐÂ×´Ì¬´¦Àí³ÌÐòÒò´ËUI»î¶¯¿ÉÒÔ¸üÐÂ  mHandler.obtainMessage(SmartCarActivity.MESSAGE\_STATE\_CHANGE, state, -1).sendToTarget();  }  /\*\*  \* Return the current connection state. \*/  public synchronized int getState() {  return mState;  }  /\*\*  \* Start the contorl service. Specifically start AcceptThread to begin a  \* session in listening (server) mode. Called by the Activity onResume() \*/    public synchronized void start() {  if (D) Log.d(TAG, "start");  // Cancel any thread attempting to make a connection  if (mConnectThread != null)  {  mConnectThread.cancel();  mConnectThread = null;    }  // Cancel any thread currently running a connection  if (mConnectedThread != null)  {  mConnectedThread.cancel();  mConnectedThread = null;  }  setState(STATE\_LISTEN);    // Start the thread to listen on a BluetoothServerSocket  if (mSecureAcceptThread == null) {  mSecureAcceptThread = new AcceptThread(true);  mSecureAcceptThread.start();  }    }    public synchronized void connect(BluetoothDevice device, boolean secure) {  if (D) Log.d(TAG, "connect to: " + device);  // Cancel any thread attempting to make a connection  if (mState == STATE\_CONNECTING) {  if (mConnectThread != null) {mConnectThread.cancel(); mConnectThread = null;}  }  // Cancel any thread currently running a connection  if (mConnectedThread != null) {mConnectedThread.cancel(); mConnectedThread = null;}  // Start the thread to connect with the given device  mConnectThread = new ConnectThread(device, secure);  mConnectThread.start();  setState(STATE\_CONNECTING);  }    /\*\*  \* Start the ConnectedThread to begin managing a Bluetooth connection  \* @param socket The BluetoothSocket on which the connection was made  \* @param device The BluetoothDevice that has been connected  \*/  public synchronized void connected(BluetoothSocket socket, BluetoothDevice  device, final String socketType) {  if (D) Log.d(TAG, "connected, Socket Type:" + socketType);  // Cancel the thread that completed the connection  if (mConnectThread != null) {mConnectThread.cancel(); mConnectThread = null;}  // Cancel any thread currently running a connection  if (mConnectedThread != null) {mConnectedThread.cancel(); mConnectedThread = null;}  // Cancel the accept thread because we only want to connect to one device  if (mSecureAcceptThread != null) {  mSecureAcceptThread.cancel();  mSecureAcceptThread = null;  }    // Start the thread to manage the connection and perform transmissions  mConnectedThread = new ConnectedThread(socket, socketType);  mConnectedThread.start();  // Send the name of the connected device back to the UI Activity  Message msg = mHandler.obtainMessage(SmartCarActivity.MESSAGE\_DEVICE\_NAME);  Bundle bundle = new Bundle();  bundle.putString(SmartCarActivity.DEVICE\_NAME, device.getName());  msg.setData(bundle);  mHandler.sendMessage(msg);  setState(STATE\_CONNECTED);  }    public synchronized void close() {  if (D) Log.d(TAG, "close: " );  if (mConnectThread != null) {  mConnectThread.cancel();  mConnectThread = null;  }  if (mConnectedThread != null) {  mConnectedThread.cancel();  mConnectedThread = null;  }    if (mSecureAcceptThread != null) {  mSecureAcceptThread.cancel();  mSecureAcceptThread = null;  }    setState(STATE\_CLOSE);  }  /\*\*  \* Stop all threads  \*/  public synchronized void stop() {  if (D) Log.d(TAG, "stop");  if (mConnectThread != null) {  mConnectThread.cancel();  mConnectThread = null;  }  if (mConnectedThread != null) {  mConnectedThread.cancel();  mConnectedThread = null;  }    if (mSecureAcceptThread != null) {  mSecureAcceptThread.cancel();  mSecureAcceptThread = null;  }  setState(STATE\_NONE);  }  /\*\*  \* Write to the ConnectedThread in an unsynchronized manner  \* @param out The bytes to write  \* @see ConnectedThread#write(byte[])  \*/  public void write(byte[] out) {  // Create temporary object  ConnectedThread r;  // Synchronize a copy of the ConnectedThread  synchronized (this) {  if (mState != STATE\_CONNECTED) return;  r = mConnectedThread;  }  // Perform the write unsynchronized  r.write(out);  }    /\*\*  \* Indicate that the connection attempt failed and notify the UI Activity.  \*/  private void connectionFailed() {  // Send a failure message back to the Activity  Message msg = mHandler.obtainMessage(SmartCarActivity.MESSAGE\_TOAST);  Bundle bundle = new Bundle();  bundle.putString(SmartCarActivity.TOAST, "²»ÄÜÁ¬½ÓÉè±¸");  msg.setData(bundle);  mHandler.sendMessage(msg);  // Start the service over to restart listening mode  BluetoothService.this.start();  }  /\*\*  \* Indicate that the connection was lost and notify the UI Activity.  \*/  private void connectionLost() {  // Send a failure message back to the Activity  Message msg = mHandler.obtainMessage(SmartCarActivity.MESSAGE\_TOAST);  Bundle bundle = new Bundle();  bundle.putString(SmartCarActivity.TOAST, "Éè±¸Á¬½Ó¶ªÊ§");  msg.setData(bundle);  mHandler.sendMessage(msg);  // Start the service over to restart listening mode  BluetoothService.this.start();  }    /\*\*  \* This thread runs while listening for incoming connections. It behaves  \* like a server-side client. It runs until a connection is accepted  \* (or until cancelled).  \*/  private class AcceptThread extends Thread {  // The local server socket  private final BluetoothServerSocket mmServerSocket;  private String mSocketType;  public AcceptThread(boolean secure) {  BluetoothServerSocket tmp = null;  mSocketType = secure ? "Secure":"Insecure";  // Create a new listening server socket  try {  if (secure) {  tmp = mBluetoothAdapter.listenUsingRfcommWithServiceRecord(  mSocketType, MY\_UUID);  }  } catch (IOException e) {  Log.e(TAG, "Socket Type: " + mSocketType + "listen() failed", e);  }  mmServerSocket = tmp;  }  public void run() {  if (D) Log.d(TAG, "Socket Type: " + mSocketType +  "BEGIN mAcceptThread" + this);  setName("AcceptThread" + mSocketType);  BluetoothSocket socket = null;  // Listen to the server socket if we're not connected  while (mState != STATE\_CONNECTED) {  try {  // This is a blocking call and will only return on a  // successful connection or an exception  socket = mmServerSocket.accept();  } catch (IOException e) {  Log.e(TAG, "Socket Type: " + mSocketType + "accept() failed", e);  break;  }  // If a connection was accepted  if (socket != null) {  synchronized (BluetoothService.this) {  switch (mState) {  case STATE\_LISTEN:  case STATE\_CONNECTING:  // Situation normal. Start the connected thread.  connected(socket, socket.getRemoteDevice(),  mSocketType);  break;  case STATE\_NONE:  case STATE\_CLOSE:  case STATE\_CONNECTED:  // Either not ready or already connected. Terminate new socket.  try {  socket.close();  } catch (IOException e) {  Log.e(TAG, "Could not close unwanted socket", e);  }  break;  }  }  }  }  if (D) Log.i(TAG, "END mAcceptThread, socket Type: " + mSocketType);  }  public void cancel() {  if (D) Log.d(TAG, "Socket Type" + mSocketType + "cancel " + this);  try {  mmServerSocket.close();  } catch (IOException e) {  Log.e(TAG, "Socket Type" + mSocketType + "close() of server failed", e);  }  }  }    /\*\*  \* This thread runs while attempting to make an outgoing connection  \* with a device. It runs straight through; the connection either  \* succeeds or fails.  \*/  private class ConnectThread extends Thread {  private final BluetoothSocket mmSocket;  private final BluetoothDevice mmDevice;  private String mSocketType;  public ConnectThread(BluetoothDevice device, boolean secure) {  mmDevice = device;  BluetoothSocket tmp = null;  mSocketType = secure ? "Secure" : "Insecure";  // Get a BluetoothSocket for a connection with the given BluetoothDevice  try  {  if (secure)  {  tmp = device.createRfcommSocketToServiceRecord(  MY\_UUID);  }  }    catch (IOException e)  {  Log.e(TAG, "Socket Type: " + mSocketType + "create() failed", e);  }  mmSocket = tmp;  }  public void run() {  Log.i(TAG, "BEGIN mConnectThread SocketType:" + mSocketType);  setName("ConnectThread" + mSocketType);  // Always cancel discovery because it will slow down a connection  mBluetoothAdapter.cancelDiscovery();  // Make a connection to the BluetoothSocket  try  {  // This is a blocking call and will only return on a successful connection or an exception  // ÕâÊÇÒ»¸ö×èÈûµ÷ÓÃ,Ö»»á·µ»Ø³É¹¦Á¬½Ó»òÒ»¸öÀýÍâ  mmSocket.connect();  }  catch (IOException e)  {  // Close the socket  try  {  mmSocket.close();  }  catch (IOException e2)  {  Log.e(TAG, "unable to close() " + mSocketType +  " socket during connection failure", e2);  }  connectionFailed();  return;  }  // Reset the ConnectThread because we're done  synchronized (BluetoothService.this)  {  mConnectThread = null;  }  // Start the connected thread  connected(mmSocket, mmDevice, mSocketType);  }  public void cancel() {  try {  mmSocket.close();  } catch (IOException e) {  Log.e(TAG, "close() of connect " + mSocketType + " socket failed", e);  }  }    }  /\*\*  \* This thread runs during a connection with a remote device.  \* It handles all incoming and outgoing transmissions.  \*/  private class ConnectedThread extends Thread {  private final BluetoothSocket mmSocket;  private final InputStream mmInStream;  private final OutputStream mmOutStream;  public ConnectedThread(BluetoothSocket socket, String socketType) {  Log.d(TAG, "create ConnectedThread: " + socketType);  mmSocket = socket;  InputStream tmpIn = null;  OutputStream tmpOut = null;  // Get the BluetoothSocket input and output streams  try {  tmpIn = socket.getInputStream();  tmpOut = socket.getOutputStream();  } catch (IOException e) {  Log.e(TAG, "temp sockets not created", e);  }  mmInStream = tmpIn;  mmOutStream = tmpOut;  }  public void run() {  Log.i(TAG, "BEGIN mConnectedThread");  byte[] buffer = new byte[1024];  int bytes;  // Keep listening to the InputStream while connected  while (true) {  try {  // Read from the InputStream  bytes = mmInStream.read(buffer);  // Send the obtained bytes to the UI Activity  mHandler.obtainMessage(SmartCarActivity.MESSAGE\_READ, bytes, -1, buffer)  .sendToTarget();  } catch (IOException e) {  Log.e(TAG, "disconnected", e);  connectionLost();  // Start the service over to restart listening mode  BluetoothService.this.start();  break;  }  }  }  /\*\*  \* Write to the connected OutStream.  \* @param buffer The bytes to write  \*/  public void write(byte[] buffer) {  try {  mmOutStream.write(buffer);  // Share the sent message back to the UI Activity  // mHandler.obtainMessage(SmartCarActivity.MESSAGE\_WRITE, -1, -1, buffer)  // .sendToTarget();  } catch (IOException e) {  Log.e(TAG, "Exception during write", e);  }  }  public void cancel() {  try {  mmSocket.close();  } catch (IOException e) {  Log.e(TAG, "close() of connect socket failed", e);  }  }  }  }  package com.xzit.sc;  import java.util.Set;  import android.app.Activity;  import android.bluetooth.BluetoothAdapter;  import android.bluetooth.BluetoothDevice;  import android.content.BroadcastReceiver;  import android.content.Context;  import android.content.Intent;  import android.content.IntentFilter;  import android.os.Bundle;  import android.util.Log;  import android.view.View;  import android.view.Window;  import android.view.View.OnClickListener;  import android.widget.AdapterView;  import android.widget.ArrayAdapter;  import android.widget.Button;  import android.widget.ListView;  import android.widget.TextView;  import android.widget.AdapterView.OnItemClickListener;  public class Scan extends Activity {  // Debugging µ÷ÊÔ  private static final String TAG = "Scan";  private static final boolean D = true;    // Return Intent extra  public static String EXTRA\_DEVICE\_ADDRESS = "device\_address";  // Member fields  private BluetoothAdapter mBtAdapter;  private ArrayAdapter<String> mPairedDevicesArrayAdapter;  private ArrayAdapter<String> mNewDevicesArrayAdapter;    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  // Setup the window  requestWindowFeature(Window.FEATURE\_INDETERMINATE\_PROGRESS);  setContentView(R.layout.scan);  // Set result CANCELED in case the user backs out  setResult(Activity.RESULT\_CANCELED);    // Initialize the button to perform device discovery  Button scanButton = (Button) findViewById(R.id.button\_scan);  scanButton.setOnClickListener(new OnClickListener() {  public void onClick(View v) {  doDiscovery();  v.setVisibility(View.GONE);  }  });  // ·µ»Ø  Button cancelButton = (Button) findViewById(R.id.button\_cancel);  cancelButton.setOnClickListener(new OnClickListener() {  public void onClick(View v) {  finish();  return;  }  });  // Initialize array adapters. One for already paired devices and  // one for newly discovered devices  mPairedDevicesArrayAdapter = new ArrayAdapter<String>(this, R.layout.device\_name);  mNewDevicesArrayAdapter = new ArrayAdapter<String>(this, R.layout.device\_name);  // Find and set up the ListView for paired devices  ListView pairedListView = (ListView) findViewById(R.id.paired\_devices);  pairedListView.setAdapter(mPairedDevicesArrayAdapter);  pairedListView.setOnItemClickListener(mDeviceClickListener);  // Find and set up the ListView for newly discovered devices  ListView newDevicesListView = (ListView) findViewById(R.id.new\_devices);  newDevicesListView.setAdapter(mNewDevicesArrayAdapter);  newDevicesListView.setOnItemClickListener(mDeviceClickListener);  // Register for broadcasts when a device is discovered  IntentFilter filter = new IntentFilter(BluetoothDevice.ACTION\_FOUND);  this.registerReceiver(mReceiver, filter);  // Register for broadcasts when discovery has finished  filter = new IntentFilter(BluetoothAdapter.ACTION\_DISCOVERY\_FINISHED);  this.registerReceiver(mReceiver, filter);  // Get the local Bluetooth adapter  mBtAdapter = BluetoothAdapter.getDefaultAdapter();  // Get a set of currently paired devices  Set<BluetoothDevice> pairedDevices = mBtAdapter.getBondedDevices();  // If there are paired devices, add each one to the ArrayAdapter  if (pairedDevices.size() > 0) {  findViewById(R.id.title\_paired\_devices).setVisibility(View.VISIBLE);  for (BluetoothDevice device : pairedDevices) {  mPairedDevicesArrayAdapter.add(device.getName() + "\n" + device.getAddress());  }  } else {  String noDevices = getResources().getText(R.string.none\_paired).toString();  mPairedDevicesArrayAdapter.add(noDevices);  }  }  @Override  protected void onDestroy() {  super.onDestroy();  // Make sure we're not doing discovery anymore  if (mBtAdapter != null) {  mBtAdapter.cancelDiscovery();  }  // Unregister broadcast listeners  this.unregisterReceiver(mReceiver);  }  /\*\*  \* Start device discover with the BluetoothAdapter  \*/  private void doDiscovery() {  if (D) Log.d(TAG, "doDiscovery()");  // Indicate scanning in the title  setProgressBarIndeterminateVisibility(true);  setTitle(R.string.scanning);  // Turn on sub-title for new devices  findViewById(R.id.title\_new\_devices).setVisibility(View.VISIBLE);  // If we're already discovering, stop it  if (mBtAdapter.isDiscovering()) {  mBtAdapter.cancelDiscovery();  }  // Request discover from BluetoothAdapter  mBtAdapter.startDiscovery();  }  // The on-click listener for all devices in the ListViews  private OnItemClickListener mDeviceClickListener = new OnItemClickListener() {  public void onItemClick(AdapterView<?> av, View v, int arg2, long arg3) {  // Cancel discovery because it's costly and we're about to connect  mBtAdapter.cancelDiscovery();  // Get the device MAC address, which is the last 17 chars in the View  String info = ((TextView) v).getText().toString();  String address = info.substring(info.length() - 17);  // Create the result Intent and include the MAC address  Intent intent = new Intent();  intent.putExtra(EXTRA\_DEVICE\_ADDRESS, address);  // Set result and finish this Activity  setResult(Activity.RESULT\_OK, intent);  finish();  }  };  // The BroadcastReceiver that listens for discovered devices and  // changes the title when discovery is finished  private final BroadcastReceiver mReceiver = new BroadcastReceiver() {  @Override  public void onReceive(Context context, Intent intent) {  String action = intent.getAction();  // When discovery finds a device  if (BluetoothDevice.ACTION\_FOUND.equals(action)) {  // Get the BluetoothDevice object from the Intent  BluetoothDevice device = intent.getParcelableExtra(BluetoothDevice.EXTRA\_DEVICE);  // If it's already paired, skip it, because it's been listed already  if (device.getBondState() != BluetoothDevice.BOND\_BONDED) {  mNewDevicesArrayAdapter.add(device.getName() + "\n" + device.getAddress());  }  // When discovery is finished, change the Activity title  } else if (BluetoothAdapter.ACTION\_DISCOVERY\_FINISHED.equals(action)) {  setProgressBarIndeterminateVisibility(false);  setTitle(R.string.select\_device);  if (mNewDevicesArrayAdapter.getCount() == 0) {  String noDevices = getResources().getText(R.string.none\_found).toString();  mNewDevicesArrayAdapter.add(noDevices);  }  }  }  };  }  <?xml version="1.0" encoding="utf-8"?>  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:background="@drawable/sc\_background"  android:orientation="vertical" >    <Button  android:id="@+id/gravity\_open"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentTop="true"  android:layout\_toLeftOf="@+id/device\_scan"  android:text="@string/gravity\_open" />  <Button  android:id="@+id/device\_scan"  android:layout\_width="65dp"  android:layout\_height="wrap\_content"  android:layout\_alignParentRight="true"  android:layout\_alignParentTop="true"  android:text="@string/device\_scan" />  <Button  android:id="@+id/exit\_button"  android:layout\_width="65dp"  android:layout\_height="wrap\_content"  android:layout\_alignParentBottom="true"  android:layout\_toRightOf="@+id/gravity\_open"  android:text="@string/exit\_button" />  <Button  android:id="@+id/gravity\_close"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignLeft="@+id/gravity\_open"  android:layout\_alignParentBottom="true"  android:text="@string/gravity\_close" />  <Button  android:id="@+id/about\_button"  android:layout\_width="65dp"  android:layout\_height="wrap\_content"  android:layout\_alignParentRight="true"  android:layout\_below="@+id/dir2"  android:text="@string/about\_button" />  <TextView  android:id="@+id/textView1"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentTop="true"  android:layout\_centerHorizontal="true"  android:layout\_gravity="center"  android:text="@string/main\_title"  android:textSize="12sp" />  <Button  android:id="@+id/dir2"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_alignBottom="@+id/device\_scan"  android:layout\_alignParentTop="true"  android:layout\_toRightOf="@+id/dir1"  android:text="@string/dir2" />  <Button  android:id="@+id/dir1"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_alignBottom="@+id/dir2"  android:layout\_alignParentLeft="true"  android:text="@string/dir1" />  <Button  android:id="@+id/dir3"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_above="@+id/about\_button"  android:layout\_toRightOf="@+id/dir2"  android:text="@string/dir3" />  <Button  android:id="@+id/device\_switch"  android:layout\_width="65dp"  android:layout\_height="wrap\_content"  android:layout\_above="@+id/gravity\_close"  android:layout\_alignParentRight="true"  android:text="@string/device\_switch" />  <Button  android:id="@+id/dir5"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignLeft="@+id/dir2"  android:layout\_alignRight="@+id/dir2"  android:layout\_below="@+id/dir2"  android:text="@string/dir5" />  <Button  android:id="@+id/dir7"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignParentLeft="true"  android:layout\_alignRight="@+id/dir1"  android:layout\_below="@+id/dir5"  android:text="@string/dir7" />  <Button  android:id="@+id/dir4"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignBottom="@+id/dir5"  android:layout\_alignParentLeft="true"  android:layout\_alignRight="@+id/dir1"  android:text="@string/dir4" />  <Button  android:id="@+id/dir9"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_alignLeft="@+id/dir3"  android:layout\_alignRight="@+id/dir3"  android:layout\_alignTop="@+id/dir8"  android:text="@string/dir9" />  <Button  android:id="@+id/dir6"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_above="@+id/dir9"  android:layout\_alignLeft="@+id/dir9"  android:layout\_alignRight="@+id/dir3"  android:text="@string/dir6" />  <Button  android:id="@+id/dir8"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_alignBottom="@+id/dir7"  android:layout\_below="@+id/dir5"  android:layout\_toLeftOf="@+id/dir9"  android:text="@string/dir8" />    </RelativeLayout>  <?xml version="1.0" encoding="utf-8"?>  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical" >  <TextView android:id="@+id/title\_paired\_devices"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/title\_paired\_devices"  android:visibility="gone"  android:background="#666"  android:textColor="#fff"  android:paddingLeft="5dp"  />  <ListView android:id="@+id/paired\_devices"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:stackFromBottom="true"  android:transcriptMode="alwaysScroll"  android:layout\_weight="1" />    <TextView android:id="@+id/title\_new\_devices"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/title\_other\_devices"  android:visibility="gone"  android:background="#666"  android:textColor="#fff"  android:paddingLeft="5dp" />    <ListView android:id="@+id/new\_devices"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:stackFromBottom="true"  android:transcriptMode="alwaysScroll"  android:layout\_weight="2" />  <Button android:id="@+id/button\_scan"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/button\_scan" />    <Button android:id="@+id/button\_cancel"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/button\_cancel" />  </LinearLayout>  <?xml version="1.0" encoding="utf-8"?>  <resources>  <!-- app\_name -->  <string name="app\_name">智能车</string>    <!-- 退出系统-->  <string name="exit\_button">退出</string>  <string name="about\_button">关于</string>  <string name="main\_title">蓝牙遥控车客户端 @夏杰</string>    <!-- 蓝牙-->  <string name="device\_scan">扫描</string>  <string name="device\_switch">打开</string>    <!-- 重力感应-->  <string name="gravity\_open">重力感应开</string>  <string name="gravity\_close">重力感应关</string>    <!-- 方向按钮 -->  <string name="dir1">左前</string>  <string name="dir2">前进</string>  <string name="dir3">右前</string>  <string name="dir4">左转</string>  <string name="dir5">停车</string>  <string name="dir6">右转</string>  <string name="dir7">左后</string>  <string name="dir8">倒车</string>  <string name="dir9">右后</string>    <!-- Scan -->  <string name="scan\_label">连接</string>  <string name="title\_paired\_devices">已配对设备</string>  <string name="title\_other\_devices">其他可用的设备</string>  <string name="button\_scan">扫描查找设备 </string>  <string name="none\_paired">没有设备已经被配对</string>  <string name="none\_found">找不到设备 </string>  <string name="scanning">扫描设备。。。</string>  <string name="select\_device">选择一个设备连接</string>  <string name="button\_cancel">取消</string>      <string name="title\_connecting">正在连接...</string>  <string name="title\_disconnected">未连接</string>  <string name="bt\_not\_enabled">蓝牙不启用</string>      <string name="discoverable">Make Discoverable</string>  <!-- About -->  <string name="about\_label">关于</string>  <string name="about\_title">使用与说明</string>  <string name="about\_text1">蓝牙连接：点击连接；扫描查找设备，点击连接蓝牙设备，等待连接完成； </string>  <string name="about\_text2">按键控制：点击方向键控制小车运动；</string>  <string name="about\_text3">重力控制：点击启用重力，控制小车运动，用完点击关闭重力；</string>  <string name="about\_text4">出品人：桂林电子科技大学，三院科协</string>  <string name="about\_text5">最终解释权归桂林电子科技大学三院科协所有</string>  <string name="about\_text6">YFRobot出品</string>  <string name="about\_return">返回</string>  </resources> |