

BÀI THỰC HÀNH SỐ 7

Mục đích:

- Giới thiệu các phương pháp giao tiếp vi điều khiển và bộ nhớ dữ liệu SRAM 62256.
- Giới thiệu các phương pháp giao tiếp vi điều khiển và bộ nhớ chương trình ROM 27256

Sau khi kết thúc học phần này, sinh viên có thể:

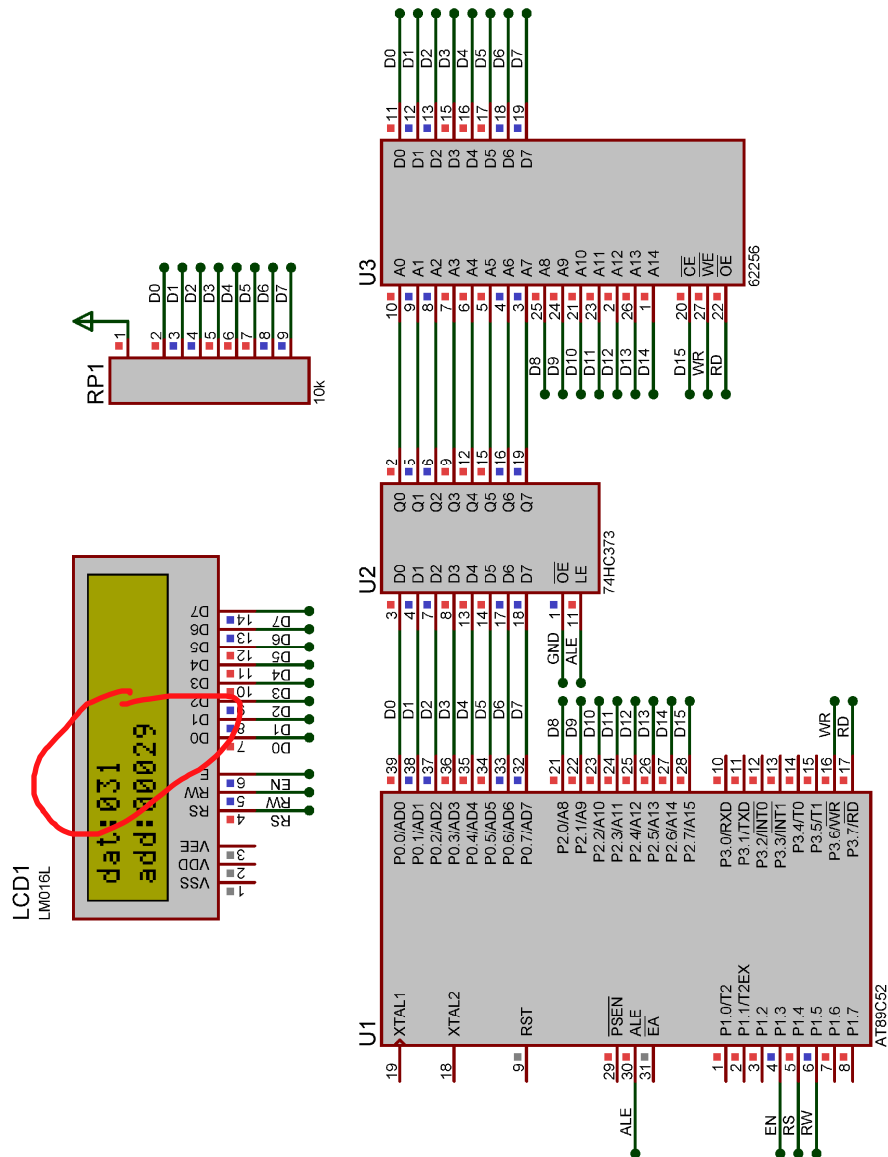
- Hiểu các phương pháp mở rộng bộ nhớ
- Thiết kế một chương trình firmware đơn giản sử dụng ngôn ngữ lập trình cấp cao C dành cho các vi điều khiển (máy tính trên chip) họ 8051 để giao tiếp với các loại bộ nhớ

Yêu cầu thực hành:

Tham khảo datasheet của bộ nhớ RAM 62256/ROM 27256, 74373 từ đó hãy trả lời các câu hỏi sau đây:

- Dung lượng bộ nhớ (Capacity) là bao nhiêu cho từng loại
- tổ chức bộ nhớ (sơ đồ khối) như thế nào?
- Số lượng chân địa chỉ là bao nhiêu?
- Số lượng chân dữ liệu là bao nhiêu?
- Trên vi điều khiển họ 8051 làm thế nào để xác định rằng Chương trình (program code) được nạp (fetched) từ bộ nhớ ROM bên ngoài (external ROM)?
- Vai trò của chân PSEN trên vi điều khiển họ 8051 là gì khi kết nối với ROM ngoại
- Trong kết nối với ROM ngoại, giải thích sự khác nhau giữa các tín hiệu RD và PSEN, chúng được sử dụng ở đâu?
- Giải thích vai trò của các cổng P0, P2 trên vi điều khiển khi kết nối với ROM ngoại
- Vai trò của IC 74373 là gì?
- Bộ nhớ tối đa bên ngoài có thể mở rộng của vi điều khiển họ 8051 là bao nhiêu cho (a) không gian bộ nhớ dữ liệu (b) không gian bộ nhớ Chương trình.
- Khi kết nối với một bộ nhớ bên ngoài, nếu chỉ các chân như P2.0 đến P2.5 được sử dụng thì kích thước bộ nhớ mở rộng được bao nhiêu?
- Khi kết nối đến bộ nhớ ngoại thì các chân P2.6 và P2.7 có được sử dụng cho mục đích điều khiển ngoại vi không?

Vấn đề 1 Giao tiếp giữa vi điều khiển họ 8051 với bộ nhớ RAM ngoại 62256. Sử dụng phần mềm Proteus ISIS vẽ mạch điện dưới đây.



TH Kiến Trúc Và Tổ Chức Máy Tính TS. Phạm Văn Khoa

Sử dụng keilC để viết chương trình với tên **RAM_62256_LCD.c** để tạo chương trình minh họa cho giao tiếp giữa vi điều khiển và RAM ngoại (đi kèm là thư viện LCD1602.h) ■

```
#include<reg52.h>
#include<LCD1602.h>
unsigned char xdata *pp;

void write_ram(unsigned int add,unsigned char dat)
{
    pp=0x0000+add;
    *pp=dat;
}

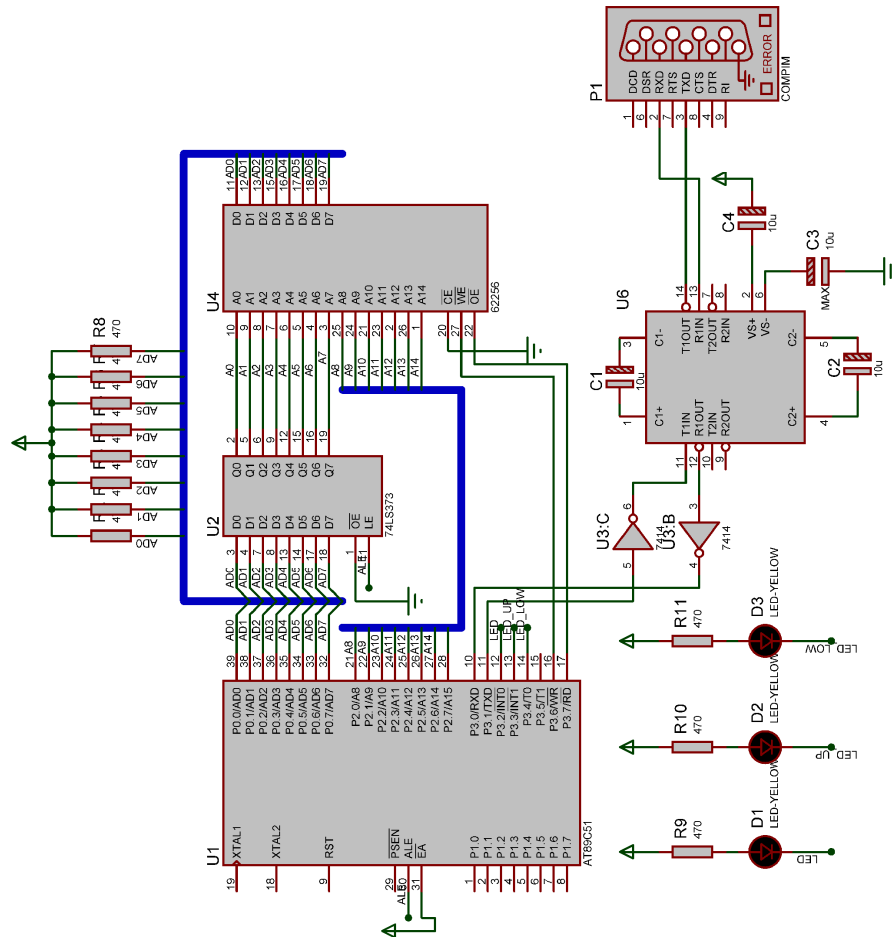
unsigned char read_ram(unsigned int add)
{
    unsigned char dat;
    pp=0x0000+add;
    dat=*pp;
    return dat;
}

void main(void)
{
    unsigned int i,j;
    unsigned char dat;
    unsigned char
line_one[]={ 'd','a','t',':',0,0,0};
    unsigned char
line_two[]={ 'a','d','d',':',0,0,0,0,0};
    init_LCD();
    for(i=0;i<32768;i++)
    {
        write_ram(i,i%256);
    }
    for(i=0;i<4;i++)
    {
        display_str(0,i,line_one[i]);
    }
}
```

```
        display_str(1,i,line_two[i]);
    }
    while(1)
    {
        for(i=0;i<32768;i++)
        {
            dat=read_ram(i);
            line_one[4]=dat/100+0x30;
            line_one[5]=dat/10%10+0x30;
            line_one[6]=dat%10+0x30;
            line_two[4]=i/10000+0x30;
            line_two[5]=i/1000%10+0x30;
            line_two[6]=i/100%10+0x30;
            line_two[7]=i/10%10+0x30;
            line_two[8]=i%10+0x30;
            for(j=4;j<7;j++)
            {
                display_str(0,j,line_one[j]);
            }
            for(j=4;j<9;j++)
            {
                display_str(1,j,line_two[j]);
            }

            delay1ms(1000);
        }
    }
}
```

Vấn đề 2: Giao tiếp giữa vi điều khiển họ 8051 với bộ nhớ RAM ngoại 62256 và truyền nhận giữa liệu giữa máy tính và vi điều khiển. Sử dụng phần mềm Proteus ISIS vẽ mạch điện dưới đây.



Hình 20: Sơ đồ nguyên lý giao tiếp bộ nhớ RAM ngoại và máy tính

TH Kiến Trúc Và Tổ Chức Máy Tính TS. Phạm Văn Khoa

Sử dụng keilC để viết chương trình với tên **RAM_62256_uart.c** để tạo chương trình minh họa cho giao tiếp giữa vi điều khiển và RAM ngoại và truyền nhận dữ liệu với máy tính

```
// RAM 62256 A0-A14
#include <AT89X51.H>

#define B9600    -3           //baud 9600   11.059MHz
#define B4800    -6           //baud 4800   11.059MHz
#define B2400    -12          //baud 2400   11.059MHz
#define B1200    -24          //baud 1200   11.059MHz
//*****Định nghĩa chân*****
#define DBLED    P3_2
#define LED_UP           P3_3
#define LED_DOWN        P3_4

unsigned char xdata *pp;

unsigned int i;
void SetupSerial(char baud);

void TxChar(unsigned char ch);
void TxStr(char *str);
void TxCrLf(void);
unsigned char RxChar(void);

void Delay(unsigned int loop);

void ResetXRam(void);
void write_ram(unsigned int add,unsigned char dat);
unsigned char read_ram(unsigned int add);

void main (void)
{
    unsigned char r_add,d_add;
```

```
SetupSerial(B9600);
DBLED = 1;
LED_UP=1;
LED_DOWN=1;

TxStr("Test External RAM 62C256");
TxCrLf();
ResetXRam();
///  
while(1)
{

    if(RI)
    {
        RI=0;
        r_add=Rxchar();
        DBLED=0;

        TxChar(r_add);
        TxCrLf();

        switch(r_add)
        {
            case 'u':
                LED_UP=0;
                d_add=read_ram(r_add);
                TxChar(r_add-32);
                break;
            case 'L':
                LED_DOWN=0;
                d_add=read_ram(r_add);
                TxChar(r_add+32);
                break;
            default:
                break;
        }
    }
}
```

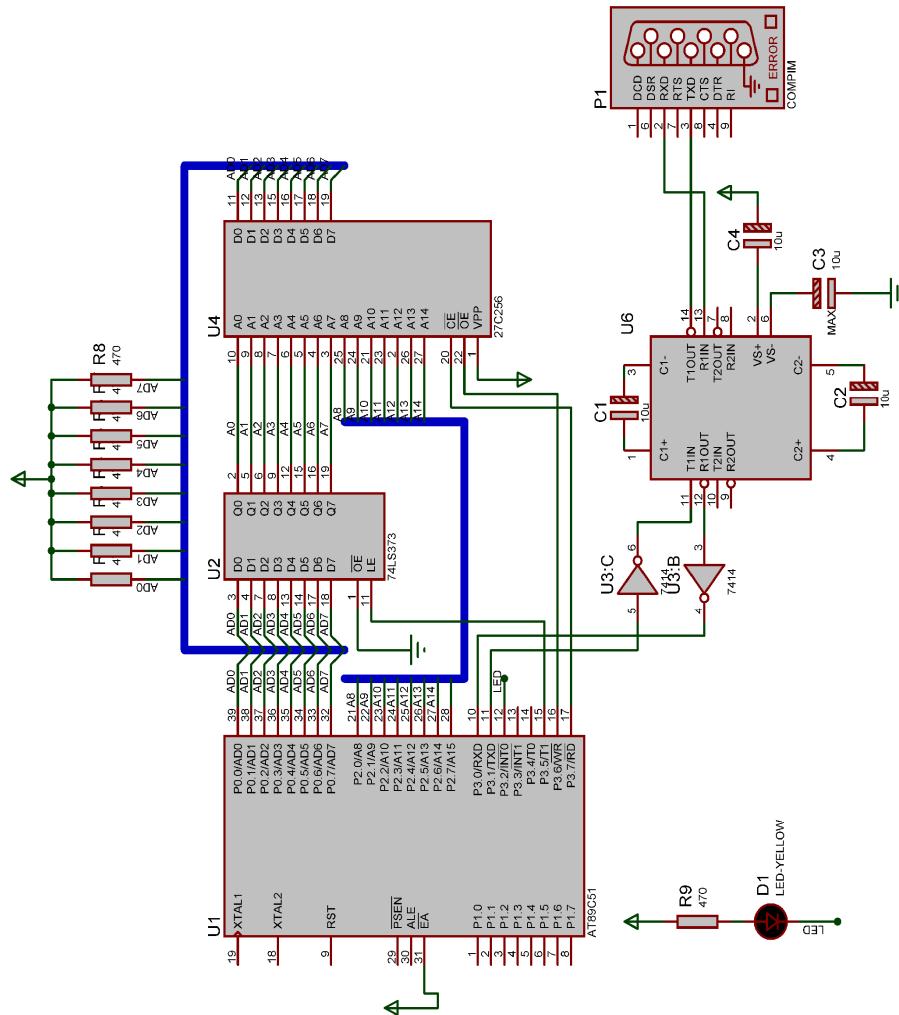


```
}  
void SetupSerial(char baud)  
{  
    SCON = 0x50;  
    TMOD = 0x20;  
    TH1 = baud;  
    TF1 = 0;  
    TR1 = 1;  
}  
//*****Transmit*****  
void TxChar(unsigned char ch)  
{  
    SBUF = ch;  
    while(TI==0);  
    TI=0;  
}  
  
void TxStr(char *str)  
{  
    while(*str!=0)  
    {  
        TxChar(*str);  
        str++;  
    }  
}  
void TxCrLf(void)  
{  
    TxChar(13);  
    TxChar(10);  
}  
  
//*****Receive*****  
unsigned char RxChar(void)  
{  
    RI = 0;  
    return(SBUF);  
}  
//*****DeLay*****  
void Delay(unsigned int loop)
```

```
{
    while(loop--);
}
//*****
void ResetXRam(void)
{
    unsigned int j;
    TxCrLf();
    TxStr("Reset Xram");
    TxCrLf();

    //for(j=0;j<32767;j++)write_ram(j,j%256);
    for(j=0;j<257;j++)write_ram(j,j%256);
    TxStr("Reset Ram OK");
}
void write_ram(unsigned int add,unsigned char dat)
{
    pp=0x0000+add;
    *pp=dat;
}
unsigned char read_ram(unsigned int add)
{
    unsigned char dat;
    pp=0x0000+add;
    dat=*pp;
    return dat;
}
```

Vấn đề 3: Giao tiếp giữa vi điều khiển họ 8051 với bộ nhớ ROM ngoại 27256 và truyền nhận giữa liệu giữa máy tính và vi điều khiển. Sử dụng phần mềm Proteus ISIS vẽ mạch điện dưới đây.



Hình 21: Sơ đồ nguyên lý giao tiếp bộ nhớ ROM ngoại

Sử dụng keilC để viết chương trình

```
#include <AT89X51.H>
#define B9600 -3 //baud 9600 11.059MHz
#define B4800 -6 //baud 4800 11.059MHz
#define B2400 -12 //baud 2400 11.059MHz
#define B1200 -24 //baud 1200 11.059MHz
#define DBLED P3_2
#define LAT P3_5
#define OE P3_6 //G
#define CE P3_7 //E
extern void SetupSerial(char baud);
extern void TxChar(unsigned char ch);
extern unsigned char RxChar(void);
extern void TxStr(char *str);
extern void TxHex(unsigned char ch);
extern void TxCrLf(void);
void SetReadXRom(void);
void ReadXRom(void);
void Delay(unsigned int Loop);
unsigned char ReadByte(unsigned int Addr);
unsigned int i;
void main (void)
{
    unsigned char ch;
    SP = 0x62;
    SetupSerial(B9600);
    TxStr("External rom");
    TxCrLf();
    while(1)
    {
        DBLED = 1;
        if(RI)
        {
            DBLED = 0;
            ch = RxChar();
            TxChar(ch);
            switch(ch) {
                case 0x12: //Ctrl_R
```

```
SetReadXRom();
ReadXRom();
break;
default:
break;
}
}
}
}
void SetReadXRom(void)
{
CE = 1;
OE = 1;
Delay(15000);
}
void ReadXRom(void)
{
unsigned char ch;
for(i=256;i<276;i++)
{
ch = ReadByte(i);
TxHex(ch);
}
}
unsigned char ReadByte(unsigned int Addr)
{
unsigned char RetVal,HiAddr,LoAddr;
LoAddr = (unsigned char) Addr;
HiAddr = (unsigned char) (Addr >> 8);
P0 = LoAddr;
LAT = 1;
LAT = 0;
HiAddr = (HiAddr & 0x7f) | (P2 & 0x80);
P2 = HiAddr;
P0 = 0xff;
CE = 0;
OE = 0;
Delay(0);
```

```
RetVal = P0;
CE = 1;
OE = 1;
return(RetVal);
}
void Delay(unsigned int loop)
{
while(loop--);
}
//siop.c
#include <AT89X52.H>
void SetupSerial(char baud);
void TxChar(unsigned char ch);
unsigned char RxChar(void);
void TxStr(char *str);
void TxHex(unsigned char ch);
void TxCrLf(void);
void SetupSerial(char baud)
{
SCON = 0x52;
TMOD = 0x20;
TH1 = baud;
TF1 = 0;
TR1 = 1;
}
void TxChar(unsigned char ch)
{
while(!TI);
TI = 0;
SBUF = ch;
}
unsigned char RxChar(void)
{
RI = 0;
return(SBUF);
}
void TxStr(char *str)
{
while(*str)
```

```
{  
TxChar(*str);  
str++;  
}  
}  
void TxNibble(unsigned char ch)  
{  
if(ch < 10)  
{  
ch = ch + 48;  
TxChar(ch);  
}  
else  
{  
ch = ch + 55;  
TxChar(ch);  
}  
}  
void TxHex(unsigned char ch)  
{  
unsigned char tmp;  
tmp = (ch & 0xf0) >> 4;  
TxNibble(tmp);  
tmp = ch & 0x0f;  
TxNibble(tmp);  
}  
void TxCrLf(void)  
{  
TxChar(13);  
TxChar(10);  
}
```