

# 位组成字和字分解成位

本文档内容仅供参考，如使用中存在问题请与技术支持取得联系。

## 1. 实现功能

通过 C 语言程序实现任意选择连续的位组成一个字，或者将一个字中连续的几个位分解出来。

## 2. 使用设备

PLC 型号：XDH-30A16-E

软件版本：XDPro 3.7.17b

## 3. 使用功能介绍

### 1) FUNC1

实现将多个连续的位组成一个字（最多16个位）。

D100 中存储需组合成字的位软元件 M 个数，D300 中存储位软元件 M 的起始地址，利用 for 循环、<<位左移、|按位或后赋值实现将 M 点状态赋值给 16 个数组成员，而后将其组合后存放在 HD100 中。

```
void FUNC1(PINT16S W,PBIT B)
{
    #define SysRegAddr_HD_D_HM_M
    INT Data = 0x00;           //定义一个16进制数0为Data
    INT n[16];                 //定义一个长度是16的数组n
    int i,j,a,d;               //定义a变量是位组成一个字的个数
    a = D[100];                //定义d变量是位组成字的位起始地址
    d = D[300];                //定义d变量是位组成字的位起始地址
    for(i=0;i<16;i++)          //将16个M点状态赋值给16个数组
    {
        if(M[i+d])
        {
            n[i] = 1;
        }
        else{
            n[i] = 0;
        }
    }
    for(j=0;j<a;j++)            //将a个位的状态组合放到寄存器HD100里面
    {
        Data |= n[j] << j;
    }
    HD[100] = Data;
}
```

## 2) FUNC2

将一个字分解成几个位（最多16个位且是连续的）。

D200 是一个字分解成位的个数，D400 是分解后的位起始地址，HD200 是需要将位提取出来的数据源。

```
void FUNC2(PINT16S W,PBIT B)
{
    #define SysRegAddr_HD_D_HM_M

    INT c[16];                //定义一个长度16的数组c
    int b,k,f;

    k = D[200];                //定义k变量为一个字分解成位的个数
    f = D[400];                //定义f变量为分解后的位起始地址

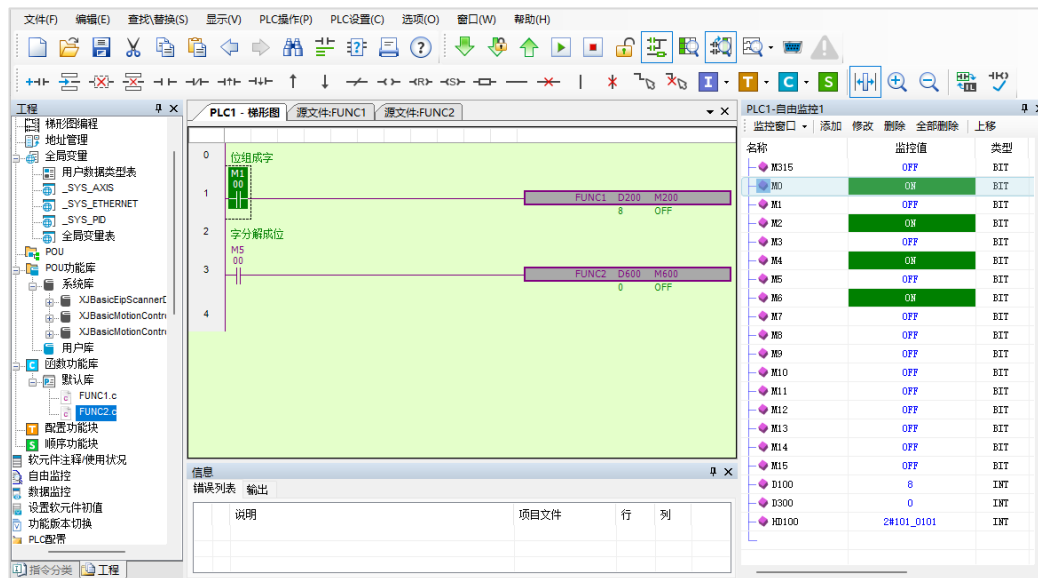
    for(b=0;b<k;b++)          //将一个字HD200的k个位取出来
    {
        c[b] = ( HD[200] & ( 0x01 << b ) ) == ( 0x01 << b ) ? 1 : 0 ;
        if(c[b]==0)
        {
            M[b + f] = 0;
        }
        else
        {
            M[b + f] = 1;
        }
    }
}
```

## 4. 操作步骤

### 1) 实现将 M0 往后连续 8 个位的状态，组合成一个字。

具体操作步骤：

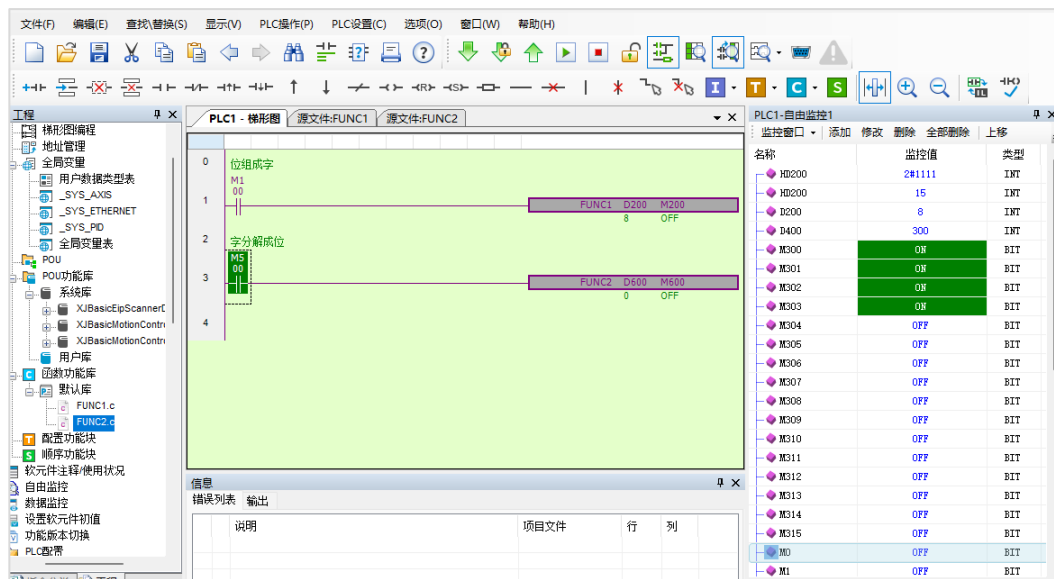
- 1、给 D100 传值 8;
- 2、给 D300 传值 0;
- 3、给 M0-M7 连续 8 个位状态;
- 4、梯形图中触发 M100,则结果存放在 HD100 里面，如截图所示：



2) 实现将一个字的连续 8 个位的状态分解到 M300 往后的点位中。

具体操作步骤:

- 1、给 D200 传值 8;
- 2、给 D400 传值 300;
- 3、给 HD200 输入想要分解位的一个字;
- 4、梯形图中触发 M500,则结果存放在 M300-M307 这几个位中。如图所示:



## 5. 注意事项

- 在触发相应 C 语言块之前,一定要给相关参数数值(D100,D200,D300,D400,HD200)。
- 本案例中位组成字,只能是连续的位,且最多 16 个位;字分解成位,只能分解传送到连续的位中,且最多 16 个位。