

# 教育机器人教师培训

## 第8课：系统时间的控制



# 亮灯时间控制（主程序）

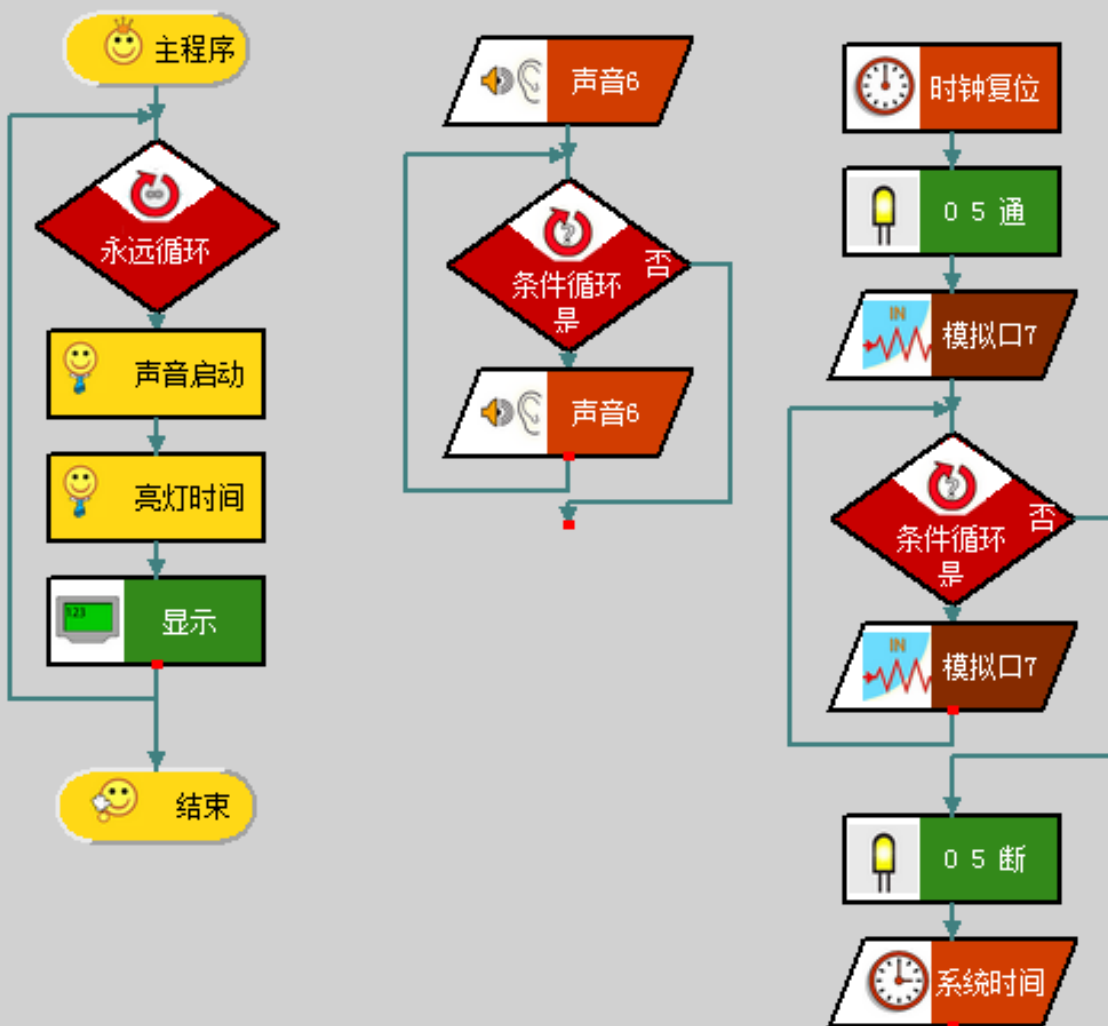
\_Build:20150104 Ccon102 [8 亮灯时间控制.flw] 主程序窗口

视图(V) 工具(T) 窗口(W) 帮助(H)



主程序名字 ProgramA

图库  
图库  
图库  
子程序  
模块  
启动



```
#include "ASEIO.h"
```

```
float tim_1=0.0f;
```

```
int ma_1=0;
```

```
int voi_1=0;
```

```
void main()
```

```
{
```

```
while(1)
```

```
{
```

```
SubRoutine_1 ();
```

```
SubRoutine_2 ();
```

```
printf( "\n\ntim_1=%f\n\n\n\n",tim_1 );
```

```
}
```

```
}
```

# 亮灯时间控制 (亮灯时间主程序)

注释：  
从声音启动到红外感知的时间

The screenshot displays a PLC programming environment with the following components:

- File Name:** VJC\_4.2\_CH\_Build:20150104 Ccon102 [8 亮灯时间控制.flw] 子程序窗口: 亮灯...
- Menu Bar:** 文件(F) 编辑(E) 视图(V) 工具(T) 窗口(W) 帮助(H)
- Toolbar:** Includes icons for file operations, search, and execution.
- Left Panel (Library):**
  - 执行器模块库 (Actuator Module Library)
  - 传感器模块库 (Sensor Module Library)
  - 控制模块库 (Control Module Library)
  - 程序模块库 (Program Module Library)
  - 任务 (Task)
  - 新建子程序 (New Subroutine)
  - 子程序返回 (Subroutine Return)
  - 结束模块 (End Module)
  - 自定义 (Custom)
  - 声音启动 (Sound Start)
  - 亮灯时间 (Light On Time)
- Main Diagram Area:**
  - Start: 亮灯时间 (Light On Time)
  - Step 1: 时钟复位 (Clock Reset)
  - Step 2: 0 5 通 (0 5 On)
  - Step 3: 模拟口7 (Analog Port 7)
  - Step 4: 条件循环 (Condition Loop) - 是 (Yes) / 否 (No)
  - Step 5: 模拟口7 (Analog Port 7)
  - Step 6: 0 5 断 (0 5 Off)
  - Step 7: 系统时间 (System Time)
  - End: 返回 (Return)
- Code Window (Right):**

```
/*亮灯时间*/  
void SubRoutine_2()  
{  
    resettime();  
    SetDO( 0x21, 1 );  
    ma_1 = AI(7);  
    while(ma_1 > 500)  
    {  
        ma_1 = AI(7);  
    }  
    SetDO( 0x21, 0 );  
    tim_1 =seconds();  
    return;  
}
```

就绪

```
float tim_1=0.0f;
int ma_1=0;
int voi_1=0;
void SubRoutine_1();
void SubRoutine_2();
/*声音启动*/
void SubRoutine_1()
{
    voi_1 = AI(6);
    while(voi_1 < 1800)
    { voi_1 = AI(6); }
    return;
}
/*亮灯时间*/
void SubRoutine_2()
{
    resettime();
    SetDO( 0x21, 1 ); ma_1 = AI(7);
    while(ma_1 > 500) { ma_1 = AI(7); }
    SetDO( 0x21, 0 );
    tim_1 =seconds();
    return;
}
```

```
void main()
{
    while(1)
    {
        SubRoutine_1 ();
        SubRoutine_2 ();
        printf( "\n\ntim_1=%f\n\n\n\n\n",tim_1 );
    }
}
```

# 编写、调试以下项目程序探索规律

## 测距小车

### (1) 活动要求

测距小车是指小车能够自动测算出自己走过的路程，并将结果显示到屏幕上。

具体目标：小车运行程序后，开始行驶，到达终点时，按下控制器的 **Enter** 按键，小车停止，在控制器的屏幕上返回路程的数值。

### (2) 活动分析

A、根据公式：路程=时间 x 速度，只要知道小车走过的时间与速度即可计算得出它的路程。

B、如何获取速度？让小车在行驶过程中保持速度不变，通过测量与计算可以得出它的速度。

C、如何获取时间？小车行驶过程中控制器可以记录小车行驶的时间，可以通过访问“系统时间变量”获取这个时间。

