



# 模块 5

测验：循线迷宫-有限状态机



# 测验：循线迷宫-有限状态机

## 问题 1 指针与字符串

编写一个比较两个以 null 结尾的字符串的函数。指向两个字符串的指针通过引用传递给函数。如果字符串不匹配，则返回参数将为 0；如果字符串匹配，则返回参数将为非零。原型是

```
int StringCompare(char *pt1, char *pt2);
```

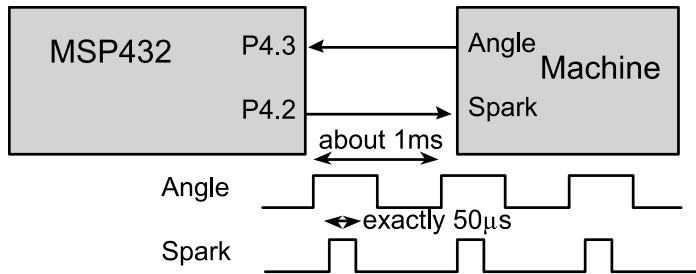
## 问题 2 数组与索引

写出一个计算两个数组的点积的函数。每个数组都有十个 32 位有符号整数。原型是：

```
int32_t DotProduct(int32_t buf1[10], int32_t buf2[10]);
```

## 问题 3 有限状态机

使用链表有限状态机设计一个基于微机的控制器。该系统具有一个输入和一个输出。



输入**角度**是一个周期信号，频率为 20 到 1000 Hz（周期为 1 到 50 ms）。每次**角度**从 0 变为 1 时，输出**触发**应为正脉冲（正好为 50µs 宽）。**角度**上升沿和**触发**脉冲开始之间的延迟应尽可能短。**角度**的周期可以在 1 毫秒到 50 毫秒之间变化。由于**角度**是输入，您无法控制它，只响应其上升沿。

- 为该系统设计一个输入，一个输出有限状态机。绘制状态转换图。使用描述性的状态名。
- 显示 C 代码以创建静态分配的链表。包含声明，将其放入微型计算机的 ROM 中。
- 显示 C 语言控制器。控制器可以没有条件语句（如果没有开关，没有条件运算符）。假设这是微型计算机执行的唯一任务，即，显示所有必要的说明。使程序在复位上自动启动。您可以在 **Clock.c** 中调用函数。

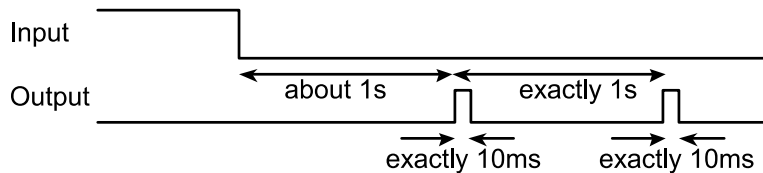
## 问题 4 有限状态机

设计具有两个输入和两个输出的 FSM。输入在端口 4 位 0,1 上，输出在端口 5 位 0,1 上。最初输出为 00。您也可以假设两个输入都是 0。机器等待第一个输入变为 1。如果 P4.1 先变为高电平，则将输出设置为 10，如果 P4.0 先变为高电平，则输出转到 01，如果两个输入同时变为高电平，则输出变为 11。

- 为该系统设计两输入，两输出有限状态机。绘制状态转换图。使用描述性的状态名。
- 显示 C 代码以创建静态分配的链表。将下一个状态实现为指针，包含语句以将其放入微型计算机的 ROM 中。
- 显示 C 语言控制器。控制器可以没有条件语句（如果没有开关，没有条件运算符）假设这是微机执行的唯一任务。即，显示所有必要的说明。使程序在复位上自动启动。您可以在 **Clock.c** 中调用函数。

## 问题 5 有限状态机

您将使用 Moore FSM 设计起搏器。有一个输入和一个输出。如果心脏自己跳动，输入将会很高。如果心脏没有自己跳动，输入将是低的。如果心脏没有击打你的机器应该放心。如果心脏单独跳动，输入将很高，输出应该很低。但是，如果输入为低电平，则应通过每 1000 ms 输出 10 ms 输出脉冲来调节心率。P4.0 输出，P4.1 输入。



- 为该系统设计一个输入，一个输出有限状态机。绘制状态转换图。使用描述性的状态名
- 显示 C 代码以创建静态分配的链表。包含声明，将其放入微型计算机的 ROM 中。
- 显示 C 语言控制器。控制器可以没有条件语句（如果没有开关，没有条件运算符）。假设这是微型计算机执行的唯一任务。即，显示所有必要的说明。使程序在复位上自动启动。您可以在 **Clock.c** 中调用函数。