

3-27 控制系统与接收机连接图

3.2.8.2 接口设计

由于本系统分布在上下两块板，两板之间存在电气连接，因此需要设计电路板间接口，上下板之间各对应引脚之间的电气连接关系如表 3-3 所示：

表 3-3 接口引脚的电气定义

Pin	Definition	Pin	Definition
1	7.5V 电源	11	右电机控制
2	3.3V 电源	12	前电机控制
3	5.0V 电源	13	后电机控制
4	单片机状态显示	14	左电机控制
5	JTAG-TDI	15	空
6	JTAG-TDO	16	接地
7	JTAG-TMS	17	空
8	JTAG-TCK	18	接地
9	GYRO-Z	19	GYRO-X
10	ZREF	20	GYRO-Y

3.3 软件设计^{[19][20][21][23][40]}

3.3.1 总体设计

本控制系统的软件部分采用 C 语言编制，主要功能是接收遥控指令，对飞

行器进行遥控操作，同时通过采集飞行器姿态信号，对飞行器进行闭环控制，提高飞行器的飞行稳定性和可靠性，使飞行器的运行安全可靠。由于四桨碟形飞行器是一个极为发散的系统，为了更好的改善其飞行稳定性，加入了角速率陀螺反馈环，并通过软件对其采集的信号进行处理，辅助飞行器的控制。

本设计要求输出精确的 PWM 脉冲，因此采用外部晶体。

主程序流程图如图 3-28 所示，系统对时钟进行初始化，选择外部晶体为系统时钟；配置交叉开关，将 P0.0~P0.3 配置给 UART0、UART1，以方便系统调试及传感器标定，将 P0.4~P0.7 配置给 CEX0~CEX3，输出所要求得到的 PWM 脉冲；初始化计数器/定时器、ADC、PCA 等内部资源，并开始接收遥控指令、采集角度信号和角速率信号，通过运算子程序计算旋翼控制量，并产生相应占空比的 PWM 脉冲，来控制旋翼速度，调节飞行器姿态。

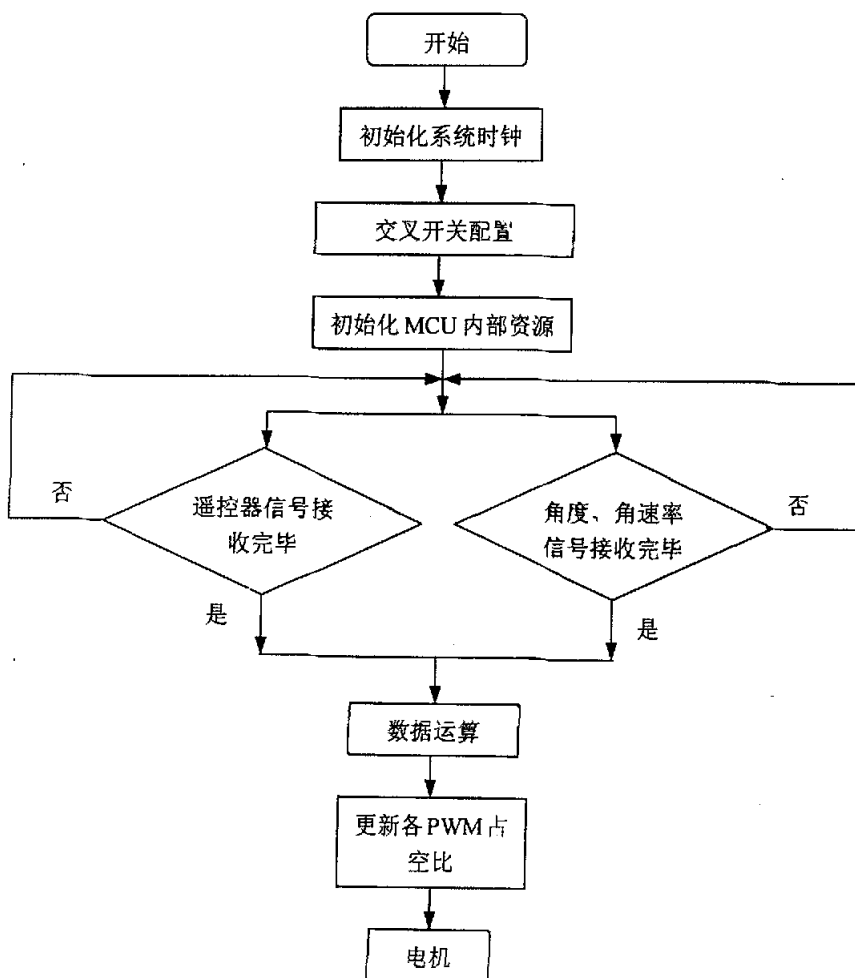


图 3-28 主程序流程框图

系统的数据输入输出如图 3-29 所示，输入信号主要有：来自遥控器的四路

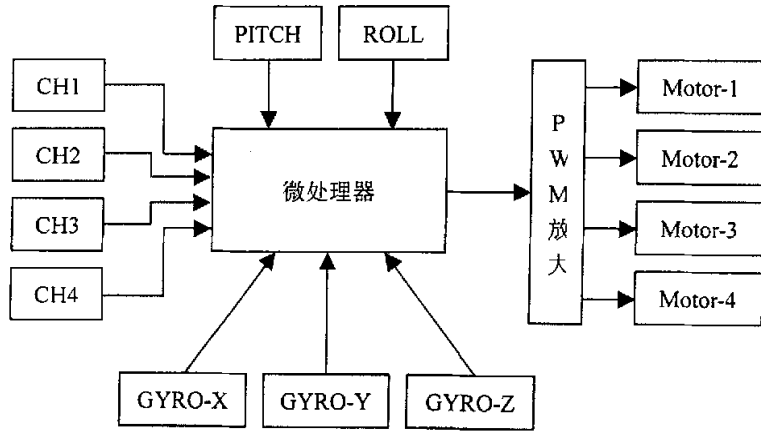


图 3-29 系统输出输出

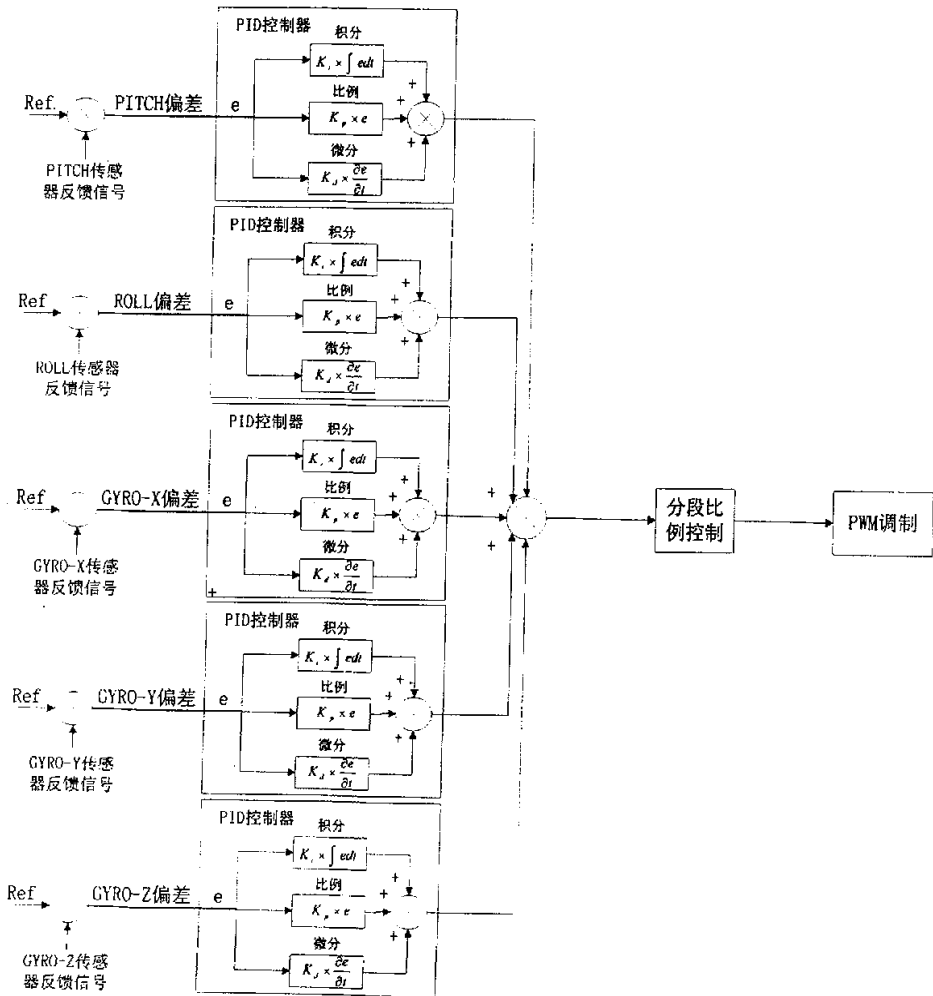


图 3-30 控制系统框图

遥控器指令，分别用来控制飞行器的升降、前飞后飞、左飞右飞、自身旋转；来

自角度传感器的两路角度信号 PITCH 和 ROLL，用来感知飞行器的当前姿态并用来与目标姿态进行比较，实现闭环控制；来自三个单轴角速率陀螺的三路角速率信号，用来增加阻尼，提高飞行器的飞行稳定性。微处理器将所获得的这 9 路信号之后，将这些信号通过 PID、及分段比例控制运算，计算得出四个旋翼的控制量，并以 PWM 占空比的形式分别通过端口 P0.4、P0.5、P0.6、P0.7 输出控制旋翼动作。

控制系统的控制运算如图 3-30 所示，将各传感器所获得的信号与遥控指令相比较，得出各个偏差 e ，通过 PID 控制算法得出各传感信号对系统的控制量，然后再将这些控制量根据控制规则进行分段比例控制，得出最终的控制量，并以 PWM 的形式输出控制各个旋翼。

3.3.2 系统初始化

由于系统需要一个精确的时基信号，因此考虑采用 24M 的外部晶体。系统初始化程序如下：

```
//初始化系统时钟，使用外部振荡器
void SYSCLK_Init(void)
{
    int i; //延时计数器
    OSCXCN=0x67; //启动外部振荡器
    for (i=0;i<256;i++); //等待振荡器起振
    while(!(OSCXCN&0x80)); //等待晶体振荡器稳定
    OSCICN=0x88; //选择外部振荡器作为系统时钟源
}
```

3.3.3 交叉开关和 I/O 口配置

C8051F021 具有丰富的内部资源，可以通过优先权交叉开关译码器（即交叉开关）将端口 0~3 的引脚分配给器件上的数字外设。分配顺序是从 P0.0 开始，可以一直分配到 P3.7。

本设计将 UART0、UART1、PCA 等内部资源分配给各端口引脚。引脚配置步骤如下：

- 交叉开关引脚分配：

当交叉开关配置寄存器 XBR0，XBR1 和 XBR2 中外设的对应允许位被设置为逻辑 1 时，交叉开关将端口引脚分配给外设，特殊功能寄存器的设置如下所示。

特殊功能寄存器 XBR0 的设置：