

## 1路编码器脉冲计数器或2路DI高速计数器, Modbus RTU模块 YL150

### 产品特点:

- 编码器解码转换成标准Modbus RTU协议
- 可用作编码器计数器或者转速测量
- 支持编码器计数, 可识别正反转
- 也可以设置作为2路独立DI高速计数器
- 计数值支持断电自动保存
- DI输入支持PNP和NPN输入
- 继电器和机械开关输入时可以设置滤波时间
- 通过RS-485接口可以清零和设置计数值
- 宽电源供电范围: 8 ~ 32VDC
- 可靠性高, 编程方便, 易于应用
- 标准DIN35导轨安装, 方便集中布线
- 用户可编程设置模块地址、波特率等



图 1 YL150 模块外观图

### 典型应用:

- 编码器脉冲信号测量
- 流量计脉冲计数或流量测量
- 生产线产品计数
- 物流包裹数量计数
- 接近开关脉冲信号测量
- 编码器信号远传到工控机
- 水表或电表脉冲计数
- 智能工厂与工业物联网

### 产品概述:

YL150产品实现传感器和主机之间的信号采集, 用来解码编码器信号。YL150系列产品可应用在 RS-485总线工业自动化控制系统, 自动化机床, 工业机器人, 三坐标定位系统, 位移测量, 行程测量, 角度测量, 转速测量, 流量测量, 产品计数等等。

产品包括信号隔离, 脉冲信号捕捉, 信号转换和RS-485串行通信。每个串口最多可接255只 YL150系列模块, 通讯方式采用ASCII码通讯协议或MODBUS RTU通讯协议, 波特率可由代码设置, 能与其他厂家的控制模块挂在同一RS-485总线上, 便于计算机编程。

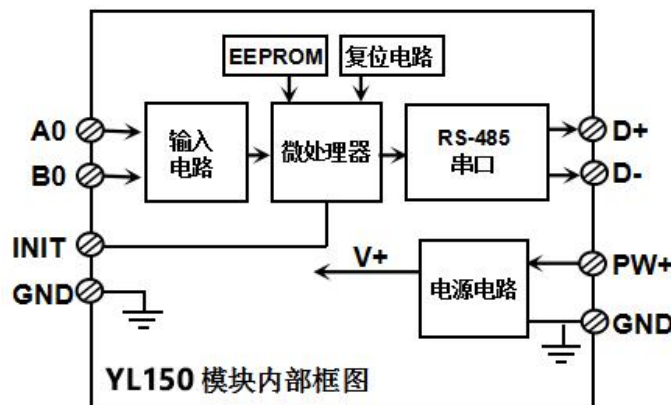


图 2 YL150 模块内部框图

YL150系列产品是基于单片机的智能监测和控制系统,所有的用户设定的地址,波特率,数据格式,校验和状态等配置信息都储存在非易失性存储器EEPROM里。

YL150系列产品按工业标准设计、制造,信号输入 / 输出之间不隔离,抗干扰能力强,可靠性高。工作温度范围- 45°C ~ +85°C。

### 功能简介:

YL150远程I/O模块,可以用来测量1路编码器信号,也可以设置作为2路独立计数器或者DI状态测量。

#### 1、信号输入

1 路编码器信号输入或 2 路独立计数器,可接干接点和湿接点,通过命令设置输入类型。

#### 2、通讯协议

通讯接口: 1 路标准的 RS-485 通讯接口。

通讯协议: 支持两种协议,命令集定义的字符协议和 MODBUS RTU 通讯协议。模块自动识别通讯协议,能实现与多种品牌的 PLC、RTU 或计算机监控系统进行网络通讯。

数据格式: 10 位。1 位起始位,8 位数据位,1 位停止位。无校验。

通讯地址 (0~255) 和波特率 (2400、4800、9600、19200、38400、57600、115200bps) 均可设定; 通讯网络最长距离可达 1200 米,通过双绞屏蔽电缆连接。

通讯接口高抗干扰设计,±15KV ESD 保护,通信响应时间小于 100ms。

#### 3、抗干扰

可根据需要设置校验和。模块内部有瞬态抑制二极管,可以有效抑制各种浪涌脉冲,保护模块,内部的数字滤波,也可以很好的抑制来自电网的工频干扰。

### 产品选型:

YL150 -   □    
          通讯接口

**485:** 输出为 RS-485 接口

选型举例: 型号: **YL150 - 485** 表示输出为 RS-485 接口

(typical @ +25°C, Vs为24VDC)

输入类型: 编码器 AB 信号输入, 1 通道 (A0/B0)。

低电平: 输入 < 1V

高电平: 输入 3.5 ~ 30V

频率范围 0-50KHz。

编码器计数范围  $\uparrow$  - 2147483647 ~  $\uparrow$  +2147483647

DI 计数器范围 0 ~  $\uparrow$  4294967295

输入电阻: 30KΩ

通 讯: 协议 RS-485 标准字符协议 和 MODBUS RTU通讯协议

波特率 (2400、4800、9600、19200、38400、57600、115200bps) 可软件选择

地址 (0~255) 可软件选择

通讯响应时间: 100 ms 最大

工作电源: +8 ~ 32VDC 宽供电范围, 内部有防反接和过压保护电路

功率消耗: 小于1W

工作温度: - 45 ~ +80°C  
 工作湿度: 10 ~ 90% (无凝露)  
 存储温度: - 45 ~ +80°C  
 存储湿度: 10 ~ 95% (无凝露)  
 外形尺寸: 106 mm x 59mm x 24mm

引脚定义:

引脚	名称	描述	引脚	名称	描述
1	PW+	电源正端	5	INIT	初始状态设置
2	GND	电源负端	6	GND	数字信号输出地
3	DATA+	RS-485 信号正端	7	A0	编码器 0 信号 A 输入端
4	DATA-	RS-485 信号负端	8	B0	编码器 0 信号 B 输入端

表1 引脚定义

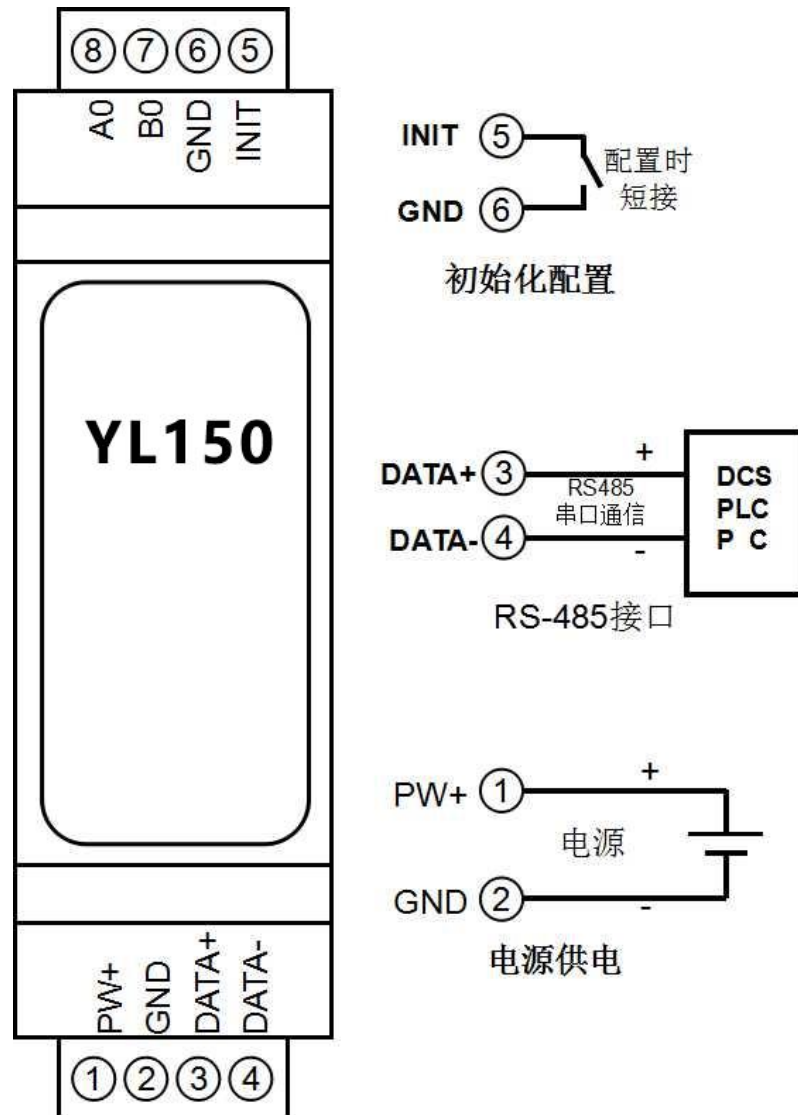
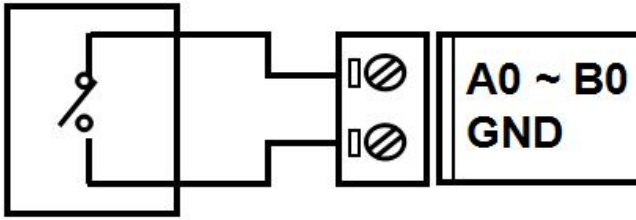
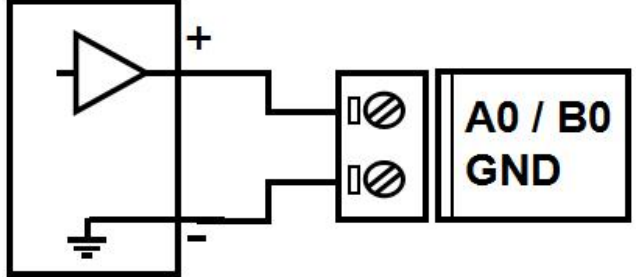


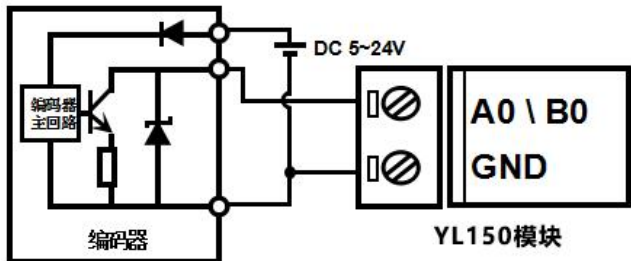
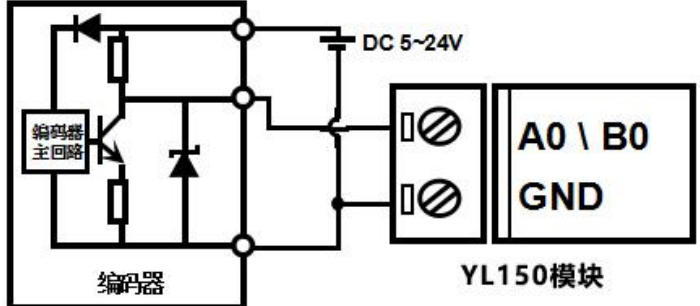
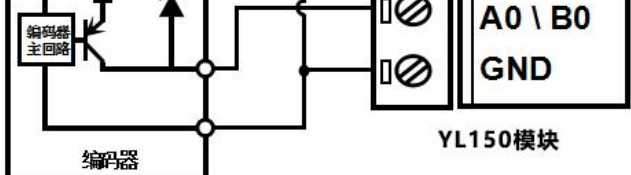
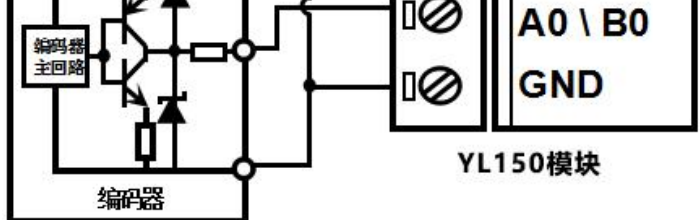
图3 YL150 模块接线图

DI 计数输入接线图

干接点输入	电平输入
	
<p>需要打开内部上拉电阻, 40082 寄存器设置为 1, 或者发送字符命令\$01Q1。</p>	<p>需要关闭内部上拉电阻, 40082 寄存器设置为 0, 或者发送字符命令\$01Q0</p>

注: 出厂默认是关闭上拉的

编码器信号输入接线图

NPN 型编码器	带上拉电阻的 NPN 型编码器
	
<p>需要打开内部上拉电阻, 40082 寄存器设置为 1, 或者发送字符命令\$01Q1。</p>	<p>需要关闭内部上拉电阻, 40082 寄存器设置为 0, 或者发送字符命令\$01Q0</p>
PNP 型编码器	推挽式编码器
	
<p>需要关闭内部上拉电阻, 40082 寄存器设置为 0, 或者发送字符命令\$01Q0</p>	<p>需要关闭内部上拉电阻, 40082 寄存器设置为 0, 或者发送字符命令\$01Q0</p>

注: 出厂默认是关闭上拉的

## YL150 字符协议命令集:

模块的出厂初始设置, 如下所示:

**地址代码为 01**

**波特率 9600 bps**

**禁止校验和**

如果使用 RS-485 网络, 必须分配一个不重复的地址代码, 地址代码取值为 16 进制数在 00 和 FF 之间, 由于新模块的地址代码都是一样的, 他们的地址将会和其他模块矛盾, 所以当你组建系统时, 你必须重新配置每一个 YL150 模块地址。可以在接好 YL150 模块电源线和 RS485 通讯线后, 通过配置命令来修改 YL150 模块的地址。波特率, 校验和状态也需要根据用户的要求而调整。而在修改波特率, 校验和状态之前, 必须让模块先进入缺省状态, 否则无法修改。

### 让模块进入缺省状态的方法:

YL150 模块都有一个特殊的标为 INIT 的管脚。将 INIT 管脚短路接到 GND 管脚后, 再接通电源, 此时模块进入缺省状态。在这个状态时, 模块的配置如下:

地址代码为 00

波特率 9600 bps

禁止校验和

这时, 可以通过配置命令来修改 YL150 模块的波特率, 校验和状态等参数。在不确定某个模块的具体配置时, 也可以将 INIT 管脚短路接到 GND 管脚, 再接通电源, 使模块进入缺省状态, 再对模块进行重新配置。

字符协议命令由一系列字符组成, 如首码、地址 ID, 变量、可选校验和字节和一个用以显示命令结束符(**cr**)。主机除了带通配符地址 “\*” 的同步的命令之外, 一次只指挥一个 YL150 模块。

命令格式: **(Leading Code)(Addr)(Command)[data][checksum](cr)**

**(Leading code)** 首码是命令中的第一个字母。所有命令都需要一个命令首码, 如%, \$, #, @, ...等。 **1- 字符**

**(Addr)** 模块的地址代码, 如果下面没有指定, 取值范围从 00 ~ FF (十六进制)。 **2- 字符**

**(Command)** 显示的是命令代码或变量值。 **变量长度**

**[data]** 一些输出命令需要的数据。 **变量长度**

**[checksum]** 括号中的Checksum (校验和) 显示的是可选参数, 只有在启用校验和时, 才需要此选项。 **2- 字符**

**(cr)** 识别用的一个控制代码符, (**cr**)作为回车结束符, 它的值为0x0D。 **1-字符**

当启用校验和(checksum)时, 就需要[Checksum]。它占2-字符。命令和应答都必须附加校验和特性。校验和用来检查所有输入命令, 来帮助你发现主机到模块命令错误和模块到主机响应的错误。校验和字符放置在命令或响应字符之后, 回车符之前。

计算方法: 两个字符, 十六进制数, 为之前所发所有字符的ASCII码数值之和, 然后与十六进制数0xFF相与所得。

应用举例: 禁止校验和(checksum)

用户命令 **\$002(cr)**

模块应答 **!00020600 (cr)**

启用校验和(checksum)

用户命令 **\$002B6 (cr)**

模块应答 **!00020600 A9 (cr)**

'\$' = 0x24    '0' = 0x30    '2' = 0x32

B6=(0x24+0x30+0x30+0x32) AND 0xFF

'!' = 0x21    '0' = 0x30    '2' = 0x32    '6' = 0x36

A9=(0x21+0x30+0x30+0x30+0x32+0x30+0x36+0x30+0x30) AND 0xFF

命令的应答：

应答信息取决于各种各样的命令。应答也由几个字符组成，包括首代码，变量和结束标识符。应答信号的首代码有两种，‘!’或‘>’表示有效的命令而‘?’则代表无效。通过检查应答信息，可以监测命令是否有效

**注意：1、**在一些情况下，许多命令用相同的命令格式。要确保你用的地址在一个命令中是正确的，假如你用错误的地址，而这个地址代表着另一个模块，那么命令会在另一个模块生效，因此产生错误。

**2、**必须用大写字母输入命令。

**3、(cr)**代表键盘上的回车符，不要直接写出来，应该是敲一下回车键（Enter 键）。

### 1、设置编码器的工作模式

说明：设置编码器工作模式，0 或 1，出厂默认为 0。工作模式修改后，必须重启模块才会生效。

**工作模式 0：**编码器 AB 信号输入

**工作模式 1：**两路独立的高速计数器输入

**注意：**下面命令备注（**工作模式 0**）的表示是仅在编码器工作模式为 0 时数据才有效。

备注（**工作模式 1**）的表示是仅在编码器工作模式为 1 时数据才有效。

命令格式：**\$AA3B** 设置编码器的工作模式。重启后生效。

参数说明：**AA** 模块地址，取值范围 00~FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

应答格式：**! AA(cr)** 表示设置成功

参数说明：**B** 代表编码器的工作模式，值为 0：工作模式 0；值为 1：工作模式 1

应用举例： 用户命令（字符格式） **\$0131**

模块应答（字符格式） **! 01(cr)**

说明：设置编码器为工作模式 1，两路高速计数器模式。

### 2、读取编码器的工作模式

说明：读取编码器的工作模式。

命令格式：**\$AA4** 读取编码器的工作模式。

参数说明：**AA** 模块地址，取值范围 00~FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

应答格式：**! B (cr)** 代表 8 个编码器通道的工作模式，8 个数，排列顺序为编码器 7~编码器 0，  
值为 0：工作模式 0；值为 1：工作模式 1

应用举例： 用户命令（字符格式） **\$014**

模块应答（字符格式） **! 1 (cr)**

说明：编码器为工作模式 1

### 3、读取开关状态命令

说明：从模块中读回所有编码器输入通道开关量状态。

命令格式：**#AA(cr)**

参数说明：**#** 分界符。十六进制为 23H

**AA** 模块地址，取值范围 00~FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

应答格式：**> CC (cr)** 命令有效。

**?01(cr)** 命令无效或非法操作。

参数说明: > 分界符。十六进制为 3EH

**CC** 代表读取到的编码器输入开关状态, 8 个数, 排列顺序为 B0A0,

值为 0: 输入低电平; 值为 1: 输入高电平

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应用举例: 用户命令 (字符格式) **#01**

模块应答 (字符格式) **>01(cr)**

说 明: 模块输入开关状态是 **01**, 排列顺序为 B0A0

A0: 高电平 B0: 低电平

#### 4、读编码器计数器数据命令 (工作模式 0)

说 明: 读取编码器计数器的数据。 '+' 表示正转, '-' 表示反转。

命令格式: **#AA2**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

**2** 表示读编码器计数器数据命令。

应答格式: **!+AAAAAAAAAA (cr)**

应答格式: **!+AAAAAAAAAA(cr)**

应用举例: 用户命令 (字符格式) **#012**

模块应答 (字符格式) **!+0012345678 (cr)**

说 明: 编码器的计数值为正转+12345678

#### 5、读编码器输入频率命令 (工作模式 0)

说 明: 读取编码器输入的频率。 '+' 表示正转, '-' 表示反转。

命令格式: **#AA3**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

**3** 表示读编码器输入频率命令。

应答格式: **!+AAAAAA.AA (cr)**

应答格式: **!+AAAAAA.AA (cr)**

应用举例: 用户命令 (字符格式) **#013**

模块应答 (字符格式) **!+001000.00 (cr)**

说 明: 编码器的输入频率值为正转+1KHz。

#### 6、读编码器输入转速命令 (工作模式 0)

说 明: 读取编码器输入的转速。 '+' 表示正转, '-' 表示反转。

命令格式: **#AA4**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

**4** 表示读编码器0~编码器7输入转速命令。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!+AAAAA (cr)**

应用举例: 用户命令 (字符格式) **#014(cr)**

模块应答 (字符格式) **!+01000 (cr)**

说明: 编码器的输入转速值为正转+1000 转。

### 7、修改编码器计数器的数值命令 (工作模式 0)

说明: 修改编码器计数器的值, 也可以设置为零重新计数。

命令格式: **\$AA1+AAAAAAAA** 修改编码器的计数值。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AA(cr)** 表示设置成功

应用举例 1: 用户命令 (字符格式) **\$011+0**

模块应答 (字符格式) **!01(cr)**

说明: 设置编码器的计数值为 0。

应用举例 2: 用户命令 (字符格式) **\$011+3000**

模块应答 (字符格式) **!01(cr)**

说明: 设置编码器的计数值为+3000。

### 8、设置编码器的每转脉冲数 (工作模式 0)

说明: 设置编码器的每转脉冲数。根据接入的编码器参数来设定, 出厂默认值为 1000, 设置正确的脉冲数后才可以读出编码器转速。

命令格式: **\$AA5AAAAA** 设置编码器的每转脉冲数。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

**5** 设置编码器的每转脉冲数命令。

**AAAAA** 代表脉冲数, 如1000, 800或者600等。

应答格式: **!AA(cr)** 表示设置成功

应用举例: 用户命令 (字符格式) **\$01500300**

模块应答 (字符格式) **!01(cr)**

说明: 设置编码器的每转脉冲数为 300。

### 9、读取编码器的每转脉冲数 (工作模式 0)

说明: 读取所有编码器的每转脉冲数。

命令格式: **\$AA6** 读取编码器的每转脉冲数。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII 码。如地址01换成十六进制为30H和31H。

应答格式: **!AAAAA (cr)** 表示编码器的每转脉冲数。

应用举例: 用户命令 (字符格式) **\$016**

模块应答 (字符格式) **!01000 (cr)**

说明: 编码器的每转脉冲数都是 1000。

### 10、读计数器数据命令 (工作模式 1)

说明: 读取计数器的数据, 可以读所有通道, 也可以读单通道。

命令格式: **#AA5**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII



码。如地址01换成十六进制为30H和31H。

**5** 表示读通道A0~通道B0计数器数据命令。排列顺序A0,B0。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AAAAAAAAAA, AAAAAAAAAA (cr)**

命令格式: **#AA5N**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

**5** 表示读计数器数据命令。

**N** 表示读通道N计数器数据命令。N取值: 01,对应A0~B0

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AAAAAAAAAA(cr)**

应用举例 1: 用户命令 (字符格式) **#015**

模块应答 (字符格式) **!0012345678, 0012345678 (cr)**

说明: 所有通道的计数值为 12345678。

应用举例 2: 用户命令 (字符格式) **#0151**

模块应答 (字符格式) **!0012345678(cr)**

说明: 通道 B0 的计数值为 12345678。

## 11、读输入频率命令 (工作模式 1)

说明: 读取输入的频率, 可以读所有通道, 也可以读单通道。

命令格式: **#AA6**

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

**6** 表示读通道A0~通道B0输入频率命令。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AAAAAA.AA,AAAAAA.AA (cr)**

命令格式: **#AA6N** 读通道N输入频率。

**AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为01, 转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

**6** 表示读输入频率命令。

**N** 表示读通道N输入频率命令。N取值: 01,对应A0~B0

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AAAAAA.AA (cr)**

应用举例 1: 用户命令 (字符格式) **#016**

模块应答 (字符格式) **!001000.00,001000.00 (cr)**

说明: 所有通道的输入频率值为 1KHz。

应用举例 2: 用户命令 (字符格式) **#0160(cr)**

模块应答 (字符格式) **!001000.00(cr)**

说明: 通道 A0 的输入频率值为 1KHz。

## 12、修改 DI 计数器的数值命令 (工作模式 1)

说明: 修改 DI 计数器的值, 也可以设置为零重新计数。

命令格式: **\$AA2N+AAAAAAAA** 修改计数器 N 的计数值, N 为计数器代号, 取值 0 或 1, 对应 A0~B0, 设置 N 为 'M' 时表示同时设置所有通道的计数值。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为 01, 转换成十六进制为每个字符的 ASCII 码。如地址 01 换成十六进制为 30H 和 31H。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **! AA(cr)** 表示设置成功

应用举例 1: 用户命令 (字符格式) **\$0121+0**

模块应答 (字符格式) **! 01(cr)**

说明: 设置通道 B0 的计数值为 0。

应用举例 2: 用户命令 (字符格式) **\$012M+0**

模块应答 (字符格式) **! 01(cr)**

说明: 设置所有通道的计数值为 0。

应用举例 3: 用户命令 (字符格式) **\$012M+3000**

模块应答 (字符格式) **! 01(cr)**

说明: 设置所有通道的计数值为 +3000。

## 13、设置 DI 计数器的计数方式 (工作模式 1)

说明: 设置 DI 计数器是上升沿计数还是下降沿计数。出厂设置为 00。默认是上升沿计数 设置在模块重启后生效。

命令格式: **\$AA7BB** 设置 DI 计数器的计数方式。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为 01, 转换成十六进制为每个字符的 ASCII 码。如地址 01 换成十六进制为 30H 和 31H。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **! AA(cr)** 表示设置成功

参数说明: **BB** 代表通道状态, 2 个数, 排列顺序为 B0A0,

值为 0: 该通道上升沿计数; 值为 1: 该通道下降沿计数

应用举例: 用户命令 (字符格式) **\$01711**

模块应答 (字符格式) **! 01(cr)**

说明: 设置 B0~A0 通道下降沿计数。

## 14、读取 DI 计数器的计数方式 (工作模式 1)

说明: 读取 DI 计数器是上升沿计数还是下降沿计数。

命令格式: **\$AA8(cr)** 读取 DI 计数器的计数方式。

参数说明: **AA** 模块地址, 取值范围 00~FF(十六进制)。出厂地址为 01, 转换成十六进制为每个字符的 ASCII 码。如地址 01 换成十六进制为 30H 和 31H。

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **! BB (cr)** 表示 DI 计数器的计数方式。

参数说明: **BB** 代表通道状态, 排列顺序为 B0A0,

值为 0: 该通道上升沿计数; 值为 1: 该通道下降沿计数

应用举例: 用户命令 (字符格式) **\$018(cr)**

模块应答 (字符格式)      **! 11 (cr)**

说 明: B1~A0 通道下降沿计数。

### 15、读 DI 输入转速命令 (工作模式 1)

说 明: 读取 DI 输入的转速, 可以读所有 DI, 也可以读单路 DI。

命令格式: **#018** 读 DI0~DI7 输入转速。

应答格式: **!AAAAA,AAAAA (cr)**

命令格式: **#018N** 读 DI 通道 N 输入转速

应答格式: **!AAAAA (cr)**

应用举例 1: 用户命令 (字符格式)      **#018**

模块应答 (字符格式)      **!01000,01000 (cr)**

说 明: 所有 DI 通道的输入转速值为 1000 转。

应用举例 2: 用户命令 (字符格式)      **#0180**

模块应答 (字符格式)      **!01000(cr)**

说 明: DI0 的输入转速值为 1000 转。

### 16、设置 DI 的每转脉冲数 (工作模式 1)

说 明: 设置 DI 的每转脉冲数。根据接入 DI 的设备参数来设定, 出厂默认值为 1000, 设置正确的脉冲数后才可以读出 DI 转速。

命令格式: **\$01DWNAAAAA** 设置DI通道N的每转脉冲数。N为计数器代号,取值0或1,对应A0~B0, **AAAAA**代表脉冲数, 如1000, 800或者600等。

应答格式: **! 01(cr)** 表示设置成功

应用举例: 用户命令 (字符格式)      **\$01DW100300**

模块应答 (字符格式)      **! 01(cr)**

说 明: 设置 DI1 的每转脉冲数为 300。

### 17、读取 DI 的每转脉冲数 (工作模式 1)

说 明: 读取所有 DI 通道的每转脉冲数。

命令格式: **\$01DR** 读取所有 DI 的每转脉冲数, 排列顺序 A0~B0。

应答格式: **!AAAAA,AAAAA**

表示 DI0~DI1 的每转脉冲数。

应用举例: 用户命令 (字符格式)      **\$01DR**

模块应答 (字符格式)      **! 01000, 01000 (cr)**

说 明: 所有 DI 通道的每转脉冲数都是 1000。

### 18、设置 DI 的滤波时间 (工作模式 1)

说 明: 设置 DI 的滤波时间。单位 mS, 出厂默认是 0。光电开关输入设置为 0, 机械开关或者继电器输入建议设置为 20~100mS。设置重启后生效。

命令格式: **\$01LWNAAAAA** 设置DI通道N的每转脉冲数。N为计数器代号,取值0或1,对应A0~B0, **AAAAA**代表滤波时间, 如0, 20或者50等。

应答格式: **! 01(cr)** 表示设置成功

应用举例: 用户命令 (字符格式)      **\$01LW100020**

模块应答 (字符格式)           **! 01(cr)**

说    明: 设置 DI1 的滤波时间为 20mS。

### 19、读取 DI 的滤波时间 (工作模式 1)

说    明: 读取所有 DI 通道的滤波时间。

命令格式: **\$01LR**  读取所有 DI 的滤波时间, 排列顺序 A0~B0。

应答格式: **! AAAAA, AAAAA**  表示 DI0~DI1 的滤波时间。

应用举例:  用户命令 (字符格式)           **\$01LR**

          模块应答 (字符格式)           **! 00020, 00020 (cr)**

说    明: 所有 DI 通道的滤波时间都是 20mS。

### 20、设置计数值断电是否自动保存

说    明: 设置计数值断电是否自动保存, 出厂默认值为 0 (不自动保存, 断电清零)。

命令格式: **\$01SW**

参数说明: **S**  设置计数值断电是否自动保存命令。

**W**  0: 不自动保存, 断电清零;  1: 断电自动保存DI计数值。

应答格式: **! 01(cr)**  表示设置成功

应用举例:  用户命令 (字符格式)           **\$01S0**

          模块应答 (字符格式)           **! 01(cr)**

说    明: 设置 DI 不保存计数值, 断电后自动清零计数。

### 21、设置 DI 的上拉开关

说    明: 设置 DI 的上拉开关, 出厂默认值为 0 (DI 关闭上拉功能)。

命令格式: **\$01QX**

参数说明: **Q**  设置DI和DO的上拉开关命令。

**X**  0: DI关闭上拉电压;  1: DI接通上拉电压。

应答格式: **! 01(cr)**  表示设置成功

应用举例:  用户命令 (字符格式)           **\$01Q1**

          模块应答 (字符格式)           **! 01(cr)**

说    明: 设置 DI 接通上拉电压。DI 是 NPN 输入时可以设置为接通 DI 上拉电压。

### 22、配置 YL150 模块命令

说    明: 对一个 YL150 模块设置地址, 波特率, 校验和状态。配置信息储存在非易失性存储器 EEPROM 里。

命令格式: **%AANNTCCFF(cr)**

参数说明: **%**  分界符。

**AA**  模块地址, 取值范围 00 ~ FF(十六进制)。

**NN**  代表新的模块 16 进制地址, 数值 NN 的范围从 00 到 FF。

**TT**  用 16 进制代表类型编码。YL150 产品必须设置为 00。

**CC**  用 16 进制代表波特率编码。

波特率代码	波特率
04	2400 baud
05	4800 baud
06	9600 baud
07	19200 baud
08	38400 baud
09	57600 baud
0A	115200 baud

表 2 波特率代码

**FF** 用 16 进制的 8 位代表数据格式, 校验和。注意从 bits2 到 bits5 不用必须设置为零。

Bit7	Bit 6	Bit 5	Bit 4	Bit 3	Bit2	Bit 1	Bit 0
------	-------	-------	-------	-------	------	-------	-------

表 3 数据格式, 校验和代码

**Bit7:** 保留位, 必须设置为零

**Bit6:** 校验和状态, 为 0: 禁止; 为 1: 允许

**Bit5-bit2:** 不用, 必须设置为零。

**Bit1-bit0:** 数据格式位。 00: 工程单位(Engineering Units)

10: 16 进制的补码(Twos complement)

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

应答格式: **!AA(cr)** 命令有效。

**?AA(cr)** 命令无效或非法操作, 或在改变波特率或校验和前, 没有安装配置跳线。

参数说明: **!** 分界符, 表示命令有效。

**?** 分界符, 表示命令无效。

**AA** 代表输入模块地址

**(cr)** 结束符, 上位机回车键, 十六进制为 0DH。

其他说明: 假如你第一次配置模块, AA=00、 NN 等于新的地址。假如重新配置模块改变地址、输入范围、数据格式, AA 等于当前已配置地址, NN 等于当前的或新的地址。假如要重新配置模块改变波特率或校验和状态, 则必须安装配置跳线, 使模块进入缺省状态, 此时模块地址为 00H, 即 AA=00H, NN 等于当前的或新的地址。

假如格式错误或通讯错误或地址不存在, 模块不响应。

应用举例: 用户命令 **%0011000600(cr)**

模块应答 **!11(cr)**

说明: **%** 分界符。

**00** 表示你想配置的YL150模块原始地址为00H。

**11** 表示新的模块 16 进制地址为 11H。

**00** 类型代码, YL150 产品必须设置为 00。

**06** 表示波特率 9600 baud。

**00** 表示数据格式为工程单位, 禁止校验和。

### 23、读配置状态命令

说明：对指定一个 YL150 模块读配置。

命令格式：**\$AA2(cr)**

参数说明：**\$** 分界符。

**AA** 模块地址，取值范围 00 ~ FF(十六进制)。

**2** 表示读配置状态命令

**(cr)** 结束符，上位机回车键，十六进制为 0DH。

应答格式：**!AATTCCFF(cr)** 命令有效。

**?AA(cr)** 命令无效或非法操作。

参数说明：**!** 分界符。

**AA** 代表输入模块地址。

**TT** 代表类型编码。

**CC** 代表波特率编码。见表 2

**FF** 见表 3

**(cr)** 结束符，上位机回车键，十六进制为 0DH。

其他说明：假如格式错误或通讯错误或地址不存在，模块不响应。

应用举例： 用户命令 **\$302(cr)**

模块应答 **!30000600(cr)**

说明：**!** 分界符。

**30** 表示YL150模块地址为30H。

**00** 表示输入类型代码。

**06** 表示波特率 9600 baud。

**00** 表示禁止校验和。

### 24、设置以上字符命令设置的所有参数恢复出厂设置。

说明：设置模块用以上字符命令设置的参数恢复为出厂设置，完成后模块自动重启。

命令格式：**\$AA900** 设置参数恢复出厂设置。

参数说明：**AA** 模块地址，取值范围 00 ~ FF(十六进制)。出厂地址为01，转换成十六进制为每个字符的ASCII码。如地址01换成十六进制为30H和31H。

**(cr)** 结束符，上位机回车键，十六进制为 0DH。

应答格式：**!AA(cr)** 表示设置成功，模块会自动重启。

应用举例： 用户命令 (字符格式) **\$01900**

模块应答 (字符格式) **!01(cr)**

说明：参数恢复出厂设置。

### Modbus RTU 通讯协议：

模块的出厂初始设置，如下所示：

**Modbus 地址为 01**

**波特率 9600 bps**

**数据格式：10 位。1 位起始位，8 位数据位，1 位停止位。无校验。**

**让模块进入缺省状态的方法：**

YL150模块都有一个特殊的标为INIT的管脚。将INIT管脚短路接到GND管脚后，再接通电源，此时模块进入缺省状态。在这个状态时，模块暂时恢复为默认的状态：地址为01，波特率为9600。在不确定某个模块的具

体配置时, 用户可以查询地址和波特率的寄存器40201-40202, 得到模块的实际地址和波特率, 也可以跟据需要修改地址和波特率。

支持Modbus RTU通讯协议, 命令格式按照标准Modbus RTU通讯协议。

### YL150 的寄存器地址说明

支持功能码01, 05和15的寄存器

地址 0X (PLC)	地址 (PC, DCS)	数据内容	属性	数据说明
00001	0000	A0 的计数方式	读/写	通道 A0 ~ B0 的计数方式 (默认值为 0) 0 为上升沿计数, 1 为下降沿计数 设置在模块重启后生效。 正常不用修改, 使用默认值即可。
00002	0001	B0 的计数方式	读/写	
00033	0032	A0 输入的开关量	只读	编码器输入点的电平状态 0 表示低电平输入, 1 表示高电平输入
00034	0033	B0 输入的开关量	只读	

支持功能码03, 06和16的寄存器

地址 4X (PLC)	地址 (PC, DCS)	数据内容	属性	数据说明
40001	0	编码器工作模式	读/写	编码器工作模式, 整数, 0 或 1, 出厂默认为 0 (修改后需 <b>重启</b> 才生效) <b>工作模式 0</b> : 编码器 AB 信号输入 <b>工作模式 1</b> : 两路独立的计数器输入 下面寄存器备注 ( <b>工作模式 0</b> ) 的表示是仅在 编码器工作模式为 0 时数据才有效。备注 ( <b>工 作模式 1</b> ) 的表示是仅在编码器工作模式为 1 时数据才有效。
40017~40018	16~17	编码器计数	读/写	
40033~40034	32~33	通道 A0 计数	读/写	
40035~40036	34~35	通道 B0 计数	读/写	
				编码器计数器 ( <b>工作模式 0</b> ) 数据为有符号的长整数, 16 进制格式, 负数采 用的是补码 (two's complement) , 正数 (0x00000000~0x7FFFFFFF) , 负数 (0xFFFFFFFF~0x80000001) , 计数器清零直接向对应寄存器写入 0, 也可以根据需要写入其他值。 低 16 位在寄存器 40017, 高 16 位在寄存器 40018
				通道 A0~B7 计数器 ( <b>工作模式 1</b> ) 数据为无符号的长整数, 16 进制格式, (0x00000000~0xFFFFFFFF) , 通道 A0 低 16 位在寄存器 40033, 通道 A0 高 16 位在寄存器 40034, 其他通道同样规律。 计数器清零直接向对应寄存器写入 0, 也可以 根据需要写入其他值。

地址 4X (PLC)	地址 (PC, DCS)	数据内容	属性	数据说明
40041	40	通道 A0 的脉冲数	读/写	通道 A0~B0 的脉冲数 <b>(工作模式 1)</b> 无符号整数 (出厂默认值为 60), 根据输入信号的每转脉冲数来设定, 设置后寄存器 40109~40110 就是对应通道的转速。
40042	41	通道 B0 的脉冲数	读/写	
40068	0067	计数清零寄存器	写	无符号整数, 默认为 0, 修改这个寄存器用于清零编码器计数器或通道计数器。修改后寄存器会自动恢复为 0。 <b>写入 10: 设置编码器计数值为 0,</b> 写入 20: 设置通道 A0 计数值为 0, 写入 21: 设置通道 B0 计数值为 0, <b>写入 22: 设置通道 A0 和 B0 计数值为 0。</b> 写入其他值无效。
40073	72	编码器 0 的脉冲数	读/写	编码器的脉冲数 <b>(工作模式 0)</b> 无符号整数 (出厂默认值为 1000), 根据编码器每转脉冲数来设定, 设置后寄存器 40101 就是对应通道的转速。
40081	80	计数值自动保存	读/写	0: 不自动保存, 断电清零; 1: 断电自动保存计数值。(默认值为 1)
40082	81	DI 的上拉开关	读/写	0: DI 关闭上拉电压; (默认值为 0) 1: DI 接通上拉电压。
40089	88	参数恢复出厂设置	读/写	设置为 FF00, 则模块所有寄存器的参数恢复为出厂设置, 完成后模块自动重启
40101	100	编码器的转速	只读	编码器的转速 <b>(工作模式 0)</b> 有符号整数, 正负表示正反转。 转速是根据寄存器 40073 设定的脉冲数换算得到。
40109	108	通道 A0 的转速	只读	通道的转速 <b>(工作模式 1)</b> 无符号整数。 转速是根据寄存器 40041~40042 设定的脉冲数换算得到。
40110	109	通道 B0 的转速	只读	
40129~40130	128~129	编码器的频率	只读	编码器的脉冲频率 <b>(工作模式 0)</b> 数据为 32 位浮点数 浮点数低 16 位在寄存器 40129 浮点数高 16 位在寄存器 40130
40145~40146	144~145	通道 A0 的频率	只读	通道的脉冲频率 <b>(工作模式 1)</b> 数据为 32 位浮点数 通道 A0 浮点数低 16 位在寄存器 40145 通道 A0 浮点数高 16 位在寄存器 40146 通道 B0 类似
40147~40148	146~147	通道 B0 的频率	只读	
40181	180	通道 A0 的滤波时间	读/写	通道的滤波时间 <b>(工作模式 1)</b> 无符号整数。单位 mS, 光电开关输入设置为 0, 机械开关或者继电器输入建议设置为 20~100mS。设置重启后生效。
40182	181	通道 B0 的滤波时间	读/写	
40201	0200	模块地址	读/写	整数, 重启后生效, 范围 0x0000-0x00FF
40202	0201	波特率	读/写	整数, 重启后生效, 范围 0x0004-0x000A 0x0004 = 2400 bps, 0x0005 = 4800 bps 0x0006 = 9600 bps, 0x0007 = 19200 bps 0x0008 = 38400 bps, 0x0009 = 57600 bps 0x000A = 115200bps
40211	0210	模块名称	只读	高位: 0x01 低位: 0x50

表 5 Modbus Rtu 寄存器说明



**通讯举例 1:** 假如模块地址为 01, 以 16 进制发送: **010300100002C5CE**, 即可取得寄存器的数据。

1	3	0	10	0	2	C5	CE
模块地址	读保持寄存器	寄存器地址高位	寄存器地址低位	寄存器数量高位	寄存器数量低位	CRC 校验低位	CRC 校验高位

假如模块回复: **010304CA90FFFC476** 即读到的数据为 0xFFFFCA90, 换成 10 进制为-13680, 即表明现在编码器 0 的计数值为-13680。

1	3	4	CA	90	FF	FF	C4	76
模块地址	读保持寄存器	数据的字节数	数据 1 高位	数据 1 低位	数据 2 高位	数据 2 低位	CRC 校验低位	CRC 校验高位

**通讯举例 2:** 假如模块地址为 01, 以 16 进制发送: **010300200002C5C1**, 即可取得寄存器的数据。

01	03	00	20	00	02	C5	C1
模块地址	读保持寄存器	寄存器地址高位	寄存器地址低位	寄存器数量高位	寄存器数量低位	CRC 校验低位	CRC 校验高位

假如模块回复: 010304CA90FFFC476 即读到的数据为 0xFFFFCA90, 换成 10 进制为 4294953616, 即表明现在通道 A0 的计数值为 4294953616。

1	3	0	20	0	2	C5	C1
模块地址	读保持寄存器	寄存器地址高位	寄存器地址低位	寄存器数量高位	寄存器数量低位	CRC 校验低位	CRC 校验高位

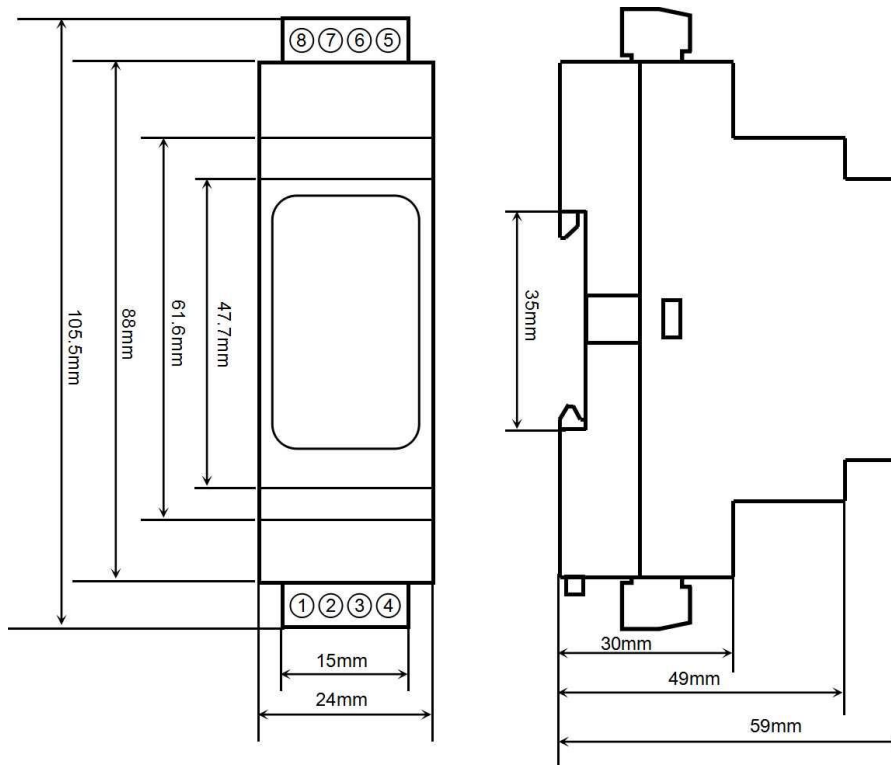
**通讯举例 3:** 假如模块地址为 01, 以 16 进制发送: **01060043000AF819**, 即清零编码器 0 的计数值。

1	6	0	43	0	0A	F8	19
模块地址	写单个保持寄存器	寄存器地址高位	寄存器地址低位	数据高位	数据低位	CRC 校验低位	CRC 校验高位

假如模块回复: 01060043000AF819 即表示设置成功, 编码器 0 的计数值修改为 0。

1	6	0	43	0	0A	F8	19
模块地址	写单个保持寄存器	寄存器地址高位	寄存器地址低位	数据高位	数据低位	CRC 校验低位	CRC 校验高位

外形尺寸: (单位: mm)



可以安装在标准 DIN35 导轨上

**保修:**

本产品自售出之日起两年内, 凡用户遵守贮存、运输及使用要求, 而产品质量低于技术指标的, 可以返厂免费维修。因违反操作规定和要求而造成损坏的, 需交纳器件费用和维修费。

**版权:**

版权 © 2021 深圳市俞霖科技开发有限公司。

如未经许可, 不得复制、分发、翻译或传输本说明书的任何部分。本说明书如有修改和更新, 恕不另行通知。

**商标:**

本说明书提及的其他商标和版权归各自的所有人所有。

版本号: V1.0

日期: 2021 年 06 月