



使用手册

UID820 CAN2.0B 总线通讯 数字 I/O 及 PWM 输入输出控制器



UID820 数字 I/O 及 PWM 输入输出控制器

[知识产权保护声明]

使用UIROBOT产品前请注意以下三点：

- UIROBOT的产品均达到UIROBOT使用手册中所述的技术功能要求。
- UIROBOT愿与那些注重知识产权保护的客户合作。
- 任何试图破坏UIROBOT器件代码保护功能的行为均可视为违反了知识产权保护法案和条例。如果这种行为导致在未经UIROBOT授权的情况下，获取软件或其他受知识产权保护的成果，UIROBOT有权依据该法案提起诉讼制止这种行为。

[免责声明]

本使用手册中所述的器件使用信息及其他内容仅为您提供便利，它们可能在未来版本中被更新。确保应用符合技术规范，是您自身应负的责任。UIROBOT对这些信息不作任何形式的声明或担保，包括但不限于使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。UIROBOT对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将UIROBOT器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障UIROBOT免于承担法律责任和赔偿。未经UIROBOT同意，不得以任何方式转让任何许可证。

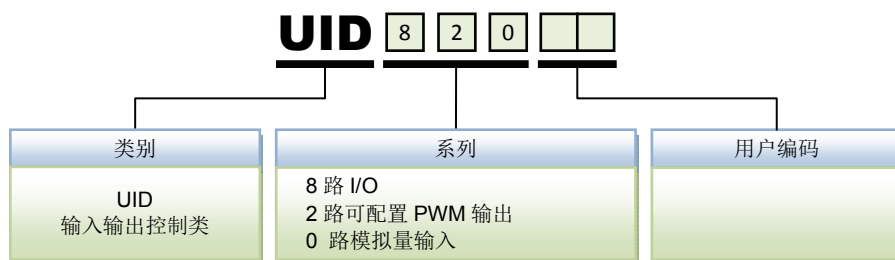
[商标和外观设计声明]

UIROBOT 的名称和徽标组合为 UIROBOT Ltd.在中国和其他国家或地区的注册商标。
UIROBOT的UIM24XXX系列步进电机（控制）驱动器和UIM25XX系列转换控制器外观设计均以申请专利保护。

[UID820 产品订购说明]

在订购 UID280 产品时请按以名下格式提供产品号，以便我们准确及时地为您提供产品：

UID820 产品牌号



UID820 数字 I/O 及 PWM 输入输出控制器

多功能 IO 端口特性

- 单机 8 路多功能 TTL 数字量输入 / 输出，可指令配置，实时调整
- 单机 2 路独立 PWM 输出，可指令配置，实时调整，基频 50Hz~10000Hz，分辨率 0.5%
- 最大 800 路联网 TTL 数字量输入/输出（100 个 UID820 组网）
- 最大 200 路联网独立 PWM 输出（100 个 UID820 组网）
- 可与 UIM242 步进电机控制器混合组网
- 配合相应的继电器模块，可对开关阀，比例阀和直流电机的执行器件进行控制
- 配合相应的电平转换模块，可接受多种行程开关，限位开关和传感器输入

嵌入式微处理机

- 内置高性能嵌入式微处理器系统
- 指令结构简单直观
- 免费提供基于 MS Windows 的 VC/VB 源代码

CAN2.0B 通讯特性

- 主动 CAN 2.0B, 全网络仅用一对双绞线（两根导线）
- 1 百万通讯比特率，10 公里通讯距离
- 可连接节点高达 100 个
- 采用差分总线，具有很强的抗噪特性

电气特性

- 宽电压输入 6~40VDC

简介

UID820 是 CAN 总线系列数字端口输入输出控制器，采用优爱宝的 simpleCAN 通讯协议。它具备 8 个方向和功能可指令配置的数字 I/O 端口。其中两个可配置为 PWM（脉宽调制）输出。PWM 波形的基频和占空比均可经由指令实时调整。UID820 可与 UIM242 系列步进电机控制驱动器混合组成控制局域网络，亦可单独组成数据采集和输入输出控制网络。

对于熟悉 CAN 通讯协议并具备 CAN 主控机的用户，可直接采用优爱宝提供的 simpleCAN 协议实现对整个电机-传感器-电磁阀网络的控制。

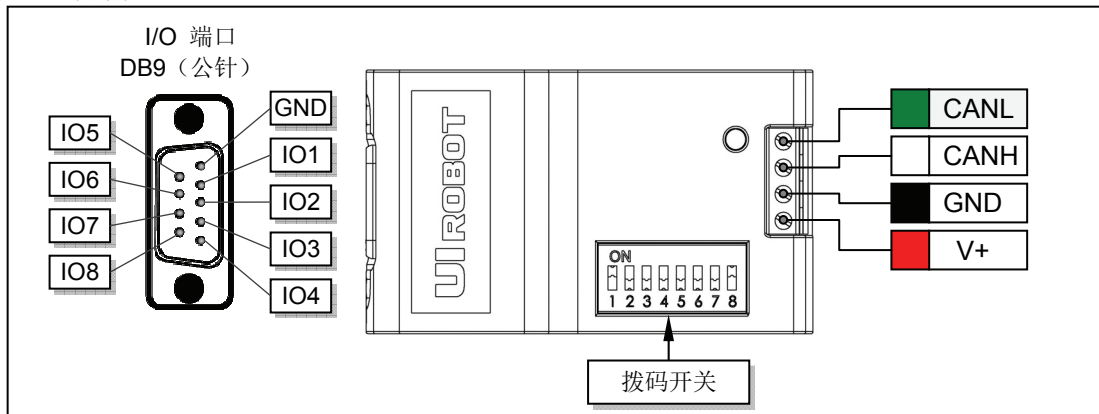
对于不想了解 CAN 协议的用户，可采用优爱宝的 UIM2501 CAN/RS232 转换器，通过基于 RS232 的简单直观的指令来控制整个电机-传感器-电磁阀网络，免去了用户直接使用 CAN 协议时面临的一系列困难。一台用户上位机只需一个 UIM2501，就可同时控制最多 100 台 UID820 数据采集器与 UIM242 驱动器。

UID820 的控制指令结构简单，高容错。数据采集器外壳为全铝合金铸件，坚固耐用，散热性能好。

UID820 数字 I/O 及 PWM 输入输出控制器

接线端口

图 0-1: 接线端口



控制端口

端口	符号	说明
1	V+	工作电压正极。电压：6 - 40V 直流
2	GND	工作电压地线，即 0V（工作电压正负极不可接错）
3	CANH	CAN 总线的高位线
4	CANL	CAN 总线的低位线

DB9 I/O 端口

标准 DB9 接插口（公针），其中 DB9 的第 1 针为地，第 2 针到第 9 针分别对应 8 路输入/输出的第 1 路到第 8 路，即 DB9 的第 2 针为第 1 路输入/输出，第 3 针为第 2 路输入/输出，依次类推。

拨码开关

UID820 带有一个 8 位的多功能拨码开关。拨码 1 到拨码 7 在上电时，提供了 UID820 的地址标识码。拨码 8 置于 ON 时（图示位置），UID820 内置的 CAN 总线终端电阻（Terminating Resistor）被使能。出厂时，UID820 地址标识码设置为 7，同时终端电阻未使能。



警告：除非 UID820 位于总线最末端，否则切勿使能终端电阻！

用户通过调整拨码开关定义 UID820 定义地址标识码。地址设定采用二进制：

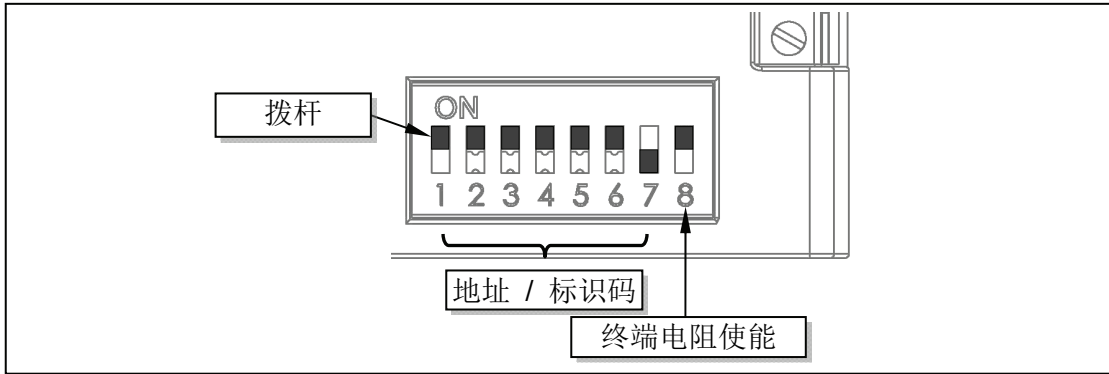
$$\text{地址/标识码} = D1 \cdot 2^0 + D2 \cdot 2^1 + D3 \cdot 2^2 + D4 \cdot 2^3 + D5 \cdot 2^4 + D6 \cdot 2^5 + D7 \cdot 2^6$$

当拨码开关 Dx ($x = 1, 2, \dots, 7$) 拨到“ON”端时， $Dx = 1$ ；反之， $Dx = 0$ 。例如当需要设定 地址 = 63 时：

$$63 = 1 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 1 \cdot 2^4 + 1 \cdot 2^5 + 0 \cdot 2^6$$

即 $D1, D2, D3, D4, D5, D6$ 拨到字母“ON”端， $D7$ 拨到数字端，如图 0-2 所示。

图 0-2：拨码开关设定

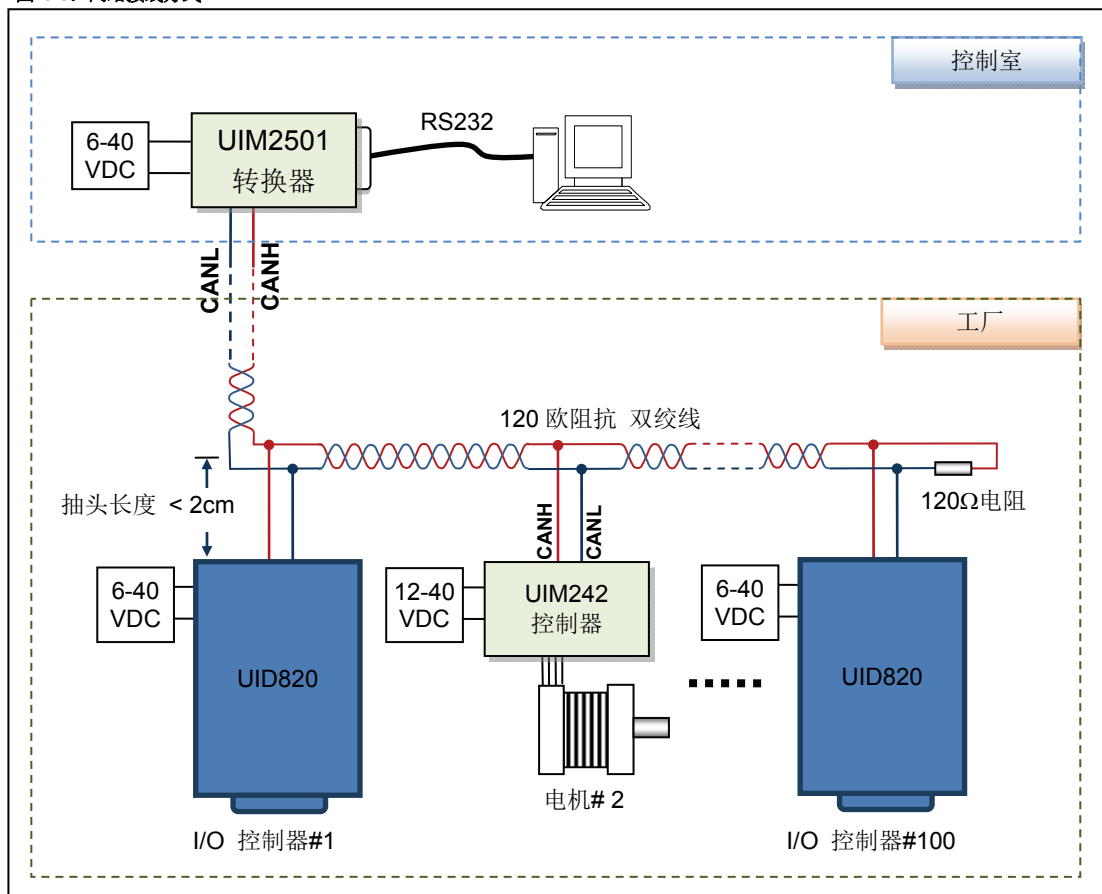


典型接线

CAN 总线提供了一个非常可靠和简洁的网络组建方案。如果采用一个 RS232/CAN 驱动转换器挂带多台数据采集器与电机驱动器的控制方式，可采用如下接线方式（图 0-3）。

注意：在多节点（即多台数据采集器/驱动器）的情况下，应该用一根双绞线将所有节点连接起来，如图 0-3 所示。一定要避免使用星形连接方式。同时，每个节点抽头线的长度不应超过 2 cm，且越短越好。在总线距离较长时，双绞线的两端应各连上一个 120 欧姆的终端电阻(Terminating Resistor)。特别是在总线距离超过 100 米时应考虑采用 CAN 总线专用的 120 欧姆阻抗屏蔽双绞线。

图 0-3: 网络接线方式



指令总表

指令	说明	反馈消息头	信息标识码
MCFG;	查询主配置寄存器	0xAA	0XB0
MCFG=X;	设置主配置寄存器	0xAA	0XB0
IOC;	查询端口功能和输入/输出方向	0xAA	0XC3
IOC =X;	设置端口功能和输入/输出方向	0xAA	0XC3
DVA;	查询端口输入逻辑值	0xAA	0XC5
DVA=X;	设置端口输出电平值	0xAA	0XC4
PMB;	查询 PWM 波形输出基频	0xAA	0XC7
PMB=X;	设置 PWM 波形输出基频	0xAA	0XC7
PMD;	查询 PWM 输出占空比	0xAA	0XC6
PMD=X;	设置 PWM 输出占空比	0xAA	0XC6
MDL=X;	查询站点 X 的型号和固件版本	0xCC	0xDE
BTR=X;	设置 CAN 通讯比特率	0xAA	0xBC
BTR;	查询当前 CAN 通讯比特率	0xAA	0xBC

UID820 数字 I/O 及 PWM 输入输出控制器

性能指标

绝对最大值（注 1）

供电电压.....	6V 至 40V
输入输出端口相对于 GND 的电压.....	-0.3V 至+5.3V
偏置电压下的环境温度.....	-40°C 至+85°C
储存温度.....	-50°C 至+150°C

注 1：如果器件工作条件超过上述“绝对最大值”，可能会对器件造成永久性损坏。上述值仅为运行条件极大值，建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下，其稳定性会受到影响。

工作电气性能（环境温度 25°C 时）

供电电压范围	6V - 40VDC
供电输入电流	100mA 最大
I/O 口输入低电平	最大 1.5V（1.5V 以下即认为低电平）
I/O 口输入高电平	最少 3.5V（3.5V 以上即认为高电平）
I/O 口输入电流	10uA 最大（即端口消耗电流；I/O 端口配置为输入时，内阻为 100K 以上）
I/O 口输出低电平	小于 0.6V
I/O 口输出高电平	大于 4.5V
I/O 口输出电流	10mA 最大（即端口驱动能力）
I/O 口短路电流	10mA
PWM 基频可调范围	50 Hz – 10K Hz
PWM 占空比分辨率	0.5%

通讯方式（环境温度 25°C 时）

通讯协议	主动 CAN 2.0B
物理连接	二线制，CANH、CANL，双绞线
CAN 总线驱动	• 支持 1 百万比特率的运行速率

使用环境及参数

冷却方式	自然冷却
使用场合	避免粉尘、油雾及腐蚀性气体
使用温度	-40 °C ~ 85 °C
使用湿度	<80%RH, 无凝露, 无结霜
使用震动	3G Max
保存温度	-50 °C ~ 150 °C

尺寸及重量

外形尺寸	66.4mm x 38mm x 18mm
重量	0.2kg

1.0 产品介绍

UID820 数字 I/O 及 PWM 输入输出控制器主要功能如下：

- 可与 UIM242XX 控制驱动器、UIM2501 转换器组成 CAN 网络，完成数字量的输入、输出。
- 2 可指令配置 PWM 波形的输出。
- 8 路可指令配置数字输入/输出。

配合 UIM2501 的使用，能让不了解 CAN 总线协议的用户能在享受 CAN 总线协议的高速（1Mbps）、长距离（10 公里）和高抗干扰等一系列优点的同时，还可以充分利用简单易用的通讯界面，进而能够专注于应用，提高设计效率缩短开发时间。

UID820 支持 1Mbps 的 CAN 通讯比特率。精简后的指令在 CAN 总线上的传输时间小于 0.1 毫秒，一般为 0.05 毫秒。通讯比特率可由用户指令修改，以适应不同距离的传输需要。

UID820 数据采集器内置嵌入式微处理系统。上位机（PC 机或控制设备）通过串行口连接到转换器与数据采集器后，向其送 ASCII 指令即可控制数据采集器的输入/输出。

指令结构简单，高容错。如果输入了错误指令，数据采集器将返回错误信息给上位机。错误指令不会被执行，避免发生事故。

该转换控制器可以使用 6V~40V 宽电压范围直流供电。

优爱宝公司同时免费提供基于 Microsoft Windows 的 VB/VC 软件控制步进电机的演示源代码和演示软件。

2.0 CAN2.0B 通讯设置

UID820 能够实现以下 CAN 通讯比特率（表 2-1），同时能够通过指令在这些 CAN 通讯比特率之间动态调整。调整波特率的目的主要是为了取得较长的通讯距离和稳定的工作状态。

用户可以用 BTR 编号来通知 UID820 切换通讯比特率。指令为 BTR = x，其中 x 就是上述通讯比特率编号。每个 UID820 出厂时 BTR 都设置为 1，即 800Kbps/50 米通讯距离。一般说来，每一条指令或者消息需要占用 64 – 128 个比特。

请注意，本文中所提到的反馈信息都是指 UID820 通过 UIM2501 发回的反馈信息。

表 2-1 CAN 总线通讯比特率

BTR 编号	比特率 (bps)	总线长度 (米)
0	1000K	25
1	800K	50
2	500K	100
3	250K	250
4	125K	500
5	50K	1000
6	20K	2500
7	10K	5000

2.1 设置CAN通讯比特率（BTR）

BTR=x;	
指令功能	设置全局网络的 CAN 通讯比特率 x。
变量范围	整数 x = 0, 1 ... 7
反馈信息	<u>0xAA</u> <u>BTR#</u> <u>0xBC</u> <u>0xFF</u>
相关备注	0xBC 是 CAN 比特率代码 的信息标识码。 BTR# 是当前的 CAN 比特率代码。

2.2 查询CAN通讯比特率

BTR;	
指令功能	查询当前网络的 CAN 通讯比特率。
变量范围	无
反馈信息	<u>0xAA</u> <u>BTR#</u> <u>0xBC</u> <u>0xFF</u>
相关备注	0xBC 是 CAN 比特率代码 的信息标识码。 BTR#是当前的 CAN 比特率代码。

3.0 指令和反馈信息结构

UID820 输入输出控制器接收客户机通过转换器 UIM2501 发来的操作信息（指令）并执行该信息所要求的操作；同时回复 ACK 信息（收到指令复述确认）；并且按客户要求返回目前各项操作参数。在没有收到客户机新的指令前，控制器完成指令后将保持现有工作状态。

对于使用自备 CAN 主机，而不用 UIM2501 RS232-CAN 协议转换器的用户，请参阅优爱宝 simpleCAN 协议编程手册来确定如何与 UID820 通讯。以下的内容都是针对使用 UIM2501 的用户而言的。

3.1 指令结构

指令是客户机向 UID820 发送的，指示完成一定功能的信息。UID820 接受的指令都遵循以下规则：

1. 单条指令总长度（包括结尾分号）不能超过 20 个字符。
2. 所有指令均为 7 位的标准 ASCII 码（1 - 127）。
3. 指令结构如下：

指令符 = 数值；或 指令符；

以上结构中，= 号可以省略。

其中，

指令符

由若干不间断的字母组成，不分大小写。

数值

由一串不间断的数字组成。有些指令没有数值，例如查询指令 SPD; STP; 等。

结束符

每句指令必须以分号，即“;”。其中的等号“=”不起作用，只是为便于读写而设置。用户可以使用冒号等代替，但不能使用“{”和“}”。

4. 指令被读入驱动器时，只有前面 3 个字母有效。比如以下指令是一样的：MCFG; 和 MCF;

3.2 反馈信息结构

反馈信息是控制驱动器向用户机发送的信息。该信息长度不固定，最大 13 字节。

UID820 通过 UIM2501 转换器发出的反馈信息使用如下结构：

消息头 驱动器标识码 信息标识码 信息数据 结束符

消息头

表示一条反馈信息的开始。有如下三种：

0xAA 表示指令确认反馈（ACK），是对收到的指令的一种重复。

0xCC 表示状态反馈，是对现状的描述。

0xEE 表示收到的信息有错误，不能被执行。

UID820 数字 I/O 及 PWM 输入输出控制器

驱动器标识码

表示当前驱动器在一个网络中的识别标号(又称站点)。对于 UIM820, 该标识范围: 5 – 125 。

信息标识码

标明了该条信息的属性。例如: 0xCC 0x05 0xE0 0xFF, 中的 0xE0 表示该信息是端口 1 出现下降沿。详细的内容在后面章节针对具体指令展开。

信息数据

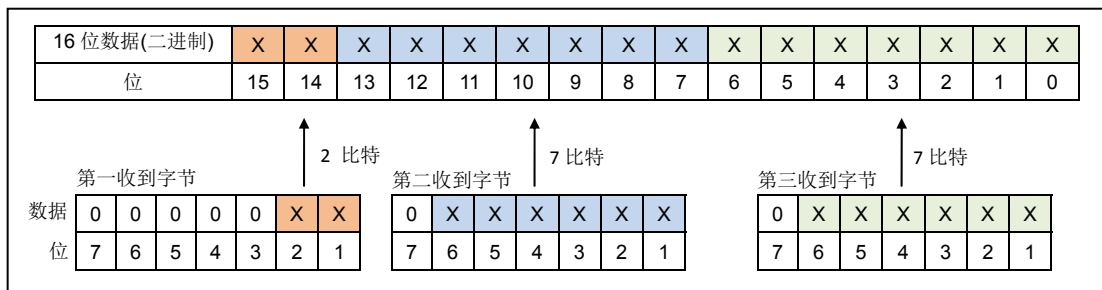
采用 7 位数据结构排列, 高位在先, 低位在后。图 2-1 展示了反馈信息中的 7 位数据字节通过移位操作转化为 16 位数据。16 位数据占用 3 个反馈数据字节。

结束符

标明一条信息的结束。UID820 控制器采用 0xFF 作为结束符。

注意, 有些反馈信息是没有信息数据的, 比如实时状态变化通知。

图 3-1: 3 个数据字节转化为 16 位数据



4.0 实时状态变化通知 RTCN

UID820 支持实时状态变化通知（即 RTCN，Real-Time Change Notification）。

与微处理器的中断相仿，实时状态变化通知是指在发生某个事件时，UID820 能够自动向用户机发送一个简短的信息。信息长度一般在 4 个字节左右。从事件发生到发送通知，时间小于 1 毫秒。传送的时间取决于通讯波特率。当 UIM2501 转换器以 115200 波特率的 RS232 传输时，传输时间小于 0.5 毫秒。所以从事件发生到用户机收到通知，时间大约 1.5 毫秒（CAN 总线传输时间小于 0.05 毫秒可以忽略）。

对于 UID820，该事件主要指当某端口配置为数字输入端口后，在端口上检测到电平出现上升沿或下降沿。

4.1 状态变化通知消息格式

当检测到输入电平发生时，UID820 自动给用户发送以下格式的实时反馈信息：

0xAA **[驱动器标识码]** **[信息标识码]** **0xFF**

驱动器标识码为驱动器的站点（由用户通过拨码设置）。

表 4-1: 实时状态变化通知事件

编号	事件名称	信息标识码	相关备注
1	IO 端口 n 下降沿	0xE0 + n*2	端口 n 检测到电平发生：高 >>>低 变化时
2	IO 端口 n 上升沿	0xE1 + n*2	端口 n 检测到电平发生：低 >>>高 变化时

注：上表中，n=1, 2,...8。

例如，当用户收到反馈信息 **0xAA 0x7 0xE8 0xFF** 时，则表示，站点为 7 的 UID820 检测到输入到端口 4 的电平发生了由高到低的变化（因为 $0xE8 = 0xE0 + 4*2$ ）。

4.2 使能 / 禁止状态变化通知

所有实时状态变化通知可被指令使能或者禁止。

使能和禁止是通过写主配置寄存器（MCFG 指令）的 **P1IE**（MCFG<0>）到 **P8IE**（MCFG<7>）位来实现的。注意，只有将端口配置为数字输入后，才能最终使用 RTCN 变化通知。对于输出端口是没有 RTCN 的。

4.3 主配置寄存器

UIM820 设有一个主配置寄存器用以使能或禁止系统的 RTCN 变化通知。一旦配置完成，主配置寄存器的参数会立刻生效。出于安全性考量，MCFG 不会保存在 EEPROM 中。每次开机必需重新设置。16 位组成，结构如下：

MCFG 寄存器位定义

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
定义	X	X	X	X	X	X	X	X	P8IE	P7IE	P6IE	P5IE	P4IE	P3IE	P2IE	P1IE

位 15- 8 保留，读作 0，写入被忽略。

位 7 - 0 **IO8 ~ IO1 变化通知使能开关**

0 = 禁止相应端口的 RTCN 变化通知

1 = 使能相应端口的 RTCN 变化通知

4.4 主配置寄存器设置指令 (MCFG)

MCFG = x;	
指令功能	设定主配置寄存器数值 x 。
变量范围	整数 $x = 0, 1 \dots 255$, 或者 16 进制数 $x = 0x0000 \dots 0x00FF$
ACK 信息	<u>0xAA</u> [UID820 标识码] <u>0xB0</u> <u>CFG2</u> <u>CFG1</u> <u>CFG0</u> <u>0xFF</u>
相关备注	<p>0xB0 是 MCFG 的信息标识码。</p> <p>CFG2 – CFG0 表示主配置寄存器数值，采用 7 位数据。转换方式参见图 3-1。</p> <p>如采用整数输入，用户需首先将上述主配置寄存器的各位填写 0 或者 1，然后再将得到的 16 位二进制数转成十进制数，然后将得到的十进制数填入语法结构中 x 的位置。如果采用 16 进制输入，则该 16 进制数必须以“0x”开头。</p>

4.5 查询主配置寄存器

MCFG;	
指令功能	查询当前主配置寄存器数值。
变量范围	无
ACK 信息	<u>0xAA</u> [UID820 标识码] <u>0xB0</u> <u>CFG2</u> <u>CFG1</u> <u>CFG0</u> <u>0xFF</u>
相关备注	参见上一节。

5.0 操作指令

用户可以将一个 UID820 作为单独器件进行操作，或者将一个或数个 UID820 接入 CAN 网络进行数字输入/输出控制。在上位机（主机）一侧，用户可以经由 UIM2501 RS232/CAN 转换器，通过基于 RS232 的指令来控制整个 CAN 网络中的每个 UID820 以及 UIM242 步进电机控制驱动器（如果有 UIM242 的话）。

依照 CAN 网络协议的规定，每个 UID820 必须有唯一的地址/标识码。标识码是上位机用来识别用户指令发往对象的依据，同时也是用户判断 UID820 数据采集器反馈来自何方的依据。如果一个步进电机控制网下属的两个或多个相同的标识码将导致错误。指定该唯一标识码的方法是通过拨码设置。详情见图 0-2。

这一章主要描述数字量输入/输出、PWM 输出设置方面的操作。

5.1 UID820 端口可配置功能

UID820 的多功能端口可由用户通过指令配置。各端口可用配置如下：

端口	TTL 输入	TTL 输入内部上拉	TTL 输出	PWM 输出	备注
IO1	✓	✓	✓	✗	
IO2	✓	✓	✓	✗	
IO3	✓	✓	✓	✗	
IO4	✓	✓	✓	✗	
IO5	✓	✓	✓	✗	
IO6	✓	✗	✓	✗	作为输入时，端口悬空会得到不可预知的结果
IO7	✓	✗	✓	✓	作为输入时，端口悬空会得到不可预知的结果
IO8	✓	✗	✓	✓	作为输入时，端口悬空会得到不可预知的结果

5.2 端口功能配置寄存器（IOC）

UID820 数据采集器设有一个寄存器用以设置端口数字量/模拟量的输入/输出。一旦配置完成，配置寄存器的参数会立刻生效并且自动存入 UID820 数据采集器上 EEPROM，掉电不会丢失。该存入过程不影响控制的实时性。主配置寄存器由 16 位组成，结构如下：

IOC 寄存器位定义

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
定义	IOC8		IOC7		IOC6		IOC5		IOC4		IOC3		IOC2		IOC1	

上表中 IOCn（n=1, 2...8）按一下方式决定端口的功能（即数字输入，数字输出或者 PWM 输出）。

IOCn = 00（二进制）时，端口 n 为 PWM 输出；

IOCn = 01（二进制）时，端口 n 为数字 TTL 电平输出；

IOCn = 11（二进制）时，端口 n 为数字 TTL 电平输入；

例如，要求：端口 1~5 为数字 TTL 电平输入，端口 6 为数字 TTL 电平输出，端口 7~8 为 PWM 输出时，

可如下配置 IOC 寄存器：

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
定义	00		00		01		11		11		11		11		11	

故此：IOC = 0000 0111 1111 1111（二进制）= 0x07FF（十六进制）

UID820 数字 I/O 及 PWM 输入输出控制器

5.3 端口功能配置指令 (IOCFG)

IOCFG = x; 或者 IOC = x;	
指令功能	设定端口功能配置寄存器 IOC 的数值 x。
变量范围	整数 x = 0、1 ... 65535, 或者 16 进制数 x = 0x0000 ... 0xFFFF
ACK 信息	0xAA [UID820 标识码] 0xC3 IOC2 IOC1 IOC0 0xFF
相关备注	0xC3 是 IOC 的信息标识码。 IOC2 – IOC0 表示端口配置寄存器数值, 采用 7 位数据结构, 转换方式参见图 3-1。 如采用整数输入, 用户需首先将上述主配置寄存器的各位填写 0 或者 1, 然后再将得到的 16 位二进制数转成十进制数, 然后将得到的十进制数填入语法结构中 x 的位置。如果采用 16 进制输入, 则该 16 进制数必须以“0x”开头。

5.4 查询功能配置寄存器

IOCFG; 或者 IOC;	
指令功能	查询当前端口功能配置寄存器数值。
变量范围	无
ACK 信息	0xAA [UID820 标识码] 0xC3 IOC2 IOC1 IOC0 0xFF
相关备注	参见上一节。

5.5 端口数字量输出控制寄存器

UID820 的 8 路端口可以同时输出高/低电平。高电平为+5V, 低电平为 0V。

DVA 寄存器位定义

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X	D8	D7	D6	D5	D4	D3	D2	D1

位 15- 8 保留, 读作 0, 写入被忽略。

位 7-0 端口 8 至端口 1 的电平输出

0 = 相应端口输出高电平 (即 5V)

1 = 相应端口输出低电平 (即 0V)

需要注意的是, 如果某个端口没有配置为数字输出, 则该指令对该端口是不起作用的。举例, 假如端口 7 已经被 IOC 指令配置为 PWM 输出 (即 IOC7 = 00 二进制)。当用户使用 DVA 指令将 D7 位设置为 0 或 1 时, UID820 不会对端口 7 做任何操作。

举例, 当需要 IO1、IO2、IO3、IO4 输出低电平, IO5、IO6、IO7、IO8 输出高电平时, 用户应如下配置 DVA 寄存器:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0

因此用户应首先保证 IOC = 0101 0101 0101 0101 (二进制) = 0x5555。

然后设置 DVA=0000 0000 1111 0000 (二进制) = 0x00F0 (16 进制)。

IOC 设置一次即可, 设置后一直存在于片上 EEPROM。每次更改端口电平, 只需更改 DVA 寄存器。

5.6 端口输出电平控制寄存器设置指令（DVA）

DVA = x;	
指令功能	设定端口输出电平控制寄存器 DVA 的数值 x。
变量范围	整数 x = 0、1 ... 255, 或者 16 进制数 x= 0x0000 ... 0x00FF
ACK 信息	0xAA [UID820 标识码] 0xC4 DVA2 DVA1 DVA0 0xFF
相关备注	0xC4 是 DVA=x 指令的信息标识码。 DVA2 – DVA0 表示 DVA 寄存器数值，采用 7 位数据结构，转换方式参见图 3-1。 如采用整数输入，用户需首先将上述主配置寄存器的各位填写 0 或者 1，然后再将得到的 16 位二进制数转成十进制数，然后将得到的十进制数填入语法结构中 x 的位置。如果采用 16 进制输入，则该 16 进制数必须以“0x”开头。

5.7 端口输入电平查询指令（DVA）

当有外部信号接入 UID820，需要知道信号的电平时可以使用该指令查询。

DVA;	
指令功能	查询当前端口输入的数字电平。
变量范围	无
ACK 信息	0xAA [UID820 标识码] 0xC5 DVA2 DVA1 DVA0 0xFF
相关备注	0xC5 是 DVA 指令的信息标识码。 DVA2 – DVA0 表示输入端口电平，采用 7 位数据结构，转换方式参见图 3-1。

需要特别说明的是，IO8、IO7、IO6 三个端口在没有外部输入信号时（端口悬空）其状态是随机的。将此三个端口对地时，其输入状态显示为 0，即位低电平；将其上拉至 5V 时，其输入状态显示为 1，即为高电平。IO1、IO2、IO3、IO4、IO5 五个端口内置有上拉电阻，端口悬空时读为 1，对地短接时读为 0。

此外如果某端口的 IOC 配置不是数字输入功能的话，则对应于该端口的 DVA 值永远为 0。

5.8 PWM输出基频设置指令（PMB）

UID820 能提供两路 PWM 输出。其 IO7、IO8 两端口可配置为 PWM 输出。PWM 输出的基频周期用 PMB 指令设置。

PMB 寄存器位定义

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
定义	PWMBase2								PWMBase1							

上表中 PWMBase1 和 PWMBase2 决定端口 7 和端口 8 的 PWM 波形的基本频率。

端口 7 的期望基频周期 = PWMBase1 * 100 微秒 (1 ≤ PWMBase1 ≤ 200)

端口 8 的期望基频周期 = PWMBase2 * 100 微秒 (1 ≤ PWMBase2 ≤ 200)

用户在设置 PMB 之前，应确保已经使用 IOC 指令将端口 7 或/和端口 8 设置为 PWM 输出。

PMB 指令执行后，PMB 寄存器的数值将会保存入片上 EEPROM，断电不会丢失。

PMB 指令执行后，PWM 波形输出会终止。用户需再次设置 PWM 占空比以启动 PWM 波形输出。

UID820 数字 I/O 及 PWM 输入输出控制器

PMB 指令格式如下：

PMB = x;	
指令功能	设定 PWM 波形的基频周期。
变量范围	整数 x = 0、1 ... 655355, 或者 16 进制数 x = 0x0000 ... 0xFFFF
ACK 信息	0xAA [UID820 标识码] 0xC7 PB2 PB1 PB0 0xFF
相关备注	0xC7 是 PMB 指令的信息标识码。 PB2 – PB0 表示 PMB 寄存器数值，采用 7 位数据结构，转换方式参见图 3-1。 如采用整数输入，用户需首先将上述主配置寄存器的各位填写 0 或者 1，然后再将得到的 16 位二进制数转成十进制数，然后将得到的十进制数填入语法结构中 x 的位置。如果采用 16 进制输入，则该 16 进制数必须以“0x”开头。

例如，当需要 IO8 的基频周期为 3ms，IO7 基频周期为 5ms 时，

$PWMBase2 = 3 \text{ 毫秒} / 0.1 \text{ 毫秒} = 30 \text{ (十进制)} = 0x1E \text{ (十六进制)}$

$PWMBase1 = 5 \text{ 毫秒} / 0.1 \text{ 毫秒} = 50 \text{ (十进制)} = 0x32 \text{ (十六进制)}$

所以 PMB = 0x1E32。

5.9 PWM 输出基频查询

PMB;	
指令功能	查询 PWM 波形的基频周期。
变量范围	无
ACK 信息	0xAA [UID820 标识码] 0xC7 PB2 PB1 PB0 0xFF
相关备注	0xC7 是 PMB 指令的信息标识码。 PB2 – PB0 表示 PMB 寄存器数值，采用 7 位数据结构，转换方式参见图 3-1。

5.10 PWM 输出占空比设置指令 (PMD)

UID820 的 PWM 输出的占空比用 PMD 指令设置。

PMD 寄存器位定义

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
定义	PWMDuty2								PWMDuty1							

上表中 PWMDuty1 和 PWMDuty2 决定端口 7 和端口 8 的 PWM 波形的占空比。分辨率是 0.5%。

端口 7 的期望占空比 = $PWMDuty1 * 0.5\%$ ($0 \leq PWMDuty1 \leq 200$)

端口 8 的期望占空比 = $PWMDuty2 * 0.5\%$ ($0 \leq PWMDuty2 \leq 200$)

出于安全考量，PMD 寄存器的数值不会保存入片上 EEPROM。

PMD 指令执行后，系统启动 PWM 波形输出。

对于一个已知系统，PMB 只需设置一次，PMD 可以不断更新。

PMD 指令格式如下:

PMD = x;	
指令功能	设定 PWM 波形的占空比。
变量范围	整数 $x = 0, 1 \dots 65535$, 或者 16 进制数 $x = 0x0000 \dots 0xFFFF$
ACK 信息	<u>0xAA</u> [UID820 标识码] <u>0xC6</u> <u>PD2</u> <u>PD1</u> <u>PD0</u> <u>0xFF</u>
相关备注	<p>0xC6 是 PMB 指令的信息标识码。</p> <p>PD2 – PD0 表示 PMD 寄存器数值, 采用 7 位数据结构, 转换方式参见图 3-1。</p> <p>如采用整数输入, 用户需首先将上述主配置寄存器的各位填写 0 或者 1, 然后再将得到的 16 位二进制数转成十进制数, 然后将得到的十进制数填入语法结构中 x 的位置。如果采用 16 进制输入, 则该 16 进制数必须以“0x”开头。</p>

例如, 当需要 IO8 占空比为 30%, IO7 占空比为 50%时

$PWMDuty2 = 30\% / 0.5\% = 60$ (十进制) = 0x3C (十六进制)

$PWMDuty1 = 50\% / 0.5\% = 100$ (十进制) = 0x64 (十六进制)

因此, $PMD = 0x3C64$ 。

5.11 PWM输出占空比查询

PMB;	
指令功能	查询 PWM 波形的基频周期。
变量范围	无
ACK 信息	<u>0xAA</u> [UID820 标识码] <u>0xC6</u> <u>PB2</u> <u>PB1</u> <u>PB0</u> <u>0xFF</u>
相关备注	<p>0xC6 是 PMD 指令的信息标识码。</p> <p>PD2 – PD0 表示 PMD 寄存器数值, 采用 7 位数据结构, 转换方式参见图 3-1。</p>

5.12 数字量及模拟量输出配置示例

描述:

配置 UID820 输出五路高电平, 一路输入, 同时输出两路周期为 5ms, 占空比为 50% 的 PWM 波形, 同时当输入状态变化时自动反馈。

要求:

IO1、IO2、IO3、IO4、IO5 输出高电平, IO6 输入, IO7、IO8 输出 PWM 波形。

实现:

1. 将 IO6 接地或接 5V。不可悬空! (注, 用户可以在完成本示例后, 尝试让 IO6 端口悬空, 以观察不可预知电平对 UID820 造成的影响)
2. 发送指令: **MCFG = 0x0020;** (配置 IO6 的变化通知。只启动 P6IE, 其他所有端口的变化通知都禁止。用户可以根据实际情况, 自行调整)。
3. 发送指令: **IOC = 0x0D55;** (配置 IOC1、IOC2、IOC3、IOC4、IOC5 为 01 (二进制) 数字输出, IOC6 为 11 (二进制) 数字输入, IOC7、IOC8 为 00 (二进制) PWM 输出)。
4. 发送指令: **DVA = 0x001F;** (D1 – D5 为 1 (高电平))。

UID820 数字 I/O 及 PWM 输入输出控制器

5. 查询 IO6 当前输入状态，发送指令：**DVA**;

6. 配置 CH7、CH8 周期为 5ms;

$PWMBase1 = 5ms / 0.1ms = 50(\text{十进制}) = 0x32$ (十六进制)

$PWMBase2 = 5ms / 0.1ms = 50(\text{十进制}) = 0x32$ (十六进制)

发送指令：**PMB=0x3232**;

7. 配置 CH7、CH8 占空比为 50%;

$PWMDuty1 = 50\% / 0.5\% = 100(\text{十进制}) = 0x64$ (十六进制)

$PWMDuty1 = 50\% / 0.5\% = 100(\text{十进制}) = 0x64$ (十六进制)

发送指令：**PMD=0X6464**;

8. 观察:

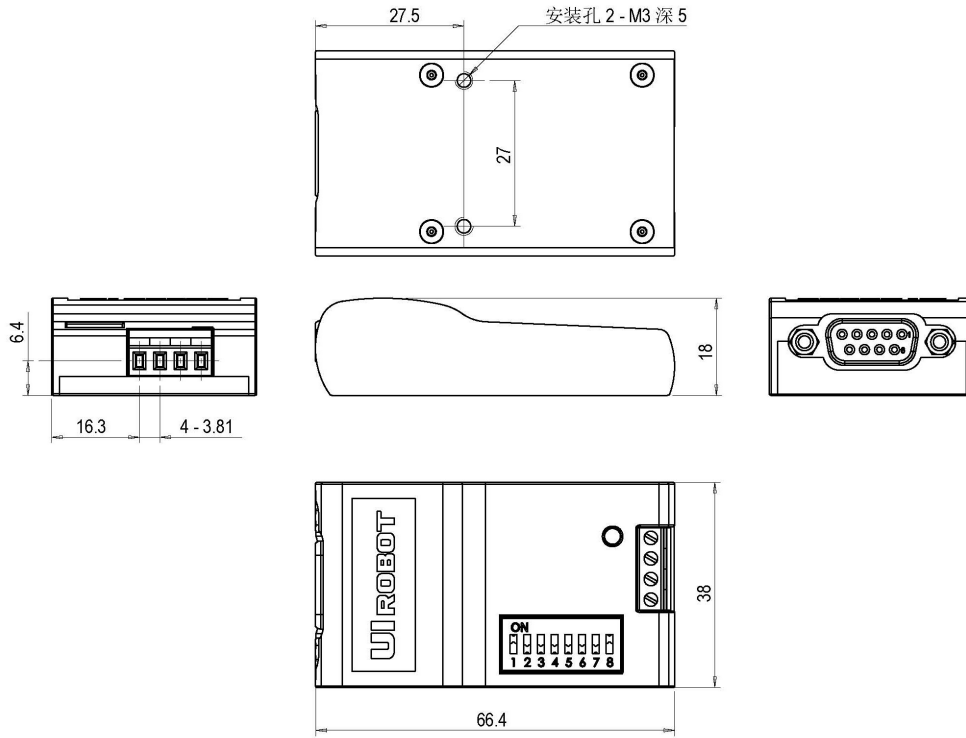
用万用表测量 IO1、IO2、IO3、IO4、IO5 相对于 GND 的电压，可得到：输出+5V 高电平；

变换加载于 IO6 上的电平，应该可以看到有时是变化通知发送到上位机；

指令 DVA；查询可知输入 IO6 端口上的输入电平；

通过示波器，应该能观测到 IO7、IO8 端口上的输出的周期为 5ms，占空比为 50%的 PWM 波形。

附录A 外形尺寸图



单位: mm