

## 第 18 章 “巾帼，英雄”

### 开启双 ID 的先河

#### 18.1 “巾帼英雄”

巾帼不让须眉，在很多场合女子比男子强，可在编程和制定规则方面，我一直认为这是男子的天下。这次将根协议从 TC0000A000 升级到 TC0000A001 是一次完美的跨越，我甚至为此而沾沾自喜。当我将这次“完美”的升级向刚刚出差回来的邓晓燕汇报后，那种得意已经荡然无存了。

邓晓燕认为将原来集中在 0xF000 段尾部的不可用空间（优先级为 1 时），均匀分配到 0x8000~0xF000 各段的尾部（优先级为 1 时），是不可取的。同时在 0x0000~0x7000 各段，以整段为单位划分等价优先层，仍旧不够方便，应当划分出更多的等价优先层。

上述两个要求几乎是背道而驰的，我用了很长时间去思考，真的想不出如何去重新划分 CAN 标识符 ID 来满足上述要求，这根本就是无解的。

当思维落入某种壁垒之中，仅靠自身的努力是很难摆脱束缚的。在一次讨论中邓晓燕说并没有人规定只能使用一种 ID 划分呀！初听，好似无理，甚至难以接受，从来没有哪个 CAN 应用层协议同时使用两种 ID 划分。

以前没有，不代表现在不能有，只要能解决问题，为何不去尝试呢？我不得不承认女子的思维比男子受到的束缚要更少，她们往往能够在困难面前另辟蹊径。至此 TTCANopen 第二次突破了传统惯性思维的束缚，破茧成蝶，展开了她美丽的翅膀。

#### 18.2 根协议再升级

这次根协议再升级，我们为 TTCANopen 准备了两种 ID 划分，第一种划分是在 TC0000A001 的基础上进行改进生成的，作用于过程变量空间，即：0x0000~07FFF，它在过程变量空间划分出 128 个等价优先层，每段含有 16 个等价优先层；第二种划分与

TC0000A000 中相同，作用于设备配置参数空间，即：0x8000~0xFFFF。

应用层协议根据 CAN 标识符中第 27 位的数值判断具体使用哪种划分，当该位为“0”时使用第一种划分；当该位为“1”时使用第二种划分。

### 18.2.1 TC0000A002 根协议（ID 分配）

本协议是对 TTCANopen 之 TC0001A000 的升级，主要包括两个 ID 划分，第一种作用于过程变量空间（0x0000~07FFF）；第二种作用于配置参数空间（0x8000~0xFFFF）。

#### 1 TTCANopen ID 划分一（即：传递过程变量的 CAN 标识符划分）

传递过程变量的 CAN 标识符划分，采用 CAN2.0b 协议扩展帧格式，其中 29 位 CAN 标识符划分见图 18-1。

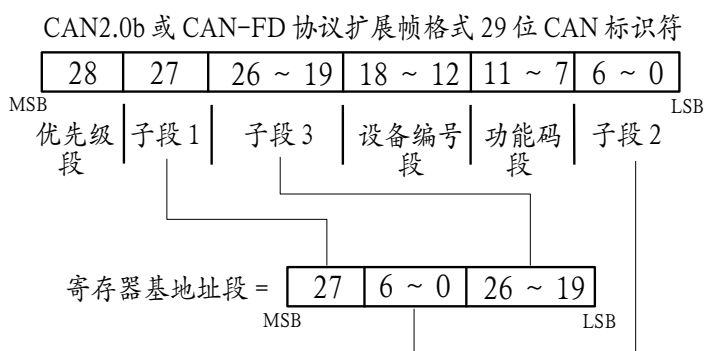


图 18-1：传递过程变量的 CAN 标识符划分

优先级段：由最高位 28 位构成。

寄存器基址段：由 27 位、6~0 位和 26~19 位构成。

设备编号段：由 18~12 位构成。

功能码段：由 11~7 位构成。

#### 2 TTCANopen ID 划分二（即：传递非过程变量的 CAN 标识符划分）

传递非过程变量的 CAN 标识符划分，采用 CAN2.0b 协议扩展帧格式，其中 29 位 CAN 标识符划分见图 18-2。

CAN2.0b 或 CAN-FD 协议扩展帧格式 29 位 CAN 标识符

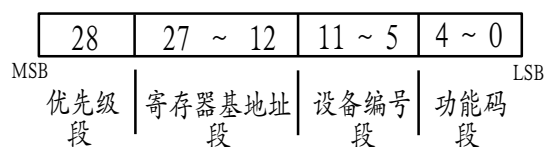


图 18-2：传递非过程变量的 CAN 标识符划分

优先级段：由最高位 28 位构成。

寄存器基地址段：由 27~12 位构成。

设备编号段：由 11~5 位构成。

功能码段：由 4~0 位构成。

### 3 各段说明

优先级段：由 CAN 标识符的最高位第 28 位组成，取值范围 0x0~0x1。应用层协议将 CAN 指令按紧急程度分为两类，一类是普通指令；一类是紧急指令。应用层协议使用优先级段为“0x1”发送普通指令，使用优先级段为“0x0”发送紧急指令；

寄存器基地址段：为十六位的寄存器基地址寻址空间，取值范围 0x0000~0xFFFF，每个地址对应一个字节寄存器，应用层协议将数据存放在对应寄存器中。一条 CAN 指令其数据场可携带不多于 8 个字节的数据（对于 CAN-FD 协议为不多于 64 个字节的数据，8 个字节以上指令能够携带的字节数不连续递增，具体为 12、16、20、24、32、48、64。），寄存器基地址标识的是 CAN 指令数据场中第 1 个数据字节的地址，其它字节的地址依序递增。当 CAN 标识符的第 27 位为“0”时，寄存器基地址段指向过程变量空间（0x0000~0x7FFF），应用层协议启用传递过程变量的 CAN 标识符划分（如图 18-1 所示），设备根据相关过程变量的种类、重要性和紧迫性，按照寄存器分段（层）管理规则，在各等价层中配置其位置，使其获得恰当的总线竞争能力；当 CAN 标识符的第 27 位为“1”时，寄存器基地址段指向非过程变量空间（0x8000~0xFFFF），应用层协议启用传递非过程变量的 CAN 标识符划分（如图 18-2 所示），设备根据配置参数空间的分层管理规则，分配各配置参数和状态参数的寄存器使用地址；

设备编号段：当 CAN 标识符的第 27 位为“0”时，应用层协议启用传递过程变量的 CAN 标识符划分，其由 CAN 标识符的第 18~12 位组成；当 CAN 标识符的第 27 位为“1”时，应用层协议启用传递非过程变量的 CAN 标识符划分，其由 CAN 标识符的第 11~5 位组成。

设备编号为设备在总线上的编号（或称设备地址），取值范围 0x00~0x7F，可为应用层协议提供 128 个设备编号，其中：0x00 为广播地址，0x01~0x03 分配给心跳器，0x04 和 0x05 分配给设备管理器，0x06 和 0x07 分配给系统主机，0x60~0x6F 分配给组地址，0x70~0x7F 系统保留；

功能码段：当 CAN 标识符的第 27 位为“0”时，应用层协议启用传递过程变量的 CAN 标识符划分，其由 CAN 标识符的第 11~7 位组成；当 CAN 标识符的第 27 位为“1”时，应用层协议启用传递非过程变量的 CAN 标识符划分，其由 CAN 标识符的第 4~0 位组成。功能码为指令功能码编号，取值范围 0x00~0x1F，其可为应用层协议提供 32 条指令功能码。

**基本指令格式表述：**使用 CAN2.0b 协议扩展帧格式的数据帧

优先级	寄存器基地址	设备地址	功能码	DLC	0~8 个字节数据
-----	--------	------	-----	-----	-----------

当指令作用于过程变量空间（ID<sub>27</sub>=0），应用层协议启用传递过程变量的 CAN 标识符划分，指令格式中的优先级对应优先级段（28 位），寄存器基地址对应寄存器基地址段（27 位、6~0 位、26~19 位），设备地址对应设备编号段（18~12 位），功能码对应功能码段（11~7 位），DLC 表示指令数据场携带的数据字节数，最后是 0~8 字节的数据场。

当指令作用于非过程变量空间或称配置参数空间（ID<sub>27</sub>=1），应用层协议启用传递非过程变量的 CAN 标识符划分，指令格式中的优先级对应优先级段（28 位），寄存器基地址对应寄存器基地址段（27~12 位），设备地址对应设备编号段（11~5 位），功能码对应功能码段（4~0 位），DLC 表示指令数据场携带的数据字节数，最后是 0~8 字节的数据场。

指令的优先级最终由 29 位 ID 值决定，值越小优先级越高（显性电平为 0）。

注：对于 CAN FD 标准，其对数据长度编码进行了扩展，当数据长度大于 8 时，使用了非线性编码，见表 18-1。其指令传输数据的字节数也相应非线性增加。

表 18-1 CAN FD 中 DLC 编码对应的数据长度

DLC	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
数据长度值	0	1	2	3	4	5	6	7	8	12	16	20	24	32	48	64

## 4 寄存器空间的划分

协议为系统和设备定义了一个 16 位寄存器寻址空间，每一个地址对应一个 8 位寄存器，凡是需要参与或可能参与 CAN 总线通讯的参数和变量，都要使用或映射到该寄存器空间，寄存器空间分配见图 18-3，其中：0x0000~0x6FFF 定义为“系统信息空间”；0x7000~0x7FFF 定义为“设备信息空间”；0x8000~0xFFFF 定义为“设备配置参数空间”。

“系统信息空间”是 TTCANopen 网络系统的全局信息映射的寄存器空间；“设备信息空间”是各设备自身信息存放的寄存器空间；“设备配置参数空间”是各设备自身配置参数存放的寄存器空间。

TTCANopen 在“系统信息空间”使用面向报文的通讯协议，其信息由寄存器地址唯一定位；在“设备信息空间”和“设备配置参数空间”使用面向节点的通讯协议，其信息由寄存器地址和设备地址共同定位。

另外，从变量特性上看，0x0000~0x7FFF 用于存储和传递过程变量，0x8000~0xFFFF 用于存储和传递非过程变量。

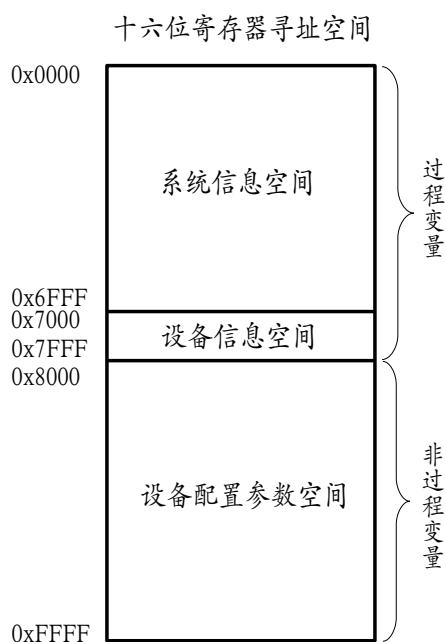


图 18-3 寄存器空间分配

为了便于各类信息的管理，协议将 16 位寻址空间划分为 16 个段，各段寄存器空间的使

用规则如下：

0x0000 段：由 TTCANopen 组织规划，分配给共性系统过程变量使用，见 TC0001；

0x1000 段：分配给 I 特性系统信息过程变量使用，见 TC0002；

0x2000 段：分配给 O 特性系统信息过程变量使用，见 TC0003；

0x3000 段：分配给 I/O 特性系统信息过程变量使用，见 TC0004；

0x4000 段：由专业标准化组织规划，分配给 I 特性系统信息过程变量见 TC0005；

0x5000 段：由专业标准化组织规划，分配给 O 特性系统信息过程变量见 TC0006；

0x6000 段：由专业标准化组织规划，分配给 I/O 特性系统信息过程变量见 TC0007；

0x7000 段：分配给设备信息过程变量使用，见 TC0008；

0x8000 段：分配给设备特征配置参数使用，见 TC0009；

0x9000 段：由标准化组织规划，对标准化设备特征配置参数进行统一分配，TC0010；

0xA000 段：保留；

0xB000 段：保留；

0xC000 段：保留；

0xD000 段：可编程类对象分配空间，见 TC0014；

0xE000 段：可编程过程变量资源配置和映射区，见 TC0015；

0xF000 段：由 TTCANopen 组织规划，分配给设备共性配置参数使用，见 TC0016。

协议将 0x0000 和 0x7000 段的最后 256 个字节（0xF00～0xFFF）分割出来做“系统保留”。

当优先级为“1”时，0xF000 段的最后 1024 个字节（0xFC00～0xFFFF）禁用。

## 5 寄存器基地址和数据字节表述

寄存器基地址用两个字节表述，高字节在前，低字节在后，如：0x1234 表示为两个字节 0x12 0x34。

指令数据场携带的数据按寄存器地址先后顺序排列，由低到高，无论数据表述的是单字节数据还是多字节数据。

## 6 兼容性要求

为了兼容早期的协议，指令的前 7 位不能出现连续隐性电平。

## 7 升级后对其它子协议的影响

对主心跳的影响，由于升级后，在 0x0000 段出现了 16 个等价优先层，这样就使得比主心跳地址 0x0010 优先级更高的空间多出了 15 倍，在其它各层也多出了 15 倍。因此，我们需要提升主心跳地址的优先级，新的主心跳地址被定义在 0x0A02 和 0x0A03 上，参见 TC0001A001。

对配置参数空间的影响，在 TC0000A001 中，被均匀分配到各段的 128 个字节不可用空间，被重新集中到 0xF000 段中（0xFC00~0xFFFF）。

## 18.3 TC0001A001 子协议（0x0000 段的分配）

设备 0x0000 段寄存器分配给系统共有的通用过程变量使用，目前，我们只在该段定义了系统心跳预报帧的信息（关于“系统心跳”请参看第六章内容），该段系统信息空间的分配由 TTCANopen 组织统一制定，见下表。

寄存器地址	参数类型	读写特性	参数描述	值域
0x0A02~0x0A05	U32	周期发送	TTCANopen 系统时 (一天中 0.1ms 的计数)	0~863999999
0x0A06~0x0A07	U16	周期发送	心跳周期 (单位 ms)	1~60000 (可被 86400000 整除)
0x0A08	U8	周期发送	CAN 接收错误计数	0~255
0x0A09	U8	周期发送	CAN 发送错误计数	0~255

（注：读写特性为“周期发送”，网络上任何设备都不能对该寄存器信息进行“读”或“写”操作。它是由指定设备（心跳器）周期发送到网络上的信息）

## 18.4 TC0017A000 子协议（推荐对过程变量的分层管理）

当过程变量空间的每一段又被划分出 16 个等价优先层后，我们就可以再对它进行一个较为合理和更加细致的分配，使我们能够从过程变量的寄存器地址上，获取更多的变量属性特征。下表是 TTCANopen 组织推荐的分配方案之一，各行业和专业组织也可以根据本专业的应用特点，自行定义分配方案。

段地址	段种层数	地址区间	说明
0x0000	1 ~ 14	0000 ~ 0DFF	系统公共特性
	15 ~ 16	0E00 ~ 0FFF	保留层
0x1000	1 ~ 2	1000 ~ 11FF	系统 本地 I 特性数字量
	3 ~ 4	1200 ~ 13FF	系统 衍生 I 特性数字量
	5 ~ 8	1400 ~ 17FF	系统 本地 I 特性模拟量
	9 ~ 12	1800 ~ 1BFF	系统 衍生 I 特性模拟量
	13 ~ 14	1C00 ~ 1DFF	系统 机动层
	15 ~ 16	1E00 ~ 1FFF	保留层
0x2000	1 ~ 2	2000 ~ 21FF	系统 本地 O 特性数字量
	3 ~ 4	2200 ~ 23FF	系统 衍生 O 特性数字量
	5 ~ 8	2400 ~ 27FF	系统 本地 O 特性模拟量
	9 ~ 12	2800 ~ 2BFF	系统 衍生 O 特性模拟量
	13 ~ 14	2C00 ~ 2DFF	系统 机动层
	15 ~ 16	2E00 ~ 2FFF	保留层
0x3000	1 ~ 2	3000 ~ 31FF	系统 本地 I/O 特性数字量
	3 ~ 4	3200 ~ 33FF	系统 衍生 I/O 特性数字量
	5 ~ 8	3400 ~ 37FF	系统 本地 I/O 特性模拟量
	9 ~ 12	3800 ~ 3BFF	系统 衍生 I/O 特性模拟量
	13 ~ 14	3C00 ~ 3DFF	系统 机动层
	15 ~ 16	3E00 ~ 3FFF	保留层
0x4000	1 ~ 2	4000 ~ 41FF	专业 本地 I 特性数字量
	3 ~ 4	4200 ~ 43FF	专业 衍生 I 特性数字量
	5 ~ 8	4400 ~ 47FF	专业 本地 I 特性模拟量
	9 ~ 12	4800 ~ 4BFF	专业 衍生 I 特性模拟量
	13 ~ 14	4C00 ~ 4DFF	专业 机动层
	15 ~ 16	4E00 ~ 4FFF	保留层
0x5000	1 ~ 2	5000 ~ 51FF	专业 本地 O 特性数字量
	3 ~ 4	5200 ~ 53FF	专业 衍生 O 特性数字量
	5 ~ 8	5400 ~ 57FF	专业 本地 O 特性模拟量
	9 ~ 12	5800 ~ 5BFF	专业 衍生 O 特性模拟量



	13 ~ 14	5C00 ~ 5DFF	专业 机动层
	15 ~ 16	5E00 ~ 5FFF	保留层
0x6000	1 ~ 2	6000 ~ 61FF	专业 本地 I/O 特性数字量
	3 ~ 4	6200 ~ 63FF	专业 衍生 I/O 特性数字量
	5 ~ 8	6400 ~ 67FF	专业 本地 I/O 特性模拟量
	9 ~ 12	6800 ~ 6BFF	专业 衍生 I/O 特性模拟量
	13 ~ 14	6C00 ~ 6DFF	专业 机动层
	15 ~ 16	6E00 ~ 6FFF	保留层
0x7000	1	7000 ~ 70FF	设备 本地 I 特性数字量
	2	7100 ~ 71FF	设备 衍生 I 特性数字量
	3	7200 ~ 72FF	设备 本地 I 特性模拟量
	4	7300 ~ 73FF	设备 衍生 I 特性模拟量
	5	7400 ~ 74FF	设备 本地 O 特性数字量
	6	7500 ~ 75FF	设备 衍生 O 特性数字量
	7	7600 ~ 76FF	设备 本地 O 特性模拟量
	8	7700 ~ 77FF	设备 衍生 O 特性模拟量
	9	7800 ~ 78FF	设备 本地 I/O 特性数字量
	10	7900 ~ 79FF	设备 衍生 I/O 特性数字量
	11	7A00 ~ 7AFF	设备 本地 I/O 特性模拟量
	12	7B00 ~ 7BFF	设备 衍生 I/O 特性模拟量
	13 ~ 14	7C00 ~ 7DFF	设备 机动层
	15 ~ 16	7E00 ~ 7FFF	保留层