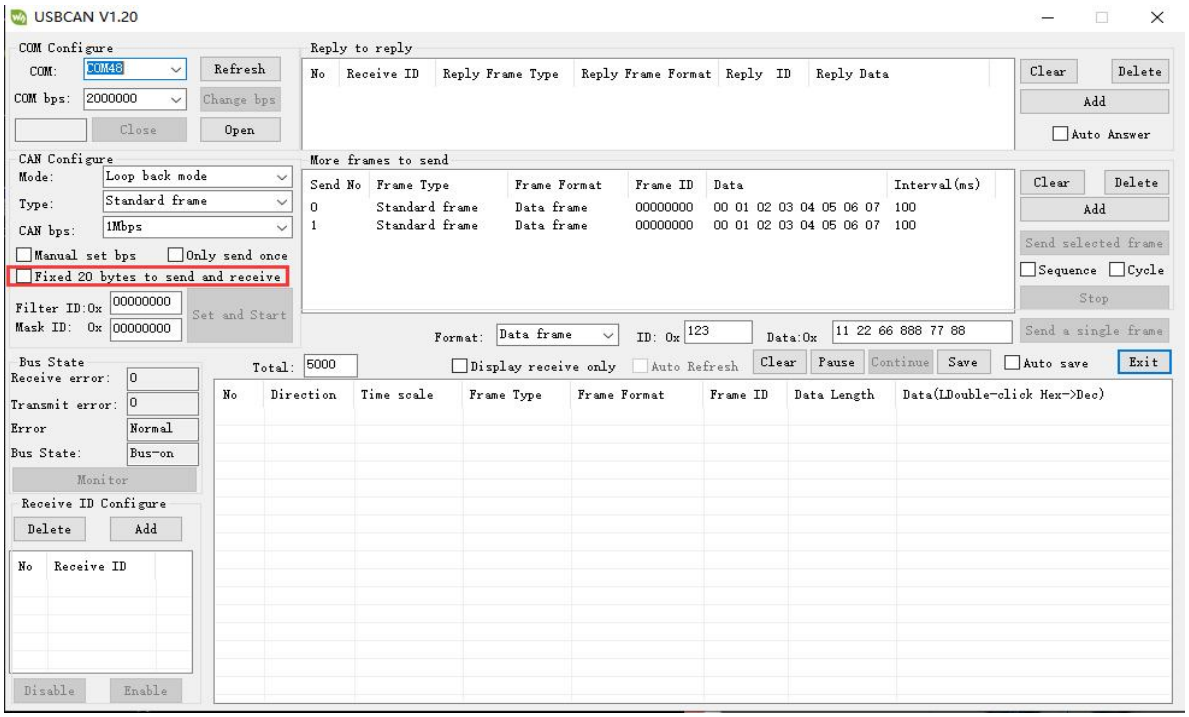
# 串行 (USB) 转换CAN协议的定义

**串行端口波特率默认为2000000，8位数据位，1位停止位，不检查。转换器内支持两种通信协议，一种是固定的20字节通信协议，另一种是可变长度通信协议。**

## 1.可变长度通信格式




**未选择软件时**

，

**(例如，在上面的框中选中了红线)，单击 “设置并开始” 按钮。转换器的内部通信是可变长度的通信协议。具体协议如下:**

<table>
<tr><td></td><td colspan="4">定义</td></tr>
<tr><td>数据包报头</td><td colspan="4">0xaa</td></tr>
<tr><td>Tyep</td><td colspan="4">0xc0类型<br>bit5 (帧类型0-标准帧 (帧ID 2字节)，1-扩展帧 (帧ID 4字节))<br>bit4 (帧格式0-数据帧，1远程帧)<br>Bit0 ~ 3帧数据长度 (0 ~ 8)</td></tr>
<tr><td rowspan="4">帧ID</td><td rowspan="4">扩展框架</td><td>1 ~ 8位 (字节1)</td><td rowspan="4">标准框架</td><td>1 ~ 8位 (字节1)</td></tr>
<tr><td>9 ~ 16位 (字节2)</td><td rowspan="3">9 ~ 11位 (字节2)</td></tr>
<tr><td>17 ~ 24位 (字节3)</td></tr>
<tr><td>25 ~ 29位 (字节4)</td></tr>
<tr><td rowspan="6">帧数据 (帧数据可能是0 ~ 8个数据)</td><td colspan="4">可以发送或接受数据1</td></tr>
<tr><td colspan="4">可以发送或接受数据2</td></tr>
<tr><td colspan="4">可以发送或接受数据3</td></tr>
<tr><td colspan="4">可以发送或接受数据4</td></tr>
<tr><td colspan="4">可以发送或接受数据5</td></tr>
<tr><td colspan="4">可以发送或接受数据6</td></tr>
</table>

| | |
|---|---|
| | 可以发送或接受数据7 |
| | 可以发送或接受数据8 |
| 结束代码 | 0x55 |
| | |
| | |
| | |
| | |
| | |

示例说明 (标准框架):

1. 例如，将CAN标准帧ID发送到0x123，相应的数据为0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88

然后串口向AA发送数据格式C8 23 01 11 22 444 44 55 66 77 88 5以下是一个解释

AA报文头

C8 Bit5为0标准帧，bit4为0，数据帧bit0 ~ 3为8，数据长度为8，发送8字节

23 01 F或帧ID 0x123

11 22 444 44 55 66 77 88为CAN发送帧数据

55结束代码

例如，接收到串行端口: C8 23 01 11 22

444 44 55 66 77

接收到的CAN为标准帧，ID为0x123，对应的数据为11 22 444 44 55 66 77 88,8字节，分辨率同上

2. 例如，将CAN标准帧ID发送到0x103，对应的数据为0x 11 0x22

然后串口将数据格式发送到:

AA C2 03 01 11 22 55

以下是一个解释:

AA报文头

C2 bit5为0标准帧，bit4为0，数据帧bit0 ~ 3为2，数据长度为2，发送2字节

帧ID 0x103的03 01 11 22发送

CAN 55结束码的帧数据

例如，接收到虚拟串行端口:

AA C2 03 01 11 22 55

接收到的CAN为标准帧，ID为0x0103，对应数据为11 22，2字节，分辨率同上

示例描述 (扩展框架):

1. 例如，CAN扩展帧ID为0x1234567，对应的数据为0x11，

0x22, 0x33, 0x44, 0x55, 0x66, 0x77, 0x88

然后串口将数据格式发送到:

E8 67 45 23 01 11 22 444 44 55 66 77 88

以下是一个解释:

AA报文头

E8 Bit5为1，扩展帧bit4为0，数据帧bit0 ~ 3为8，数据长度为8，发送8

字节

帧ID 0x1234567的67 45 23 01

11 22 444 44 55 66 77 88发送CAN帧数据

55结束代码

例如，接收到串行端口:

E8 67 45 23 01 11 22 444 44 55 66 77 88

接受的CAN是扩展帧，ID为0x1234567，对应的数据为11 22 3344 55 66 77 88,8字

节，解析如上

2. 例如，CAN扩展帧ID为0x1033021，对应的数据为0x 11 0x22:

然后串口将数据格式发送到: AA E2 21 30 03

01 11 22 55

以下是一个解释:

AA数据包报头

E2 Bit5为0，扩展帧bit4为0，数据帧bit0 ~ 3为2，数据长度为2，发送2字节

帧ID的21 30 03 01 0x1033021

11 22发送CAN帧数据

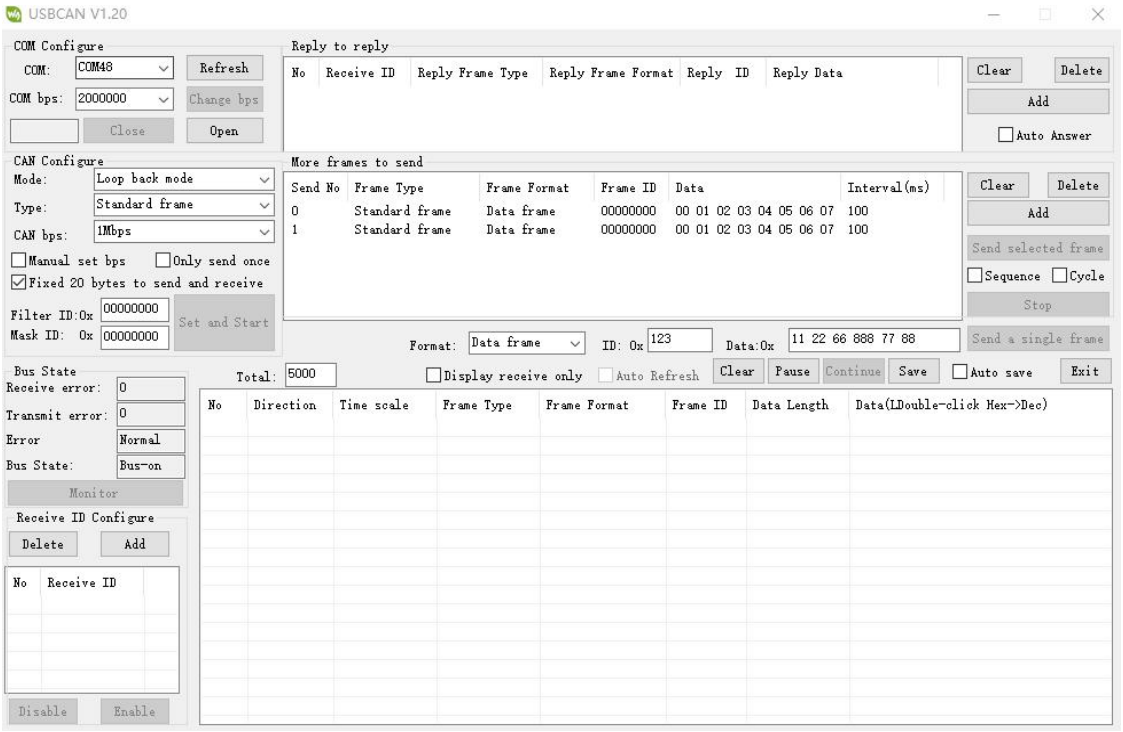
55结束代码

例如，接收到串口AA E2 21 30 03 01 11

22

接受的CAN是扩展帧，ID为0x1033021，对应数据为11 22,2字节，解析如上

# 2.固定的20字节收发器协议




**选择** **软件后（例如，在上面的框中选中了红线），单击 “设置和开始” 按钮。转换器内部通信的定义是固定的20字节收发器定义。具体协议如下。**



| | | 定义 |
|---|---|---|
| 0 | 数据包报头 | 0xaa |
| 1 | 数据包报头 | 0x55 |
| 2 | 类型 | 0x01-数据 |
| 3 | 框架类型 | 0x01-标准框架，0x02-扩展框架 |
| 4 | 框架格式 | 0x01-数据帧，0x02-远程帧 |
| 5 | 帧ID数据1 | 1～8位，前面高字节，后面低字节 |
| 6 | 帧ID数据2 | 9～16位，前面高字节，后面低字节 |
| 7 | 帧ID数据3 | 17～24位，前面高字节，后面低字节 |
| 8 | 帧ID数据4 | 25～32位，前面高字节，后面低字节 |
| 9 | 帧数据长度 | 发送或接受的CAN总线的数据长度 |

| | | |
|---|---|---|
| 10 | 帧数据1 | 可以发送或接受数据1 |
| 11 | 帧数据2 | 可以发送或接受数据2 |
| 12 | 帧数据3 | 可以发送或接受数据3 |
| 13 | 帧数据4 | 可以发送或接受数据4 |
| 14 | 帧数据5 | 可以发送或接受数据5 |
| 15 | 帧数据6 | 可以发送或接受数据 |
| 16 | 帧数据7 | 可以发送或接受数据7 |
| 17 | 帧数据8 | 可以发送或接受数据8 |
| 18 | 储备 | 0x00 |
| 19 | 检查代码 | 从帧类型到错误代码，累加和低8位<br>(红色部分) |

例如，我们接受或发送数据标准帧ID到0x123，数据为11 22 444 44 55 66 77 88

配置如下

| | | 数据 |
|---|---|---|
| 0 | 数据包报头 | 0xaa |
| 1 | 数据包报头 | 0x55 |
| 2 | 类型 | 0x01 |
| 3 | 框架类型 | 0x01 |
| 4 | 框架格式 | 0x01 |
| 5 | 帧ID数据1 | 0x23 |
| 6 | 帧ID数据2 | 0x01 |
| 7 | 帧ID数据3 | 0x00 |
| 8 | 帧ID数据4 | 0x00 |
| 9 | 帧数据<br>长度 | 0x08 |
| 10 | 帧数据1 | 11 |
| 11 | 帧数据2 | 22 |
| 12 | 帧数据3 | 444 |
| 13 | 帧数据4 | 44 |
| 14 | 帧数据5 | 55 |
| 15 | 帧数据6 | 66 |
| 16 | 帧数据7 | 77 |
| 17 | 帧数据8 | 88 |

| | | |
|---|---|---|
| 18 | 储备 | 0x00 |
| 19 | 检查代码 | **0x93** (0x01 + 0x01 + 0x01 + 0x23 + 0x01 + 0x00 + 0x00 + 0X08 + 0x11 + 0x22 + 0x33 + 0x44 + 0x55 + 0x66 + 0x77 + 0x88 + 0x00 = 0x293<br>**低8位为0x93)** |

例如，我们接受或发送数据扩展帧ID到0x12345678，数据是01 02 03 04 05 06 07 08，它配置如下

| | | 数据 |
|---|---|---|
| 0 | 数据包报头 | 0xaa |
| 1 | 数据包报头 | 0x55 |
| 2 | 类型 | 0x01 |
| 3 | 框架类型 | 0x02 |
| 4 | 框架格式 | 0x01 |
| 5 | 帧ID数据1 | 0x78 |
| 6 | 帧ID数据2 | 0x56 |
| 7 | 帧ID数据3 | 0x34 |
| 8 | 帧ID数据4 | 0x12 |
| 9 | 帧数据长度 | 0x08 |
| 10 | 帧数据1 | 01 |
| 11 | 帧数据2 | 02 |
| 12 | 帧数据3 | 03 |
| 13 | 帧数据4 | 04 |
| 14 | 帧数据5 | 05 |
| 15 | 帧数据6 | 06 |
| 16 | 帧数据7 | 07 |
| 17 | 帧数据8 | 08 |
| 18 | 储备 | 0x00 |
| 19 | 检查代码 | **0x44** (0x01 + 0x02 + 0x01 + 0x78 + 0x56 + 0x34 + 0x12 + 0x08 + 0x01 + 0x02 + 0x03 + 0x04 + 0x05 + 0x06 + 0x07 + 0x08 + 0x00 = 0x144<br>**低8位为0x44**) |