# CAN communication protocol

## Use standard ID🡪 this is Can open stack;

## Use extend ID (CAN 2.0 B) physical layer.

1. The extend ID has following definitions: //total 29 bits

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SRC | DST | CMD Num | CMD Type | Frame Type | Ex |
| 8 Bits | 8 Bits | 5 Bits | 2 Bits | 2 Bits | 4 Bits (16 different ID) |

1. SRC: the address which sends the data.
   1. 0xFF: no resp is required;
2. DST: the address which receives the data
   1. 0xFF: broadcast
3. CMD Num：(little endian)
   1. 0x00 Token:

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | - |  |  |  |  |  |  |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte71 | Byte 8 |
| Time Out0 | Time Out1 | Frames | During time 0 | During time 1 | Device type | Device status |  |

* 1. 0x01 Poll device(little endian)/recognize new device

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| - | - |  |  |  |  |  |  |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte71 | Byte 8 |
| Time Out0 | Time Out1 | Frames | During time 0 | During time 1 | Device type | Device status |  |

Device status:

|  |  |  |  |
| --- | --- | --- | --- |
| 0x00 | 0x01 | 0x02 | 0x03 |
| On line | Off line | Reset, | ?? |

* 1. 0x04 Put Data: (little endian)

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SubId | ObjId | AtrID | Len | Data0 | Data1 | Data2 | Data3 |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SubId | ObjId | AtrID | Len | status | -NULL | -NULL | -NULL |

* 1. 0x05 Get Data: (little endian)

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SubId | ObjId | AtrID | - | - | - | - | - |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SubId | ObjId | AtrID | Len | Data0 | Data1 | Data2 | Data3 |

* 1. 0x06 Put Mem: (little endian) (max 32bytes for normal cmds, but 0x2000 for bootloader)

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Adr Byte 0 | Adr Byte 1 | Adr Byte 2 | Adr Byte 3 | Len Byte 0 | Len Byte 1 | Data0 | Data1 |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Adr Byte 0 | Adr Byte 1 | Adr Byte 2 | Adr Byte 3 | Len Byte 0 | Len Byte 1 | status | - |

* 1. 0x07 Get Mem: (little endian)

Request:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte71 | Byte 8 |
| Adr 0 | Adr 1 | Adr 2 | Adr 3 | Len 0 | Len 1 |  |  |

Response:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte71 | Byte 8 |
| Adr 0 | Adr 1 | Adr 2 | Adr 3 | Len 0 | Len 1 | Data 0 | Data 1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte71 | Byte 8 |
| Data 2 | Data 3 | Data 4 | Data 5 | Data 6 | Data 7 | Data 8 | Data 9 |

1. CMD Type:

|  |  |
| --- | --- |
| 0x01 | response |
| 0x02 | request |
| 0x00 | Broadcast |
| 0x03 | Todo |

1. Frame type

|  |  |
| --- | --- |
| 0x01 | multi frames start 🡪 the data part not have the frame ID, the EX is 0; |
| 0x02 | Multi frames 🡪 the EX parts will increase by 1; |
| 0x03 | multi frames ends🡪 the EX parts will increase by 1; |
| 0x00 | No multi frames |

1. Ex
   1. ID which is used to identify this is the response to the request id;
   2. the EX parts will increase from 0 when multi frames;
2. Data - Status – error bytes:

0x00: OK;

0xF1: Error CMD;

0xF2: Error frame Number – multi frames;

0xF3: Error Data Length

0xF4: max data length

Other: the error code defined in t\_data\_obj