

Annex One: ID Alignment Details

Manual Version: V2.05

Version Update Date: 2021.07.14

一、Alignment

The valid ID for the standard frame is bit 11, and bit 29 for the extended frame. When it comes to store ID by using 32-bit unsigned integer, there might be two storage methods: Right-aligned method (Direct ID mode) and left-aligned method (SJA1000 / register mode). *Alignment methods involved with the interface function library of this USBCAN device are compatible with the interface function library of USBCAN ZLG Series device.*

1.1 Right-aligned method

Right-aligned method, that is, ID.0 as the least significant bit of ID is aligned with Bit.0. As the following table shown:

| unsigned int (bit31~bit0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|---|---|----------|
| high | | | | | | | | | | | | | | | | | | | | | | | | | | | | low | Bit | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
| | | | | | | | | | | | | | | | | | | | | | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Standard |
| | | | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Extended |

1.2 Left-aligned method

Left-aligned method, that is, ID.10 (standard frame) as the most significant bit of ID and ID.28 (extended frame) are aligned with Bit.31. As the following table shown:

| unsigned int (bit31～bit0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|---|-----|-----|---|----------|
| high | | | | | | | | | | | | | | | | | | | | | | | | | | | | | low | Bit | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
| 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | Standard |
| 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | Extended |

二、ID Right-aligned Method (Direct ID Method)

CAN message ID is represented by unsigned int in the interface function library with a total of 32 bits. Among of which, the valid ID of the standard frame is 11 bit, and 29 bit for extended frame with right-aligned method (direct ID method), that is, ID.0 as the least significant bit is aligned with Bit.0. As the following table shown:

| ID(bit31～bit0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|---|---|----------|
| high | | | | | | | | | | | | | | | | | | | | | | | | | | | | low | Bit | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
| | | | | | | | | | | | | | | | | | | | | | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Standard |
| | | | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Extended |

Note: In the table, the bit in dark background cell represents valid bit of the ID, ID.0 as the least significant bit of ID is aligned with Bit.0.

E.g.:

1. Standard frame ID: Standard frame ID is 00 00 01 23(HEX), 11 bit is the valid, actual ID value is 00 00 01 23H.
2. Extended frame ID: Extended frame ID is 1F 01 02 03(HEX), 29 bit is the valid, actual ID value is 1F 01 02 03H.

三、AccCode/AccMask Left-aligned Method (Register Mode)

AccCode (filtering acceptance yards) / AccMask (filter mask) is represented in the interface function library with unsigned integer with a total of 32 bits, including standard frame with valid ID is bit 11, and the extended frame bit 29 by using the left-aligned method (register mode), that is, ID.10 (standard frame) as the most significant bit of the ID and ID.28 (extended frame) are aligned with Bit.31. As the following table shown:

| AccCode/AccMask (bit31～bit0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|---|---|----------|
| high | | | | | | | | | | | | | | | | | | | | | | | | | | | | low | Bit | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
| 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | Standard |
| 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | Extended |

Note: In the table, the bit in dark background cell represents valid bit of the ID, the most significant bit of ID is aligned with Bit31.

E.g.:

1. Standard frame ID: Standard frame ID is 00 00 01 23(HEX), 11 bit is the valid, so the actual value should be shifted to the left by 21 bits, obtaining AccCode/AccMask value 24 60 00 00H.

2. Extended frame ID: Extended frame ID is 1F 01 02 03(HEX), 29 bit is the valid, so the actual value should be shifted to the left by 3 bits, obtaining AccCode/AccMask value F8 08 10 18H.