Application Report

Selective Wake Configuration Guide: TCAN1145-Q1 and TCAN1146-Q1

Chris Ayoub

Transceiver Interface

ABSTRACT

The TCAN1145-Q1 and TCAN1146-Q1 CAN transceivers come equipped with selective wake mode which allows partial networking to be implemented. This application note details how to set up and implement the selective wake mode on these devices.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

1.1 What is Partial Networking?

Partial networking is a function used to save power when utilizing CAN devices. In a normal CAN system, if the bus is sleeping (low-power mode) a wake-up pattern will wake up every node. The normal wake-up pattern is a filtered dominant pulse, followed by a filtered recessive pulse, followed by another filtered dominant pulse. Once this wake-up pattern is sent to the CAN bus, every single node recognizes this request and transitions out of low power mode. They will all be consuming current at normal levels. This level of current consumption is much higher during normal operation compared to when the nodes are sleeping.

With a partial networking system, sleep is broken up into two parts, deep sleep (CAN bus GND bias) and regular sleep (CAN bus 2.5V bias). During deep sleep, all the transceivers can be configured for Partial Networking. Once configured, all the transceivers are waiting for the normal wake-up pattern: a filtered dominant pulse, followed by a filtered recessive pulse, followed by another filtered dominant pulse. Once the normal wake up pattern is sent and received, the bus bias of each transceiver is set to 2.5 V from GND. This bias is necessary for the WUF receiver to decode CAN frames. The transceiver is then waiting for a wake-up frame (WUF), but before it can correctly receive and interpret the WUF, the receiver has to synchronize to the bus data rate. This is typically done by the controller sending CAN frames and the receiver analyzes the data stream by measuring the length of time between bit edges. The CAN standard allows for up to 4 CAN frames at 500 kbps or 8 CAN frames at 1 Mbps to lock in to the data rate. After these frames, the receiver is looking for the WUF on the bus.

Each device has its own unique WUF that will wake it up into standby mode. Once the CAN Bus receives this WUF, only the devices programmed to listen to that specific WUF are woken up, the rest continue to look for their own WUF, and after the $t_{SILENCE}$ timer expires they will all go back into sleep. Keep in mind that since the other transceivers that did wake up, if they are communicating the CAN bus will stay biased to 2.5 V and will no longer be in the deep sleep state. Similar to before, the level of current consumption during normal operation is much higher compared to when the nodes are in deep or regular sleep.

1.2 Why Use Partial Networking?

The main benefit of partial networking is decreased power consumption. CAN devices can generally be seen in three states: sleep mode (low current consumption), standby mode (higher current consumption), and normal mode (highest current consumption). Because each node uses significantly less current when it is asleep, the longer it is in sleep mode, the more power that can be saved. This is especially important in an automotive setting. All internal combustion vehicles use an alternator to recharge their battery while in use. If the vehicle is drawing more power than the alternator can supply, the battery will die. It is also important that when the engine is not running, current is minimized to maintain the charge on the battery. The decreased power consumption from partial networking also has a direct correlation to emissions in vehicles. Reducing the current draw of a car by 10 A leads to a lower CO2 emission of 3.5 g/km.[(2)]

2 Wake-Up Frames

2.1 Wake-Up Frame Structure

The Wake-Up Frame (WUF) is structured like a classical CAN frame defined in the ISO11898-1:2015 standard. It consists of the ID field and the Data field as well as the other required bits and fields like start-of-frame and CRC. The Device ID and Mask ID can be either 11 bits or 29 bits long, just like classical CAN frames, and the data payload can be up to 8 bytes.

2.2 Device ID and ID Mask

Within a partial networking system every CAN transceiver with selective wake is programmed with their own unique Device ID. This ID is always sent with the WUF. When a WUF is sent every selective wake transceiver can see it on the bus and they are all comparing it to their programmed Device ID and Mask ID. If the ID found in the WUF matches with the ID of the transceiver (and optionally, the DLC code and data mask have a match, see explanation in *Data Mask Bytes*), the transceiver enters standby mode. The wake request is reflected on the RXD pin, which directs the MCU to transition the device into normal mode. A Mask ID can be set for any number of transceivers when it is required that they exit sleep at the same time. In the Mask ID registers, a 1 represents a *do not care* bit. When a Mask ID is set for a transceiver, it declares what bits of the ID are unimportant for the transceiver to turn on. Any device whose ID matches with this mask can enter normal mode. Use an example with three CAN nodes.

Table 2-1. Example CAN Devices

CAN Node	ID
1	**00000000000000000000000000001**
2	**00000000000000000000000000010**
3	**11100000000000000000000000010**

If the ID Mask of CAN Node 2 is **00000000000000000000000000111**, it handles the last three bits of the ID field, in the WUF, as do not care bits. Any ID field whose first 26 bits are **00000000000000000000000000** wakes up Node 2. Therefore, when an ID of **00000000000000000000000000001** is sent in the WUF both Node 1 and 2 enter normal mode. However, when an ID of **00000000000000000000000000101** is sent, only Node 2 enters normal mode. In this case, Node 1 and 3 continue to sleep. Keep in mind that any part of the ID field can be masked out; not just the least significant bits. If the ID mask of every node is set to **11100000000000000000000000011**, then when any of the node IDs are sent in the WUF every node wakes up.

In an automotive setting there may be nodes that must wake up at the same time to perform a function. For instance, waking up the seat heater node to start heating the seats requires waking up the dashboard node to turn on the seat heater light at the same time.

2.3 Data Mask Bytes

Any node on a CAN bus can also be configured to turn on when a specific piece of data is sent in the WUF. Remember that every device must be configured to wake up to a specific ID. Every WUF must contain an ID field and every transceiver always first checks the ID field before exiting sleep mode. If an ID is not set on a transceiver it continues to sleep through the Data field of the WUF and does not enter normal mode. Even if the Data field matches with the Data Mask set on the transceiver, if the ID is not set on the device, it does not exit sleep mode. A Data Mask is typically used when multiple nodes must turn on at the same time.

In the classical CAN it is possible to program up to 8 bytes of data. Whenever a transceiver is expected to receive data to turn on, the Data Length Code (DLC) must be set to alert the transceiver of how many data bytes it should expect to receive. A DLC of at least 0001b must be set in order to set a Data Mask.

The Data Mask works a little differently than the ID Mask. For every byte received from the WUF, at least one bit must be 1b that matches a corresponding 1b bit in the same Data Mask byte. However, where the ID Mask and Data Mask are similar is that every 0b in the Data mask must correspond with a 0b in the received data. For example:

	Byte 1									**Byte 0**							
Configured Data Mask	1	1	0	1	0	0	0	0		0	0	1	0	1	1	0	0
Matching Data Frame	0	0	0	1	0	0	0	0		0	0	1	0	1	0	0	0
Non-Matching Data Frame	0	0	0	0	0	0	0	0		0	0	1	0	0	0	1	1

The DLC of this Data Mask would be 0010b which corresponds to two bytes. It may seem that byte 1 of the Non-Matching Data Frame does match up with the Mask since a 1 in the Mask typically specifies a do not care bit. However, as previously stated only one bit of the received data must be a 1b that corresponds with a 1b in the same bit of the Data Mask. As seen in the Matching Data frame, bit 4 is a 1b that matches with a 1b in bit 4 of the Data Mask.

3 Important Registers

- 10h : Mode configuration
  - Used to enable selective wake and to put the device into different modes of operation

Table 3-1. 10h

Bit	Field	Type	Reset	Description
7	SW_EN	R/W	0b	Selective wake enable for TCAN1145-Q1 and TCAN1146-Q1 otherwise reserved 0b = Disabled 1b = Enabled
6	DTO_DIS	R/W 0b	0b	Dominant time out disabled 0b = Enabled 1b = Disabled
5	FD_EN	R/W	0b	Fault detection enable for TCAN1144-Q1 and TCAN1146-Q1 otherwise reserved 0b = Disabled 1b = Enabled
4-3	RSVD	R	0b	Reserved
2-0	MODE_SEL	R/W	100b	Mode of operation select 1b = Sleep 100b = Standby 101b = Listen 111b = Normal NOTE: The current mode will be read back and all other values are reserved

- 30h-33h: Device ID selection
  - Used to set communication with the desired device ID
  - 32h: This register contains the IDE bit which is used to change the IDs from standard IDs (11 bits) to extended IDs (29 bits). This will also make it so that the Mask IDs are also changed.

Table 3-2. 30h

Bit	Field	Type	Reset	Description
7-0	Ext_ID_17:10	R/W	0b	Extended ID bits 17:10

Table 3-3. 31h

Bit	Field	Type	Reset	Description
7-0	Ext_ID_9:2	R/W	0b	Extended ID bits 9:2

Table 3-4. 32h

Bit	Field	Type	Reset	Description
7-6	Ext_ID_1:0	R/W	0b	Extended ID bits 1:0
5	IDE	R/W	0b	Extended ID field 0b = Standard ID (11-bits) 1b = Extended ID (29-bits)
4-0	ID_10:6__EXT_ID_28:24	R/W	0b	ID[10:6] and Extended ID[28:24]

Table 3-5. 33h

Bit	Field	Type	Reset	Description
7-2	ID_5:0__EXT_ID_23:18	R/W	0b	ID[5:0] and Extended ID [23:18]
1-0	RESERVED	R	0b	Reserved

- 34h-38h: Mask ID selection
  - Used to set the mask ID for the devices you would like to communicate with

Table 3-6. 34h

Bit	Field	Type	Reset	Description
7-2	Reserved	R	0b	Reserved
1-0	EXT_ID_MASK_17:16	R/W	0b	Extended ID Mask 17:16

Table 3-7. 35h

Bit	Field	Type	Reset	Description
7-0	EXT_ID_MASK_15:8	R/W	0b	Extended ID Mask 15:8

Table 3-8. 36h

Bit	Field	Type	Reset	Description
7-0	EXT_ID_MASK_7:0	R/W	0b	Extended ID Mask 7:0

Table 3-9. 37h

Bit	Field	Type	Reset	Description
7-0	ID_MASK_10:3__EXT_ID _MASK_28:21	R/W	0b	ID Mask 10:3 and Extended ID Mask 28:21 (Base ID)

Table 3-10. 38h

Bit	Field	Type	Reset	Description
7-5	SW_ID_Mask_5	R/W	0b	ID Mask 2:0 and Extended ID Mask 20:18 (Base ID)
4-1	DLC	R/W	0b	DLC [3:0]
0	DATA_MASK_EN	R/W	0b	Data mask enable 0b = DLC field and Data field are not compared and assumed valid. Remote frames are allowed. 1b = DLC field must match DLC[3:0] register and data field bytes are compared with DATAx registers for a matching 1. Remote frames are ignored

- 38h-40h: Data selection
  - Used to set the bits for the data that you would like to send
  - 38h: This register contains the bits for DLC and the bit for Data mask enable

Table 3-11. 39h + Formula

Offset = 39h + (y × 1h); where y = 0h to 7h for TCAN1145-Q1 and TCAN1146-Q1

Bit	Field	Type	Reset	Description
7-0	DATAx	R/W	0b	CAN data byte x

- 44h: General CAN data rate configuration
  - Used to set all data rate properties including frequency and CAN FD data rate ratio vs CAN data rate
    - CAN FD allows for devices to dynamically switch to different data-rates and message lengths

Table 3-12. 44h

Bit	Field	Type	Reset	Description
7	SW_FD_PASSIVE	R/W	0b	Selective Wake FD Passive: this bit modifies the behavior of the error counter when CAN with flexible data rate frames are seen. 0b = CAN with flexible data rate frame will be counted as an error frame 1b = CAN with flexible data rate frame are ignored (passive)
6-4	CAN_DR	R/W	101b	CAN bus data rate 0b = 50 kbps 1b = 100 kbps 10b = 125 kbps 11b = 250 kbps 100b = Reserved 101b = 500 kbps 110b = Reserved 111b = 1 Mbps
3-2	FD_DR	R/W	0b	CAN bus FD data rate ratio verses CAN data rate 0b = CAN FD <= 4x CAN data rate 1b = CAN FD => 5x and <= 10x CAN data rate 10b = Reserved 11b = Reserved
1-0	RESERVED	R	0b	Reserved

- 46h-47h: Selective Wake configuration
  - 46h: Used to set the threshold of the frame error counter. Once the threshold is passed the FRAME_OVF flag is thrown and the selective wake feature will be turned off.
  - 47h: Mainly a read only register used to make sure CAN frames are being decoded correctly
    - Bit 7 can be written to, letting the device know all the selective wake registers have been configured.
      - This flag is called the SWCFG bit

Table 3-13. 46h

Bit	Field	Type	Reset	Description
7-0	FRAM_CNT_THRESHOLD	R/W	00011111b	Frame Error Counter Threshold: these bits set the point at which the error counter reaches its maximum and on the next error frame will overflow and set the FRAME_OVF flag. Default is 31 so the 32nd error will set the overflow flag

Table 3-14. 47h

Bit	Field	Type	Reset	Description
7	SWCFG	RH/W	0b	Select wake configuration complete 0b = SW registers not configured 1b = SW registers configured (make this the last step in configuring and turning on selective wake) NOTE: Writing to any of these wake configuration registers (30h - 44h, 46h) clears the SWCFG bit.
6	CAN_SYNC_FD	RH	0b	The device is properly decoding CAN FD frames if frame detection is enabled. This flag is updated after every received frame. By polling this flag the system may determine if the device is properly decoding CAN FD frames, up to but not including the data field. This flag is self-clearing.
5	CAN_SYNC	RH	0b	Synchronized to CAN data: this flag indicates the device is properly decoding CAN frames if frame detection is enabled. This flag is updated after every received frame. By polling this flag the system may determine if the device is properly decoding CAN frames. This flag is self-clearing.
4-0	RESERVED	R	0b	Reserved

- 51h: Contains FRAME_OVF flag
  – Bit 3 of this register is the flag used to make sure the number of errors have not exceeded the threshold

Table 3-15. 51h

Bit	Field	Type	Reset	Description
7	WD	R/W1C	0b	Watchdog event interrupt. NOTE: This interrupt bit will be set for every watchdog error event and does not rely upon the Watchdog error counter
6	CANINT	R/W1C	0b	CAN bus wake up interrupt
5	LWU	R/W1C	0b	Local wake up
4	WKERR	R/W1C	0b	Wake error bit is set when the SWE timer has expired and the state machine has returned to Sleep mode
3	FRAME_OVF	R/W1C	0b	Frame error counter overflow
2	CANSLNT	R/W1C	0b	CAN silent
1	CANTO	R/W1C	0b	CAN timeout
0	CANDOM	R/W1C	0b	CAN bus stuck dominant

- 53h: Contains the SWERR flag
  – Bit 6 of this register is the flag used to make sure there is not an error setting up selective wake mode
    - This bit must be cleared after setting the SWCFG bit

Table 3-16. 53h

Bit	Field	Type	Reset	Description
7	SPIERR	R/W1C	0b	State when SPI status bit sets
6	SWERR	RH	0b	Logical OR of (SW_EN=1 and NOT(SWCFG)) and FRAME_OVF. Selective Wake may not be enable while SWERR is set
5	FSM	R/W1C	0b	Entered fail-safe mode. Can be cleared while in fail-safe mode.
4-1	RSVD	R	0b	Reserved
0	CRC_EEPROM	R/W1C	0b	EEPROM CRC error

4 Configuring Partial Networking

Configuring The Wake-Up Frame

1. Write all control registers for frame detection (selective wake).
   - Set all bits of **(30h–46h)** to desired specifications
2. Read all SW registers **(30h–46h)** that were just written to confirm that they were written to correctly.
3. Set SWFCG bit to a 1 **(47h[7])**
4. Selective wake enabled (SW_EN) is set, and set device into desired mode
   - Set SW_EN bit to a 1 **(10h[7])**
   - SPI write to **(10h[2:0])** = 100b for standby mode, **(10h[2:0])** = 001b for sleep mode.

Note

The wake-up frame (WUF) must be configured for the device to detect that it is being called to wake up. The above process of setting the SWCFG must be done first to load the WUF parameters into the device. If a SWERR interrupt then occurs from the Frame Overflow flag, the Frame Overflow interrupt needs to be cleared, and then the SWCFG bit must be set again to 1.

Note

If the FRAME_OVF flag is set, or if a fault condition forces the device into sleep mode (fail-safe mode disabled), or into fail-safe mode, SW_EN is disabled turning off the selective wake function. The FRAME_OVF flag must be cleared and the SW_EN and SWCFG bits must be set again to re-enable selective wake.

5 Example Configuration

Table 5-1 provides examples of what occurs under different conditions.

Table 5-1. Example Configuration

Register	Bit	What Happened
32h	[5]=0b	Selects standard ID field (11 bits)
32h	[4:0]=00000b	Sets bits 10-6 of device ID to 00000b
33h	[7:2]=000011b	Sets bits 5-0 of device ID to 000011b Final Device ID = 00000000011b
38h	[4:1]=0010b	Sets DLC to be 2 bytes of Data
38h	[0]=1b	Enables Data Mask
39h	[7:0]=00101100b	Sets Byte 0 of Data Mask to 00101100b
3Ah	[7:0]=11010000b	Sets Byte 1 of Data Mask to 11010000b
44h	[7]=1b	CAN with flexible data rate frame are ignored (passive)
44h	[6:4]=101b	CAN bus data rate is set to 500 Kbps
44h	[3:2]=0b	CAN FD <= 4x CAN data rate
45h	[7:0]=0b	Resets the error counter
46h	[7:0]= 00011111b	Sets the error counter threshold back to default (31). On the 32nd Error the frame overflow (FRAME_OVF) flag will be set.
47h	[7]=1b	The SWCFG bit acknowledges that all SW registers are configured
10h	[7]=1b	Enables Selective Wake (SW)
10h	[2:0]=001b	Puts device in sleep mode

At the end of this configuration the Device ID is 00000000011b, and the Data Mask is 11010000b 00101100b. With this ID and data payload defined, stuff bits, and the rest of the CAN frame as defined by ISO11898-1:2015, the entire CAN WUF is 000001000001110000011011010000010101100101000110111011101b.

In Figure 5-1, the RXD pin, INH pin, and CANH – CANL are captured. RXD and INH both indicate when the correct WUF is received (RXD biases low and INH biases high). The orange waveform is CANH – CANL, the yellow waveform is RXD, and the light blue waveform is INH. Five CAN frames are sent as shown in the green rectangle. The first one is the wake-up signal to the CAN bus so that the bias moves from GND to 2.5 V, and the next four are the synchronization waveforms. After these, the WUF receiver is ready to receive the WUF, which is inside the purple rectangle in Figure 5-1. Immediately after the last bit of the pink rectangle, the INH pin biases from low to high, indicating a state change from sleep mode to standby mode and a successful wake using the WUF. The INH rising edge is circled in red.

Figure 5-1. Example Configuration WUF Oscilloscope Capture

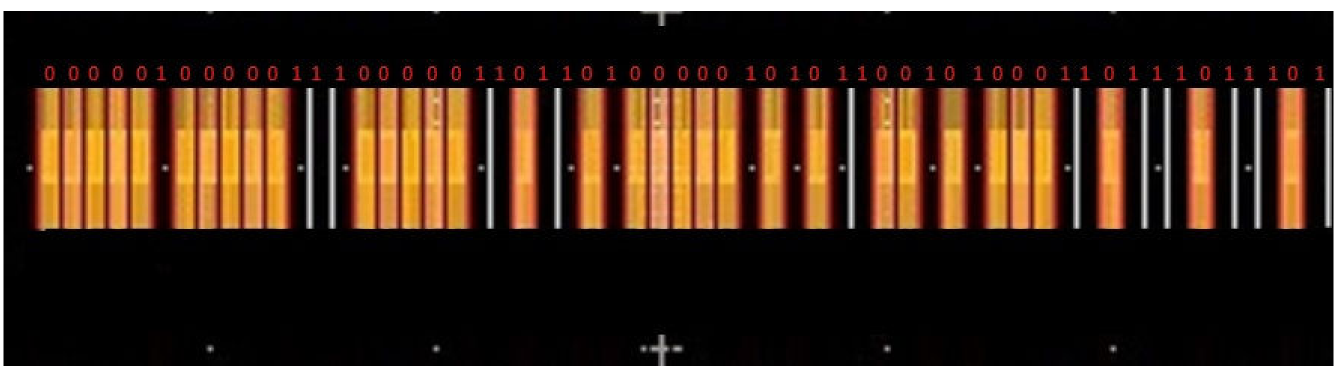

Figure 5-2. Example Configuration WUF Oscilloscope Capture Zoomed in

In Figure 5-2, the wake-up frame portion of the CAN message has been magnified to see the individual bits. Lines have also been added to separate the individual bits for easier reading and each bit has been labeled with a 1 or a 0 depending on the logic level, with 0 interpreted as a dominant bit, and 1 interpreted as recessive. This is because the orange waveform is CANH – CANL, and a dominant bit, which is interpreted as 0, occurs when the difference between CANH and CANL is greater than 0.9 V, and recessive, which is interpreted as a 1, when the difference between CANH and CANL is less than 0.5 V. As can be seen, the WUF that woke the device matches the CAN frame that results from the example configuration in Table 5-1.

6 Summary

Partial networking is an extremely important function within CAN systems. It allows for a reduction of power consumption, which has a direct effect on vehicle emissions. Since there are many applications for partial networking, it is important that any engineer working with CAN is aware of and understands partial networking. This application report serves as a good guide to any engineer learning partial networking and implementing it using the TCAN1145-Q1.

7 References

1. Texas Instruments, *TCAN114x-Q1 Enhanced CAN FD Transceiver with Partial Networking Data Sheet*
2. *CO_2 Emissions From Cars: the facts*