GCAN-601

Embedded UART-CAN

User Manual

Document version 4.50(2017/12/01)

Revision History:

Version	Date	Reason
V1.00	2013/06/16	Create the file
V2.01	2013/12/20	Modify working converter parameter
V3.01	2015/04/22	Add portion parameter
V3.50	2016/07/16	Add OBD II function
V4.01	2017/01/13	Add CANopen function
V4.50	2017/12/01	Optimize the file

Contents

1 Introduction

1.1Overview

The GCAN-601 is a communication converter that integrates with one CAN-Bus interface, one UART interface. The GCAN-601 can be directly connected to the UART interface of MCU, which can make a CAN-Bus communication without changing its original hardware.

In normal operation, the GCAN-601 module is in a state of real-time monitoring of the CAN bus and the UART bus. Once a data is received on one bus, it is immediately parsed and loaded into its own buffer. According to the set working mode, the data on the bus is sent to the other bus to realize the conversion of the data format.

1.2 Properties at a glance

- Standard UART level, can be directly connected with MCU
- Support serial baud rate 600bps~921600bps
- Supply three kinds of conversion modes: transparent conversion, transparent conversion with identity, format conversion
- Configure three kinds of conversion directions: bidirectional conversion, CAN→ serial interface, serial interface→ CAN
- CAN-Bus support CAN2.0A and CAN2.0B frame format, conform to ISO/DIS11898 specification
- Support CAN-Bus communication baud rate: 5Kbps~1Mbps
- Use 3.3V DC power supply
- Save configured parameters in storage, call the latest parameters automatically after supplying power
- Working temperature: -40.00℃~ + 85.00℃
- Dimensions: 22mm(L) ×13mm(W)

2 Usage

2.1 Interface definition and package size

2.1.1 Interface definition

GCAN-601 module interface definition as shown in Figure 2.1, the specific meaning of each pin is shown in Table 2.1.

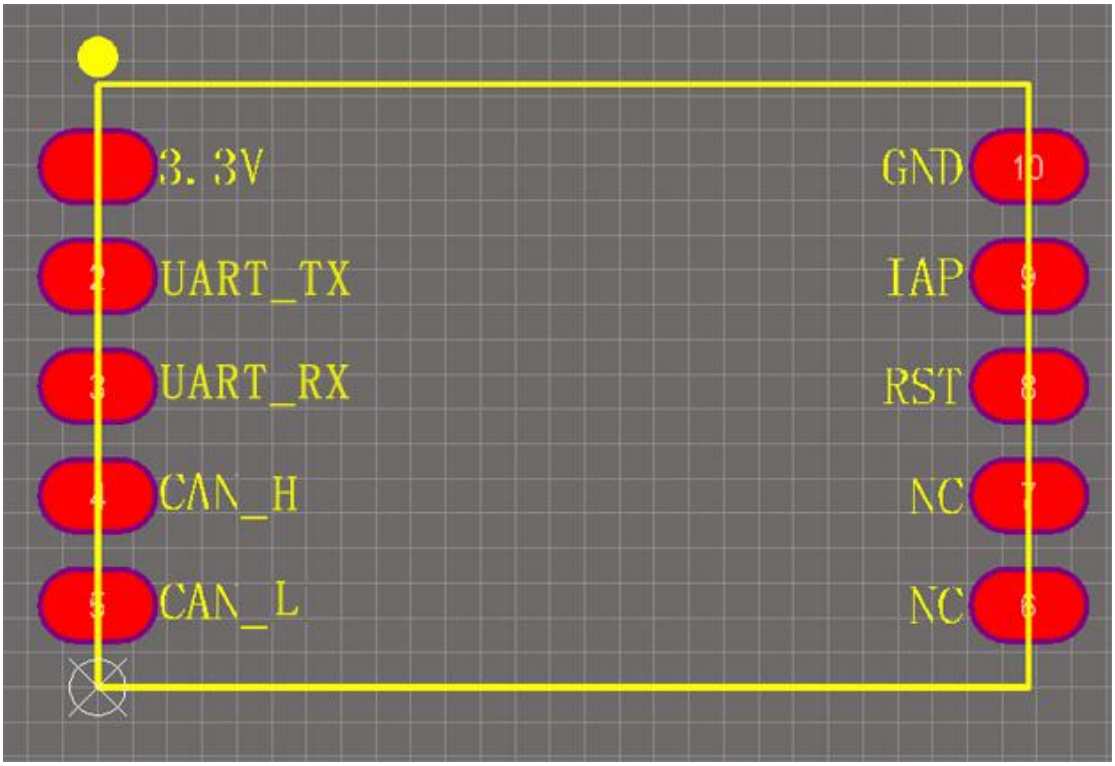


Figure 2.1 GCAN-601 interface definition

Pin	Name	Definition
1	3.3V	Power +3.3V interface
2	UART_TX	UART TX pin
3	UART_RX	UART RX pin
4	CAN_H	Connect CAN_H
5	CAN_L	Connect CAN_L
6	GND	GND interface
7	IAP	Update kernel
8	RST	Reset pin
9/10	NC	No connection

Table2.1 GCAN-601 pin definition

2.1.2 Package size

The package size of GCAN-601 module is shown in Figure 2.2

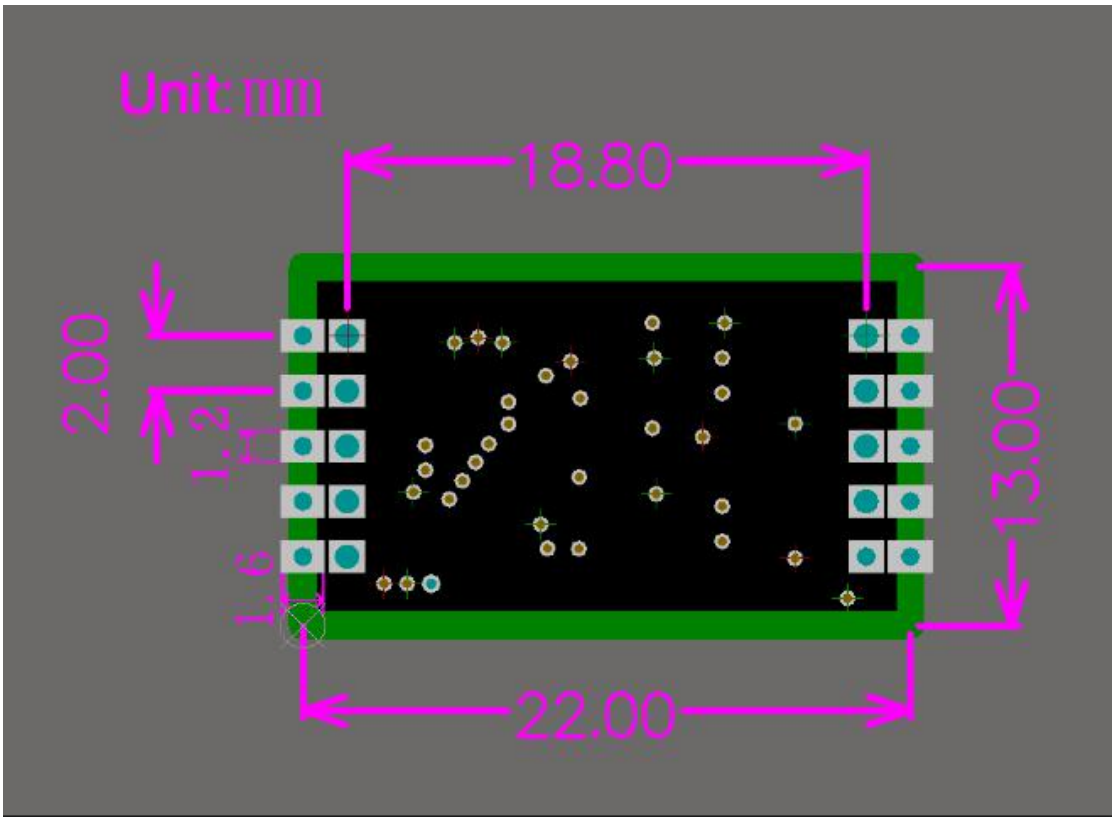
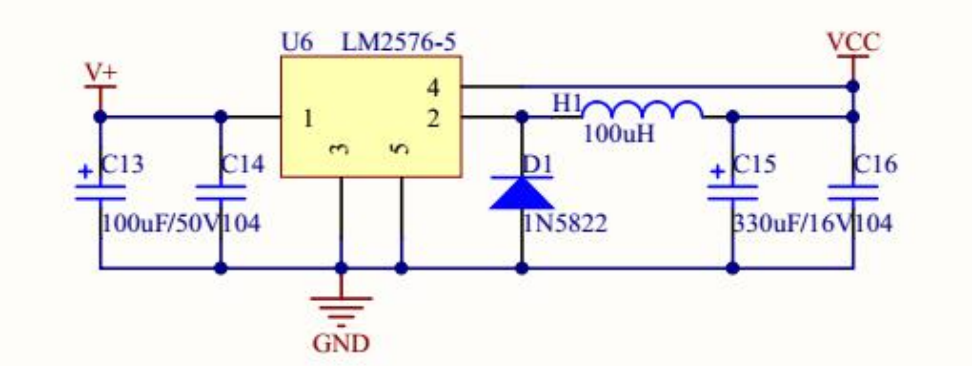


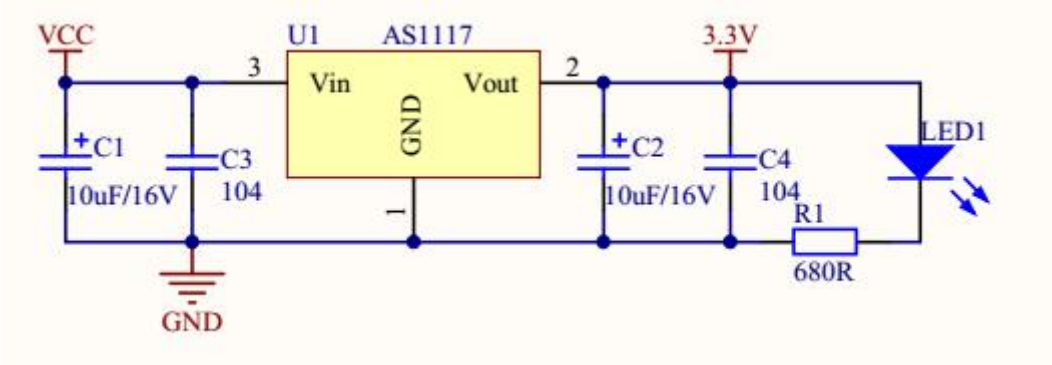
Figure 2.2 GCAN-601 package size

2.2 Typical application circuit

2.2.1 Power module

The power circuit mainly includes LM2576 power module and AMS1117 module. LM2576 power module uses for power supply, rated voltage 9-30V. The AS1117 module is mainly used for voltage conversion, and converts 5V to 3.3V to power the GCAN-601 module.

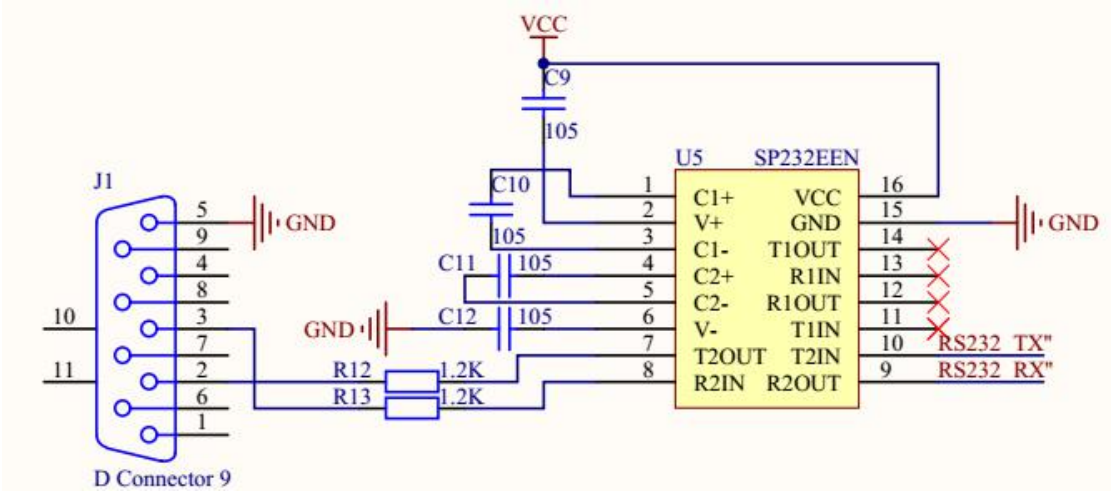




2.2.2 Serial communication

Serial communication mainly adopts RS232 transceiver SP232EEN.

Note : the TXIN of the RS232 transceiver chip needs to be connected to the TX interface of the GCAN-601 module, RXOUT is connected to the RX interface of the GCAN-601 module, and the VCC is 5V.



2.3 Working mode

GCAN-601 has three operating modes, normal mode, configuration mode and upgrade kernel.

Mode	Function
Normal mode	General mode of module leaving the factory
Configuration mode	In normal mode, connect the IAP pin to 3.3V and the module will enter the configuration mode. (RET button / re-powered to return to normal mode.)
Upgrade kernel	Connect the IAP pin to 3.3V, and then power up to enter

	IAP mode. The kernel can be upgraded through the UART. (This mode should be used under the guidance of our engineer.)

2.4 Serial connection

The GCAN-601 uses a standard UART level (3.3V), so the module can be directly connected to the MCU's UART interface.

2.5 CAN connection

The GCAN-601 must be installed two 120 ohms terminal resistance at the farthest end of CAN-Bus network; If the node number is more than 2, intermediate nodes needn't to install 120 ohms terminal resistance. For branch connections, the length should not exceed 3 meters.

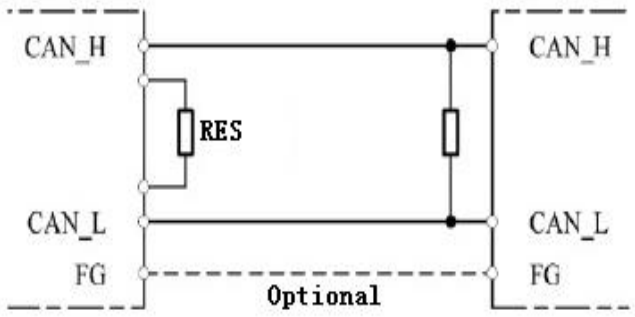


Figure2.3 GCAN-601 connect other nodes devices

Note: CAN-Bus cable can be used with ordinary twisted-pair and shielded twisted-pair.

Baud rate	Distance
1 Mbit/s	40m
500 kbit/s	110m
250 kbit/s	240m
125 kbit/s	500m
50 kbit/s	1.3km
20 kbit/s	3.3km
10 kbit/s	6.6km
5 kbit/s	13km

Table2.2 the relationship between CAN-Bus length and baud rate

3 Configuration

The GCAN-601 module can be configured in two ways, namely: using computer software to configure through the computer serial port and serial port AT command. Here mainly uses the computer configuration as an example to describe in details the various operating parameters of the module.

3.1 Configure preparation

The GCAN-601 module is powered on, and the IAP pin is connected to 3.3V and the module enters the configuration mode. Users can connect to PC using USB to UART serial cable or standard serial cable + adapter board. Enter the Device Manager to find the serial port number. As shown in Figure 3.1.

Figure 4.1 setting manager

Note: User can modify the serial port number in the setting manager.

3.2 Connect software

When the GCAN-601 module enters the configuration mode and is connected to the PC through the serial port, open the "CANCOM-Config" software on the CD to configure the module. The software interface is shown in Figure 3.2.

Figure 4.2 the main screen of CANCOM-Config

Select the serial interface number of the adapter, and click “connect” to make a connection. If user don’t know the serial port number, can enter the setting manager to view on the computer.

Click “connect”, popup “read parameters successfully”, indicating that have already read module configuration state.

3.3 Data conversion mode

Support three kinds of working mode: transparent conversion, transparent conversion with identity, format conversion.

Support three kinds of conversion directions: bidirectional conversion, CAN→ serial interface, serial interface→ CAN.

Figure 3.3 the setting of working mode

Note: Select the "convert direction", can eliminate the data interference on the Bus side don't need to convert.

3.3.1 Transparent conversion

Transparent conversion is that make a kind of Bus data format convert the other format, and keep the data without additional data and data modified.

"**Permit frame information transmits to serial frame**" is used in "Transparent conversion" mode only. If you check it, the converter will add frame information of CAN message into the first byte of serial frame in working mode.

"**Permit frame ID transmits to serial frame**" is used in "Transparent conversion" mode only. If you check it, the converter will add frame ID of CAN message between frame data and frame information in working mode. (In the state of permitting to convert frame information)

If you check the "**Permit frame ID transmits to serial frame**", you can set "send identifier", and the identifier is the frame ID of serial convert CAN-Bus.

3.3.2 Transparent conversion with identity

Transparent conversion with identity is a special transparent conversion.

In this kind of working mode, can convert the first two bytes(standard frame) or the first four bytes(extend frame) of serial frame as CAN ID.

"The interval time between serial frames" is used in "Transparent conversion with identity" mode only. The scope of interval time is the time from 1 bit to 20 bits between the two serial frames.

Note: The real interval time can't less than the setting.

3.3.3 Format conversion

Format conversion is the simplest mode. The data format is 13 bytes and contains all CAN frame information.

In the format conversion mode, the original CAN data frame can be transmitted and received through the serial port. Format conversion uses a common data definition, which is very convenient for users to carry out secondary development.

3.3.4 Configure transmission direction

Three kinds of conversion direction: bidirectional conversion, CAN→ serial interface, serial interface→ CAN.

3.4 Configure serial parameters

The GCAN-601 module supports the serial port baud rate from 600bps to 115200bps, and other parameters can set it in your need.

Figure3.4 the setting of serial interface parameters

3.5 Configure CAN parameters

Support CAN baud rate: 1000K, 500K, 250K, 200K, 125K, 100K, 50K, 20K, 10K.

Support CAN frame type: standard frame, extend frame.

“Frame type” parameter is valid in “Transparent conversion” or “Transparent conversion with identity” mode. “Transmit identifier” parameter is valid in “Transparent conversion” mode only.

The GCAN-601 module also can filter the data.

Figure 4.5 the setting of CAN parameters

3.6 Finish the configuration

Configured the GCAN-601 module, can click “write configure” to download the parameters.

Downloaded the parameters, need to repower for the module, and the new configuration will be valid.

Note: Click “default” to recover all configuration parameters into default status. Default working mode is transparent conversion mode, and permit CAN frame ID and frame information to transfer the serial frame. (The frame type of CAN interface is standard frame). The baud rate of default serial port is 115200bps, CAN is 1000kbps.

3.7 Use AT command to configure the parameters

The GCAN-601 module is not only configured by software, but also by AT command. The AT commands are all transmitted in the form of ASC codes. All English letters must be in all uppercase letters, and all commands end in carriage return and line

break: \r\n(0x0D 0x0A). In configuration mode, the serial port baud rate needs to be set to 57600, the data length is 8, the parity check is none, and the stop bit is 1.

3.7.1 Read the module information

Use ATI instructions to read device information, for example:

Send the instruction
>ATI
Receive
>$GCAN=GC000000002,18051401,UARTCAN

3.7.2 Restore the default setting

In transparent mode, users can restore default settings by ATDE commands. The serial port baud rate needs to be set to 115200, and the CAN parameter baud rate is set to 1000k.

For example:
Send the instruction
>ATDE
Receive
>ATEDOK

3.7.3 Conversion mode

User can use the ATM command to set conversion modes.

Number	Conversion mode
1	Transparent conversion
2	Transparent conversion with identity
3	Custom conversion

For example:
Send the instruction
>ATM
Receive
>ATM=1
Send the instruction
>ATM=1
Receive
> ATMOK

3.7.4 Set the transmission direction

User can use the ATD command to set the transmission direction.

Number	Transmission direction
1	Bidirectional
2	Serial→CAN
3	CAN→Serial

For example:

Send the instruction

>ATD

Receive

>ATD=1

Send the instruction

>ATD=1

Receive

> ATDOK

3.7.5 CAN frame information to be forwarded to serial frame

User can use the ATCI command to allow CAN frame information to be forwarded to the serial frame, which is only used in "**transparent conversion**" mode.

For example:

Send the instruction

>ATM

Receive

>ATCI=1

Send the instruction

>ATCI=1

Receive

> ATCIOK

3.7.6 CAN frame identification to be forwarded to serial frame

User can use the ATCID command to allow the CAN frame identification to be forwarded to the serial frame, which is only used in "**transparent conversion**" mode.

For example:

Send the instruction

>ATCID

Receive

> ATCID =1

Send the instruction

> ATCID =1

Receive

> ATCIOK

3.7.7 Configure CAN frame identity in the position of serial frame

User can configure the CAN frame identification in the position of the serial frame using the ATSA command, which is only used in the "**transparent conversion with identity**" mode.

For example:

Send the instruction

>ATSA

Receive

> ATSA =1

Send the instruction

> ATSA =1

Receive

> ATSAOK

3.7.8 Configure frame ID length

User can configure the frame ID length using the ATAL command.

Standard frame can be filled with 1 to 2 bytes, corresponding to ID1, ID0 of CAN message.

The extended frame can be filled with 1 to 4 bytes, corresponding to ID3, ID2, ID1, ID0. ID3 is the highest bit and ID0 is the lowest bit.

For example:

Send the instruction

>ATAL

Receive

> ATAL =2

Send the instruction

> ATAL =2

Receive

> ATALOK

3.7.9 Set UART baud rate

User can set the serial UART baud rate using the ATRSB instruction.

Number	Baud rate
1	115200
2	57600
3	56000
4	43000
5	38400
6	19200
7	9600
8	4800
9	2400
10	1200
11	600
12	921600
13	460800
14	230400

For example:

Send the instruction

>ATRSB

Receive

> ATRSB =1

Send the instruction

> ATRSB =2

Receive

> ATRSBOK

3.7.10 Set CAN baud rate

User can set the CAN baud rate using the ATCB command.

Number	Baud rate
1	1000k
2	500k
3	250k

4	200k
5	125k
6	100k
7	50k
8	20k
9	10k
10	5k

For example:

Send the instruction

>ATCB

Receive

> ATCB =1

Send the instruction

> ATRC =1

Receive

> ATCTOK

3.7.11 Configure frame mode

Users can configure frame mode using ATCT command.

Number	Mode
1	Standard frame
2	Extended frame

For example:

Send the instruction

>ATCT

Receive

> ATCT =1

Send the instruction

> ATCT =1

Receive

> ATCTOK

3.7.12 Send frame identifier

User can send frame identifier using the ATSS command, which is only used in "**transparent conversion**" mode.

For example:

Send the instruction

>ATSS

Receive

> ATSS =127

Send the instruction

> ATSS =100

Receive

> ATSSOK

3.7.13 Filter enable

User can use the ATEF instruction to enable filtering.

For example:

Send the instruction

>ATEF

Receive

> ATEF =1

Send the instruction

> ATEF =1

Receive

> ATEFOK

3.7.14 Set the filter group number

User can set the filter group number using the ATFGN instruction, up to 12 groups.

For example:

Send the instruction

>ATFGN

Receive

> ATFGN =2

Send the instruction

> ATFGN =1

Receive

> ATFGNOK

3.7.15 Set the check mode

User can set the check mode using the ATRSL instruction.

Number	Mode
1	No parity
2	Odd parity
3	Even parity

For example:

Send the instruction

>ATRSL

Receive

> ATRSL =1

Send the instruction

> ATRSL =1

Receive

> ATRSLOK

3.7.16 Set filter mode

Users can use the ATFilter command to filter the setting mode.

For example：

Send the instruction

>ATFilter

Receive

>ATFGN=2

mfnum=1//number

mfex=1//extend

mfbagin=22//start ID

mfend=33//end ID

Send the instruction

>ATFilter=1;1;22;33;

Receive

> mfnum=1

mfex=1

mfbagin=22

mfend=33

ATFilter

4 Application sample

4.1 Transparent conversion

Details see the "**3.3.1 Transparent conversion".**

4.1.1 Frame format

Serial frame: data flow or data with protocol, communication format: 1 start, 8 data, 1 stop

CAN-Bus frame: keep the CAN message format

4.1.2 Conversion mode

1. Serial frame convert CAN message

All of serial frame data convert into data area of CAN message. Frame ID is configured by user, as shown figure 4.1.

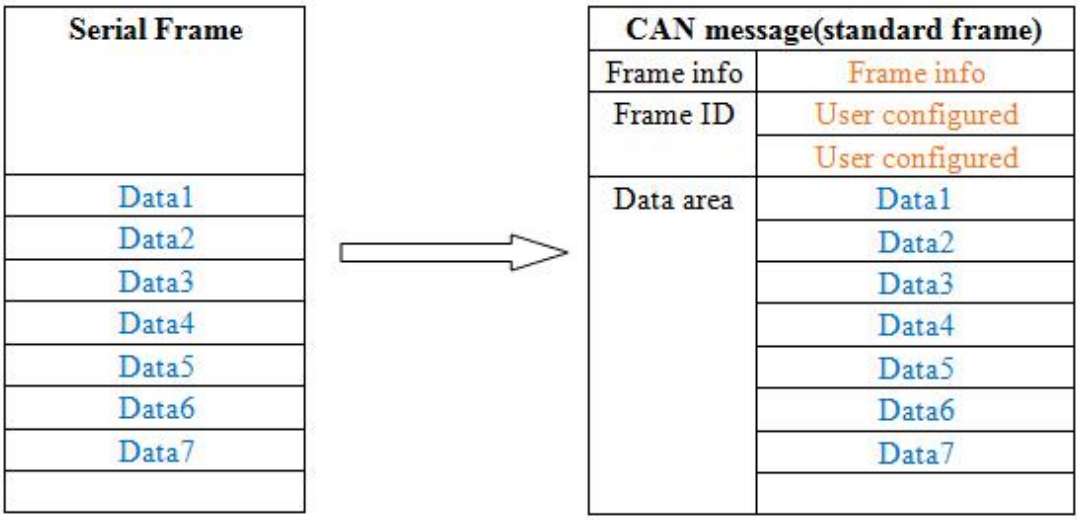


Figure 4.1 serial frame convert CAN message (transparent conversion)

2. CAN message convert serial frame

Check "**Permit frame information transmits to serial frame**" and "**Permit frame ID transmits to serial frame**", as shown figure 4.2.

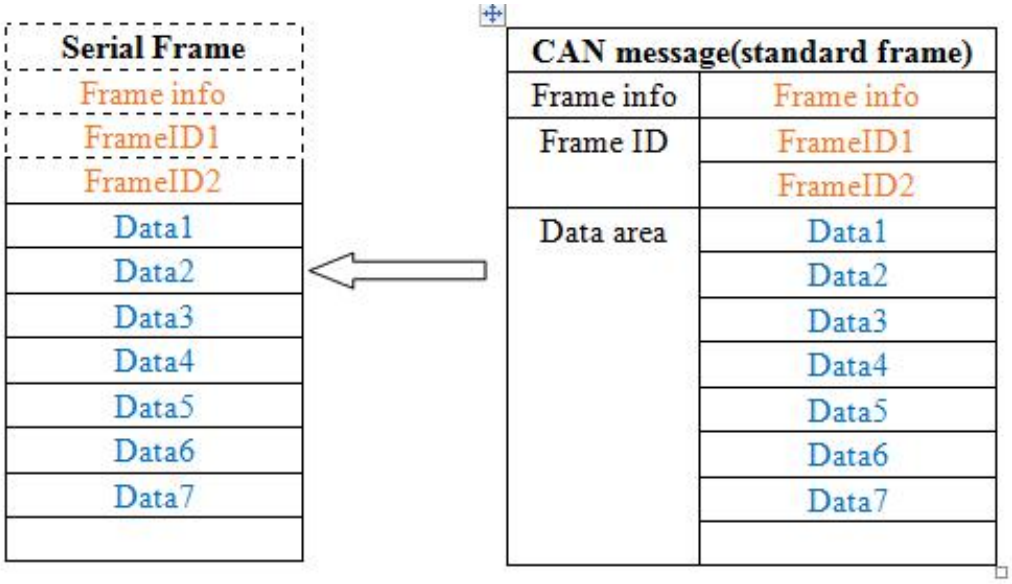


Figure 4.2 CAN message convert serial frame (transparent conversion)

4.2 Transparent conversion with identity

Details see the **"3.3.2 Transparent conversion with identity".**

4.2.1 Frame format

1. Serial frame

Details see the "**3.3.2 Transparent conversion with identity".**

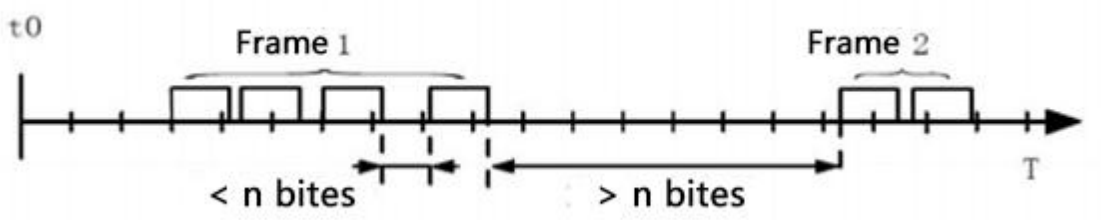


Figure 4.3 time format of serial frame (transparent conversion with identity)

2. CAN-Bus frame

Keep CAN message format, just convert corresponding to frame ID of CAN into the serial frame.

4.2.2 Conversion mode

1. Serial frame convert CAN message

The initial position ranges from 1~7, length ranges from 1~2 (standard frame) or 1~4 (extend frame).

Suggested values:

For standard frame: initial position 1, standard frame 2.

For extend frame: initial position 1, extend frame 4.

For example of standard frame:

Serial Frame	
Adr1	Data1
Adr2	Data2
Adr3	Data3
Adr4	Data4
Adr5	Data5
Adr6	Data6
Adr7	Data7
Adr8	Data8
Adr9	Data9
Adr10	Data10
…	…
Adr n	Data n

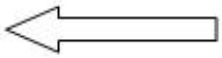
	CAN message 1	CAN message…	CAN message x
Frame info	User configured	User configured	User configured
Frame ID1	Data1	Data1	Data1
Frame ID2	Data2	Data2	Data2
Data area	Data3	Data…	Data n-3
	Data4	Data…	Data n-2
	Data5	Data…	Data n-1
	Data6	Data…	Data n
	Data7	Data…	
	Data8	Data…	
	Data9	Data…	
	Data10	Data…	

Figure 4.4 serial frame convert CAN message (transparent conversion with identity)

2. CAN message convert serial frame

For example of standard frame:

Serial Frame
Data1
Data2
Data3
Data4
Data5
Data6
Data7
Data8
Data9
Data10



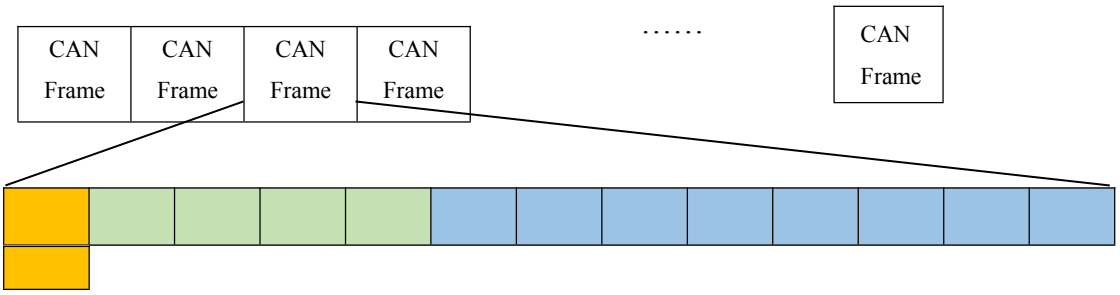
CAN message(standard frame)	
Frame info	Frame info
Frame ID	Frame ID1
	Frame ID2
Data area	Data3
	Data4
	Data5
	Data6
	Data7
	Data8
	Data9
	Data10

Figure4.5 CAN message convert serial frame (transparent conversion with identity)

4.3 Format conversion

The GCAN-601 module has data conversion format, as shown the picture.

Every CAN frame contains 13 bytes, and 13 bytes contains CAN frame + frame ID + frame data.



Frame information: 1 byte length, can identify some information of CAN frame, such as type, length so on.

FF	RTR	Retention	Retention	D3	D2	D1	D0

BIT7 BIT0

FF: the identifier of standard frame and extend frame, 1 extend frame, 0 standard frame.

RTR: the identifier of remote frame and data frame, 1 remote frame, 0 data frame.

Retention: retention 0, can't write 1.

D3~D0: data length, identity the data length of CAN frame

Frame ID: 4 bytes length, 11 bits of standard frame are valid, 29 bits of extend frame are valid.

low byte high byte low byte high byte

12h	34h	56h	78h

Extend frame: 0x12345678

00h	00h	01h	23h

Standard frame: 0x123

Frame data: 8 bytes length, the valid length decide by frame information D3~D0.

DATA1 DATA8

11h	22h	33h	44h	55h	66h	77h	88h

As above is the valid data of 8 bites.

DATA1 DATA8

11h	22h	33h	44h	55h	00h	00h	00h

As above is the valid data of 5 bites.

For example:

88h	12h	34h	56h	78h	11h	22h	33h	44h	55h	66h	77h	88h

A extend frame data, frame ID 0x12345678, contains the valid data of 8 bytes.

05h	00h	00h	01h	23h	11h	22h	33h	44h	55h	00h	00h	00h

A standard frame data, frame ID 0x123, contains the valid data of 5 bytes.

Note: Every frame must be 13 bytes, must complement 0 if don't have enough 13 bytes. Otherwise lead to make a communication error.

5 Technical specifications

Connection mode	
Serial port	PCB pad/2.08 raw pin
CAN port	PCB pad/2.08 raw pin
Interface feature	
Serial interface	UART
Serial port baud rate	600bps~921600bps
CAN interface	ISO 11898 standard, CAN2.0A/B
CAN baud rate	1000K、500K、250K、200K、125K、100K、50K、20K、10K、5K
CAN terminal resistant	Hasn't integrated, you can add it between CAN_H and CAN_L in your need
Power supply	
Voltage	+3.3V DC
Current	100mA（MAX）
Environmental test	
Working temperature	-40℃~+85℃
Working humidness	15%~90%RH, without condensation
Basic information	
Dimension	22mm *13mm
Weight	10g

Appendix : CAN2.0B protocol frame format

CAN2.0B standard frame

CAN standard frame information is 11 bytes, including two parts: information and data parts. The three bytes at the beginning are the information parts.

Byte \ Bit	7	6	5	4	3	2	1	0
1	FF	RTR	×	×	DLC			
2	(Message identifier)				ID.10—ID.3			
3	ID.2—ID.0			×	×	×	×	×
4	data 1							
5	data 2							
6	data 3							
7	data 4							
8	data 5							
9	data 6							
10	data 7							
11	data 8							

Byte 1: frame information

Bit7 (FF): frame format

FF = 0: standard frame

Bit6 (RTR): type of frame

RTR=0: data frame

RTR=1: remote frame

(DLC): the actual data length in the data frame.

Byte 2 and byte 3: message id and 11 are valid.

Byte 4 ~ 11 is the actual data of the data frame, remote frame (RTR=1) is invalid.

CAN2.0B extend frame

CAN extend frame information is 13 bytes, including two parts, information and data.

The five bytes at the beginning are the information parts.

Byte \ Bit	7	6	5	4	3	2	1	0
1	FF	RTR	×	×	DLC			
2	(Message identifier) ID.28—ID.21							
3	ID.20—ID.13							
4	ID.12—ID.5							
5	ID.4—ID.0					×	×	×
6	data 1							
7	data 2							
8	data 3							
9	data 4							
10	data 5							
11	data 6							
12	data 7							
13	data 8							

Byte 1: frame information

Bit 7(FF): frame format

FF = 1, in the extend frame

Bit6 (RTR): type of frame

RTR=0: data frame

RTR=1: remote frame

(DLC): the actual data length in the data frame.

Byte 2 ~ 5: message id numbers and 29 are valid.

Byte 6 ~ 13 is the actual data of the data frame, remote frame (RTR=1) is invalid.

Sales and service

Shenyang Guangcheng Technology Co., Ltd.

Address: Industrial Design Center, No. 42 Chongshan Middle Road, Huanggu District, Shenyang City, Liaoning Province.

QQ: 2881884588

E-mail: 2881884588@qq.com

Tel: +86-24-31230060

Website: www1.gcanbox.com

Sales and service Tel: +86-18309815706

After - sales service telephone Number: +86-13840170070

WeChat Number: 13840170070