

# DFL-CAN232 CANBUS to RS232 Bidirectional Conversion IC

(Pb Free )

## Features:

- Support CAN2.0A and CAN2.0B Protocol.
- One CAN bus interface with configurable baudrate between 20K to 1M
- One RS232 interface with configurable baudrate between 1200 to 230400
- Support the transparent transform with or without message ID
- Support dynamic change of CAN Bus baudrate, mask and filter
- Support 8 discrete digital Inputs and 8 discrete digital outputs.
- Discrete digital input can be sampled periodically or when change happens
- Discrete digital input support remote frame response
- Discrete digital output can be protected when CAN bus fault occurs and protection value can be configured

## Application fields:

- CANopen, DeviceNET, OBDII ISO 15765 or J1939 protocol interpreter or ECU Simulator.
- Use RS232 to connect CAN Bus network
- Link 2 CAN BUS to increase communication distance
- I/O extension using CAN bus
- I/O extension using RS232
- PNP ( or NPN ) Proximity Sensors are changed to NPN ( or PNP ) Proximity Sensors

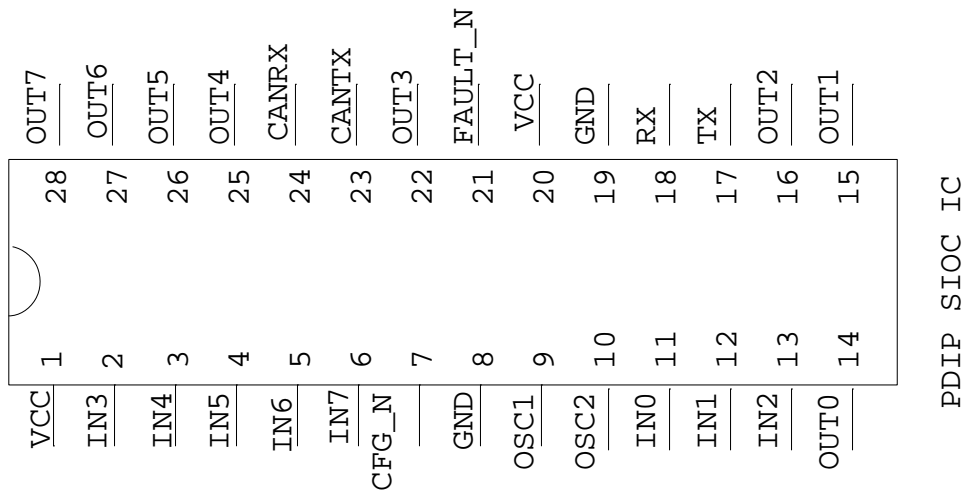
## 1 Introduction

DFL-CAN232 is an IC, which implements the bidirectional transparent transfer between CAN BUS 2.0A/B and RS232. The Formatting CAN BUS frame can be automatically produced, the user only needs to give the data to the IC's UART from host RS232, Similarly, and the IC's UART only transmits the data to host RS232 by hiding the formatting bytes. That is called transfer without message ID. The message ID can be configurable. Of course, the user can transmit the formatting frame to the IC's UART from host RS232, and the IC's UART transmits the CAN bus formatting frame to host RS232 without any hiding bytes. That is called transfer with message ID. DFL-CAN232 contains 8 discrete digital inputs and 8 discrete digital outputs. The input can be configured as periodic transmitting or transmitting only when input is changed. The period value is 1 ms to 65535ms. Of course, the user can ask the input value at any time using remote frame. The Output can be a special value when

the CAN BUS has fault or kept the same value as before.

## 2 Pins Assignment

DFL-CAN232 has 28pin PDIP and SOIC package.



The figure above is the pins assignment of DFL-CAN232.

Pin1	VCC	Pin15	OUT1
Pin2	IN3	Pin16	OUT2
Pin3	IN4	Pin17	TX
Pin4	IN5	Pin18	RX
Pin5	IN6	Pin19	GND
Pin6	IN7	Pin20	VCC
Pin7	CFG_N	Pin21	Fault_N
Pin8	GND	Pin22	OUT3
Pin9	OSC1	Pin23	CANTX

Pin10	OSC2	Pin24	CANRX
Pin11	IN0	Pin25	OUT4
Pin12	IN1	Pin26	OUT5
Pin13	IN2	Pin27	OUT6
Pin14	OUT0	Pin28	OUT7

Vcc: Power pin. Connect to the positive side of DC power supply which is from 4.2 to 5.5 VDC

IN3: Input pin. It is a general purpose digital input pin, which is bit 3 of a input byte

IN4: Input pin. It is a general purpose digital input pin, which is bit 4 of a input byte

IN5: Input pin. It is a general purpose digital input pin, which is bit 5 of a input byte

IN6: Input pin. It is a general purpose digital input pin, which is bit 6 of a input byte

IN7: Input pin. It is a general purpose digital input pin, which is bit 7 (MSB) of a input byte

CFG\_N: Input pin. It is configuration digital input. The IC will enter configuration mode if this pin is low level when power on. The only way to exit configuration mode is power off , and this pin keep high level when power on again.

The IC will enter the normal mode if this pin is high level when power on. This is the IC 's normal running method. In the normal mode, IC will change the CAN Bus Baudrate and MASK and Filter if the pin changes to low level. IC will use the new value when pin returns to high level.

GND: Power pin. Connect to the negative side of DC power supply

OSC1: input pin. It 's one pin of 4MHz Parallel-cut crystal or resonator, or a direct clock input

OSC2: Output pin. It 's other pin of 4MHz Parallel-cut crystal or resonator, or leave it unconnected if OSC1 driven with a clock input

IN0: Input pin. It is a general purpose digital input pin, which is bit 0 (LSB) of a input byte

IN1: Input pin. It is a general purpose digital input pin, which is bit 1 of a input byte

IN2: Input pin. It is a general purpose digital input pin, which is bit 2 of a input byte

OUT0: Output pin. It is a general purpose digital output pin, which is bit 0 (LSB) of a output byte

Maximum source and sink current are 25mA

OUT1: Output pin. It is a general purpose digital output pin, which is bit 1 of a output byte

Maximum source and sink current are 25mA

OUT2: Output pin. It is a general purpose digital output pin, which is bit 2 of a output byte

Maximum source and sink current are 25mA

TX: Output pin. It is an UART 's transmitting pin.

RX: Input pin. It is an UART 's receiving pin. Configuration information and CAN BUS message are passed from the host computer to UART.

Fault\_N: Output pin. It will output low level when CAN BUS has Fault or CAN BUS has transmitting fault for

more than 4-5ms. The pin will return to high level when fault disappears. The maximum time of transmitting fault detection for CAN BUS is 500ms.

OUT3: Output pin. It is a general purpose digital output pin, which is bit 3 of an output byte.

Maximum source and sink current are 25mA

CANTX: Output pin. It is CAN Bus Transmitting pin. User has to connect it to transceiver's transmitting pin such as MCP2551's TXD

CANRX: Input pin. It is CAN Bus receiving pin. User has to connect it to transceiver's receiving pin such as MCP2551's RXD

OUT4: Output pin. It is a general purpose digital output pin, which is bit 4 of an output byte.

Maximum source and sink current are 25mA

OUT5: Output pin. It is a general purpose digital output pin, which is bit 5 of an output byte.

Maximum source and sink current are 25mA

OUT6: Output pin. It is a general purpose digital output pin, which is bit 6 of an output byte.

Maximum source and sink current are 25mA

OUT7: Output pin. It is a general purpose digital output pin, which is bit 7 (MSB) of an output byte.

Maximum source and sink current are 25mA

### 3 IC's configuration

The software for IC configuration is free. The customer can ask the software using email. The running environment is Windows 9.x/windows XP/Windows 2000/Windows Vista. Firstly, user has to install the configuration software on PC. Secondly, CFG\_N is connected to ground and connect PC's RS232 to IC's TXD and RXD (cross link) by max232 chip. Thirdly, Power on the IC and run the configuration software. The following window will be shown up: (These display values are the IC's Default configuration)

**RS232 Config**

RS232 Port: COM1

Baud Rate: 9600

Receiving contains Message ID and DLC

Transmitting contains Message ID and DLC

**Notes:**

Here, the transmitting and receiving mean viewpoint is IC chip not this PC

**CAN Config**

Baud Rate: 250K

**CAN Transmitting**

Frame Type: Standard

Frame Message ID: Hex: 00 00 03 22

Immediately 1 byte Transmit

Single or Multi-bytes Transmit

**CAN Receiving**

Frame Type: Standard

Mask: 00 00 07 ff

Filter1: 00 00 03 33

Filter2: 00 00 03 33

**I/O Config**

Input Frame Type: Standard

Input Frame ID Hex: 00 00 07 22

Output Frame Type: Standard

Output Frame ID Hex: 00 00 05 22

**I/O Feature**

Input Periodic Transmitting

Period ms Count: 300

Input Changes Cause Transmitting

Output Protection during CAN Bus Fault

Output Protection Value (Hex): 00

Default Config Program Config

Step1: Choose the RS232 Port, which connect to IC.

You can choose one from dropdown list. The one you choose must be connected to UART port of IC.

Step2: Configure the Baudrate of IC's UART.

You can choose one from dropdown list, which contains 1200, 2400, 4800, 9600, 19200, 57600, 115200, and 230400. If you choose too low baudrate of UART, compared to baudrate of CAN Bus, You maybe lose some message from CAN Bus occasionally when CAN bus is in the heavy traffic. It is not often happening because our IC contains 64 bytes of ring buffers for UART.

**Note:** When the baudrate of RS232 is 57600 or more, the data you sent from RS232 outside must be separated, the time between bytes must be over 1 millisecond around. If this IC is busy with receiving CAN bus message, you had better keep the data you sent from RS232 outside to separate no matter what baudrate of RS232 you choose. The time between bytes must be over 1 millisecond around.

Step3: Click on "Receiving contains Message ID and DLC" check box to select or deselect whether the contents of IC UART receiver contains CAN BUS's frame message, that is Message ID and DLC. The contents of IC UART receiver will transmit out through CAN bus.

If the user choose "Receiving contains Message ID and DLC", it's user's responsibility to produce the frame of CAN bus. The standard frame's structure of CAN bus is in table 2, and the extended frame's structure of CAN bus is in table 3. The interval of between adjacent bytes in the frame must be less than 50 ms; otherwise IC will take it the beginning byte of new frame. We recommend to keep at least 1ms' interval between adjacent bytes in the frame when heavy CAN BUS traffic. Of cause, when light traffic, you can keep no any interval. We also recommend to keep at least 1 ms' interval between different data when you deselect "Receiving contains Message ID and DLC" and it's heavy CAN BUS traffic. Similarly, you might keep no any interval when light traffic

Table 2 the standard frame's structure of CAN bus

		7	6	5	4	3	2	1	0
Byte 1	Frame SID1	SID.10	SID.9	SID.8	SID.7	SID.6	SID.5	SID.4	SID.3
Byte 2	Frame SID2	SID.2	SID.1	SID.0	X	0	X	X	X
Byte 3	Frame information	X	RR	X	X	DLC.3	DLC.2	DLC.1	DLC.0
Byte 4	Data 1	Data							
Byte 5	Data 2	Data							
Byte 6	Data 3	Data							
Byte 7	Data 4	Data							
Byte 8	Data 5	Data							
Byte 9	Data 6	Data							
Byte 10	Data 7	Data							
Byte 11	Data 8	Data							

**Notes:** 1. SID.10 is MSB of 11bits SID (Standard ID). SID.0 is LSB of 11bits SID (Standard ID)

2. X denotes any value.

3. RR denotes Remote frame. RR=0 denotes data frame, 1 denotes remote frame. The DLC must be 0 when RR=1

4. DLC.3 is MSB of 4 bits data length. DLC.0 is LSB of 4 bits data length. Maximum of DLC is 8

5. How many bytes for standard frame depend on the DLC and RR The range is minimum 3 bytes to maximum 11 bytes

Table 3 the extended frame's structure of CAN bus

		7	6	5	4	3	2	1	0
Byte 1	Frame SID1	SID.10	SID.9	SID.8	SID.7	SID.6	SID.5	SID.4	SID.3
Byte 2	Frame SID2	SID.2	SID.1	SID.0	X	1	X	EID.17	EID.16
Byte 3	Frame EID1	EID.15	EID.14	EID.13	EID.12	EID.11	EID.10	EID.9	EID.8

Byte 4	Frame EID2	EID.7	EID.6	EID.5	EID.4	EID.3	EID.2	EID.1	EID.0
Byte 5	Frame information	X	RR	X	X	DLC.3	DLC.2	DLC.1	DLC.0
Byte 6	Data 1	Data							
Byte 7	Data 2	Data							
Byte 8	Data 3	Data							
Byte 9	Data 4	Data							
Byte 10	Data 5	Data							
Byte 11	Data 6	Data							
Byte 12	Data 7	Data							
Byte 13	Data 8	Data							

**Notes:** 1. SID.10 is MSB of 29 bits message ID. EID.0 is LSB of 29 bits message ID

2. The 29 bits 'message ID from MSB to LSB is SID.10, ... , SID.0, EID.17, EID.16,..., EID.0

3. X denotes any value.

4. RR denotes Remote frame. The DLC must be 0 when RR=1

5. DLC.3 is MSB of 4 bits data length. DLC.0 is LSB of 4 bits data length. Maximum of DLC is 8

6. How many bytes for standard frame depend on the DLC and RR The range is minimum 5 bytes to maximum 13 bytes

Step 4: Click on "Transmitting contains Message ID and DLC" check box to select or deselect whether the contents of IC UART transmitter contains CAN BUS 's frame message, that is Message ID and DLC. The contents of IC UART transmitter buffer will send to Host computer through RS232.

Step 5: Choose the baudrate of CAN bus

You can choose one from dropdown list, which contains 20K, 50K, 100K, 125K, 200K, 250K, 500K, 1000K

Step 6: Choose the type of frame from UART to CAN bus.

If you choose "Receiving contains Message ID and DLC" in step 3, please skip this step.

You can choose "Standard" or "Extended" from dropdown list. "Standard" means 11bit message ID frame.

"Extended" means 29bit message ID frame.

Step 7: Give the message ID of fame from UART to CAN bus

If you choose "Receiving contains Message ID and DLC" in step 3, please skip this step.

The IC will automatically add the message ID to UART 's receiving data. Pay more attention to 29bits of message ID, which is SID.10 –SID.0 EID.17 –EID.0. That is to say, SID.10 is the MSB of 29 bits of message ID, and EID.0 is the LSB of 29 bits of message ID.

Step 8: Choose one from option buttons "Immediately 1 byte transmit" and "Single or multi-bytes transmit"

If you choose "Receiving contains Message ID and DLC" in step 3, please skip this step.

If you choose "Immediately 1 byte transmit", 1 byte will be put into a CAN bus frame. The frame will contain only a data byte, and it will be sent out immediately through CAN bus. In this way, the efficiency of transmitting is not high.

If you choose "Single or multi-bytes transmit", IC will wait a proper time for more bytes from UART. So,

Perhaps, more bytes (maximum is 8) will be put into a CAN bus frame. In this way, the efficiency of transmitting is higher, but there is a bit of delay.

Step 9: Choose the type of frame from CAN Bus to UART

You can choose one from dropdown list, which contains Standard, Extended, ALL.

If you choose the Standard or Extended, the IC will not transmit ( by UART) the same frame as the one receiving by UART even though it matches the filters.

If you choose all, the IC will transmit any frame (By UART) even though one it receives by UART. In this situation, the IC ignores the mask and filters.

**Notes 1:** If you want that IC 's UART can send input frame or output frame, you have to make sure filter1 contains input frame or output frame. You cannot only let filter2 contains input frame or output frame. Otherwise, no guarantee.

**In any situation,** CAN Bus can transmit input frame, and output can be controlled by CAN Bus.

**Notes 2:** If you want to receive remote input frame from CAN Bus (not from IC 's UART), you must make sure filter1 contains the input frame.

**Notes 3:** If remote input frame is from IC 's UART, the CAN BUS will send data input frame firstly and then remote input frame. It 's not like the regular way, first remote frame second data frame. But it does not matter because the purpose is only telling other CAN bus nodes that the input data is from the requisition of remote frame.

**Notes 4:** If you choose the Standard or Extended, and input frame or output frame match the filter1, then IC 's UART can send input frame or output frame even though CAN bus has fault.

Step 10: Give the mask and filter1 and filter2

If you choose 摸?in the step 9, please skip this step. But, we strongly recommend using filter to decrease communication traffic.

The mask and filter1 and filter 2 decide which frame will be received by IC 's CAN bus.

The bit of mask set to 1 means that the correspond bit of message ID must be the same as correspond bit of filter in order to receive the message from CAN bus. Of course, any filter matching is OK no matter it is from filter1 or filter2. See table 4

Table 4 Accept/refuse table

Mask bit n	Filter 1 bit n	Filter 2 bit n	ID bit n	Accept or refuse bit n
------------	----------------	----------------	----------	------------------------



0	X	X	X	accept
1	0	X	0	accept
1	X	0	0	accept
1	0	0	1	Refuse
1	1	X	1	accept
1	X	1	1	accept
1	1	1	0	Refuse

Notes: X means any value

An exception is that output port can be controlled by output frame no matter whether to contain in the filters.

Step 11: Choose the type of transmitting frame for input.

You can choose "Standard" or "Extended" from dropdown list. "Standard" means 11bit message ID frame.

"Extended" means 29bit message ID frame.

Step 12: Give the message ID of input

Step 13: Choose the type of frame for output.

You can choose "Standard" or "Extended" from dropdown list. "Standard" means 11bit message ID frame.

"Extended" means 29bit message ID frame.

Step 14: Give the message ID of output

If the message ID of output is the same as input, the input will be directly passed to output according to choice of step 15. Any nodes in the CAN BUS cannot send this message ID data.

Step 15: Choose one from option buttons "Input periodic transmitting" and "Input changes cause transmitting"

If you choose "Input periodic transmitting", you must input periodic time, it's between the minimum 1 millisecond and maximum 65535 milliseconds.

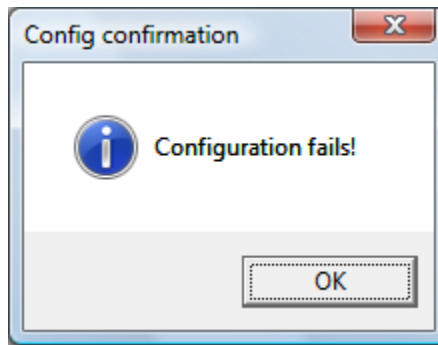
If you choose "Input changes cause transmitting", the IC has the debounce function. You don't need to worry about switch's debounce when you press a button which is connected to discrete input pin. But the input value has about 15ms delay.

**Note:** If you choose "Input changes cause transmitting" and your inputs are connected to logic 1 or 0 permanently by hardware, then the IC will never send the input values to CAN BUS.

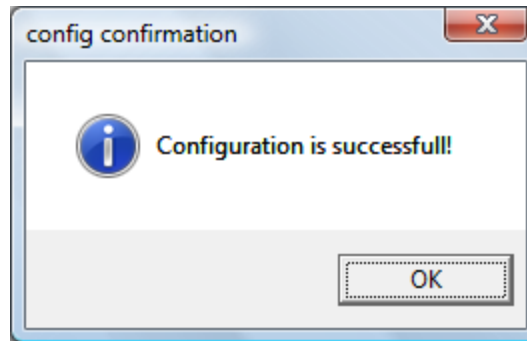
Step 16: Click on "Output protection during CANbus fault" check box to select or deselect whether output port will be special setting value when CAN bus fault occurs.

If you choose "Output protection during CANbus fault" you must input the special setting value.

Step 17: Click on the button "Program Config" wait for a short time, if the following windows occur:



It means some errors happen; you have to try clicking on the button "Program Config" until the following windows occur:



Step 18: Power off the IC, disconnect the UART of IC to RS232 of PC. Let CFG\_N be connected to high level. Now the IC completes the configuration.

We can configure the IC again by CFG\_N changed to low level from high level when the IC is running in the normal mode. This is called dynamic configuration. It is useful for some protocol such as ISO15765, which supports 250k/500k baudrate 11bit/29 bits ID CAN bus. Pay more attention, dynamic configuration information will be lost when power off.

The approach of dynamic configuration will be shown as following:

Firstly, Host controller or PC forces the IC's CFG\_N pin to low level, and hold it.

Secondly, host microcontroller or PC sends 14 bytes of configuration data through UART after IC's CFG\_N has low level for at least 8 milliseconds.

**Note: the interval of between the adjacent bytes among 14 bytes is minimum of 1 ms in order to transmit correctly in heavy traffic.**

Thirdly, if the host microcontroller or PC receives hexadecimal 55, it means the dynamic configuration is successful. If the host microcontroller or PC receives hexadecimal AA or receives nothing for long time, it means the dynamic configuration fails, you have to let host microcontroller or PC sends 14 bytes of configuration data again until success. Host controller or PC can let the IC's CFG\_N pin return to high level after successful dynamic configuration in order to return to the normal mode.

The 14 bytes of configuration data is explained below:

		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	Can Baudrate and Receiving frame type	M1	M0	X	X	X	B2	B1	B0
Byte 2	Mask	Mk.28	Mk.27	Mk.26	Mk.25	Mk.24	Mk.23	Mk.22	Mk.21
Byte 3	Mask	Mk.20	Mk.19	Mk.18	0	1	0	Mk.17	Mk.16
Byte 4	Mask	Mk.15	Mk.14	Mk.13	Mk.12	Mk.11	Mk.10	Mk.9	Mk.8
Byte 5	Mask	Mk.7	Mk.6	Mk.5	Mk.4	Mk.3	Mk.2	Mk.1	Mk.0
Byte 6	Filter1	F1.28	F1.27	F1.26	F1.25	F1.24	F1.23	F1.22	F1.21
Byte 7	Filter1	F1.20	F1.19	F1.18	0	EIDE	0	F1.17	F1.16
Byte 8	Filter1	F1.15	F1.14	F1.13	F1.12	F1.11	F1.10	F1.9	F1.8
Byte 9	Filter1	F1.7	F1.6	F1.5	F1.4	F1.3	F1.2	F1.1	F1.0
Byte 10	Filter2	F2.28	F2.27	F2.26	F2.25	F2.24	F2.23	F2.22	F2.21
Byte 11	Filter2	F2.20	F2.19	F2.18	0	EIDE	0	F2.17	F2.16
Byte 12	Filter2	F2.15	F2.14	F2.13	F2.12	F2.11	F2.10	F2.9	F2.8
Byte 13	Filter2	F2.7	F2.6	F2.5	F2.4	F2.3	F2.2	F2.1	F2.0
Byte 14	Check Sum	SUM(Byte1 +Byte2+.....+Byte12+Byte13)							

X means any value. M1 M0 is the choice of receiving frame of CAN bus.

M1 M0 =01 means only receiving standard frame.

M1 M0 =10 means only receiving extended frame.

M1 M0 =11 means receiving all frame.

Mk.28 is the MSB of mask.

Mk.0 is the LSB of mask. If M1 M0 =01 (only receiving standard frame), you only need to set MK.28 to MK.18, don't care MK.17 to MK.0, and MK.17 to MK.0 can be any value.

F1.28 is the MSB of Filter1.

F1.0 is the LSB of Filter1. If M1 M0 =01 (only receiving standard frame), you only need to set F1.28 to F1.18, don't care F1.17 to F1.0. But you must set EIDE to 0. If M1 M0 =10 (only receiving extended frame), you must set EIDE to 1

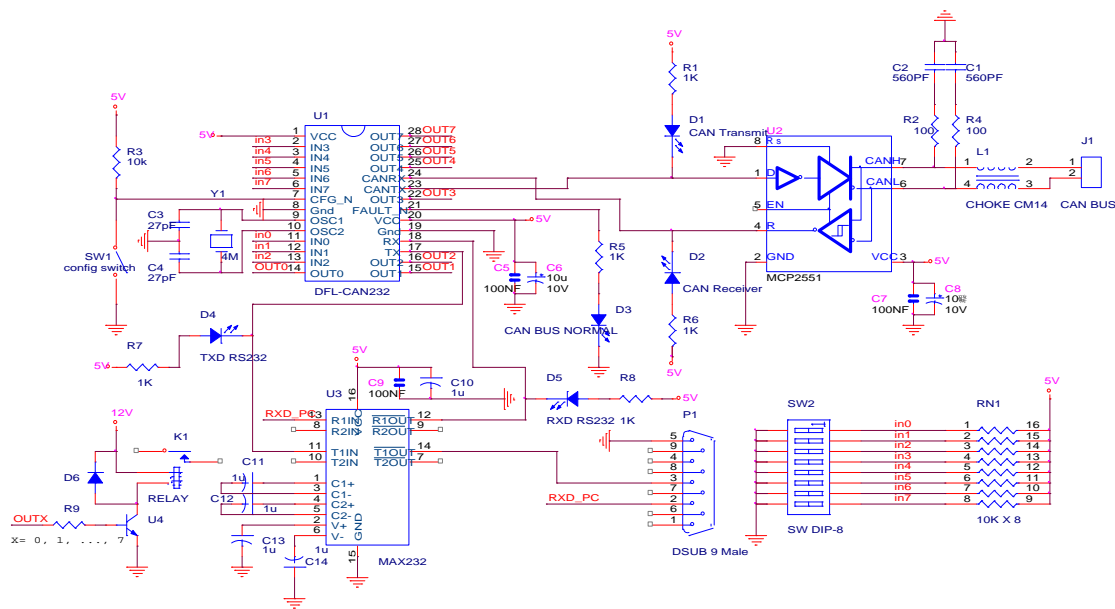
F2.28 is the MSB of Filter2.

F2.0 is the LSB of Filter2. If M1 M0 =01 (only receiving standard frame), you only need to set F2.28 to F2.18, don't care F2.17 to F2.0. But you must set EIDE to 0. If M1 M0 =10 (only receiving extended frame), you must set EIDE to 1

**Note:** When the type of frame from CAN Bus to UART is "All" , you can not use dynamic configuration. I suggest that you can make the mask to be 0 and use "Standard" or "Extended" receiving frame. In this situation, you can use the dynamic configuration.

## 4 Reference circuits

The general circuit is shown below:



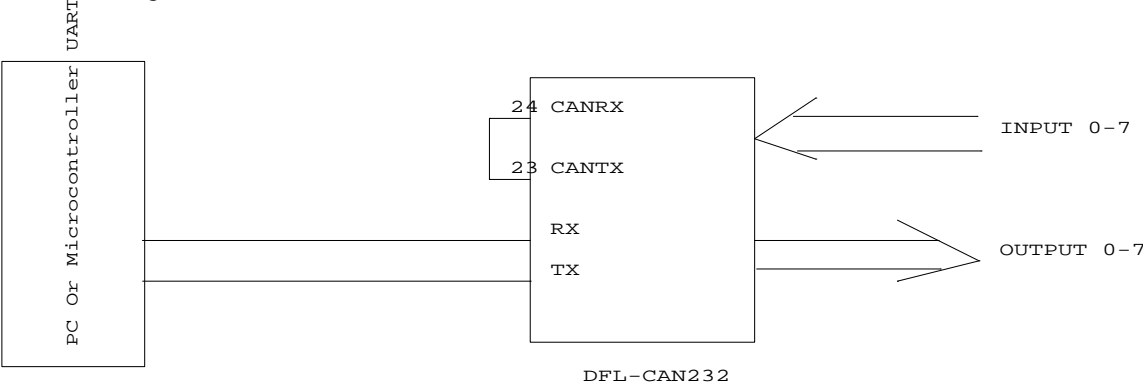
**Notes:** 1 If the CAN bus node is last one; you must connect a 120 ohm 's resistor between CANH and CANL.

2 Under the less interference environment, it is acceptable not to use common mode choke L1

If you connect IC 's UART to microcontroller instead of PC, you don t need MAX232, directly connect to microcontroller 's UART.

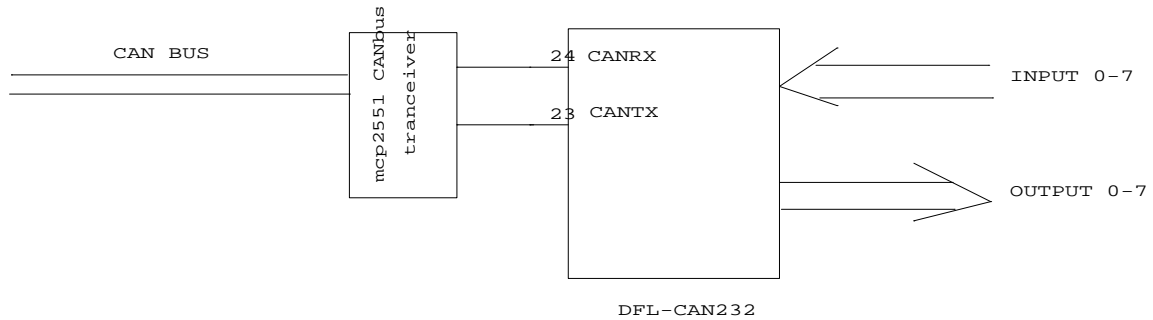
If you don t want the CAN bus function, you only want UART to INPUT/OUTPUT extension.

The block diagram of circuit is shown below:

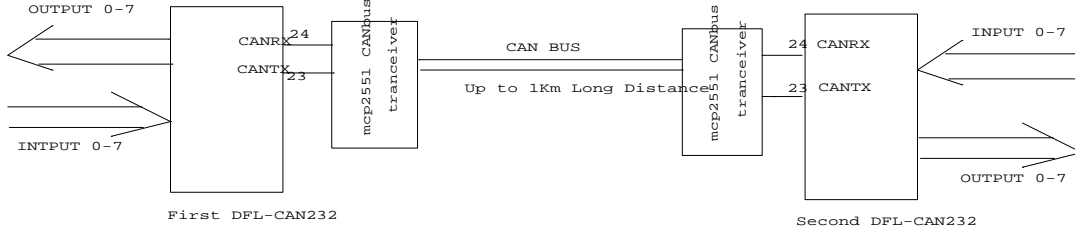


You must configured CAN bus transmitting frame 's message ID is equal to (I/O) output frame 's message ID, and CAN bus receiving frame 's message ID is equal to (I/O) input frame 's message ID. You have to choose the type of frame from CAN Bus to UART as Standard or extended. You cannot choose the type of frame from CAN Bus to UART as ALL. If you have more than 8 inputs or/and outputs, you can use more boards (8 inputs, 8 outputs) to link together by CAN bus. In this situation, you must configure UART receiving contains message ID and DLC in order to distinguish different boards

If you only want CAN Bus as Input/Output extension, the following block diagram will be used.

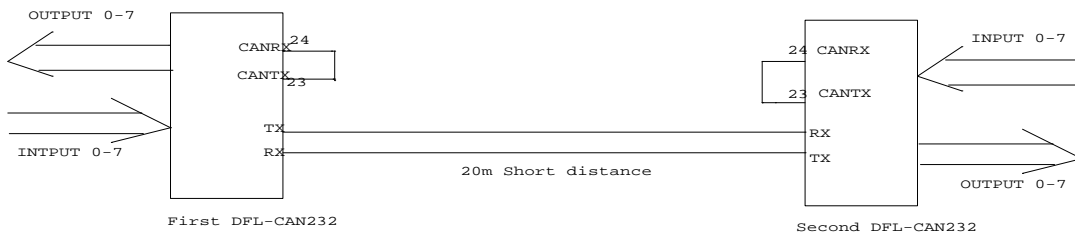


If you want to get input/output from long distance, you can put DFL-RS232 chip in 2 different places and use CAN bus to connect them. The block diagram is shown below.



You must configure the message ID of first DFL-CAN232's (I/O) input frame is equal to the message ID of second DFL-RS232's (I/O) output frame, and the message ID of first DFL-CAN232's (I/O) output frame is equal to the message ID of second DFL-RS232's (I/O) input frame. Mask and Filter setting can be any value because it is only related to RS232-CAN transfer.

If you want to get input/output from not too long distance, another cheap way is UART link, or RS232 link (using max232 to change UART voltage level). The following block diagram will be used.



You must configure:

- 1 the message ID of first DFL-CAN232's (I/O) input frame is equal to the message ID of first DFL-CAN232's receiving frame
- 2 the message ID of second DFL-CAN232's transmitting frame is equal to the message ID of second DFL-CAN232's (I/O) output frame
- 3 the message ID of first DFL-CAN232's (I/O) output frame is equal to the message ID of first DFL-CAN232's

transmitting frame

- 4 the message ID of second DFL-CAN232's (I/O) input frame is equal to the message ID of second DFL-CAN232's receiving frame
- 5 RS232's configuration must be the same baudrate. No need to contain ID.

If you want an adaptor from PNP (or NPN) proximity sensors to NPN (or PNP) proximity sensors, you will set the message ID of DFL-CAN232's (I/O) input frame to be equal to the message ID of DFL-CAN232's (I/O) output frame. In this way, DFL-CAN232's discrete inputs directly pass to their discrete outputs according to period milliseconds count. Please use our UART-CAN board (Parts No: DFL232CANBV1) for the adaptor.

## 5 Electrical Specifications

Absolute Maximum ratings

Oscillator Frequency... .. 4MHz

VCC... .. 7.5V

Ambient Temperature under bias.....-40 to +125

Max output current sunk by any I/O Pin... .. 25mA

Max output current sourced by any I/O Pin... .. 25mA

Max output current sunk by Fault\_N Pin... .. 25mA

Max output current sourced by Config\_N Pin... .. 25mA

Max output current sunk by Config\_N Pin... .. 25mA

Max output current sourced by Fault\_N Pin... .. 25mA

Max output current sunk by all Pins except Vcc and Vss... .. 200mA

Max output current sourced by by all Pins except Vcc and Vss... .. 200mA

DC Characteristics:

Standard operating Temperature: -40 to 85

Supply Current: maximum: 20mA, typical: 16mA

Vcc: 4.2VDC to 5.5VDCI

VIL ----- Input low voltage: max=0.3Vcc

VIH----- Input high voltage: min=0.7Vcc

VOH-----Output high voltage: min=VCC-0.7

VOL-----Output low voltage: max=0.6 when Vcc=4.5V

Packaging Information

DFL-CAN232/P is PDIP (300mils) packaging

DFL-CAN232/S is SIOC (7.5mm) packaging

28-Lead Plastic Dual In-Line (P)-300 mil Body [PDIP]

28-Lead Plastic Small Outline (SO)-Wide, 7.5 mm Body [SOIC]

## **IMPORTANT NOTICE**

The information in this manual is subject to change without notice.

Dafulai's products are not authorized for use as critical components in life support devices or systems. Life support devices or systems are those which are intended to support or sustain life and whose failure to perform can be reasonably expected to result in a significant injury or death to the user. Critical components are those whose failure to perform can be reasonably expected to cause failure of a life support device or system or affect its safety or effectiveness.

## **COPYRIGHT**

The product may not be duplicated without authorization. Dafulai Company holds all copyright. Unauthorized duplication will be subject to penalty.