

CAN Bus logger with SD-card

0. What is it?

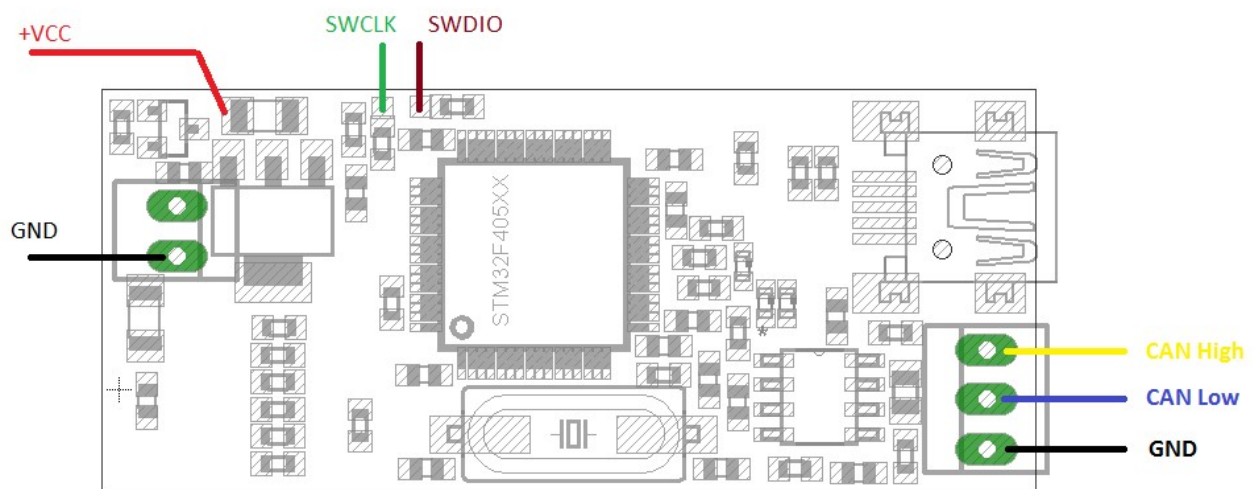
This is just simple logger which writes everything from CAN bus to text file on a micro SD card. It has following features:

- Easy to use: Only one start/stop button and all the settings are stored in configuration text file on the SD card.
- Optional message filtering based on ID mask matching.
- Selectable listen-only mode (without CAN bus acknowledge).
- Three LEDs for indication of logger.

Device specification:

Parameter	Value
Power supply voltage	5V-20V
Current consumption	60mA at 5V input
CAN baud rate	up to 1Mbps (any non-standard baud rate supported)
CAN ID mask filters	1
Microcontroller	STM32F405RGT6
CAN transceiver	SN65HVD232DR
PCB size	48.26 mm x 20.85 mm (1.9 in x 0.82 in)

Board connections:



LEDs:

- **Green:** power on, also toggling every time when the CAN receives and accepts messages.
- **Blue:** blinks each time when block of data has been written to the SD card.
- **Red:** fault indication (see below).

1. How to start

- Connect CAN bus.
- Connect power supply (make sure the correct polarity and voltage range is connected!)
- Place Config.txt file to the root folder of SD card (here is [example](#) of the file).
- Insert SD card.
- Press "START" button to start log
- The blue LED should blink periodically (with speed dependant from writing rate).
- Press "START" button again to stop the log.
- The log file placed to root folder and has name in format: HH-MM-SS.csv where HH-MM-SS is time from power cycling of the device.

2. Configuration file format

All numbers in configuration text should be in decimals (even filter data mask, apologies for the inconvenience).

Parameter	Meaning
baud	CAN bus baud rate in kbps
ack_en	If 1 then CAN logger is responding with ACK slot on reception of a valid CAN frame. Set to 0 for silent (listen-only) mode.
id_filter_mask	Bit mask for ID filtering
id_filter_value	Expected value for ID
log_std	Messages with standard (11 bit) ID are accepted if set to 1
log_ext	Messages with extended (24 bit) ID are accepted if set to 1
timestamp	If 1 then every record in log file has a time stamp (in milliseconds)

The ID filter acceptance criterion is:

Message ID *bitwise and* id_filter_mask = id_filter_value *bitwise and* id_filter_mask

For example if id_filter_mask = 10 = 1010 binary and the id_filter_value = 2 = 0010 binary. It means that the bit #1 and bit #3 of CAN identifier will be checked, and the bit #1 is expected to be 1 and bit #3 to be 0. Thus only identifiers with a binary ending of ...0X1X will be accepted by CAN1, i.e. in hex 0x?2, 0x?3, 0x?6, 0x?7.

Set id_filter_mask = 0 for disabling ID filter.

3. Log file format

Timestamp, ID, Data0, Data1, ...,										
78824,	38,	80,	00,	08,	00,	00,	00,	C7		
78825,	244,	10,	00,	00,	00,	00,	00,	5E		
78826,	3A,	00,	00,	00,	00,	04,	00,	45		
78827,	348,	00,	01,	00,	00,	00,	52			
78828,	3E,	1D,	E6,	44						
78829,	3B,	00,	00,	00,	E3,	23				
78829,	20,	00,	00,	07						
78830,	30,	04,	00,	00,	00,	20,	00,	5C		
78830,	39,	23,	00,	10,	70					
78830,	B1,	00,	00,	00,	00,	11,	C8			
78830,	B4,	00,	00,	00,	00,	00,	00,	BC		
78831,	120,	00,	00,	00,	00,	10,	10,	00,	49	
78833,	38,	80,	00,	08,	00,	00,	00,	C7		
78835,	3A,	00,	00,	00,	00,	04,	00,	45		
...		

Diagram labels:

- Header line (red line pointing to the first row)
- Data (byte#0 - byte#6) (blue line pointing to the data bytes)
- Message ID (green line pointing to the ID field)
- Timestamp (in ms) (black line pointing to the timestamp field)

4. Faults and indication

- If **red** LED and **blue** LED are both on just after pressing "START" button, then this is a configuration text file problem; check configuration file.
- If **red** LED is on during logging, then this is either data buffer overflow or SD card problem. Check if there is enough free space on SD card or SD card has acceptable writing speed rate.

5. CAN bus termination

Please note that 120 ohm termination resistor is permanently connected on board. There is no configurable option for it.

If the bus already has termination at the both ends, then you may consider removing on-board termination resistor. However, for most application it's ok to leave it (for example when connecting to vehicle OBD-II diagnostic connector).