



RKS+CAN ASCII-Interface Description



The RKS+CAN interface allows to be used as a virtual serial port with a simple character (ASCII) based interface. A command ends with [CR], the reply of the interface is ... [CR] (0x0C) on success or [BELL] (0x07) on failure.

Supported commands:

1. Setup CAN bitrate, short form:
 $S_n[CR]$ where $n \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$
 0 = 10kBit/s
 1 = 20kBit/s
 2 = 50kBit/s
 3 = 100kBit/s
 4 = 125kBit/s
 5 = 250kBit/s
 6 = 500kBit/s
 7 = 800kBit/s
 8 = 1000kBit/s

The command can only be sent when the CAN channel is closed.

Example: S6[CR] to setup the CAN for 500kBit/s

2. Setup CAN bitrate, SJA 1000 form:
 $snnmm[CR]$ where $nn = BTR0$ and $mm = BTR1$ hexadecimal values for a 16MHz clock.

The command can only be sent when the CAN channel is closed.

The RKS+CAN hardware does not use a SJA1000 hardware but provides this interface for compatibility purposes. Example: s0714[CR] for 250kBit/s. For details how to calculate the values, see SJA 1000 datasheet or Google/MSN.

3. Setup CAN bitrate, easy form:
 $synnnnn[CR]$ where $nnnnn$ = hexadecimal value of the bitrate.

The command can only be sent when the CAN channel is closed.
Example: sy14585[CR] for 83,333kBit/s.

4. Open CAN channel:
O[CR], if no bitrate has been set before, it defaults to 500kBit/s.
5. Close CAN channel:
C[CR], close previously opened CAN channel.
6. Get interface serial, compatibility form:
N[CR], returns the compatibility form of the interface serial number e.g. N1234[CR].
7. Get interface version numbering, compatibility form:
V[CR], returns the compatibility form of the interface software and hardware versions, e.g. V1002[CR].
8. Get RKS-CAN interface serial:
x[CR], returns full interface serial.
9. Read error flags:
F[CR], returns Fxx where xx is a 8-bit hexadecimal number representing the error flags.

Bit	Represents
1	CAN receive FIFO queue full
2	CAN transmit FIFO queue full
3	Error warning (EI), see SJA 1000 datasheet
4	Data Overrun (DOI), see SJA 1000 datasheet
5	unused
6	Error Passive (EPI), see SJA 1000 datasheet
7	Arbitration Lost (ALI), see SJA 1000 datasheet
8	Bus Error (BEI), see SJA 1000 datasheet

Example: F01[CR] if CAN receive FIFO is full.

10. Send 11 bit CAN frame:
tiiiIdddddddddddddd[CR], sends a CAN frame with hexadecimal ID iii, length I and data bytes dd (according to length). Example:
t280411223344[CR] sends 4 bytes, CAN ID 0x280, bytes 0x11, 0x22, 0x33, 0x44.
11. Send 29 bit CAN frame:
TiiiiiiiIdddddddddddddd[CR], sends a CAN frame with hexadecimal ID iiiiii, length I and data bytes dd (according to length). Example:
T0000E3883223344[CR] sends 3 bytes, CAN ID 0xE388, bytes 0x22, 0x33, 0x44.

12. Send 11 bit CAN RTR frame:
`riiil[CR]`, sends a CAN frame with hexadecimal ID `iii` and length `l` (normally zero). Example: `r2880[CR]` sends RTR (Remote Transmission Request) message for CAN ID `0x288`.
13. Send 29 bit CAN RTR frame:
`Riiiiiiiil[CR]`, sends a CAN frame with hexadecimal ID `iiiiiii`, length `l`.
 Example: `R000E3880` sends RTR message for CAN ID `0xE388`.
14. Receive 11 or 29 bit CAN frames:
 When the CAN channel is open and there is traffic on the bus, the received CAN messages are reported in the same form as they are sent. See commands 10 and 11. If hardware timestamps are activated, they are added on the end of the message.
15. Hardware timestamps command:
`Zn[CR]` where `n = 0` means hardware timestamps are off, `n = 1` means timestamps on, measured in milliseconds going from 0 to 60000 ms. If `n = 2`, hardware timestamps in 10 us units are sent, going from 0 to 600 ms. The timestamps are cat on the end of received messages as two hexadecimal bytes in ASCII form.
16. Set acceptance code register, SJA 1000 compatibility:
`Mxxxxxxxx[CR]` allows to set up CAN message filtering according to SJA 1000 datasheet.
17. Set acceptance mask register, SJA 1000 compatibility:
`mxxxxxxxx[CR]` allows to set up CAN message filtering according to SJA 1000 datasheet.
18. Set listen only mode:
`Lx[CR]` enables or disables listen only mode. In listen only mode, the CAN controller does not acknowledge received messages. Because of this, listen only mode does only make sense if there are at least two CAN nodes connected to a bus.
 Listening to only one CAN controller results in error frames as the messages are never acknowledged!

 The command can only be sent when the CAN channel is closed.
 Example: `L1[CR]` enables listen only mode, `L0[CR]` disables it.
19. Get Ub voltage:
`U[CR]` measures the voltage on the OBD2 port, it returns `Uxxx[CR]` where `xxx` is the hexadecimal value of the voltage in 10mV units. E.g. `U561[CR]` is a measurement of 13770mV.

You find further info for using the system at the following internet addresses:

<https://www.canhack.de>

<https://www.canhack.de/viewtopic.php?t=137>

<https://shop.dieselschrauber.org/can-usb-interface-kit-p-313.php>

Kaufmann Automotive GmbH
Süsswinkelstrasse 9
CH-9453 Eichberg