MIC331X/341X

Multiprotocol OBD to UART Interpreter Datasheet

1 Description

The MIC3X1X is an OBD to UART interpreter IC designed to provide bi-directional half-duplex communication with the vehicle's On-Board Diagnostic system (OBD-II). It supports all legislated OBD-II protocols, as well as two proprietary networks: GM Single Wire CAN (GMLAN), and Ford Medium Speed CAN (MS CAN).

A wealth of information can be obtained by tapping into the OBD bus, including the status of the malfunction indicator light (MIL), diagnostic trouble codes (DTCs), inspection and maintenance (I/M) information, freeze frames, VIN, hundreds of real-time parameters, and more.The MIC3X1X is fully compatible with the de facto industry standard ELM327 command set. Based on a 32-bit processor core, the MIC3X1X offers more features and better performance than any other ELM327 compatible IC.

2 Feature

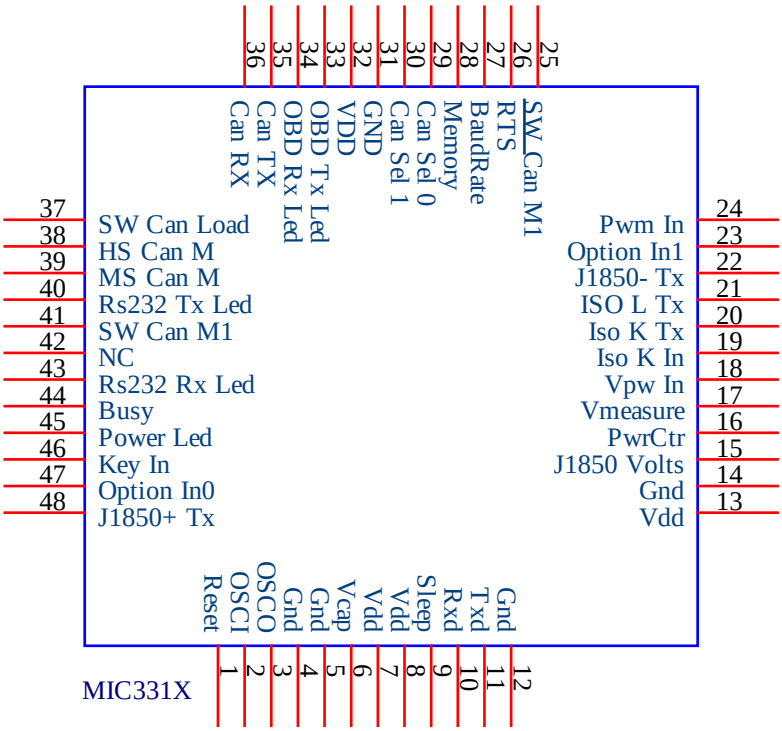
- **Rich instruction set, compatible with more APP software.**
  - -- Fully compatible with ELM's AT command set;
  - -- Fully compatible with STN's ST command set;
  - -- Expanded 6 AT commands for HS-CAN, MS-CAN, SW-CAN;
  - --64 VT macro commands, some instructions are equivalent to the effect of several AT and ST instructions, simplifying the communication steps.
  - -- Independent FAP/FAB filter , CAN protocol 48/3 groups, non-CAN protocol 3/3 groups.
  - -- Independent FAFC/FCID filter, 8/8 groups.
- **Multiple protocols**
  - --15 ELM format protocols;
  - -- 23 STN format protocols;
  - -- Up to 64 user-defined protocols.
- **Multiple fieldbus.**
  - -- SAE J1850 VPW
  - -- SAE J1850 PWM
  - -- ISO 9141-2
  - -- ISO 14230-4
  - -- ISO 15765-4 CAN
  - -- SAE J1939 CAN
  - -- ISO 11898 (raw CAN)
  - -- GMLAN Single Wire CAN (GMW3089)
  - -- Ford Medium Speed CAN (MS CAN)

- **Multiple power management solutions**
  - -- Compatible with ELM and STN power management solutions.
  - -- Support for instructions, UART silence, battery low voltage, OBD bus silence, etc. Up to 5 ways to go to sleep;
  - -- Support UART active, battery voltage drop, OBD bus active, key, etc. Up to 5 ways to wake up from sleep mode.
  - -- Device sleep power consumption is less than 3 mA.
- **OBD request byte up to 1024 bytes, can meet the needs of some special long frame communication.**
- **UART data buffer up to 2K bytes, baudrate increased to 500000bps.**
- **User-operated storage area**
  - -- 256-byte storage unit
  - -- More than 100,000 erasable times
  - -- Read and write operations on any unit in the storage area by using the VT command
- **Multiple WM sequence settings**
  - -- Support AT command to manually set WM sequence
  - -- Support for specifying a protocol or a full protocol-wide temporary WM sequence
  - -- Support for a permanent link between a protocol and a private WM sequence
- **Secure bootloader for easy firmware updates**

3 Typical Applications

- Vehicle telematics
- Fleet management and tracking applications
- Usage-based insurance (UBI)
- OBD data loggers
- Automotive diagnostic scan tools and code readers
- Digital dashboards

4.1 Connection Diagram



4.2 Pin Descriptions

Reset (pin 1)

A momentary (>20µsec) logic low applied to this input will reset the MIC3X1X. If unused, this pin should be connected to a logic high (Vdd) level.

OSCI (pin 2) and OSCO (pin 3)

A 16.000 MHz oscillator crystal is connected between these two pins. Loading capacitors as required by the crystal (typically 20pF each) will also need to be connected between each of these pins and circuit common (Gnd). Note that If an external clock is used, pin is connect to the OSCI pin and e OSCO leftfloating. Also, note that this oscillator is turned off when in the Low Power of operation.

Gnd

Groundreference for logic and I/O pins.

VDD

Positive 3.0 – 5V supply for logic and I/O pins.

Vcap(pin 6)

CPU logic filter capacitor connection. Connect to a low-ESR (< 1 Ω) tantalum or ceramic capacitor. Minimum value is 0.47 µF; typical value is 1 µF.

Pin Descriptions (continued)

Sleep (pin 9)

External sleep control input. When enabled in firmware, puts the device into low-power sleep mode. Polarity of this pin can be configured in firmware; default configuration is active low. Pull up to VDD if unused.

Rxd (pin 10)

This is the RS232 receive data input. The signal level is compatible with most interface ICs (when at idle, the level should be high), but can be used with other interfaces as well, since the input has Schmitt trigger wave shaping.

Txd (pin 11)

This is the RS232 data transmit output. The signal level is compatible with most interface ICs (the output is high when idle), and there is sufficient current drive to allow interfacing using only a PNP transistor, if desired.

J1850 Volts (pin 15)

This output can be used to control a voltage supply for the J1850 Bus+ output. The pin normally outputs a logic high level when a nominal 8V is required (for J1850 VPW), and a low level for 5V (for J1850 PWM), but this can be changed with PP 12. If this switching capability is not required for your application, this output can be left open-circuited.

PwrCtrl (pin 16)

The normal state of the pin will be as set by PP 0E bit 6, and the pin will remain in that state until the V_Link01 switches to the Low Power mode of operation, when the output changes to the opposite level. This output is typically used to control enable inputs, but may also be used for relay circuits, etc. with suitable buffering.

Vmeasure (pin 17)

This analog input is used to measure a 0 to 5V signal that is applied to it. Care must be taken to prevent the voltage from going outside of the supply levels of the MIC3X1X, or damage may occur. If it is not used, this pin should be tied to either VDD or VSS.

VPW In (pin 18)

This is the active high input for the J1850 VPW data signal. When at rest (bus recessive) this pin should be at a low logic level. This input has Schmitt trigger wave shaping, so no special amplification is required.

ISO K In (pin 19)

This is the active low input for the ISO 9141 and ISO 14230 data signal. It is derived from the K Line, and should be at a high logic level when at rest (bus recessive). No special amplification is required, as this input has Schmitt trigger wave shaping.

ISO K Tx (pin 20) and ISO L Tx (pin 21)

These are the active high output signals which are used to drive the ISO 9141 and ISO 14230 buses to an active (dominant) level. Many new vehicles do not require the L Line – if yours does not, you can simply leave pin 22 open-circuited.

J1850 - Tx (pin 22)

This active high output is used to drive the J1850 Bus- Line to an active (dominant) level

Pin Descriptions (continued)

for J1850 PWM applications. If unused, this output can be left open-circuited.

Option In1 (pin 23) Option In0(pin 47)

Select module type：

In1	In0	Module
H	H	BT3.0
H	L	BT4.0
L	H	WIFI
L	L	USB

PWM In (pin 24)

This is the active low input for the J1850 PWM data signal. It should normally be at a high level when at rest (ie. bus recessive). This input has Schmitt trigger wave shaping, so no special amplification is required.

SW CAN M1 (pin 25) SW CAN M0 (pin 41)

Single-wire CAN transceiver operating mode selection outputs, connect to MODE0, MODE1 pins of a single-wire CAN transceiver IC:

M0	M1	Mode
L	L	Sleep Mode
H	L	High-Speed
L	H	High-Voltage Wake-Up
H	H	Normal Mode

$\overline{\text{RTS}}$ / IgnMon (pin 26)

This input pin can serve one of two functions, depending on how the Power Control options (PP 0E) are set.

If both bit 7 and bit 2 of PP 0E are ‘1’s, this pin will act as an Ignition Monitor. This will result in a switch to the Low Power mode of operation, should the IgnMon signal go to a low level, as would happen if the vehicle’s ignition were turned off. An internal ‘debounce’ timer is used to ensure that the MIC3X1X does not shut down for noise at the input.

When the voltage at pin 26 is again restored to a high level, and a time of 1 or 5 seconds (as set by PP 0E bit 1) passes, the MIC3X1X will perform a ‘Warm Start’ and return to normal operation. A low to high transition at pin 26 will in fact restore normal operation, regardless of the setting of PP 0E bit 2, or whether pin 26 was the initial cause for the low power mode. This feature allows a system to control how and when it switches to low power standby operation, but still have automatic wakeup by the ignition voltage, or even by a pushbutton.

If either bit 7 or bit 2 of PP 0E are ‘0’, this pin will function as an active low ‘Request To Send’ input. This can be used to interrupt the OBD processing in order to send a new command, or as previously mentioned, to highlight the fact that the ignition has been turned off. Normally kept at a high level, this input is brought low for attention, and should remain so until the Busy line (pin 44) indicates that the V_ Link01 is no longer busy, or until a prompt character is received (if pin 44 is being used for power control).

This input has Schmitt trigger wave shaping. By default, pin 26 acts as the RTS interrupt input.

PwrCtrl / Busy (pin 44)

This output pin can serve one of two functions, depending on how the Power Control options (PP 0E) are set.If bit 7 of PP 0E is a ‘1’ (the default), this pin willfunction as a Power Control output. The normal state of the pin will be as set by PP 0E bit 6, and the pin will remain in that state until the V_ Link01 switches to the Low Power mode of operation, when the output changes to the opposite level.

Pin Descriptions (continued)

This output is typically used to control enable inputs, but may also be used for relay circuits, etc. with suitable buffering. If bit 7 of PP 0E is a '0', pin 44 will function as a 'Busy' output, showing when the MIC3X1X is actively processing a command (the output will be at a high level), or when it is idle, ready to receive commands (the output will be low). By default, bit 7 of PP 0E is '1', so pin 44 provides the Power Control function.

Baud Rate (pin 27)

This input controls the baud rate of the RS232 interface. If it is at a high level during power-up or reset, the baud rate will be set to 38400 (or the rate that has been set by PP 0C). If at a low level, the baud rate will be initialized to 9600 bps.

Memory (pin 28)

This input controls the default state of the memory option. If this pin is at a high level during power-up or reset, the memory function will be enabled by default. If it is at a low level, then the default will be to have it disabled. Memory can always be enabled or disabled with the AT M1 and AT M0 commands.

CAN SEL1 (pin 30) CAN SEL0 (pin 29)

CAN input channel selection:

SEL1	SEL0	Can channel
L	L	HS CAN
L	H	MS CAN
H	L	SW CAN
H	H	Reserved

CAN Tx (pin 35) and CAN Rx (pin 36)

These are the two CAN interface signals that must be connected to a CAN transceiver IC (see the Example Applications section for more information).If unused, pin 36 must be connected to a logic high (VDD) level.

SW CAN Load (pin 37)

Single-wire CAN high-speed tool load enable output. The pin outputs logic high when high-speed tool load is enabled via the STCSWM command. Leave unconnected if unused.

HS CAN M (pin 38) MS CAN M (pin 39)

High-Speed transceiver operating mode selection outputs, connect to Rs pins of a CAN transceiver IC:

Rs	MODE
L	HIGH-SPEED
H	SLEEP

Power Led (pin 45)

Powe LED output. This pin will output constant high when the device is running and will output low when in sleep mode.. Leave unconnected if unused.

Key In (pin 46)

When the Bluetooth module is not connected, if the falling edge of this pin is detected, enable the Bluetooth module in the discoverable mode 300s.

J1850 + Tx (pin 48)

This active high output is used to drive the J1850 Bus+ Line to an active level. Note that this signal does not have to be used

Pin Descriptions (continued)

for the Bus- Line, since a separate J1850 Bus- drive output is provided on pin 22. output can be left open-circuited.

Rs232 RxLed (pin 43) Rs232 TxLed (pin 40)

OBD RxLed (pin 34) and OBD TxLed (pin 33)

These four output pins are normally high, and are driven to low levels when the MIC3X1X is transmitting or receiving data. These outputs are suitable for directly driving most LEDs through current limiting resistors, or interfacing to other logic circuits. If unused, these pins may be left open-circuited.

Power LED (pin 45)

This pin outputs a low level when the J1962 connector pin 16 has a 12V voltage, and outputs a high level when the J1962 connector pin 16 has no 12V voltage.

5.1 Communicating with the MIC3X1X

The MIC3X1X uses a three-wire UART connection that is CMOS/TTL compatible. The UART settings are:

- 38400 baud (default)
- 8 data bits
- No parity bit
- One stop bit
- No handshaking

The baud rate is software-selectable

5.2 AT Commands Sumary

Fully support to v2.2 version, see Appendix A: **MIC3X1X command list**

5.3 ST Commands Sumary

ST to VT have a one-to-one replacement instruction. see Appendix A: **MIC3X1X command list**

5.4 VT Command Sumary

The VT instruction can realize all the functions of the AT and ST instructions, with powerful functions and more convenient use.

1	D
2	I
3	P1hh
4	P2 hh
5	PROI
6	PRON
7	PROT
8	VERS
9	PC
10	PO
11	PBR baud
12	PBRD
13	CFG_CAN <PROTOCOL>,<OPTION>,<BAUDRATE>,<TYPE>,[TM]
14	SET_CAN <PROTOCOL>,<OPTION>,<BAUDRATE>,<TYPE>,[TM]
15	CFG_ISO <PROTOCOL>,<OPTION>,<BAUDRATE>,[IIA]
16	CAN_WM <No>,<PROTOCOL>,<HEADER>,<DATA1~8>,<PERIOD>,<MODE>
17	ISO_WM <No>,<PROTOCOL>,<HEADER>,<DATA1~5>,<PERIOD>,<CTRL>
18	DEL_CAN_WM <No>
19	DEL_ISO_WM <No>
20	DISP_CAN_WM <No>
21	DISP_ISO_WM <No>

VT Command Sumary (continued)

22	SET_FM <FILTER>,<MASK>
23	SET_CAN_FC<DATA1~5>,<MODE>,[<HEADER>,<FILTER>,<MASK>]
24	SET_HD <HEADR>,[RECIVIER],[TIMEOUT]
25	FCST hh
26	SDST hh
27	ISOFI tl, th, data
28	FBA [pattern], [mask]
29	FBA
30	FPA [pattern], [mask]
31	FPA
32	FCPA [pattern], [mask]
33	FCPA
34	FCTRA tx, rx
35	FCTRA
36	M <FILTER>,<MASK>
37	MFCA
38	SHOW_BUS [CAN_TYPE]
39	AT_PROTOCOL_ALL
40	ST_PROTOCOL_ALL
41	VT_PROTOCOL_ALL
42	UCS_ATI <ascii>
43	UCS_ATDI <ascii>
44	UCS_STDI <ascii>
45	UCS_STI <ascii>
46	UCS_MFR <ascii>
47	UCS_STSN <ascii>
48	SET_UDS <ascii>
49	SET_MAC <hh…hhh>
50	RD_MAC
51	RD_UDS
52	WT_EE <POS>,<DATA[1~8]>
53	RD_EE <POS>,<LEN>
54	POWERMANAGE
55	PDVS <VOLTS>, <TIMER>
56	VDWK <VOLTS>, <TIMER>
57	VLCW [+/-]<VOLTS>, <TIMER>
58	VLRD
59	SWGP <switch 1>, … <switch n>
60	TOST <TimoutType 1:ms>, … <TimoutType n:ms>
61	TP_RTS TotalLen, BlockSize[,EnChk]

VT Command Sumary (continued)

62	TP_DT ii hh ... hh [chksum]
63	TP_END
64	TP_ABORT

5.5 VT Command Description

1) D

Overview:

Output the name of the product design company.

Example:

>VTD
JINXUSOLU

2) I

Overview:

Output the adapter device name defined by the [VT] command.

Example:

>VTI
xxxxxxxx v2.2.84

3) P1hh

Overview:

Switch the protocol defined by [VT] Instruction.

Parameter:

1hh: [VT] Instruction defined protocol number. The protocol number defined by the [VT] instruction starts with the number 1.

Example:

>VTP 10C
OK

>ATDP
10C: ISO 15765 (500K/11B), DLC:F, HS_CAN, WM_No:2 [VT]

4) P2hh

Overview:

Switch the protocol defined by [ST] Instruction.

Parameter:

hh: [ST] Instruction defined protocol number.

Example:

>VTP2 33
OK

>ATDP
HS CAN (ISO 15765, 500K/11B) [ST]

5) PROI

Overview:

Display [VT] defines the name of the manufacturer.

Example:

>VTPROI
xxxxx

6) PRON

Overview:

Display current protocol number, similar to ATDPN command.

7) PROT

Overview:

Display a description of the current protocol, similar to the ATDP command.

Examples:

```
>ATSP6
OK
>ATDP
ISO 15765-4 (CAN 11/500)
>ATDPN
6
>ATDP
ISO 15765-4 (CAN 11/500)
>STPR
6 [AT]
>STPRS
ISO 15765-4 (CAN 11/500) [AT]
>VTPRON
6 [AT]
>VTPROT
ISO 15765-4 (CAN 11/500) [AT]
```

```
>VTP101
OK
>ATDPN
101 [VT]
>ATDP
ISO 15765 (33.3K/11B), DLC:F, SWCAN, MODE3 [VT]
>STPR
101 [VT]
>STPRS
ISO 15765 (33.3K/11B), DLC:F,SWCAN, MODE3 [VT]
>VTPRON
101
>VTPROT
ISO 15765 (33.3K/11B), DLC:F, SWCAN, MODE3
>
```

```
>STP11
OK
>ATDP
SAE J1850 PWM [ST]
>ATDPN
11 [ST]
>STPR
11
>STPRS
SAE J1850 PWM
>VTPRON
11 [ST]
>VTPROT
SAE J1850 PWM [ST]
```

```
>VTP11A
OK
>ATDPN
11A [VT]
>ATDP
11A: ISO 15765 (500K/11B), DLC:V, HS_CAN, WM_NO:1 [VT]
>STPR
11A [VT]
>STPRS
11A: ISO 15765 (500K/11B), DLC:V, HS_CAN, WM_NO:1 [VT]
>VTPRON
11A
>VTPROT
11A: ISO 15765 (500K/11B), DLC:V, HS_CAN, WM_NO:1
```

8) VERS

Overview:

Multi-protocol conversion firmware version.

Example:

>VTVERS
MIC3x1X V2.0.84 (x = 3 or 4) (X = 1 or 2 or 3)

9) PC

Overview:

Close the current OBD protocol. It is equivalent to the "STPC" command.

Example:

>VTPC
OK

10) PO

Overview:

Open the current OBD protocol.

Example:

>VTPO
OK

11) PBR baud

Overview:

Set the baud rate of the current OBD protocol.

Parameter:

baud: baud rate, effective value (1,500000)

Example:

>VTPBR 500000
OK

12) PBRD

Overview:

Display the baud rate of the current OBD protocol.

Example:

>VTPBRD
500000

13) CFG_CAN <PROTOCOL>, <OPTION>, <BAUDRATE>, <TYPE>, [TM]

Overview:

Configure the physical layer to the protocol defined by the CAN standard for a certain [VT] and save it. Deleting this custom protocol when only the first parameter is included.

Due to the variety of CAN buses available on different vehicles, this requires us to do a lot of configuration work before accessing the CAN bus of different brands of vehicles.

In order to simplify the configuration process, we set the protocol area defined by [VT], numbered from 101~140.

The user can configure different data link layer protocols and different physical interfaces in advance, and define these configurations as a protocol number (we call such a protocol number called [VT] protocol number), so as to achieve the handover protocol. Simplify the purpose of the configuration process.

When a WM sequence has been connected to a certain number of protocols, the linked WM sequence will be deleted automatically when the protocol configuration is changed.

Parameters:

PROTOCOL: The protocol number ranges from 101 to 140

OPTION: The data link layer of CAN is defined. The definition of each bit is the same as the definition of PP2C.

BAUDRATE: The baud rate of the CAN bus is defined in the same way as PP2D.

TYPE: Refers to one of HS_CAN, MS_CAN, SW_CAN

TM(Optional): Indicates the mode when SWCAN is working, ranging from 0 to 7.

Examples:

example 1:

>VTCFG_CAN 101, 81, 0F, SW_CAN, 3

OK

It indicates that the data link with the baud rate of 33.3KHZ conforming to the ISO15765 standard is defined to the protocol number 101, the physical link to the SWCAN port, and the SWCAN working mode is 3.

example 2:

>VTCFG_CAN 102, 81, 01, HS_CAN

OK

Indicates that the data link with the baud rate of 500KHZ conforming to the ISO15765 standard is defined to protocol number 102, physical link to HSCAN port.

example 3:

>VTCFG_CAN 102

OK

Indicates that the custom protocol 102 is deleted in the storage area.

14) SET_CAN <PROTOCOL>, <OPTION>, <BAUDRATE>, <TYPE>, [TM]

Overview:

The data link layer of the protocol with the specified physical layer is CAN bus is configured. After the modification, the definition related to the protocol will no longer be the default configuration, and

the configuration will be invalid after reset.

When there is only the first parameter, it means to delete the current configuration and restore the default configuration.

For users accustomed to using the AT command, this command contains most of the configuration of the CAN bus physical layer and data link layer.

Use the SET_CAN command to reconfigure a protocol defined by [AT], which simplifies the configuration process while preserving most of the original process of calling AT commands.

Note: If the current protocol is the protocol that SET_CAN will set, while the bus data is active, it will result in incorrect results.

We recommend using this command before switching protocols; or after switching protocols and when connection data has not yet been initialized.

Parameters:

PROTOCOL: [AT] defines one of protocol B to protocol F

OPTION: The data link layer of CAN is defined. The definition of each bit is the same as the definition of PP2C. XX means ignore this parameter setting.

BAUDRATE: The baud rate of the CAN bus is defined in the same way as PP2D. XX means ignore this parameter setting.

TYPE: Refers to one of HS_CAN, MS_CAN, SW_CAN. XX means ignore this parameter setting.

TM(Optional): Indicates the mode when SWCAN is working, ranging from 0 to 7. X means ignore this parameter setting.

Examples:

example 1:

>VTSET_CAN F, 91, 06, SW_CAN, 3

OK

It indicates that the data link with the baud rate of 33.3KHZ conforming to the ISO15765 standard is defined to the protocol F, the physical link to the SWCAN port, and the SWCAN working mode is 3.

>ATSPF

OK

>ATDP

ISO 15765-4 (CAN 11/95)

example 2:

>VTSET_CAN B, 81, 0F, SW_CAN, 3

OK

Indicates that the OPTION word of protocol B defined by [AT] is set to 0x81, the baud rate control word is 0x0F, and is connected to the SW_CAN interface. The working mode control word of SWCAN is 0x03.

Equivalent to:

>ATPB 81 0F

OK

>ATLNKSW PB

OK

>ATTM 3

OK

example 3:

>VTSET_CAN F
OK
>ATSPF
OK

>ATDP
ISO 15765-4 (CAN 11/33)

Indicates that the [AT] definition protocol F deletes the current configuration and restores the default configuration after power-on.

15) CFG_ISO <PROTOCOL>, <OPTION>, <BAUDRATE>, [IIA]

Overview:

Configure the physical layer to be a protocol defined by the ISO standard for a certain [VT] and save it. Deleting this custom protocol when only the first parameter is included.

Due to the various specifications of the ISO bus on different vehicles, this requires us to do a lot of configuration work before accessing the ISO/KWP bus of different brands of vehicles.

In order to simplify the configuration process, we set the protocol area defined by [VT], numbered from 101~140.

The user can configure protocols of different data link layers in advance, and define these configurations as a protocol number, thereby achieving the purpose of simplifying the configuration process when switching protocols.

When a WM sequence has been connected to a certain number of protocols, the linked WM sequence will be deleted automatically when the protocol configuration is changed.

Parameters:

PROTOCOL: The protocol number ranges from 101 to 140.

OPTION: Define the data link layer of ISO

b7: IsoHeader	0: No	1: Yes	b6: IsoSi	0: No	1: Yes
b5: IsoFi	0: No	1: Yes	b4: IsoInit	0: No	1: Yes
b3: Keep-alives	0: No	1: Yes	b2: Data Format	0: ISO9141	1: ISO14230
b1: ---			b0: ---		

BAUDRATE: ISO bus baud rate.

IIA(Optional): ISO initialization address, if not entered, the default value is 0x33.

Examples:

example 1:

>VTCFG_ISO 103, B8, 10
OK

Indicates that the data link with the baud rate of 10.4KHZ conforming to the ISO14230 standard is defined to the protocol 103, and the fast start allows the wake/hold sequence.

example 2:

>VTCFG_ISO 103
OK

Indicates that the custom protocol 103 is deleted in the storage area.

16) CAN_WM <No>, <PROTOCOL>, <HEADER>, <DATA1~8>, <PERIOD>, <MODE>

Overview:

The physical layer defined for a [VT] sets the wake/hold sequence for the CAN standard protocol.

The purpose of this instruction is to simplify the setup process of the periodic wake/hold sequence on the CAN bus with one instruction.

If there is a storage requirement, the memory area allows up to 8 sets of wake/hold sequences to be connected to a [VT] defined protocol.

If there is no storage requirement, <No> to write a 0 indicates a temporary wake/hold sequence.

If the current protocol meets the WM condition and the protocol is active, the WM sequence takes effect immediately.

Parameters:

No: The serial number (0~8) is provided for use when saving the sequence, 0 means temporary use and not stored.

PROTOCOL: a [VT] definition protocol, The protocol number ranges from 101 to 140. XX Indicates that the agreement is not distinguished (only valid when the number is 0).

HEADER: Wake/keep sequence CAN ID.

format: XX means to use the protocol default HEADER.
hhh (11bit CAN ID)
hhhhhh (not CAN ID)
hhhhhhhh (29bit CAN ID)

DATA: 1~8 byte buffer. format: hh ~ hh hh hh hh hh hh hh hh

PERIOD: Wake-up/hold sequence transmission period, unit 20ms. format: hh. XX means to use the default value (PP22).

MODE: 0,1,2,9,A. 0,1,2 Equivalent to Definition in WM0,1,2. When bit3 is valid, it indicates that the ID of the WM message can be automatically changed according to the ATSH instruction, and at the same time, it can add the extension address (ATCEAhh) by itself.

FORMAT: h

input error:

1: Number exists;
2: The protocol type is incorrect.;
3: Header format error;
4: Wrong number of data;
5: Cycle setting error;
6: Mode setting error;
7: Missing comma;
8: Number input error;
9: Protocol input error;
10:Agreement does not exist;

Examples:

example 1:

>VTCAN_WM 1, 101, 7DF, 01 3E 00 00 00 00 00 00, 64, 2
OK

Set a header to 7DF, the data content is 01, 3E, 00, 00, 00, 00, 00, 00, and the interval is 2000ms.

The wake-up/hold sequence of mode 2 is connected to protocol 101, and the link information is stored.

example 2:

>VTCAN_WM 0, XX, 7DF, 01 3E 00 00 00 00 00 00, 64, 1

OK

equivalent to:

>AT WH 7DF
OK
>AT WM 01 3E 00 00 00 00 00 00
OK
>AT WT 64
OK
>AT WM 1
OK

Set a wakeup/hold sequence that is currently used by the 7DF, not linked to the specified protocol or save the configuration. The configuration is automatically cleared after reset.

The CAN WM sequence works with any 11-bit CAN protocol, and the 29-bit CAN still uses the default header.

example 3:

>VTCAN_WM 0, 103, XX, 01 3E 00 00 00 00 00 00, 64, 2
OK

equivalent to:

>AT WM 01 3E 00 00 00 00 00 00
OK
>AT WT 64
OK
>AT WM 2
OK

Set a header to the currently used wake/hold sequence for sending the request frame header. Do not link to the specified protocol or save the configuration. The configuration is automatically cleared after reset.

This CAN WM sequence works only under the [VT]103 protocol.

17) ISO_WM <No>, <PROTOCOL>, <HEADER>, <DATA1~5>, <PERIOD>, <CTRL>

Overview:

The physical layer defined for a [VT] sets the wake/hold sequence for the ISO standard protocol.

The purpose of this instruction is to simplify the setup process of the periodic wake/hold sequence on the ISO bus with one instruction.

If there is a storage requirement, the memory area allows up to 8 sets of wake/hold sequences to be connected to a [VT] defined protocol.

If there is no storage requirement, <PROTOCOL> is written to XX to indicate that the wake-up/hold sequence is used only if the CAN protocol specified by VTP1hh is used.

The WM sequence takes effect immediately if the current protocol is WM compliant and the protocol is active.

Parameters:

No: The serial number (0~8) is provided for use when saving the sequence, 0 means temporary use and not stored.

PROTOCOL: a [VT] definition protocol, The protocol number ranges from 101 to 140. XX means no distinction between agreements.

HEADER: Device header address for wake/hold sequence. XX is the wrong usage in this command. The header of the WM sequence must be specified in ISO/KWP.

DATA: 1~5 byte buffer

PERIOD: Wake-up/hold sequence transmission period, unit 20ms. XX means to use the

default value (PP17).

CTRL: 0/1:Disable/enable ISO wake/hold sequence

Examples:

example 1:

>VTISO_WM 1, 103, C133F1, 3E 00, 64, 1

OK

Set a wakeup/hold sequence with C1FF1, data content 01, 3E, and interval 2000ms, connect to protocol 103, enable the sequence, and store the link information.

example 2:

>VTISO_WM 0, XX, C133F1, 3E 00, 64, 1

OK

Represents the definition of a temporary protocol with the [VT] defined 103 protocol link WM sequence (not stored), equivalent to:

>AT WM C1 33 F1 3E 00 >AT SW 64

OK OK

This WM works only when the protocol is [VT]103

18) DEL_CAN_WM <No>

Overview:

In addition to the ability to specify the sequence number to set the wake/hold sequence; delete its contents; remove its link to a [VT] protocol.If the current protocol has a WM sequence and meets the set WM condition, the WM sequence is stopped.

Parameter:

No: The sequence number is provided for use when saving the sequence, 0 or no input. This parameter indicates that the current sequence is deleted.

Examples:

example 1:

>VT DEL_CAN_WM 1

OK

Delete the WM sequence configuration with sequence number 1 in the CAN WM sequence storage area.

example 2: **equivalent to:**

>VT DEL_CAN_WM >VT DEL_CAN_WM 0

OK OK

Delete the current (unstored) settings for the wake/hold sequence configuration.

19) DEL_ISO_WM <No>

Overview:

In addition to the ability to specify the sequence number to set the wake/hold sequence; delete its contents; remove its link to a [VT] protocol.

If the current protocol has a WM sequence and meets the set WM condition, it switches to the system default WM sequence.

Parameter:

No: The sequence number is provided for use when saving the sequence, 0 or no input. This parameter indicates that the current sequence is deleted.

Examples:

example 1:

>VT DEL_ISO_WM 1

OK

Delete the WM sequence configuration with sequence number 1 in the ISO WM sequence storage area.

example 2:

>VT DEL_ISO_WM

OK

equivalent to:

>VT DEL_ISO_WM 0

OK

Delete the current (unstored) settings for the wake/hold sequence configuration.

20) DISP_CAN_WM <No>

Overview:

Display the wake-up/hold sequence associated with the CAN bus with the specified sequence number.

Parameter:

No: Sequence number, 1~8 means the saved sequence, 0 means the current sequence. F means all numbers. format: h

Examples:

example 1:

>VT DISP_CAN_WM 1

No:1; P:101; H:7DF; D:01 3E 00 00 00 00 00 00; T:64, M:2

Get the configuration details of the WM sequence with CAN WM sequence storage area number 1.

example 2:

>VT DISP_CAN_WM

H:7DF; D:01 3E 00 00 00 00 00 00; T:64, M:2

Get the configuration details of the current CAN WM sequence.

example 3:

>VT DISP_CAN_WM F

No:1; P:101; H:7DF; D:01 3E 00 00 00 00 00 00; T:64, M:2

No:2; P:105; H:7DF; D:02 3E 80; T:92, M:A

ND

ND

ND

ND

ND

ND

Get the configuration details of all WM sequences in the CAN WM sequence storage area.

21) DISP_ISO_WM <No>

Overview:

Displays the wakeup/hold sequence associated with the ISO/KWP bus with the specified sequence number.

Parameter:

No: Sequence number, 1~8 means the saved sequence, 0 means the current sequence. F means all numbers. format: h

Examples:

example 1:

>VT DISP_ISO_WM 1

P:103; H:C133F1; D:[3E 00]; T:64; C:1

Get the configuration details of the WM sequence with ISO WM sequence storage area number 1.

example 2

>VT DISP_ISO_WM F

P:103; H:C133F1; D:[3E 00]; T:64; C:1

ND

ND

ND

ND

ND

ND

ND

Get the configuration details of all numbered WM sequences in the ISO WM sequence storage area.

22) SET_FM <FILTER>, <MASK>

Overview:

Set FILTER and MASK.

The purpose of this instruction is to set any format FILTER and MASK with one instruction.

The address format of FILTER and MASK must be the same. If the format of FILTER is hhh-hh, the format of MASK must also be hhh-hh.

Parameters:

FILTER:

hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address); hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address). hhhhhh(Non-CAN format)

Note: You cannot use hhhhhh to define FILTER under CAN protocol, which will be regarded as the device header of non-CAN protocol.

MASK:

hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address); hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address). hhhhhh(Non-CAN format)

Note: You cannot use hhhhhh to define MASK under CAN protocol, which will be regarded as the device header of non-CAN protocol.

Examples:

example 1:

>VTSET_FM 7DF, 0FF
OK

equivalent to:

>ATCF 7DF
OK
>ATCM 0FF
OK

example 2:

>VTSET_FM 7DF-F1, 0FF-FF
OK

Equivalent to:

>ATCF 7DF
OK
>ATCM 0FF
OK
>ATCER F1
OK

Note: Due to the function of CAN Mix ID transmission, two independent filtering mechanisms are adopted for the 11bit CAN ID and 29bit CAN ID inside the MIC3X1X.
That is to say, setting 11-bit FILTER and MASK only works for 11-bit CAN ID request. Similarly, 29-bit CAN ID is also the working mode.
The following VTSET_CAN_FC, VTSET_HD commands are the same processing.

Examples:

>0100
18 DA F1 11 06 41 00 BE 3F A8 13
18 DA F1 1D 06 41 00 BF BE F9 80
18 DA F1 19 06 41 00 00 00 00 00
> VTSET_FM 7DF, 700
OK
>0100
18 DA F1 11 06 41 00 BE 3F A8 13
18 DA F1 1D 06 41 00 BF BE F9 80
18 DA F1 19 06 41 00 00 00 00 00
> VTSET_FM 18daf11D, 1fffffff
OK
>0100
18 DA F1 1D 06 41 00 BF BE F9 80

23) SET_CAN_FC <DATA1~5>, <MODE>, [<HEADER>]

Overview:

Set the FC frame content of the ISO15765-4 protocol.
The purpose of this instruction is to set the HEADER, data content and mode of the FC frame of ISO 15765 with one instruction.

Parameters:

DATA: FC frame data content(1~5bytes). format: hh ~ hh hh hh hh hh
MODE: Similar to the definition of AT FC SM h. format: h (0~3). When MODE is 3, the CAN-ID of the FC frame is the same as that set by VTSET_HD.
HEADER(Optional): Sending device header address. format:
hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address);
hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address).
XX(ignore).

Examples:

example 1:

>VTSET_CAN_FC F1 30 FF 08 00, 2, 7DF
OK

equivalent to:

>AT FC SH 7DF
OK
>AT FC SD F1 30 FF 08 00
OK
>AT FC SM2
OK

example 2:

>VTSET_CAN_FC F1 30 FF 08 00, 2
OK

equivalent to:

>AT FC SD F1 30 FF 08 00
OK
>AT FC SM2
OK

example 3:

>VTSET_HD 7DF-40, 7E8-20, FF
OK
>VTSET_CAN_FC 40 30 FF 08, 1, 7DF
OK
>22F101
7E8 20 10 AC 62 F1 01 01 01
7E8 20 21 00 13 09 11 18 8B
7E8 20 22 00 00 00 00 00 00
7E8 20 23 00 00 00 06 00 00
7E8 20 24 1F FD 06 01 2C 08
7E8 20 25 00 00 27 18 06 01
7E8 20 26 2C 08 00 00 27 19
7E8 20 27 06 01 2C 08 00 00
7E8 20 28 27 20 06 01 2C 08
7E8 20 29 00 00 27 1C 06 01
7E8 20 2A 2C 08 00 00 27 27
7E8 20 2B 06 01 2C 08 00 00
7E8 20 2C 27 21 06 01 2C 08
7E8 20 2D 00 00 27 22 06 01
7E8 20 2E 2C 08 00 00 27 1F
7E8 20 2F 06 01 2C 08 00 00
7E8 20 20 27 1A 06 01 2C 08
7E8 20 21 00 00 27 1E 06 01
7E8 20 22 2C 0C 00 00 27 0C
7E8 20 23 06 01 19 0C 00 00
7E8 20 24 27 14 06 01 19 0C
7E8 20 25 00 00 27 12 06 01
7E8 20 26 19 0C 00 00 27 0B
7E8 20 27 06 01 19 A0 00 00
7E8 20 28 25 DB 04 10 02 A1
7E8 20 29 00 00 24 A7 FF 03
7E8 20 2A E0 01 00 00 24 7A
7E8 20 2B 01 03 02 05 00 00
7E8 20 2C 1E F6 06 1C 0F FF

equivalent to:

>AT ST FF
OK
>AT FC SH 7DF
OK
>AT FC SD 40 30 FF 08
OK
>AT FC SM 1
OK
>ATCRA 7E8
OK
>ATCEA 40
OK
>ATCER 20
OK
>22F101
7E8 20 10 AC 62 F1 01 01 01
7E8 20 21 00 13 09 11 18 8B
7E8 20 22 00 00 00 00 00 00
7E8 20 23 00 00 00 06 00 00
7E8 20 24 1F FD 06 01 2C 08
7E8 20 25 00 00 27 18 06 01
7E8 20 26 2C 08 00 00 27 19

7E8 20 27 06 01 2C 08 00 00
7E8 20 28 27 20 06 01 2C 08
7E8 20 29 00 00 27 1C 06 01
7E8 20 2A 2C 08 00 00 27 27
7E8 20 2B 06 01 2C 08 00 00
7E8 20 2C 27 21 06 01 2C 08
7E8 20 2D 00 00 27 22 06 01
7E8 20 2E 2C 08 00 00 27 1F
7E8 20 2F 06 01 2C 08 00 00
7E8 20 20 27 1A 06 01 2C 08
7E8 20 21 00 00 27 1E 06 01
7E8 20 22 2C 0C 00 00 27 0C
7E8 20 23 06 01 19 0C 00 00
7E8 20 24 27 14 06 01 19 0C
7E8 20 25 00 00 27 12 06 01
7E8 20 26 19 0C 00 00 27 0B
7E8 20 27 06 01 19 A0 00 00
7E8 20 28 25 DB 04 10 02 A1
7E8 20 29 00 00 24 A7 FF 03
7E8 20 2A E0 01 00 00 24 7A
7E8 20 2B 01 03 02 05 00 00
7E8 20 2C 1E F6 06 1C 0F FF

example 4:

>VT SET_HD 7E0-10, 7E8-20, FF
OK
>VT SET_CAN_FC 30 FF 50, 3
OK
>22F190
014
0: 62 F1 90 57 42
1: 41 4A 42 39 43 35
2: 36 4A 42 30 33 35
3: 36 35 35 FF FF FF

Indicates that the setting request device address is 7E0-10, the receiving device address is 7E8-20, the receiving timeout is 0xFF*4.096ms (in the ATAT0 state), the FC frame sending ID is 7E0-10, and the data content is 30 FF 32.
In order to be compatible with the EL command of the ELM327, the AT FC SM 3 command is invalid. It must be implemented by the VT SET_CAN_FC command.

example 5:

>VT SET_HD 7E0-10, 7E8-20, FF
OK
>VT SET_CAN_FC XX, 3
OK
>22F190
014
0: 62 F1 90 57 42
1: 41 4A 42 39 43 35
2: 36 4A 42 30 33 35
3: 36 35 35 FF FF FF

Same function as in Example 6, the difference between this instruction and Example 6 is that the contents of the previously set FC SD [1~5] data buffer are not modified.

example 6:

>VT SET_HD 7E0-10, 7E8-F1, FF
OK
>VT SET_CAN_FC 30 FF 08, 1, 7E0-10
OK
>22F190
014
0: 62 F1 90 57 42
1: 41 4A 42 39 43 35
2: 36 4A 42 30 33 35
3: 36 35 35 FF FF FF

equivalent to:

>AT FC SH 7E0	>22F190
OK	014
>AT FC SD 10 30 FF 08	0: 62 F1 90 57 42
OK	1: 41 4A 42 39 43 35
>AT FC SM 1	2: 36 4A 42 30 33 35
OK	3: 36 35 35 FF FF FF

...

Indicates that the FC frame is returned only for the FF frame with the header 7E8F1, and the content of the FC frame is "30 FF 08".

24) SET_HD <HEADR>, [RECEIVER], [TIMEOUT]

Overview:

Set the HEADER and unique receiving address of the diagnostic device.

Parameters:

HEADR: Device header address, format:
hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address);
hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address).
hhhhhh(24bit J1850VPW/ISO header)
XX means not changing the original value. ** Ignore a certain byte.
Note: You cannot use hhhhhh to define HEADER under CAN protocol, which will be regarded as the device header of non-CAN protocol.

RECEIVER: receiving address(Optional), format:
hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address);
hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address).
**hhhh(Ignore the upper 8 bits of HEADER)

Note: You cannot use hhhhhh to define HEADER under CAN protocol, which will be regarded as the device header of non-CAN protocol.

TIMEOUT(Optional): Reply timeout value(Optional), unit: 4.096ms(0~0xFF).
The condition that this parameter is confirmed is that the first two of the parameters command have been entered (including XX).

Examples:

example 1:

>VT SET_HD 7DF, 7E8, 50

OK

equivalent to:

>AT SH 7DF
OK
>AT CRA 7E8
OK
>AT ST 50
OK

example 2:

```
>VT SET_HD 7DF-10, 7E8-F1, 50
OK
```

equivalent to:

```
>AT SH 7DF
OK
>AT CEA 10
OK
>AT CRA 7E8
OK
>AT CER F1
OK
>AT ST 50
OK
```

example 3:

```
>VT SET_HD XX, 7E8-F1, 50
OK
```

equivalent to:

```
>AT CRA 7E8
OK
>AT CER F1
OK
>AT ST 50
OK
```

example 4:

```
>VT SET_HD XX, 7E8-F1
OK
```

equivalent to:

```
>AT CRA 7E8
OK
>AT CER F1
OK
```

example 5:

```
>VT SET_HD 7DF-10
OK
```

equivalent to:

```
>AT SH 7DF
OK
>AT CEA 10
OK
```

example 6:

```
>VT SET_HD XX, XX, 50
OK
```

equivalent to:

```
>AT ST 50
OK
```

example 7:

```
>VT SET_HD XX, 416B10, 50
OK
```

equivalent to:

```
>ST FAP 416B10, FFFFFF
OK
>AT ST 50
OK
```

The header of the transmitted frame uses the default address, the receiving address is 41, 6B, 10; the receiving timeout is 200ms.

example 8:

>VT SET_HD CF33F1, **F110, 50
OK

equivalent to:

>AT SH CF33F1
OK
>ST FAP 00F110, 00FFFF
OK
>AT ST 50
OK

The header of the transmitted frame is CF33F1, the receiving address is XX, 6B, 10; The receiving timeout is 200ms.

Note: When not in the extended CAN address mode, it is wrong to ignore the send address and directly set the receive address to the extended CAN address.

Examples:

>ATZ
ELM327 v2.2
>ATH1
OK
>0100
7E8 06 41 00 BE 3F A8 13
7E9 06 41 00 BF BE F9 80
7EC 06 41 00 00 00 00 00
>VT SET_HD XX, 7E8-20
?

The prompt here indicates that the receiver cannot be set to the extended address mode when it is not in the extended address state.

>ATCEA50
OK
>VT SET_HD XX, 7E8-F1
OK
>22F190
7E8 F1 10 14 62 F1 90 57 42
7E8 F1 21 41 4A 42 39 43 35
7E8 F1 22 36 4A 42 30 33 35
7E8 F1 23 36 35 35 FF FF FF

> ATCEA
OK

>22F190
NO DATA

>0100
7E8 06 41 00 BE 3F A8 13
>VT SET_HD 7DF-40, 7E8-F1
OK
>22F190
7E8 F1 10 14 62 F1 90 57 42
7E8 F1 21 41 4A 42 39 43 35
7E8 F1 22 36 4A 42 30 33 35
7E8 F1 23 36 35 35 FF FF FF

Only when the sender is set to the extended address format (hhh-hh; hhhhhhhh-hh), it indicates that the CAN extended address mode is set. At this time, the receiver can be
set to the extended address format.

>VT SET_HD 7DF-40, 7E8
?
Indicates that the sender and receiver address formats do not match.
>VT SET_HD 7DF-40, 7E8-** OK

```
>ATCER20
OK
>22F190
NO DATA
>ATCERF1
OK
```

```
>22F190
7E8 F1 10 14 62 F1 90 57 42
7E8 F1 21 41 4A 42 39 43 35
7E8 F1 22 36 4A 42 30 33 35
7E8 F1 23 36 35 35 FF FF FF
```

When in the CAN extended address state, if the receiving address setting is ignored, the receiving filter is used as the receiving filtering address by the setting value or default value of the CER command.

example 9:

Mainly to explain the role of '*'. A '*' means to ignore the 4-digit FILTER bit. XX means no action, keep the original settings.

```
>VTSET_HD 7DF, 7E8
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
```

```
>VTSET_HD 7DF
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
```

```
>VTSET_HD 7DF, 7E*
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
7E9 10 2D 46 89 11 12 13 14
7E9 21 05 06 07 08 09 0A 0B
7E9 22 0C 0D 0E 0F 10 11 12
7E9 23 13 14 15 16 17 18 19
7E9 24 1A 1B 1C 1D 1E 1F 20
7E9 25 21 22 23 24 25 26 27
7E9 26 28 29 2A 2B 00 00 00
```

```
>VTSET_HD 7DF, 7**
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
7E9 10 2D 46 89 11 12 13 14
7E9 21 05 06 07 08 09 0A 0B
7E9 22 0C 0D 0E 0F 10 11 12
7E9 23 13 14 15 16 17 18 19
7E9 24 1A 1B 1C 1D 1E 1F 20
7E9 25 21 22 23 24 25 26 27
7E9 26 28 29 2A 2B 00 00 00
```

```
>VTSET_HD 7DF, 7E8
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
>VTSET_HD 7DF, XX
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
>VTSET_HD 7DF, ***
OK
>0189
7E8 10 2B 46 89 01 02 03 04
7E8 21 05 06 07 08 09 0A 0B
7E8 22 0C 0D 0E 0F 10 11 12
7E8 23 13 14 15 16 17 18 19
7E8 24 1A 1B 1C 1D 1E 1F 20
7E8 25 21 22 23 24 25 26 27
7E8 26 28 29 00 00 00 00 00
7E9 10 2D 46 89 11 12 13 14
7E9 21 05 06 07 08 09 0A 0B
7E9 22 0C 0D 0E 0F 10 11 12
7E9 23 13 14 15 16 17 18 19
7E9 24 1A 1B 1C 1D 1E 1F 20
7E9 25 21 22 23 24 25 26 27
7E9 26 28 29 2A 2B 00 00 00
```

25) FCST hh

Overview:

Set the timeout time of the FC frame of CAN multi-frame request information.

Parameters:

hh: The timeout for waiting for FC frame response is: hh*4.096ms. The hh input must be greater than or equal to 0x14. The default is 0x7A (500ms) after reset.

Example:

```
>VT FCST 30
OK
```

26) SDST hh

Overview:

The raw CAN multi-frame sending time interval.

Parameter:

hh: The unit is ms, and the default value is 0 after power-on.

Example:

```
>VT SDST 30
OK
```

27) ISOFI tl, th, data

Overview:

Set the KWP(fast) protocol initialization sequence.

Parameters:

tl: Duration of bus low level during initialization (1~10 bytes).

th: The duration of the bus high level during initialization.

data: Initialization request message (1~8 bytes).

Example:

> VTISOFI 25, 25, C133F18166
OK

Equivalent to:

> STIFI25,25,C133F18166
OK

28) FBA [pattern], [mask]

Overview:

Add block filter.

Parameters:

pattern: address. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

mask: Address mask. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

Example:

>VTFBA 7E8, 7FF
OK

Equivalent to:

> STFBA 7E8,7FF or STFAB 7E8,7FF
OK

29) FBA

Overview:

Clear all block filters.

Example:

>VTFBA
OK

Equivalent to:

> STFBC or STFCB
OK

30) FPA [pattern], [mask]

Overview:

Add pass filter.

Parameters:

pattern: address. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

mask: Address mask. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

Example:

>VTFPA 7E8, 7FF
OK

Equivalent to:

> STFPA 7E8,7FF or STFAP 7E8,7FF
OK

31) FPA

Overview:

Clear all pass filters.

Example:

>VTFPA
OK

Equivalent to:

> STFPC or STFCP
OK

32) FCPA [pattern], [mask]

Overview:

Add can flow control filter.

Parameters:

pattern: address. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

mask: Address mask. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

Example:

>VTFCPA 7E8, 7FF
OK

Equivalent to:

> STFAFC 7E8,7FF or STFFCA 7E8,7FF
OK

33) FCPA

Overview:

Clear all can flow control filters.

Example:

>VTFCPA
OK

Equivalent to:

> STFCFC or STFFCC
OK

34) FCTRA tx, rx

Overview:

Add CAN flow control address pair.

Parameters:

tx: FC frame sender address. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

rx: FF,CF frame sender address. Format: hhh,hhhhh,hhhhhhhh,hhhhhhhhhh.

Examples:

example1:

>VTFCTRA 6F140, 640F1
OK

equivalent to:

> STCFCPA 6F140, 640F1
OK

example2:

>VTFCTRA 7E0, 7E8
OK

equivalent to:

> STCFCPA 7E0, 7E8
OK

35) FCTRA

Overview:

Clear all CAN flow control address pairs.

Example:

>VTFCTRA
OK

Equivalent to:

> STCFCPC or STCCFCP
OK

36) M <FILTER>, <MASK>

Overview:

Monitor active data on a protocol bus that meets the filter criteria.

Parameters:

FILTER:

Receive ID(Optional), format:
hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address);
hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address).
**hhhh(Ignore the upper 8-bit ISO HEADER)

MASK(Optional):

hhh(11bit CAN ID); hhh-hh(11bit CAN ID+Extended address);
hhhhhhhh(29bit CAN ID); hhhhhhhh-hh(29bit CAN ID+Extended address).
hhhhhh(24bit J1850VPW/ISO HEADER)

Examples:

example 1:

```
>VT M 7DF, 7FF
OK
```

equivalent to:

```
>ST FAP 7DF, 7FF
OK
>ST M
```

example 2:

```
>VT M 7E8-F1, 7FF-FF
OK
```

equivalent to:

```
>ST FAP 7E8, 7FF
OK
>AT CER F1
OK
>ST M
```

example 3:

```
>VT M 416B10
OK
```

equivalent to:

```
>ST FAP 416B10, FFFFFF
OK
>ST M
```

example 4:

```
>VT M **F110, 00FFFF
OK
```

equivalent to:

```
>ST FAP 00F110, 00FFFF
OK
>ST M
```

Monitor the data sent by the ECU with the receiving address (XX6B10).

37) MFCA

Overview:

Use the current filter settings and flow control settings to monitor the OBD bus.

Example:

```
>VTMFCA
```

Equivalent to:

```
>ST M
```

38) SHOW_BUS [CAN_TYPE]

Overview:

Measure the external bus and estimate the possible physical layer protocols and frequencies.
Returns the string "Inativly" if there is no active protocol.
The default measurement is the bus activity of the HSCAN link. If you measure the link status of other CANs, you need to specify the CAN bus of which link in the parameter "CAN_TYPE".
One of the benefits of this is that the user has the opportunity to avoid using the wrong baud rate to communicate on the CAN bus, causing false alarms on other nodes on the CAN bus.
Note: For the ISO bus, it cannot be determined whether it belongs to slow init or fast init.

Parameter:

CAN_TYPE(Optional): Refers to one of HS_CAN, MS_CAN, SW_CAN.

Examples:

example 1:

>VT SHOW_BUS
Inativly
Monitoring includes HSCAN, VPWM, ISO bus activity and bus frequency. The measurement results are: No active protocol.

example 2:

>VT SHOW_BUS
P: PWM
Monitoring includes HSCAN, VPWM, ISO bus activity and bus frequency. The measurement result is: PWM.

example 3:

>VT SHOW_BUS
P: VPW
Monitoring includes HSCAN, VPWM, ISO bus activity and bus frequency. The measurement result is: VPW.

example 4:

>VT SHOW_BUS
P: ISO; F:10.4K
Monitoring includes HSCAN, VPWM, ISO bus activity and bus frequency. The measurement results are: ISO, frequency 10.4K.

example 5:

>VT SHOW_BUS
P: HSCAN; F:500K
Monitoring includes HSCAN, VPWM, ISO bus activity and bus frequency. The measurement results are: HSCAN, frequency 500K.

example 6:

>VT SHOW_BUS MS_CAN
P: MSCAN; F:125K
Monitoring includes MSCAN, VPWM, ISO bus activity and bus frequency. The measurement results are: MSCAN, frequency 125K.

39) AT_PROTOCOL_ALL

Overview:

Display all protocol details defined by [AT].

The suffix of a protocol is "WM_No: h", which indicates that the protocol is linked to a WM sequence, and the index number is h;

If the protocol is CAN protocol, you can see the details of this WM sequence through the VTDISP_CAN_WM h command;

If the protocol is a KWP protocol, you can see the details of this WM sequence through the VTDISP_ISO_WM h command.

The content after a serial number is the string "ND", which indicates that the serial number has not been defined by the user as a fixed protocol.

40) ST_PROTOCOL_ALL

Overview:

Display all protocol details defined by [ST].

41) VT_PROTOCOL_ALL

Overview:

Display all protocol details defined by [VT].

42) UCS_ATI <ascii>

Overview:

Set the string content displayed by ATI. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Example:

>VT UCS_ATI Abc1357 v1.35
OK
>ATI
Abc1357 v1.35

43) UCS_ATDI <ascii>

Overview:

Set the string content displayed by AT@1. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Example:

>VT UCS_ATDI This is an OBD communication converter
OK
>AT@1
This is an OBD communication converter.

44) UCS_STDI <ascii>

Overview:

Set STDI to display string content. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Examples:

>VT UCS_STDI Xxyyzz vx.x
OK
>STDI
Xxyyzz vx.x

45) UCS_STI <ascii>

Overview:

Set STI to display string content. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Examples:

>VT UCS_STI Xyz1234567 va.b
OK
>STI
Xyz1234567 va.b

46) UCS_MFR <ascii>

Overview:

Set STMFR to display string content. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Examples:

>VT UCS_MFR genric
OK
>STMFR
genric

47) UCS_STSN <ascii>

Overview:

Set STSN to display string content. Allow up to 47 ascii characters to be entered.

Note: This command needs to be entered when no selected protocol or WM sequence is executed, otherwise there is a risk of losing space character information.

Examples:

>VT UCS_STSN 1122334455667788
OK
>STSN

1122334455667788

48) SET_UDS <ascii>

Overview:

Set a unique device ID number, which can only be set once at the factory. Enter up to 15 characters.

Note: This command is entered when no protocol is selected or no WM sequence is executed, otherwise there is a risk of losing space character information.

Examples:

>VT SET_UDS abcdefghijklmn1
OK
>RD_UDS
abcdefghijklmn1

49) SET_MAC

Overview:

Set the 6-byte or 12-byte MAC address content, which can only be set once at the factory. Character range ('0'~'9'), ('A'~'F')

Examples:

>VT SET_MAC 11 22 33 44 55 66
OK
or
>VT SET_MAC 12 23 34 45 56 67 78 89 9A AB BC CD
OK

50) RD_MAC

Overview:

Read the contents of the MAC address. If there is MAC related information set, it will be displayed, otherwise it will display '?'.

Examples:

>VT RD_MAC
12-23-34-45-56-67-78-89-9A-AB-BC-CD

51) RD_UDS

Overview:

Read unique device ID number.

Examples:

> VT RD_UDS
abcdefghijklmn1

52) WT_EE <POS>, <DATA[1~8]>

Overview:

Write data to a user-defined EEPROM of a specified length.

Since many APP developers have proposed to have a piece of space in their own device, they can be used to perform actions such as recording and encryption.

So, we reserved 256 bytes of EEPROM space for the user to use.

Note: The MIC3X1X does not do deliberate protection, so according to the characteristics of the EEPROM, we recommend that the user do not write more than 100,000 times.

Parameters:

POS： The format is hh, the location in the user-defined EEPROM (00~FF)

DATA: 1~8 bytes of data, format: hh~hh hh hh hh hh hh hh hh

Examples:

```
>VT WT_EE 00, 01 02 03 04 050 6 07
OK

>VT WT_EE 08, 11 12 13 14 15 16 17
OK
>VT WT_EE 10, 21 22 23 24 35 26 27 28
OK
>VT WT_EE 18, 61 62 63 64 56 66 67 78
OK

>VT RD_EE 01, 15
02 03 04 05 06 07 00 11
12 13 14 15 16 17 00 21
22 23 24 35 26
```

53) RD_EE <POS>, <LEN>

Overview:

Read data for a user-defined EEPROM location of a specified length.

Parameters:

POS： The format is hh, the location in the user-defined EEPROM (00~FF)

LEN: The format is hh, data length, 01~FF

Examples:

```
>VT RD_EE 00, 20
01 02 03 04 05 06 07 00
11 12 13 14 15 16 17 00
21 22 23 24 35 26 27 28
61 62 63 64 56 66 67 78
```

54) POWERMANAGE

Overview:

Read the power management settings.

Example:

```
>VT PowerManage
SLEEP:
UART            (OFF)
OBD             (OFF)
VOLTAGE          (0V)
IGN            (OFF)
WAKE:
UART            (OFF)
OBD             (ON)
VOL DEEP DROP (4.0V)
IGN            (OFF)
```

"**SLEEP**:" Indicates the conditions for entering sleep.

UART: Indicates the switch that enters sleep when the serial port is silent;

OBD: Indicates the switch that enters sleep when the OBD data bus is silent;

VOLTAGE: Indicates the voltage threshold when entering sleep;

IGN: Indicates the switch that enters sleep according to the IGN pin state.

"**WAKE**:" Indicates the condition to wake from sleep.

UART: Indicates the switch setting that wakes up the MIC3X1X from sleep due to the active serial port;

OBD: Indicates the switch setting that causes the MIC3X1X to wake up from sleep due to the active OBD data bus;

VOL DEEP DROP: Indicates the set value of the battery voltage drop used to wake up the IC331X from sleep;

IGN: Enable wake-up switch when IGN pin voltage is opposite to sleep voltage;

55) PDVS <VOLTS>, <TIMER>

Overview:

Set the conditions for entering sleep. When the MIC3X1X enters the LOW POWER mode, the battery voltage is lower than VOLTS and continues for TIMER seconds, the MIC3X1X goes to sleep.

Parameters:

VOLTS：Voltage value. format: d.d or dd.d; range: (1.0~25.0). unit: volt; Carry system: Decimal.

TIMER: duration. format:d~ddddd; range:(1~65535); unit: sec; Carry system: Decimal.

Example:

```
>VTPDVS 6.8, 20
OK
```

It means that in the standby state, the voltage of 6.8V is continuously detected for 20 seconds, and the MIC3X1X enters the sleep state.

56) VDWK <VOLTS>, <TIMER>

Overview:

After the MIC3X1X enters the LOW POWER mode, the battery voltage drops below VOLTS and continues for MINMER milliseconds.

The MIC3X1X wakes up from the LOW POWER state. It enters the normal working state.

Parameters:

VOLTS： Voltage value. format: 0.d or d.d; range: (0.1~9.9). unit: volt; Carry system: Decimal.

TIMER: duration. format: d~dd; range: (1~100). unit: ms; Carry system: Decimal.

Examples:

>VTVDWK 2.5, 20

OK

Indicates that the battery voltage wakes up from a low power state when there is a drop of more than 2.5 volts for more than 20ms.

57) VLCW [+/-]<VOLTS>, <TIMER>

Overview:

Wake up the device by the voltage changing.

Parameters:

none: Disable voltage change wake-up function.

+: The device wakes up through voltage rise, -: Device wakes up by voltage drop, Unsigned bit:The device wakes up when the voltage rises or falls.

VOLTS： value. format: d.d or d.d; Range: (0.1~8.0). unit: V.

TIMER: Sampling interval. format: d~dd; Range: (1~65535). unit: ms.

Examples:

>VTVLCW +0.5, 2000

OK

It means that the battery voltage is sampled at a sampling rate of 0.5HZ. When the battery voltage rises by more than 0.5V, the MIC342X wakes up from the low power consumption state.

58) VLRD

Overview:

The battery voltage is displayed every 4ms. After receiving any character, it is in standby mode.

Parameters:

none

Examples:

>VTVDRD

12.1V

59) SWGP <switch 1>, … <switch n>

Overview:

Group settings for the switch class AT command.

The AT commands are as follows: see SWGP _table

Example:

>VT SWGP AL, KW0, CAF1, E0, CSM1, S1, AT0, GT1

OK

Equivalent to:

>ATAL
OK
>ATKW0
OK
>ATCAF1
OK
>ATE0
OK
>ATCSM1
OK
>ATS1
OK
>ATAT0
OK
>STCSEGT1
OK
>

SWGP _table			
parameter	instruction	parameter	instruction
AL	ATAL	NL	ATNL
D0	ATD0	D1	ATD1
E0	ATE0	E1	ATE1
H0	ATH0	H1	ATH1
L0	ATL0	L1	ATL1
M0	ATM0	M1	ATM1
R0	ATR0	R1	ATR1
S0	ATS0	S1	ATS1
V0	ATV0	V1	ATV1
W0	ATW0	W1	ATW1
AT0	ATAT0	AT1	ATAT1
AT2	ATAT2		
KW0	ATKW0	KW1	ATKW1
CAF0	ATCAF0	CAF1	ATCAF1
CFC0	ATCFC0	CFC1	ATCFC1
CSM0	ATCSM0	CSM1	ATCSM1
JHF0	ATJHF0	JHF1	ATJHF1
GT1	STCSEGT1	GT0	STCSEGT0
GR1	STCSEGR1	GR0	STCSEGR0

parameter	Instructions for use
BZF0	When GR1 is valid, if 7F xx 78 is received, it will be displayed as a new message
BZF1	When GR1 is valid, if 7F xx 78 information is received, it will not be displayed
CRF0	Do not add any check or format bytes to the response message received by CAN
CRF1	Add length and check byte to CAN reply message
FCDA1	Allow to extract BS, ST information from ATFCSD command or VTSET_CAN_FC command.
FCDA0	It is not allowed to extract BS, ST information from ATFCSD command or VTSET_CAN_FC command.

60) TOST TimoutType:ms

Overview:

Set a certain type or a group of timeout parameters. Multiple parameters can be sent at the same time.

Parameters:

TimoutType contains various timeout types, which are described as follows:

IP1X ---- The maximum inter-byte time (P1max) of the message received by the ISO/KWP protocol;

IP4 ---- The inter-byte time of the ISO/KWP protocol transmission message (P4);

REP ---- timeout between message request and reply;

REQ ---- message sending timeout;

REPQ ---- the interval between the last message reply and the next request;

Examples:

example1:

>VTTOST IP1X: 100

OK

Equivalent to:

>STIP1X 100

OK

61) TP_RTS TotalLen, BlockSize[,EnChk]

Overview:

Request to establish a multi-packet data transmission pipeline. This command must be executed with the long byte command disabled (STCSEGT0 or VTSWGPGT0), otherwise return? If the total number of bytes in the request is less than the size of the packet, return ?.

Parameters:

TotalLen: The size of the data to be transmitted. This parameter is greater than or equal to 50 and less than 750.Other values are invalid.

BlockSize: The size of each packet of data, the value of this parameter does not exceed 50, otherwise it is considered invalid.

EnChk: This parameter is optional.A value of 1 indicates that a check byte (sum check) needs to be added at the end of each packet of data.

Examples:

>VT TP_RTS 255, 50, 1

OK

Request to open a batch data transmission channel. The total length of the data to be transmitted is 255 bytes, with a maximum of 50 bytes per packet of data, allowing the parity of each packet of data.

62) TP_DT ii hh ... hh [chksum]

Overview:

Single packet data transmission.

Parameters:

ii: Package number

hh: data

chksum: Checksum

Examples:

>VT TP_DT 00 2E3044025AFF135A5A5A5A0000004201164401100200000010900000000000000000E890000000000C000000015AFF135A 6D

OK

A packet with a serial number of 0 (one byte for the packet serial number), 50 bytes of data, and a one-byte checksum. A reply of OK indicates that the check passes.

63) TP_END [,h]

Overview:

If each packet is received to indicate that the data transmission is complete, close the transmission pipeline and send a CAN multi-frame request; if some data packets are missed, return ?.

Parameters:

h: indicates that a maximum of several reply messages are received.If there is no parameter after the command, it indicates that an unlimited number of reply messages are allowed to be received until the reply times out.

Examples:

example1:

>VT TP_END

640F1036E3044

example2:

>VT TP_END,2

640F17F6E78

640F1036E3044

64) TP_ABORT

Overview:

Abandon the transmission and close the transmission pipeline.

Example:

>VT TP_ABORT

OK

5.6 Supplemental AT command

1) DPCLK

Overview:

Display the physical connection of the CAN part of each [AT] definition protocol.

Examples:

>AT DPCLK
P6: HS_CAN
P7: HS_CAN
P8: HS_CAN
P9: HS_CAN
PA: HS_CAN
PB: MS_CAN
PC: MS_CAN
PD: MS_CAN
PE: MS_CAN
PF: SW_CAN

Indicates that the physical connection of protocol 6~A is HSCAN, the physical connection of protocol B~E is MSCAN, and the physical connection of protocol F is SWCAN.

2) LNKHS Ph

Overview:

Link the HSCAN physical interface to a protocol defined by [AT].

Examples:

>ATLNKHS P7
OK
> AT DPCLK
P6: HS_CAN
P7: HS_CAN
P8: HS_CAN
P9: HS_CAN
PA: HS_CAN
PB: MS_CAN
PC: MS_CAN
PD: MS_CAN
PE: MS_CAN
PF: SW_CAN

3) LNKMS Ph

Overview:

Link the MSCAN physical interface to a protocol defined by [AT].

Examples:

>atlnkms p8
OK
>atdpclk
P6: HS_CAN
P7: HS_CAN
P8: MS_CAN
P9: HS_CAN
PA: HS_CAN
PB: MS_CAN
PC: MS_CAN
PD: MS_CAN
PE: MS_CAN
PF: SW_CAN

4) LNKSW Ph

Overview:

Link the SWCAN physical interface to a protocol defined by [AT].

Example:

> ATLNKSW PF
OK
>atdpclk
P6: HS_CAN
P7: HS_CAN
P8: MS_CAN
P9: MS_CAN
PA: HS_CAN
PB: MS_CAN
PC: MS_CAN
PD: MS_CAN
PE: MS_CAN
PF: SW_CAN

5) SWCL0/1

Overview:

Set the Rtool switch.
1: connected; 0: disconnected.

Example:

>ATSWCL 1
OK

6) TM h

Overview:

Set the SWCAN transceiver working mode.

The range of h is (0~7), and the part larger than 3 is equivalent to performing the ATSWCL1 action at the same time.

Example:

>ATTM7

OK

Equivalent to:

>ATTM3

OK

>ATSWCL1

OK

5.7 Advanced Features

(1) Periodic (Wakeup) Messages

Some applications require that there be periodic messages sent by the test equipment (scan tool) in order to maintain a connection.

If these messages do not arrive in a timely fashion, the ECU will either revert to a default mode of operation, or close the connection and go into a low power 'sleep' mode.

In order to stop the ECU from doing this, you will need to send what we term as 'wakeup' messages. Some texts also refer to these as CAN periodic messages.

The MIC3X1X takes a series of convenient and flexible actions to help users quickly implement the ability to send WM sequences on a regular basis.

Mainly by CAN_WM, ISO_WM two VT instructions to achieve. The following is an introduction to these two instructions:

CAN_WM <No>, <PROTOCOL>, <HEADER>, <DATA1~8>, <PERIOD>, <MODE>

ISO_WM <No>, <PROTOCOL>, <HEADER>, <DATA1~5>, <PERIOD>, <CTRL>

Overview:

The purpose of these commands is to simplify the setup of the periodic wake/hold sequence on the CAN/ISO bus with one instruction.

This instruction can implement two aspects of WM sequence setting action.

<1> As a temporarily set WM sequence, only the current protocol determines that it is a CAN or ISO related protocol, otherwise the entire setting is invalid;

<2> As a defined [VT] defined protocol private WM sequence.

When the parameter <No> is 0, the WM sequence is only used as a temporary WM sequence, and is invalid when the device is powered on or performs a reset action;

When the parameter <No> is 1~8, it defines a protocol private WM sequence for a defined [VT].

The private WM sequence is valid only when the VTP1xx command is executed to switch to this protocol.

A maximum of 8 groups can be set for a private WM sequence.

If [VT] defines a protocol deletion action or deletes a WM sequence action, such a link is always valid and is not affected by the reset action.

When both the temporary WM sequence is defined and the current protocol has its own private WM sequence, the private WM sequence replaces the temporary WM for wake/hold actions.

The temporary sequence needs to be deleted before the user uses ATWM, WD, WH, WT, etc., because the priority of the temporary sequence is higher than the sequence set by these AT commands.

If the current protocol is a protocol defined by [VT] and has a private WM sequence, it is also necessary to delete the WM sequence of the protocol link to use the above AT command related to wakeup/hold.

Note: When using ATSP, STP two instructions to switch protocols, the protocol private WM sequence defined by [VT] will be disabled.

At this time, if there is a temporary WM sequence, the temporary sequence is valid; if there is no temporary WM sequence: the target protocol is CAN protocol, there is no temporary WM sequence; the ISO protocol uses the default WM sequence (if there is a default WM in this

protocol).

For a detailed introduction and examples of these two commands, Please refer to the chapter "VT Command description".

(2) Low voltage sleep function

The VTPDVS command can set a voltage threshold to ensure that the diagnostic device sleeps.

The original intention of this command is that when the vehicle battery voltage is too low, the device can ignore the normal request of the user and directly enter the low power state.

When the user uses the VTPDVS command to set the voltage below the set voltage so that the MIC3X1X enters a low power state, it may occur that the device is always in a sleep state because the set voltage is higher than the normal voltage of the vehicle battery.

At this point, the user needs to re-power the diagnostic device. Within 4 minutes after power-on, the set voltage can be modified without ignoring the timeout parameter of the command.

The timeout condition of the command takes effect after 4 minutes. It is generally recommended that the user set the timeout parameter of the command to be greater than 1000 seconds.

Therefore, after ensuring that the voltage parameters are set incorrectly, there is still enough time to remedy.

(3) Multi-byte request function

The ELM327 supports only up to 8 bytes of OBD requests.

However, in some special cases the length of the request byte is greater than 8 bytes.

At this time, users are more difficult to use such special requests.

The MIC3X1X has allowed more than 8 bytes of requests for this type of application scenario, and we also recommend that the user request information not exceed 256 bytes in length.

In the J1850 VPWM and ISO/KWP protocols, a single long frame request is used to implement a request exceeding 8 bytes;

In the CAN protocol, it is a multi-frame transmission method.

The flow control handshake/session mechanism is automatically completed under the ISO15765 protocol, which is as convenient as a request of less than 8 bytes.

Overview:

example 1:

```
>VT SET_HD 6F1-63, 663-F1
OK
>VT SET_CAN_FC 30 FF 19, 3
OK
>31 01 0F 1F 0B 00 3A 00 06
663 F1 10 12 71 01 0F 1F 0B
663 F1 21 00 3A 1D 05 FD 91
663 F1 22 5F 4D E2 00 3A 00
663 F1 23 06 FF FF FF FF FF
```

example 2:

```
>VT SET_HD 735-40, 73D-F1
OK
>VT SET_CAN_FC 30 FF 19, 3
OK
>34 CF 00 00 00 09 00 00 4D 05 05 03 00 00 00
05 1A 17
73D F1 10 11 74 08 09 0A 0B
73D F1 21 0C 0D 0E 0F FE ED
73D F1 22 DC CB BA A9 98 87
73D F1 23 76 65 54 43 32 21
73D F1 24 10 55 55 55 55 55
```

(4) ISO15765 FC frame related operations.

The MIC3X1X supports four modes of FC frame reply format. They are mode 0~ mode 3.

Among them: Mode 0 is the default mode. The whole operation mode is the same as ELM327.

Mode 1 is the same as ELM327 in AT command mode. There is a slight difference when the user sets the head address of the FC frame using the VT SET_CAN_FC command.

If the HEADER of the set FC frame contains an extended address, the MIC3X1X will automatically add the address value to the data area of the FC frame, and the ELM327 does not have such a function.

Overview:

example 1:

>VT SET_CAN_FC 30 FF 19, 735-40, 1
OK

Here the HEADER of the frame is: 735; The data is: 40 30 FF 19.

example 2:

>AT FC SH 735
OK
>AT FC SD 40 30 FF 19
OK
>AT FC SM 1
OK

The execution result of Example 2 is exactly the same as that of Example 1.

Mode 2 operates in the same way as the ELM327.

When the HEADER of the FC is set to the format with the extended address in the VT SET_CAN_FC command, the extended address has no practical meaning and will be ignored in the FC frame.

Mode 3 can only be set by the VT SET_CAN_FC command, in which the HEADER of the FC frame is the ATSH, ATCEA or VTSET_HD command setting value.

Overview:

example 1：

>VT SET_HD 735-40
OK
>VT SET_CAN_FC 30 FF 19, 3
OK

Here the HEADER of the frame is: 735; The data is: 40 30 FF 19.

example 2：

>AT SH 735
OK
>AT CEA 40
OK
>AT FC SD 30 FF 19
OK
>VT SET_CAN_FC, XX, 3
OK

The execution result of Example 2 is exactly the same as that of Example 1.

(5) Custom protocol

The MIC3X1X has 64 groups of customizable protocols.
This approach is mainly to allow the user to configure the used protocol once, and do not need to repeatedly configure the PP zone parameters in the future use.
Custom protocols must use the VTP1hh command for command switching; all [VT] custom protocol details can be viewed with the VT VT_PROTOCOL_ALL command.
The MIC3X1X has 8 customizable custom WM sequences for the ISO/KWP protocol that can be linked to [VT] custom protocols that conform to the ISO/KWP protocol.
There are also up to 8 sets of custom CAN protocol WM sequences that can be linked to [VT] custom protocols that conform to the CAN protocol.
That is, the MIC3X1X allows up to 16 sets of [VT] custom protocols to have private Wake/keep sequences.

(6) Switch command group

There are many commands in the AT command that indicate the state of the switch, which is more ambiguous each time the user sets it.

To this end, the MIC3X1X has specially opened a command SWGP to operate these switch information in one operation.

These switch information contains: AL, D0/1, E0/1, H0/1, L0/1, M0/1, NL, R0/1, S0/1, V0/1, W1/0, AT0/1/2, KW0/1, CAF0/1, CFC0/1, CSM0/1, JHF0/1, GT0/1, GR0/1,PCB0/1, IAT0/1, FC0/1, IMCS0/1, BZF0/1, CRF0/1

(7) Storage unit

Some users want a storage area to store private information.
The MIC3X1X opens up 256 bytes of EEPROM space for easy reading and writing.
Since the number of EEPROM safe reads and writes is 100,000 times, it is recommended that the user erase the number of times each unit of the memory area does not exceed 100,000 times.

(8) Configuration item

The configuration area is compatible with ELM327 V2.2 and ELM329 V2.2 with 53 configuration items.

(9) Command set

There are three sets of command sets inside the MIC3X1X, which are the AT, ST and VT command sets.

The AT and ST command sets are compatible with ELM, the command set of STN, and the VT command set is a set of macro instructions. Some instructions are powerful, and the function of one instruction is equivalent to the execution effect of several AT or ST commands.

The VT instruction is constantly evolving, and its purpose is to reduce the number of instructions, which is convenient for users. Please refer to the chapter "VT Command description".

(10) TP transmission protocol

In some APPs, up to several kilobytes of data are written to the ECU. Because APP and adapter generally communicate through Bluetooth or WIFI interface.Generally, Bluetooth and WIFI use a segmented packet transmission mechanism when communicating. As many as 1K or so ASCII characters are generally broken into several packets for transmission. If the wireless communication is in a harsh environment, there may be packet loss. Therefore, there is a risk that the user's request data is defective, resulting in unpredictable consequences. Of course, in this case, the user can use the ATE1 command to verify that the data transmission is completely correct. However, such a large data packet often requires multiple attempts to succeed in a harsh environment, so it will affect the execution efficiency of the APP software.

MIC3X1X defines a multi-packet transmission protocol to implement packet processing when many bytes of data are requested, and establishes a confirmation mechanism. This transmission protocol enables APPs to write up to 4128 bytes of data to the ECU.Divide a large set of request data into several small data packets for segmented transmission. Divide a large group of request data into several small data packets and send them in segments. Each sub-data packet is smaller than the smallest data packet of Bluetooth or WIFI, and the maximum data per packet in MIC3X1X does not exceed 90 bytes.A check mechanism is established at the same time as the transmission. If a packet transmission fails, you only need to retransmit the data of the sub packet.After the adapter receives all the packet data and completes the verification, it can communicate with the vehicle data. This mechanism not only increases the anti-interference performance, but also improves the execution efficiency of the APP.

The transmission protocol is divided into three parts under normal communication conditions, namely creating a multi-packet transmission channel (TP.RTS), transmitting sub-packet data (TP.DT), closing the transmission channel and sending a request (TP.END).In addition, if there are more important tasks to be processed in the multi-packet transmission process, the multi-packet transmission can be aborted (TP.ABORT).The above 4 parts are implemented by 4 VT commands: TP_RTS TotalLen, BlockSize [, EnChk]; TP_DT ii hh ... hh [chksum]; TP_END; TP_ABORT. They represent the four processes of creating, transmitting, closing and aborting.

The function of the TP_RTS command is to create a multi-packet transmission channel.It declares the maximum amount of data to be transmitted at one time, the size of the sub-packets, and whether to perform a verification action on each sub-packet.

The TP_DT command implements the transmission and verification of sub-packet data (if verification is allowed). The check in the sub-packet uses a simple sum check. The value of the check word is equal to the sum of all transmitted data in the packet except the packet sequence number byte.

The TP_END command realizes the missing packet check, implements the data request action to the ECU and specifies the number of received reply messages when all sub-packages are received.

The function of the TP_ABORT command is to abort the current multi-packet transmission

process at any time.

The data request to the ECU will only occur after receiving the TP_END command and confirming that all sub-packets have been received correctly.

We recommend that the verification mechanism be allowed when the pipeline is established.This approach will increase the security of the content of the subpackage.

If the verification fails (return?) When a sub-packet is transmitted, the user can use the TP_DT command to resend the packet data before closing or terminating the transmission channel. There is no limit to the number of retransmissions.

(11) ISO15765 receiving information format 2

The purpose is to format the ISO15765 multi-frame response into a line of information, thereby simplifying the steps for users to process this type of information.

Enter this mode through STCSEGR1 or VTSWGPGR1, and exit this mode through STCSEGR0 or VTSWGPGR0. The switch variables GR, BZF, and CRF affect the formatting method.

example:

(1) Assuming that when H1, CER is valid, the received data is as follows:

00 00 07 E8 40 10 40 00 01 02 03 04
00 00 07 E8 40 21 05 06 07 08 09 0A
00 00 07 E8 40 22 0B 0C 0D 0E 0F 10
00 00 07 E8 40 23 11 12 13 14 15 16
00 00 07 E8 40 24 17 18 19 1A 1B 1C
00 00 07 E8 40 25 1D 1E 1F 20 21 22
00 00 07 E8 40 26 23 24 25 26 27 28
00 00 07 E8 40 27 29 2A 2B 2C 2D 2E
00 00 07 E8 40 28 2F 30 31 32 33 34
00 00 07 E8 40 29 35 36 37 38 39 3A
00 00 07 E8 40 2A 3B 3C 3D 3E 3F 00

After executing VTSWGP H0, GR1, S0:

000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F20212223242526272
8292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F

After executing VTSWGP H1, GR1, S1, CRF1:

7E8 40 00 40 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F 07 E0

After executing VTSWGP H1, GR1, S0, CRF1:

7E8-40,0040,000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F2021222
32425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F,07E0

Divided into 4 sections and separated by','. They are address section, length section, data section, and

check section.

(2) Assuming that the above information has one more frame of busy information before the eply, the switch variable BZF can be enabled to ignore the printing of the busy signal, as follows:

00 00 07 E8 40 03 7F 00 78

After executing VTSWGP H1, GR1, S0, CRF0, BZF0:

7E8-40,0003,7F0078,00F7

7E8-40,0040,000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F2021222
32425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F,07E0

After executing VTSWGP H1, GR1, S0, CRF0, BZF1:

7E8-40,0040,000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F2021222
32425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F,07E0

(12) About FAP, FAB, FAFC, FC ID

(1) Flow control default

When no flow control related commands are executed, the default is ELM working mode, and the entire operation mode of the CAN protocol is consistent with ELM regulations.

When more than one flow control command is executed, the flow control mode takes effect.

The flow control commands are as follows:

STFCP, STFAP, STFCFC, STFAFC.

(2) FAP quantity

Under the non-CAN protocol, there are 3 groups, more than 3 groups are invalid; under the CAN protocol, there are 16 groups, if more than 16 groups, the error "OUT OF MEMORY" will be reported.

(3) FAB quantity

Under the non-CAN protocol, there are 3 groups, more than 3 groups are invalid; under the CAN protocol, there are 16 groups, if more than 16 groups, the error "OUT OF MEMORY" will be reported.

(4) FAFC quantity

Under the non-CAN protocol, there are 3 groups, more than 3 groups are invalid; under the CAN protocol, there are 8 groups, if more than 8 groups, the error "OUT OF MEMORY" will be reported

(5) FC ID number

It is a 32-unit FIFO, setting more than 8 groups will overwrite the previous setting.

(13) ISO15765 FC frame control

Under the ISO15765 protocol, the MIC3X1X sends out the FC frame content immediately after receiving the FF frame, and no FC frame is sent after the received CF frame.

Some users need to strictly follow the provisions of the ISO15765 protocol, and the interval of subsequent FC frames is controlled according to the BS byte in the FC frame.
For this reason, MIC3X1X has added the operation of FC frame analysis.
The specific implementation process is as follows:

example 1:

>VTSWGP FCDA1
OK
>VTSET_CAN_FC 300800, 1, 7E0
OK
>0601
...

Equivalent to:

>VTSWGP FCDA1
OK
>ATFCSH7E0
OK
>ATFCSD 300800
OK
>ATFCSM1
OK
>0601
...

The switch FCDA is set to 1, which means that it is allowed to parse the set FC frame data. The result of this example is to send FC frames every 8 CF frames after receiving the FF frame. The header of the FC frame is 7E0, and the content of the FC frame is 0x30, 0x08, 0x00.

example 2:

>VTSWGP S0,H1,AL,AT0,FCDA1
OK
>VTSET_HD 6F1-40, 640-F1, FF
OK
>VTSET_CAN_FC 300208, 3
OK
>22F101
...

Equivalent to:

>ATS0
OK
>ATH1
OK
>ATAT0
OK
>VTSWGP FCDA1
OK
>ATSTFF
OK
>ATFCSH6F1
OK
>ATSH6F1
OK
>VTFCTRA
OK
>VTFCTRA 6F140, 640F1
OK
>ATFCSD40300208
OK
>ATFCSM1
OK
>ATCEA40
OK
>ATCERF1
OK
>ATCRA640
OK
>22F101
...

The execution result of this example is to send FC frames every 2 CF frames after receiving the FF frame, the interval of CF frames is 8ms. The header of the FC frame is 6F140, and the content of the FC frame
It is 0x30, 0x02, 0x08.

After the MIC3X1X is reset, the FCDA switch is set to 0. In this case, the FC frame is only sent once after receiving the FF frame.

5.8 MIC3X1X_PP_area

PP	Description	values	Default	Type
0x00	Perform an AT MA command after powerup or reset	00=ON;FF=OFF	FF(OFF)	R
0x01	Printing of header bytes(AT H default setting)	00=ON;FF=OFF	FF(OFF)	D
0x02	Allow long messages (AT AL default setting)	00=ON;FF=OFF	FF(OFF)	D
0x03	NO DATA timeout time (AT ST default setting) setting = value*4.096msec	00 to FF	0x32(205ms)	D
0x04	Default Adaptive Timing mode (AT AT setting)	00 to 02	0x01	D
0x05				
0x06	OBD Source (Tester) Address. Not used for J1939 protocols.	00 to FF	0xF1	R
0x07	Last Protocol to try during automatic searchs	01 t0 0C	0x09	I
0x08	Display a false ID ('ELM327 V2.2')	00=ON;FF=OFF	FF	R
0x09	Character echo (AT E default setting)	00=ON;FF=OFF	0x00(ON)	R
0x0A	Linefeed Character	00 to FF	0x0A	R
0x0B				
0x0C	RS232 baudrate divisor when pin27 is high	00 to FF	0x68(38.4)	P
0x0D	Carriage Return Character	00 t0 FF	0x0D	R
0x0E	Power Control options. Each bit controls an option, as follows: b7: Master enable. 0: off 1: on if 0, pins 26 and 44 perform as described for v1.0 to v1.3a (must be 1 to any Low Power functions) b6: Pin 44 full power level. 0: low 1: high normal output level, is inverted when in low power mode b5: Auto LP control. 0: disabled 1: enabled allows low power mode if the RS232 activity stops b4: Auto LP timeout. 0: 5mins 1: 20mins no RS232 activity timeout setting b3: Auto LP warning. 0: disabled 1: enabled if enabled, says'ACT ALERT' 1 minute before RS232 timeout b2: Ignition control 0: disabled 1: enabled allows low power mode if the IgnMon input goes low b1: Ignition delay 0: 1sec 1: 5sec delay after IgnMon(pin 26) returns to a high level, before normal operation resumes b0: reserved for future - leave set at 0	00 t0 FF	0x9A 0b10011010	R
0x0F	Activity Monitor options. Each bit controls an option, as follows: b7: mornitor master control 0: disabled 1: enabled must be 1 to allow b3 to b6 b6: allow wake from Low Power 0: no 1: yes wakes on shift from no activity to activity b5: Auto LP control. 0: disabled 1: enabled allows low power mode if the OBD activity stops.	00 t0 FF	0xD5 0b11010101	D

MIC3X1X_PP_area (continued)

PP	Description	values	Default	Type
0x0F	b4: Auto LP timeout. 0: 30secs 1: 150secs no OBD activity timeout setting b3: Auto LP warning. 0: disabled 1: enabled if enabled, says'ACT ALERT' on timeout b2: reserved for future - leave set at 1 b1: add exclamation mark 0: no 1: yes if 1, sends '!'before ACT ALERT and LP ALERT b0: LP LED 0: disabled 1: enabled if 1, the OBD Tx LED flashs when in Low Power mode (one 16msec flash repeated every 4 seconds)	00 t0 FF	0xD5 0b11010101	D
0x10	J1850 voltage setting time seeting(in ms) = (PP 10 value)*4.096	00 t0 FF	0x0D(53 ms)	D
0x11	J1850 Break Signal monitor enable (reports BUS ERROR if break signal duration limits are exceeded)	00=ON;FF=OFF	00(ON)	D
0x12	J1850 Volts (pin 15) output polarity normal = Low output for 5V, High output for 8V invert = High output for 5V, Low output for 8V	00=invert FF=normal	0xFF (normal)	R
0x13	Time delay added between protocols 1&2 during a search setting (in ms) = 150 + (PP13 value)*4.096	00 to FF	0x55 (498 ms)	I
0x14	ISO/KWP final stop bit width (provides P4 interbyte time) setting (in us) = 98 + (PP14 value)*64	00 to FF	0x50 (5.2ms)	D
0x15	ISO/KWP maximum inter-byte time(P1), and also used for the minimum inter-message time(P2). setting (in ms) = (PP15 value)*2.112	00 to FF	0x0A (21 ms)	D
0x16	Default ISO/KWP baud rate (AT IB default setting) Note: 4800,12500 and 15625 baud can not be a defaults	00=96 FF=10	FF (10.4K)	R
0x17	ISO/KWP wakeup message rate (AT SW default setting) setting (in msec) = (PP17 value)*20.48	00 to FF	0x92 (3.0sec)	D
0x18	ISO/KWP delay before a fast init, if a slow init has taken place setting (in msec) = 1000 + (PP18 value)*20.48	00 to FF	0x31 (2.0sec)	I
0x19	ISO/KWP delay before a slow init, if a fast init has taken place setting (in msec) = 1000 + (PP19 value)*20.48	00 to FF	0x31 (2.0sec)	I
0x1A	Protocol 5 fast initiation active time (TiniL) setting (in msec) = (PP1A value)*2.5	00 to FF	0x0A (25 msec)	D
0x1B	Protocol 5 fast initiation active time (TiniH) Setting (in msec) = (PP1B value)*2.5	00 to FF	0x0A (25 msec)	D
0x1C	ISO/KWP outputs used for initiation (b7 to b2 are not used) b1: L line(pin 21) 0: disabled 1: enabled b0: K line(pin 20) 0: disabled 1: enabled if disabled, an output will remain low during protocol initiations	00 to FF	0x03 (0b00000011)	D

MIC3X1X_PP_area (continued)

PP	Description	values	Default	Type
0x1D	ISO/KWP P3 time (delay before sending requests)	00 to FF	0x0F	D
	Ave time (in msec) = (PP1D value - 0.5)*4.096		(59 msec)	
0x1E	ISO/KWP K line minimum quiet time before an init can begin (W5)	00 to FF	0x4A	D
	setting (in msec) = (PP1E value)*4.096		(303 msec)	
0x1F	KWP byte count includes the checksum byte?	FF=NO;00=YES	FF(NO)	R
0x20	Default (Single Wire) Transceiver Mode	00 to 03	0x03	D
	M0-M1 pin setting during normal CAN operation		(normal)	
0x21	Default CAN Silent Monitoring setting (for AT CSM)	FF=ON;00=OFF	0xFF(ON)	R
0x22	CAN Wakeup message rate (AT SW default setting)	00 to FF	0x62	D
	setting = value*20.48ms		(2.0 sec)	
0x23	Default Wakeup Mode (AT WM setting)	00 to 02	0x00(OFF)	D
0x24	CAN auto formatting (AT CAF default setting)	00=ON;FF=OFF	0x00(ON)	D
0x25	CAN auto flow control(AT CFC default setting)	00=ON;FF=OFF	0x00(ON)	D
0x26	CAN filler byte (used to pad out messages)	00 to FF	0x00	D
0x27				
0x28	CAN Filter settings (controls CAN sends while searching)		0xFF (0b11111111)	D
	The bits of this byte control options, as follows:			
	b7: 500 kbps match 0: ignored 1: required	00 to FF		
	b6: 250 kbps match 0: ignored 1: required			
	b5 to b1: reserved for future - leave set to 1			
	b0: send if bus is quiet 0: not allowed 1: allowed			
0x29	Printing of the CAN data length (DLC) when printing header bytes	00=ON;FF=OFF	0xFF	D
	(AT D0/D1 default setting)		(OFF)	
0x2A	CAN Error Checking (applies to protocols 6 to C)		0x3C (0b00111100)	D
	Each bit of this byte controls an option, as follows:			
	b7: ISO15765 Data Length 0: accept any 1: must be 8 bytes			
	b6: ISO15765 PCI = 00 0: allowed 1: not allowed			
	b5: Search after ERR94 0: normal 1: CAN is blocked			
	b4: Search after LV RESET 0: normal 1: CAN is blocked	00 to FF		
	b3: Wiring Test 0: bypass 1: perform			
	Processing 7F xx 78's:			
	b2: enabled (CAN&KWP) 0: no 1: yes			
	b1: valid Modes (xx values) 0: all 1: only 00 to 0F			
	b0: valid CAN protocols 0: all 1: only ISO15765			
0x2B	Protocol A (SAE J1939) CAN baudrate divisor	01 to 40	0x02	R
	baudrate (in kbps) = 500 / (PP2B value)		(250 Kbps)	

MIC3X1X1_PP_area (continued)

PP	Description	values	Default	Type
0x2C	Protocol B (USER1) CAN options.			R
	Each bit of this byte controls an option, as follows:			
	b7: Transmit ID Length 0: 29 bit ID 1: 11 bit ID			
	b6: Data Length 0: fixed 8 byte 1: variable DLC			
	b5: Receive ID Length 0: as set by b7 1: both 11 and 29 bit			
	b4: baudrate multiplier 0: x1 1: x 8/7			
	b3: reserved for future - leave set at 0	00 to FF	0xE0	
	b2, b1 and b0 determine the data formatting options:		(0b11100000)	
	b2 b1 b0 Data Format			
	0 0 0 none			
	0 0 1 ISO 15765-4			
	0 1 0 SAE J1939			
	Other combinations are reserved for futrue updates - results will			
	be unpredictable if you should select one of them.			
0x2D	Protocol B (USER1) CAN baudrate divisor. See PP2B for a description.	01 to 40	0x04(125kbps)	R
0x2E	Protocol C (USER2) CAN options. See PP2C for a description.	00 to FF	0x80	R
0x2F	Protocol C (USER2) CAN baudrate divisor. See PP2B for a description.	01 to 40	0x0A(50kbps)	R
0x30	Protocol D (USER3) CAN options. See PP2C for a description.	00 to FF	0x42	R
0x31	Protocol D (USER3) CAN baudrate divisor. See PP2B for a description.	01 to 40	0x01(500kbps)	R
0x32	Protocol E (USER4) CAN options. See PP2C for a description.	00 to FF	0xF0	R
0x33	Protocol E (USER4) CAN baudrate divisor. See PP2B for a description.	01 to 40	0x06(95.2kbps)	R
0x34	Protocol F (USER5) CAN options. See PP2C for a description.	00 to FF	0xE0	R
0x35	Protocol F (USER5) CAN baudrate divisor. See PP2B for a description.	01 to 40	0x0F(33.3kbps)	R

6.1 Absolute Maximum Ratings

Storage Temperature ..... -65°C to +150°C

Ambient Temperature with Power Applied ..... -40°C to +85°C

Voltage on VDD with respect to VSS ..... -0.3V to +5.5V

Voltage on any other pin with respect to VSS..... -0.3V to (VDD + 0.3V)

Note:

These values are given as a design guideline only. The ability to operate to these levels is neither inferred nor recommended, and stresses beyond those listed here will likely damage the device.

6.2 Electrical Characteristics

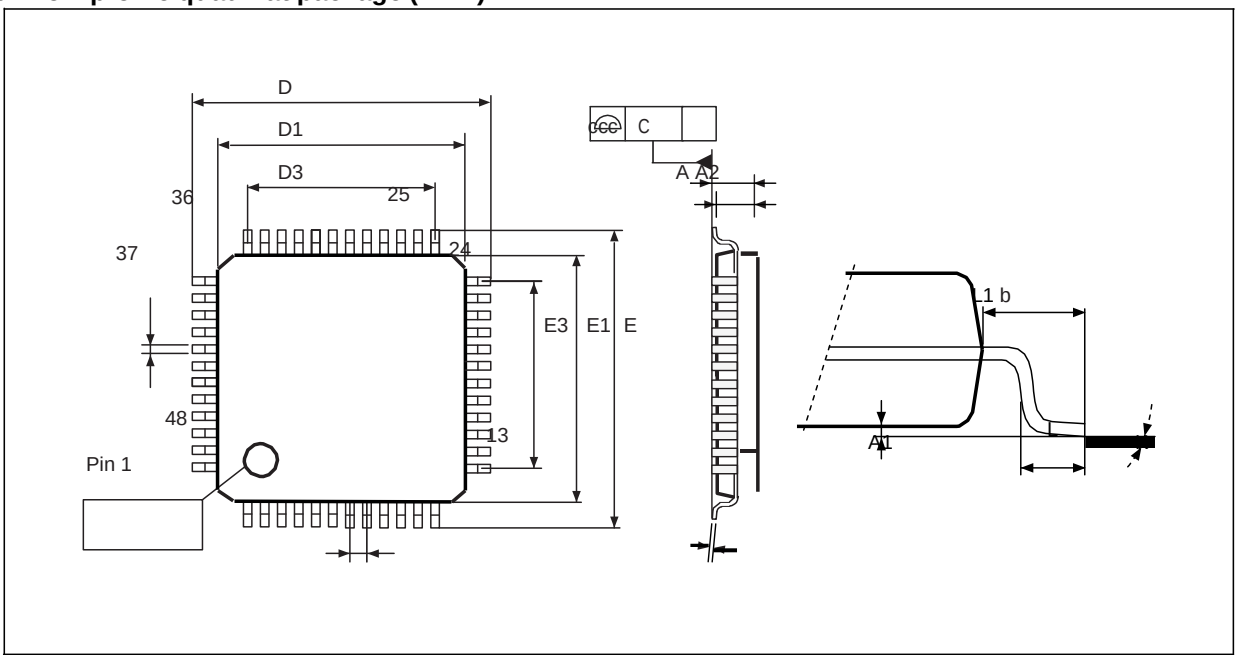
All values are for operation at 25°C and a 5V supply, unless otherwise noted.

Characteristic	Minimum	Typical	Maximum	Units	Conditions
Supply voltage, VDD	3.0	5	5.5	V	
VDD rate of rise	0.05			V/ms	see note 1
Average current, IDD normal		12		mA	MIC3X1X device only - does not include any load currents
low Power		1		mA	
Input logic levels low	VSS		0.8	V	
high	3		VDD	V	
Schmitt trigger rising input thresholds rising		2.9	5	V	
falling	1	1.5		V	
Output low voltage		0.3		V	current (sink) = 10 mA
Output high voltage		4.4		V	current (source) = 10 mA
Brown-out reset voltage	2.65	2.79	2.93	V	
A/D conversion time		9		msec	AT RV to beginning of response
Pin 43 wake pulse duration	3			msec	to wake from Low Power mode
IgnMon debounce time	50	65		msec	
AT LP to PwrCtrl output time		1		sec	
LP ALERT to PwrCtrl output time		2		sec	

1. This spec must be met in order to ensure that a correct power on reset occurs. It is quite easily achieved using most common types of supplies, but may be violated if one uses a slowly varying supply voltage, as may be obtained through direct connection to solar cells or some charge pump circuits.

6.3 Packaging Diagrams and Parameters

48-pin low profile quad flat package (7 x 7)

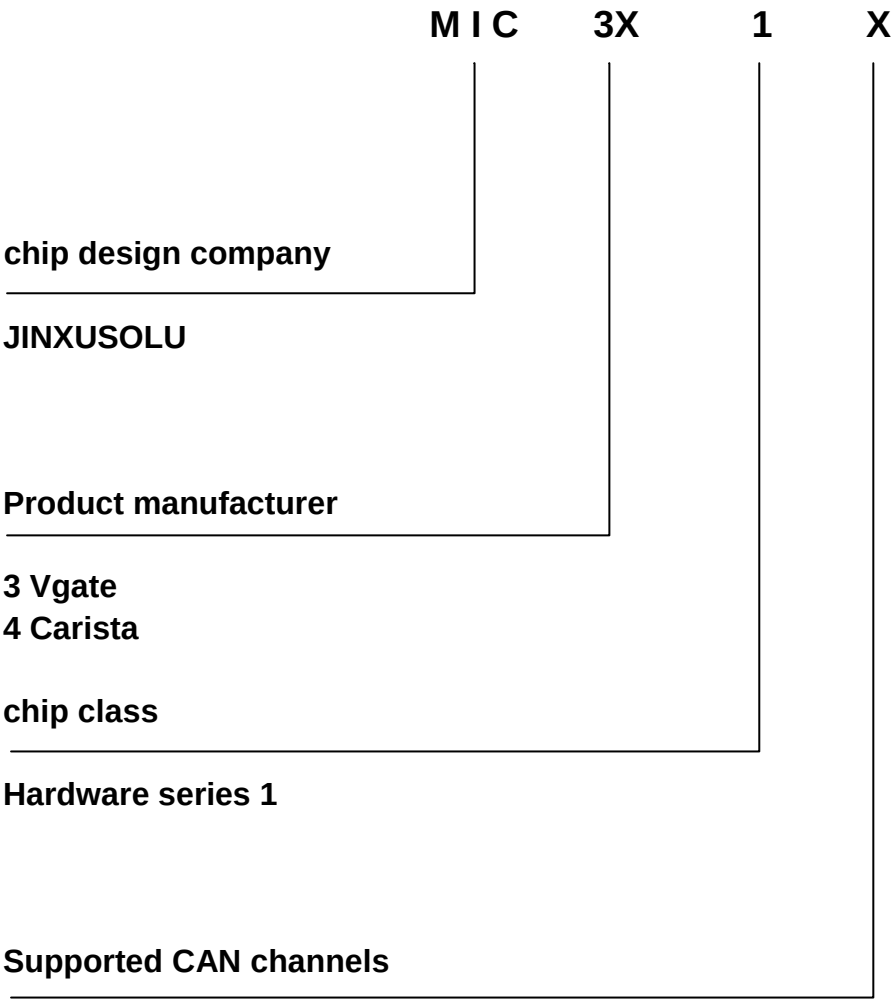


48-pin low profile quad flat package mechanical data

Symbol	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A			1.600			0.0630
A1	0.050		0.150	0.0020		0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.170	0.220	0.270	0.0067	0.0087	0.0106
c	0.090		0.200	0.0035		0.0079
D	8.800	9.000	9.200	0.3465	0.3543	0.3622
D1	6.800	7.000	7.200	0.2677	0.2756	0.2835
D3		5.500			0.2165	
E	8.800	9.000	9.200	0.3465	0.3543	0.3622
E1	6.800	7.000	7.200	0.2677	0.2756	0.2835
E3		5.500			0.2165	
e		0.500			0.0197	
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1		1.000			0.0394	
k	0.0°	3.5°	7.0°	0.0°	3.5°	7.0°
ccc			0.080			0.0031

1. Values in inches are converted from mm and rounded to four decimal places.

6.4 Ordering information



*** VTI command reply product information.**

*** VTVERS command reply chip information.**

MIC3X1X Command List

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
1	<CR>	AT	V	V	X	V	V	X	repeat the last command
2	.[1~8bytes]	AT	X	V	X	X	X	V	send bytes with the 11 bit ID
3	:[1~8bytes]	AT	X	V	X	X	X	V	send bytes with the 29 bit ID
4	AL	AT	V	V	V	V	V	V	Allow Long (>7 byte) messages
5	AMC	AT	V	X	V	X	X	V	display Activity Monitor Count
6	AMT hh	AT	V	X	V	X	X	V	set the Activity Mon Timeout to hh
7	AR	AT	V	X	V	V	V	V	Automatically Receive
8	AT0, AT1, AT2	AT	V	V	V	V	V	V	Adaptive Timing off, auto1*, auto2
9	@1	AT	V	V	V	V	V	V	display the device description
10	@2	AT	V	V	V	V	V	V	display the device identifier
11	@3 ccccccccccc	AT	V	V	V	V	V	V	store the @2 identifier
12	BD	AT	V	V	V	V	V	V	perform a Buffer Dump
13	BI	AT	V	V	V	V	V	V	Bypass the Initialization sequence
14	BRD hh	AT	V	V	V	V	V	V	try Baud Rate Divisor hh
15	BRT hh	AT	V	V	V	V	V	V	set Baud Rate Timeout
16	C0,C1	AT	X	V	X	X	X	V	Control Output off*, or on
17	CA	AT	X	V	X	X	X	V	is there CAN Activity at pin 11?
18	CAF0, CAF1	AT	V	V	V	V	V	V	Automatic Formatting off, or on
19	CEA	AT	V	V	V	X	X	V	turn off CAN Extended Addressing
20	CEA hh	AT	V	V	V	X	X	V	use CAN Extended Address hh
21	CER hh	AT	V	V	V	X	X	V	set CAN Extended Rx address to hh
22	CF hhh	AT	V	V	V	V	V	V	set the CAN ID Filter to hhh
23	CF hhhhhhhh	AT	V	V	V	V	V	V	set the CAN ID Filter to hhhhhhhh
24	CFC0, CFC1	AT	V	V	V	V	V	V	Flow Controls off, or on

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
25	CM hhh	AT	V	V	V	V	V	V	set the ID Mask to hhh
26	CM hhhhhhhh	AT	V	V	V	V	V	V	set the ID Mask to hhhhhhhh
27	CP hh	AT	V	V	V	V	V	V	set CAN Priority to hh (29 bit)
28	CRA	AT	V	V	V	V	V	V	reset the Receive Address filters
29	CRA hhh	AT	V	V	V	V	V	V	set CAN Receive Address to hhh
30	CRA hhhhhhhh	AT	V	V	V	V	V	V	set the Rx Address to hhhhhhhh
31	CS	AT	V	V	V	V	V	V	CAN Status counts and frequency
32	CSM0, CSM1	AT	V	V	V	X	X	V	Silent Monitoring off, or on
33	CTM1	AT	V	X	V	X	X	V	set Timer Multiplier to 1
34	CTM5	AT	V	X	V	X	X	V	set Timer Multiplier to 5
35	CV dddd	AT	V	V	V	V	V	V	Calibrate the Voltage to dd.dd volts
36	CV 0000	AT	V	V	V	V	V	V	restore CV value to factory setting
37	D	AT	V	V	V	V	V	V	set all to Defaults
38	D0, D1	AT	V	V	V	V	V	V	display of the DLC off*, or on
39	DM1	AT	V	V	V	V	V	V	monitor for DM1 messages
40	DP	AT	V	V	V	V	V	V	Describe the current Protocol
41	DPCLK	AT	X	X	X	X	X	V	Display current HSCAN, MSCAN, SWCAN link status
42	DPN	AT	V	V	V	V	V	V	Describe the Protocol by Number
43	E0, E1	AT	V	V	V	V	V	V	Echo off, or on
44	FB hh	AT	X	V	X	X	X	V	set the CAN Filler Byte to hh
45	FC SM h	AT	V	V	V	V	V	V	Flow Control, Set the Mode to h
46	FC SH hhh	AT	V	V	V	V	V	V	Flow Control, Set the Header to hhh
47	FC SH hhhhhhhh	AT	V	V	V	V	V	V	Flow Control, Set the Header to hhhhhhhh
48	FC SD[1~5bytes]	AT	V	V	V	V	V	V	Flow Control, Set Data to [...]

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
49	FE	AT	V	V	V	V	V	V	Forget Events
50	FI	AT	V	X	V	V	V	V	perform a Fast Initiation
51	H0, H1	AT	V	V	V	V	V	V	Headers off, or on
52	I	AT	V	V	V	V	V	V	print the version ID
53	IA	AT	X	V	X	X	X	V	Is this protocol Active?
54	IB 10	AT	V	X	V	V	V	V	set the ISO Baud rate to 10400
55	IB 12	AT	V	X	V	X	X	V	set the ISO Baud rate to 12500
56	IB 15	AT	V	X	V	X	X	V	set the ISO Baud rate to 15625
57	IB 48	AT	V	X	V	X	X	V	set the ISO Baud rate to 4800
58	IB 96	AT	V	X	V	V	V	V	set the ISO Baud rate to 9600
59	IFR0,1,2	AT	V	X	V	V	V	V	IFRs off, auto, or on, if not monitoring
60	IFR4,5,6	AT	V	X	V	X	X	V	IFRs off, auto, or on, at all times
61	IFR H,S	AT	V	X	V	V	V	V	IFR value from Header* or Source
62	IGN	AT	V	V	V	V	V	V	read the IgnMon input level
63	IIA hh	AT	V	X	V	V	V	V	set ISO (slow) Init Address to hh
64	IN1	AT	X	V	X	X	X	V	read INput 1 level
65	IN2	AT	X	V	X	X	X	V	read INput 2 level
66	JE	AT	V	V	V	V	V	V	use J1939 Elm data format
67	JHF0, JHF1	AT	V	V	V	X	X	V	Header Formatting off, or on
68	JS	AT	V	V	V	V	V	V	use J1939 SAE data format
69	JTM1	AT	V	V	V	X	X	V	set Timer Multiplier to 1
70	JTM5	AT	V	V	V	X	X	V	set Timer Multiplier to 5
71	KW	AT	V	X	V	V	V	V	display the Key Words
72	KW0，KW1	AT	V	X	V	V	V	V	Key Word checking off, or on

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
73	L0,L1	AT	V	V	V	V	V	V	Linefeeds off, or on
74	LNKHS Ph	AT	X	X	X	X	X	V	Physically link HSCAN to a specified protocol
75	LNKMS Ph	AT	X	X	X	X	X	V	Physically link MSCAN to a specified protocol
76	LNKSW Ph	AT	X	X	X	X	X	V	Physically link SWCAN to a specified protocol
77	LP	AT	V	V	V	V	V	V	go to Low Power mode
78	M0, M1	AT	V	V	V	V	V	V	Memory off, or on
79	MA	AT	V	V	V	V	V	V	Monitor All
80	MIX SM h	AT	X	X	X	X	X	V	Set the mix ID sending mode
81	MP hhhh	AT	V	V	V	V	V	V	Monitor for PGN 0hhhh
82	MP hhhh n	AT	V	V	V	X	X	V	Monitor for PGN 0hhhh and get n messages
83	MP hhhhhh	AT	V	V	V	V	V	V	Monitor for PGN hhhhhh
84	MP hhhhhh n	AT	V	V	V	X	X	V	Monitor for PGN hhhhhh and get n messages
85	MR hh	AT	V	X	V	V	V	V	Monitor for Receiver = hh
86	MT hh	AT	V	X	V	V	V	V	Monitor for Transmitter = hh
87	NL	AT	V	V	V	V	V	V	Normal Length messages
88	PB xx yy	AT	V	V	V	X	X	V	Protocol B options and baud rate
89	PC	AT	V	V	V	V	V	V	Protocol Close
90	PP xx OFF	AT	V	V	V	V	V	V	disable Prog Parameter xx
91	PP FF OFF	AT	V	V	V	V	V	V	all Prog Parameters disabled
92	PP xx ON	AT	V	V	V	V	V	V	enable Prog Parameter xx
93	PP FF ON	AT	V	V	V	V	V	V	all Prog Parameters enabled
94	PP xx SV yy	AT	V	V	V	V	V	V	for PP xx, Set the Value to yy
95	PPS	AT	V	V	V	V	V	V	print a PP Summary
96	R0, R1	AT	V	V	V	V	V	V	Responses off, or on

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
97	RA hh	AT	V	X	V	V	V	V	set the Receive Address to hh
98	RD	AT	V	V	V	V	V	V	Read the stored Data byte
99	RTR	AT	V	V	V	V	V	V	send an RTR message
100	RV	AT	V	V	V	V	V	V	Read the input Voltage
101	S0, S1	AT	V	V	V	V	V	V	printing of Spaces off, or on
102	SD hh	AT	V	V	V	V	V	V	Save Data byte hh
103	SH xyz	AT	V	V	V	V	V	V	Set Header to xyz
104	SH xxyyzz	AT	V	V	V	V	V	V	Set Header to xxyyzz
105	SH wwxxyyzz	AT	V	V	V	X	X	V	Set Header to wwxxyyzz
106	SI	AT	V	X	V	V	V	V	perform a Slow (5 baud) Initiation
107	SP h	AT	V	V	V	V	V	V	Set Protocol to h and save it
108	SP Ah	AT	V	V	V	V	V	V	Set Protocol to Auto, h and save it
109	SP 00	AT	V	V	V	V	V	V	Erase stored protocol
110	SR hh	AT	V	V	V	V	V	V	Set the Receive address to hh
111	SS	AT	V	V	V	V	V	V	use Standard Search order (J1978)
112	ST hh	AT	V	V	V	V	V	V	Set Timeout to hh x 4 msec
113	STM1	AT	X	V	X	X	X	V	Set Timer Multiplier to 1
114	STM5	AT	X	V	X	X	X	V	Set Timer Multiplier to 5
115	SW hh	AT	V	V	V	V	V	V	Set Wakeup interval to hhx20 msec
116	SW 00	AT	V	V	V	V	V	V	turn off Wakeup messages w0
117	SWCL0, SWCL1	AT	X	X	X	X	X	V	SWCAN LOAD select line output 0,1
118	TA hh	AT	V	V	V	V	V	V	set Tester Address to hh
119	TM0,1,2,3,4,5,6,7	AT	X	V	X	X	X	V	Set the transceiver mode (corresponding to M0, M1,M2)
120	TP h	AT	V	V	V	V	V	V	Try Protocol h

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
121	TP Ah	AT	V	V	V	V	V	V	Try Protocol h with Auto search
122	V0, V1	AT	V	V	V	V	V	V	use of Variable DLC off, or on
123	W0,W1	AT	X	V	X	X	X	V	Wakeup messages off, or on
124	WD [1~8bytes]	AT	X	V	X	X	X	V	set the Wakeup Data bytes
125	WH hhh	AT	X	V	X	X	X	V	set the Wakeup Header (11 bit)
126	WH hhhhhhhh	AT	X	V	X	X	X	V	set the Wakeup Header (29 bit)
127	WM [1~6 bytes]	AT	V	V	V	V	V	V	set the Wakeup Message
128	WM0,1,2	AT	X	V	X	X	X	V	set the Wakeup Mode to 0, 1 or 2
129	WS	AT	V	V	V	V	V	V	Warm Start (quick software reset)
130	WT hh	AT	X	V	X	X	X	V	set Wakeup Time to hh x 20 msec
131	Z	AT	V	V	V	V	V	V	reset all
132	BR baud	ST	X	X	X	V	V	V	Switch UART baud rate in software-friendly way
133	BRT ms	ST	X	X	X	V	V	V	Set UART baud rate switch timeout
134	S@1 ascii	ST	X	X	X	V	V	V	Set AT@1 device description string
135	SATI ascii	ST	X	X	X	V	V	V	Set ATI device ID string
136	SBR baud	ST	X	X	X	V	V	V	Switch UART baud rate in terminal-friendly way
137	WBR	ST	X	X	X	V	V	V	Write current UART baud rate to NVM
138	DI	ST	X	X	X	V	V	V	Print device hardware ID string (e.g., “OBDLink r1.7”)
139	I	ST	X	X	X	V	V	V	Print firmware ID string (e.g., “STN1100 v1.2.3”)
140	MFR	ST	X	X	X	V	V	V	Print device manufacturer ID string
141	SN	ST	X	X	X	V	V	V	Print device serial number
142	SLCS	ST	X	X	X	V	V	V	Print active PowerSave configuration summary
143	SLEEP [delay]	ST	X	X	X	V	V	V	Enter sleep mode with optional delay
144	SLLT	ST	X	X	X	V	V	V	Report last sleep/wakeup triggers

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
145	SLPCP 0/1	ST	X	X	X	V	V	V	Set PWR_CTRL output polarity
146	SLU sleep, wakeup	ST	X	X	X	V	V	V	UART sleep/wakeup triggers on/off
147	SLUIT sec	ST	X	X	X	V	V	V	Set UART inactivity timeout
148	SLUWP min, max	ST	X	X	X	V	V	V	Set UART wakeup pulse timing
149	SLVG on|off	ST	X	X	X	V	V	V	Voltage change wakeup trigger on/off
150	SLVGW [+|-]volts, ms	ST	X	X	X	V	V	V	Set configuration of the voltage change wakeup trigger
151	SLVL sleep, wakeup	ST	X	X	X	V	V	V	Voltage level sleep/wakeup triggers on/off
152	SLVLS <|> volts|0xhhh, sec	ST	X	X	X	V	V	V	Set configuration of the voltage level sleep trigger
153	SLVLW <|> volts|0xhhh, sec	ST	X	X	X	V	V	V	Set configuration of the voltage level wakeup trigger
154	SLX sleep, wakeup	ST	X	X	X	V	V	V	External sleep trigger on/off
155	SLXP 0/1	ST	X	X	X	V	V	V	Set polarity of the external sleep control input
156	SLXS	ST	X	X	X	V	V	V	Print external SLEEP input status
157	SLXST ms	ST	X	X	X	V	V	V	Set minimum active time for external sleep trigger before entering sleep
158	SLXWT ms	ST	X	X	X	V	V	V	Set minimum inactive time for external sleep trigger before wakeup
159	IAT 0/1	ST	X	X	X	V	V	V	Turn adaptive maximum interbyte timing (P1 max) off/on
160	IBR baud	ST	X	X	X	V	V	V	Set ISO baud rate
161	IMCS 0/1	ST	X	X	X	V	V	V	Turn ISO manual checksum off/on
162	IP1X ms	ST	X	X	X	V	V	V	Set maximum interbyte time for receiving messages (P1 max)
163	IP4 ms	ST	X	X	X	V	V	V	Set interbyte time for transmitting messages (P4)
164	IFI ms, ms, message	ST	X	X	X	V	V	V	Perform custom ISO fast initialization

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
165	PCB 0/1	ST	X	X	X	V	V	V	Turn automatic check byte calculation and checking off/on*(only for ISO/KWP)
166	CMM mode	ST	X	X	X	V	V	V	Set CAN monitoring mode
167	CSTM ms	ST	X	X	X	V	V	V	Set delay offset for STmin
168	CAFCP ttt, rrr	ST	X	X	X	V	V	V	Add flow control 11bit ID pair
169	CCFCP	ST	X	X	X	V	V	V	Clear all Flow Control 11bit ID Pairs
170	CFCPA ttt, rrr	ST	X	X	X	V	V	V	Add flow control 11-29 bit ID pair
171	CFCPC	ST	X	X	X	V	V	V	Clear all Flow Control 11-29 bit ID Pairs
172	FAP [pattern], [mask]	ST	X	X	X	V	V	V	Add pass filter
173	FPA [pattern], [mask]	ST	X	X	X	X	V	V	Add pass filter
174	FAB [pattern], [mask]	ST	X	X	X	V	V	V	Add block filter
175	FBA [pattern], [mask]	ST	X	X	X	X	V	V	Add block filter
176	FAFC [pattern], [mask]	ST	X	X	X	V	V	V	Add flow control filter
177	FFCA [pattern], [mask]	ST	X	X	X	X	V	V	Add CAN flow control filter
178	FCP	ST	X	X	X	V	V	V	Clear all Pass filters
179	FPC	ST	X	X	X	X	V	V	Clear all pass filters
180	FCB	ST	X	X	X	V	V	V	Clear all Block filters
181	FBC	ST	X	X	X	X	V	V	Clear all block filters
182	FCFC	ST	X	X	X	V	V	V	Clear all Flow Control filters
183	FFCC	ST	X	X	X	X	V	V	Clear all CAN flow control filters
184	FA	ST	X	X	X	V	V	V	Enable automatic filtering
185	FAC	ST	X	X	X	V	V	V	Clear all filters

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
186	STP hh	ST	X	X	X	X	V	V	Set protocol to hh
187	STPR	ST	X	X	X	X	V	V	Report protocol number
188	STPRS	ST	X	X	X	X	V	V	Report protocol string
189	STPBR baud	ST	X	X	X	V	V	V	Set OBD protocol baud rate.
190	STPBRR	ST	X	X	X	X	V	V	Report current OBD protocol baud rate
191	STCSWM mode	ST	X	X	X	X	V	V	Set SW CAN mode
192	STPRBR	ST	X	X	X	X	V	V	Report current OBD protocol baud rate
193	VRX	ST	X	X	X	V	V	V	Get the value of the voltage ADC of vbatt
194	VCAL volts	ST	X	X	X	V	V	V	Corrected voltage value, similar to ATCVhhhh
195	STCSEGT 0|1	ST	X	X	X	V	V	V	Turn CAN Tx segmentation on/off
196	STCSEGR 0|1	ST	X	X	X	V	V	V	Turn CAN Rx segmentation off*/on
197	PC	ST	X	X	X	V	V	V	Close current protocol
198	PO	ST	X	X	X	V	V	V	Open current protocol
199	PTO ms	ST	X	X	X	V	V	V	Set OBD request timeout
200	PTOT ms	ST	X	X	X	V	V	V	Set message transmission timeout
201	PTRQ ms	ST	X	X	X	V	V	V	Set minimum time between last response and next request
202	D	VT	X	X	X	X	X	V	Display solution designer
203	I	VT	X	X	V	X	X	V	Displays the adapter device name defined by [VT].
204	P1hh	VT	X	X	X	X	X	V	Switch to the protocol defined by [VT].
205	P2 hh	VT	X	X	X	X	X	V	Set one protocol [ST]
206	PROI	VT	X	X	V	X	X	V	Displays the name of the manufacturer defined by [VT].
207	PRON	VT	X	X	V	X	X	V	Displays the current protocol number, similar to the ATDPN command.

MIC3X1X Command List (continued)

index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
208	PROT	VT	X	X	V	X	X	V	Displays a description of the current protocol, similar to the ATDP command.
209	VERS	VT	X	X	V	X	X	V	Display multi-protocol conversion chip name.
210	PC	VT	X	X	X	X	X	V	Close current protocol
211	PO	VT	X	X	X	X	X	V	Open current protocol
212	PBR baud	VT	X	X	X	X	X	V	Set current OBD protocol baud rate
213	PBRD	VT	X	X	X	X	X	V	Report actual OBD protocol baud rate
214	CFG_CAN <PROTOCOL>,<OPTION>,<BAUDRATE>,<TYPE>,[TM]	VT	X	X	X	X	X	V	Configure the physical layer to be a protocol defined by the CAN standard for a certain [VT].
215	SET_CAN <PROTOCOL>,<OPTION>,<BAUDRATE>,<TYPE>,[TM]	VT	X	X	X	X	X	V	Configure the data link layer of the protocol with the specified physical layer as the CAN bus.
216	CFG_ISO <PROTOCOL>,<OPTION>,<BAUDRATE>,[IIA]	VT	X	X	X	X	X	V	Configure the physical layer to be a protocol defined by the ISO standard for a certain [VT].
217	CAN_WM <No>,<PROTOCOL>,<HEADER>,<DATA1~8>,<PERIOD>,<MODE>	VT	X	X	X	X	X	V	The physical layer defined for a [VT] sets the WM sequence for the CAN standard protocol.
218	ISO_WM <No>,<PROTOCOL>,<HEADER>,<DATA1~5>,<PERIOD>,<CTRL>	VT	X	X	X	X	X	V	The physical layer defined for a [VT] sets the WM sequence for the ISO standard protocol.

MIC3X1X Command List (continued)

Index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
219	DEL_CAN_WM <No>	VT	X	X	X	X	X	V	Delete the set WM sequence of the specified sequence number; delete its contents; release its link to a [VT] protocol.
220	DEL_ISO_WM <No>	VT	X	X	X	X	X	V	Delete the set WM sequence of the specified sequence number; delete its contents; release its link to a [VT] protocol.
221	DISP_CAN_WM <No>	VT	X	X	X	X	X	V	Display all set WM sequences related to the CAN protocol.
222	DISP_ISO_WM <No>	VT	X	X	X	X	X	V	Display all set WM sequences related to the ISO protocol.
223	SET_FM <FILTER>,<MASK>	VT	X	X	X	X	X	V	Set FILTER and MASK for each protocol.
224	SET_CAN_FC<DATA1~5>,<MODE>, [<HEADER>,<FILTER>,<MASK>]	VT	X	X	X	X	X	V	Set the FC frame content of the ISO15765-4 protocol.
225	SET_HD <HEADR>,[RECIVIER],[TIMEOUT]	VT	X	X	X	X	X	V	Set the HEADER and unique receiving address of a protocol.
226	FCST hh	VT	X	X	X	X	X	V	Set the timeout period for multi-frame message FC frames
227	SDST hh	VT	X	X	X	X	X	V	Set the raw CAN multi-frame transmission interval
228	ISOFI tl, th, data	VT	X	X	X	X	X	V	Perform custom ISO fast initialization
229	FBA [pattern], [mask]	VT	X	X	X	X	X	V	Add block filter
230	FBA	VT	X	X	X	X	X	V	Clear all block filters
231	FPA [pattern], [mask]	VT	X	X	X	X	X	V	Add pass filter
232	FPA	VT	X	X	X	X	X	V	Clear all pass filters

MIC3X1X Command List (continued)

Index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
233	FCPA [pattern], [mask]	VT	X	X	X	X	X	V	Add CAN flow control filter
234	FCPA	VT	X	X	X	X	X	V	Clear all CAN flow control filters
235	FCTRA tx, rx	VT	X	X	X	X	X	V	Add flow control address pair
236	FCTRA	VT	X	X	X	X	X	V	Clear all flow control address pairs
237	M <FILTER>,<MASK>	VT	X	X	X	X	X	V	Monitor active data on a protocol bus that meets the HEADER filter criteria.
238	MFCA	VT	X	X	X	X	X	V	Monitor OBD bus using current filters
239	SHOW_BUS [CAN_TYPE]	VT	X	X	X	X	X	V	Measure the OBD bus and estimate the possible physical layer protocols and frequencies.
240	AT_PROTOCOL_ALL	VT	X	X	X	X	X	V	Show all protocol details defined by [AT].
241	ST_PROTOCOL_ALL	VT	X	X	X	X	X	V	Show all protocol details defined by [ST].
242	VT_PROTOCOL_ALL	VT	X	X	X	X	X	V	Show all protocol details defined by [VT].
243	UCS_ATI <ascii>	VT	X	X	V	X	X	V	Set the string content displayed by ATI.
244	UCS_ATDI <ascii>	VT	X	X	V	X	X	V	Set the string content displayed by AT@1.
245	UCS_STDI <ascii>	VT	X	X	X	X	X	V	Set the string displayed by the STDI command.
246	UCS_STI <ascii>	VT	X	X	X	X	X	V	Set the string displayed by the STI command
247	UCS_MFR <ascii>	VT	X	X	X	X	X	V	Set the string displayed by the STMFR command
248	UCS_STSN <ascii>	VT	X	X	X	X	X	V	Set the string displayed by the STSN command
249	SET_UDS <ascii>	VT	X	X	X	X	X	V	Set the unique device ID number.
250	SET_MAC <hh…hhh>	VT	X	X	X	X	X	V	Set the 6-byte or 12-byte MAC address
251	RD_MAC	VT	X	X	X	X	X	V	Read the contents of the MAC address.
252	RD_UDS	VT	X	X	X	X	X	V	Read the unique device ID number.

MIC3X1X Command List (continued)

Index	command	define	ELM327	ELM329	IcarPro	Stn1100	Stn1151	MIC3X1X	Explanation
253	WT_EE <POS>,<DATA[1~8]>	VT	X	X	X	X	X	V	Write data to a user-defined EEPROM
254	RD_EE <POS>,<LEN>	VT	X	X	X	X	X	V	Read data of the user-defined EEPROM location
255	POWERMANAGE	VT	X	X	X	X	X	V	View sleep and wake source settings
256	PDVS <VOLTS>, <TIMER>	VT	X	X	X	X	X	V	Set the voltage value of low voltage for sleep
257	VDWK <VOLTS>, <TIMER>	VT	X	X	X	X	X	V	Setting the value of voltage drop
258	VLCW [+/-]<VOLTS>, <TIMER>	VT	X	X	X	X	X	V	Wake up the device by the voltage changing.
259	VLRD	VT	X	X	X	X	X	V	Read the voltage value at 4 ms intervals
260	SWGP <switch 1>, … <switch n>	VT	X	X	X	X	X	V	Set batch switch command
261	TOST <TimoutType 1:ms>, … <TimoutType n:ms>	VT	X	X	X	X	X	V	Set one group timeout
262	TP_RTS TotalLen, BlockSize[,EnChk]	VT	X	X	X	X	X	V	Request multi-packet data transmission
263	TP_DT ii hh ... hh [chksum]	VT	X	X	X	X	X	V	Single packet data transmission
264	TP_END	VT	X	X	X	X	X	V	Multi-packet data transmission completed
265	TP_ABORT	VT	X	X	X	X	X	V	Abandon multi-packet transmission

ST to VT Commands conversion table

Table 1 – General ST Commands		
Command	**Description**	VT command compatibility status
STCALSTAT	Read voltage calibration status	Not compatible
STRSTNVM	Reset NVM to factory defaults	Not compatible
STSAVCAL	Save all calibration values	Not compatible
STUIL *0|1*	Disable/Enable* LEDs	Not compatible
Table 2 – UART Specific ST Commands		
Command	**Description**	VT command compatibility status
STBR *baud*	Switch UART baud rate in software-friendly way	Not compatible
STBRT *ms*	Set UART baud rate switch timeout	Not compatible
STSBR *baud*	Switch UART baud rate in terminal-friendly way	Not compatible
STWBR	Write current UART baud rate to NVM	Not compatible
Table 3 – Device ID ST Commands		
Command	**Description**	VT command compatibility status
STDI	Print device hardware ID string (e.g., “OBDLink r1.7”)	Not compatible
STDICES	Print engine start count	Not compatible
STDICPO	Print POR (Power on Reset) count	Not compatible
STDITPO	Print POR timer	Not compatible
STI	Print firmware ID string (e.g., “STN1100 v1.2.3”)	Not compatible
STIX	Print extended firmware ID string	Not compatible
STMFR	Print device manufacturer ID string	VT PROI
STSATI *ascii*	Set ATI device ID string	VT UCS_ATI <ascii>
STSDI *ascii*	Set device hardware ID string	Not compatible
STSN	Print device serial number	Not compatible
STS@1 *ascii*	Set AT@1 device description string	VT UCS_ATDI <ascii>

Table 4 – Voltage Reading ST Commands		
Command	**Description**	VT command compatibility status
STVCAL [*volts* [*, offset*]]	Calibrate voltage measurement	This type of instruction can be implemented with ATCV, ATRV commands
STVR [*precision*]	Read voltage in volts	
STVRX	Read voltage in ADC steps	

Table 5 – OBD Protocol ST Commands		
Command	**Description**	VT command compatibility status
STP *p*	Set current protocol	VT P2 hh
STPBR *baud*	Set current OBD protocol baud rate	VT PBR baud
STPBRR	Report actual OBD protocol baud rate	VT PBRD
STPC	Close current protocol	VT PC
STPCB *0|1*	Turn automatic check byte calculation and checking off/on*	VT SWGP PCB 0/1
STPO	Open current protocol	VT PO
STPR	Report current protocol number	VT PRON
STPRS	Report current protocol string	VT PROT
STPTO *ms*	Set OBD request timeout	VT TOST REP:ms
STPTOT *ms*	Set message transmission timeout	VT TOST REQ:ms
STPTRQ *ms*	Set minimum time between last response and next request	VT TOST REPQ:ms
STPX param1 [, param2, …]	Send arbitrary message	Not compatible
Table 6 – ISO Specific ST Commands		
Command	**Description**	VT command compatibility status
STIAT *0|1*	Turn adaptive maximum interbyte timing (P_1 max) off/on*	VT SWGP IAT0/1
STIFI *ms, ms, message*	Perform custom ISO fast initialization	VT ISOFI tl, th, data
STIP1X *ms*	Set maximum interbyte time for receiving messages (P_1 max)	VT TOST IP1X:ms
STIP4 *ms*	Set interbyte time for transmitting messages (P_4)	VT TOST IP4 ms

Table 7 – CAN Specific ST Commands		
Command	**Description**	VT command compatibility status
STCAF *format [, tt]*	Set CAN addressing format	Can use VTSWGP CAF0/1 command to achieve
STCFCPA *txadd[ext]*, *rxadd[ext]*	Add flow control address pair	VT FCTRA tx, rx
STCFCPC	Clear all flow control address pairs	VT FCTRA
STCMM *mode*	Set CAN monitoring mode	Can use VTSWGP CSM0/1 command to achieve
STCSEGR *0|1*	Turn CAN Rx segmentation off*/on	VT SWGP GR 0/1
STCSEGT *0|1*	Turn CAN Tx segmentation off*/on	VT SWGP GT 0/1
STCSTM *ms*	Set delay offset for STmin	VT SDST hh
STCSWM *mode*	Set Single Wire CAN transceiver mode	AT TM h
STCTOR *fcTimeout, cfTimeout*	Set CAN FC and CF Rx timeouts	VT FCST hh
STCTR *hhhhhh*	Set CAN timing configuration registers	Not compatible
STCTRR	Read CAN timing configuration	Not compatible
Table 8 – Filtering ST Commands		
Command	**Description**	VT command compatibility status
STFA	Enable automatic filtering	VT SWGP FC1
STFAC	Clear all filters	VT SWGP FC0
STFBA [*pattern*], [*mask*]	Add block filter	VT FBA [pattern], [mask]
STFBC	Clear all block filters	VT FBA
STFFCA [*pattern*], [*mask*]	Add CAN flow control filter	VT FCPA [pattern], [mask]
STFFCC	Clear all CAN flow control filters	VT FCPA
STFPA [*pattern*], [*mask*]	Add pass filter	VT FPA [pattern], [mask]
STFPC	Clear all pass filters	VT FPA
STFPGA *pgn* [*, tgt address*]	Add SAE J1939 PGN filter	Not compatible
STFPGC	Clear all SAE J1939 PGN filters	Not compatible

Table 9 – Monitoring ST Commands		
Command	**Description**	VT command compatibility status
STM	Monitor OBD bus using current filters	VT MFCA
STMA	Monitor all messages on OBD bus	Can be achieved with ATMA
Table 10 – PowerSave ST Commands		
Command	**Description**	VT command compatibility status
STSLCS	Print active PowerSave configuration summary	For the power management part, please refer to the compatible ELM part, which is more comprehensive, simpler and more practical than STN. Refer to programmable parameters, ATLP, VTPDVS, VTVDWK, etc. can realize this kind of function.
STSLEEP [*delay*]	Enter sleep mode with optional delay	
STSLLT	Report last sleep/wakeup triggers	
STSLPCP *0*|*1*	Set PWR_CTRL output polarity	
STSLU *sleep*, *wakeup*	UART sleep/wakeup triggers on/off	
STSLUIT *sec*	Set UART inactivity timeout	
STSLUWP *min*, *max*	Set UART wakeup pulse timing	
STSLVG *on*|*off*	Voltage change wakeup trigger on/off	
STSLVGW [+|-]*volts*, *ms*	Set configuration of the voltage change wakeup trigger	
STSLVL *sleep*, *wakeup*	Voltage level sleep/wakeup triggers on/off	
STSLVLS <|> *volts*|*0xhhh*, *sec*	Set configuration of the voltage level sleep trigger	
STSLVLW <|> *volts*|*0xhhh*, *sec*	Set configuration of the voltage level wakeup trigger	
STSLX *sleep*, *wakeup*	External sleep trigger on/off	
STSLXP *0*|*1*	Set polarity of the external sleep control input	
STSLXS	Print external SLEEP input status	
STSLXST *ms*	Set minimum active time for external sleep trigger before entering sleep	
STSLXWT *ms*	Set minimum inactive time for external sleep trigger before wakeup	

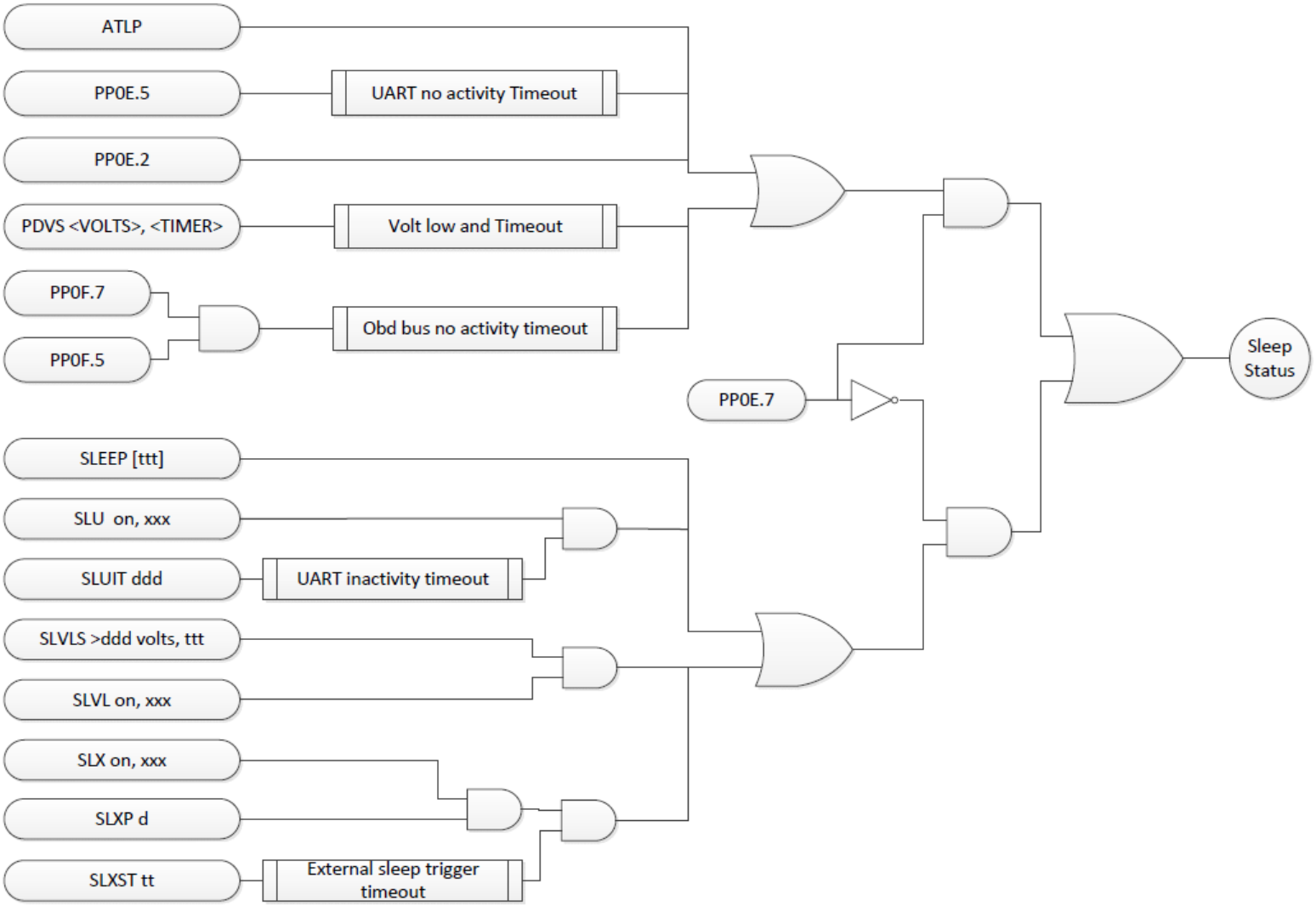
Table 11 – Bluetooth Commands		
Command	**Description**	VT command compatibility status
STBTCOD *hhhhhh*	Set Bluetooth modem CoD	Not compatible
STBTDN *ascii*	Set Bluetooth broadcasting device name	Not compatible

Table 12 – General Purpose I/O ST Commands		
Command	**Description**	VT command compatibility status
STGPC *pin1*:*options* [, …, *pinN*:*options*]	Configure I/O pins	Not compatible
STGPIR *pin1* [, …, *pinN*]	Read inputs	Not compatible
STGPIRH *pin1* [, …, *pinN*]	Read inputs, report value as hex	Not compatible
STGPOR *pin1* [, …, *pinN*]	Read output latches	Not compatible
STGPOW *pin1*:*state* [, …, *pinN*:*state*]	Write output latches	Not compatible

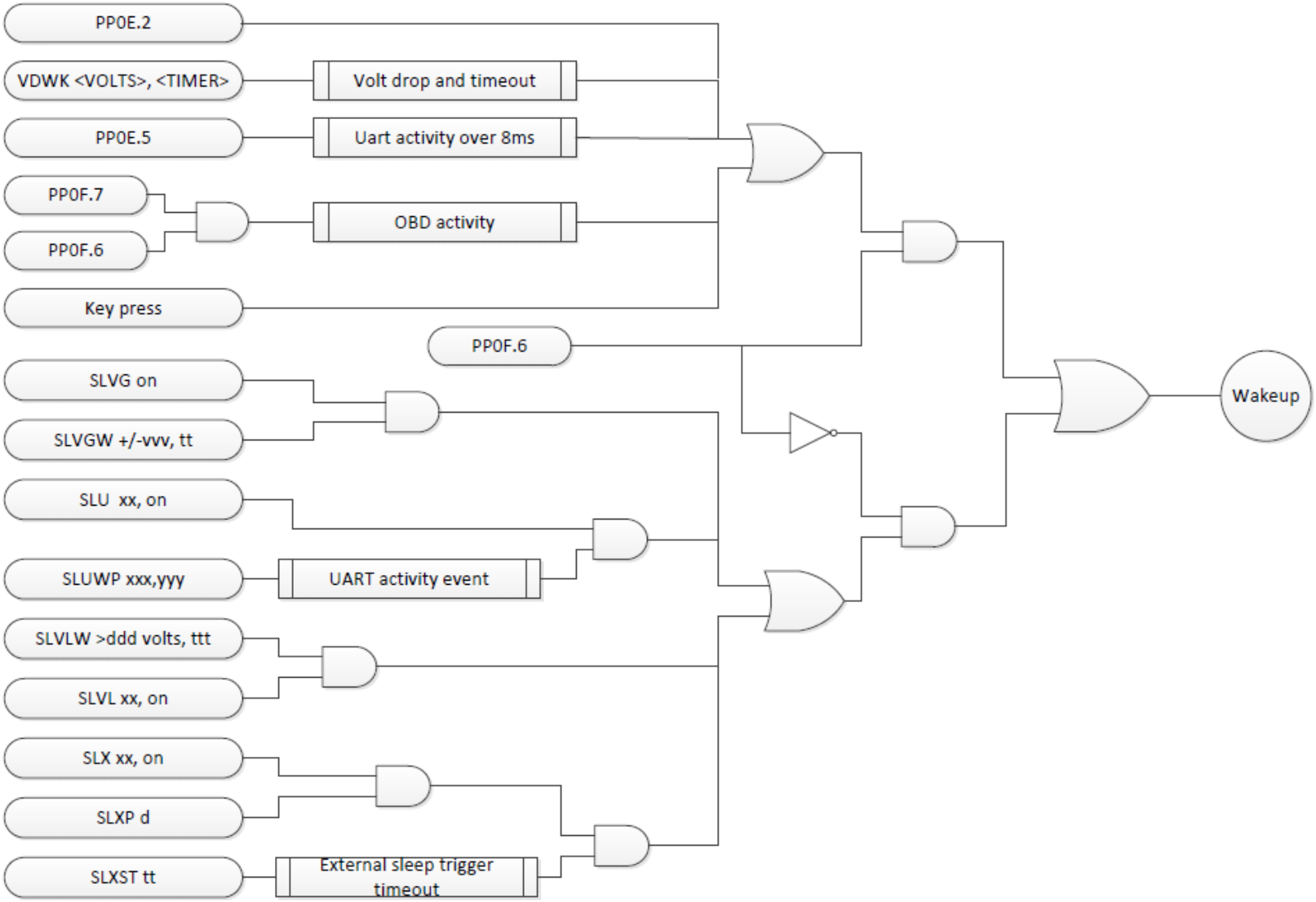
Table 13 – Periodic Messaging ST Commands		
Command	**Description**	VT command compatibility status
STPPMA *period, header, data*	Add a periodic message	Please refer to the instructions for using AT or VT to implement periodic messages
STPPMC	Clear all periodic messages	
STPPMD *handle*	Delete a periodic message	

Deprecated ST Commands		
Command	**Description**	VT command compatibility status
STCAFCP	Add CAN flow control address pair	VT FCTRA tx, rx
STCCFCP	Clear all CAN flow control address pairs	VT FCTRA
STFAB	Add block filter	VT FBA [pattern], [mask]
STFAFC	Add CAN flow control filter	VT FCPA [pattern], [mask]
STFAP	Add pass filter	VT FPA [pattern], [mask]
STFAPG	Add SAE J1939 PGN filter	Not compatible
STFCA	Clear all filters	VT SWGP FC 0/1
STFCB	Clear all block filters	VT FBA
STFCFC	Clear all CAN flow control filters	VT FCPA
STFCP	Clear all pass filters	VT FPA
STFCPG	Clear all SAE J1939 PGN filters	Not compatible
STIBR	Set ISO baud rate	VT PBR baud
STIMCS	Turn ISO manual checksum off/on	VT SWGP IMCS 0/1
STPRBR	Report actual OBD protocol baud rate	VT PBRD

Power Manage Diagram: Sleep



Power Manage Diagram: Wakeup



Revision History

MIC3X1X V2.0.84

Changes:

-- Added pass filter increased to 48 groups.

MIC3X1X V2.0.82

Commands:

Add VT command:

-- VLCW, VLRD

MIC3X1X V2.0.80

Changes:

-- SWGPGR1 The response speed of formatted characters has been optimized.

MIC3X1X V2.0.78

Changes:

-- 7Fxx78 is not limited by the number of reply lines.

MIC3X1X V2.0.76

Changes:

-- The CAN controller adds a pre-buffering mechanism.

MIC3X1X V2.0.74

Changes:

-- Modified the FAP printout function of VPW protocol and PWM protocol.

-- Removed the two-character limit between ATNL and ATAL, and set it to characters less than 128 bytes uniformly.

MIC3X1X V2.0.72

Changes:

-- ISO15765 accepts the formatter's support for extended addresses.

MIC3X1X V2.0.68

Changes:

-- Improve OBD-command interruption latency.

-- Modified the function of the VTSET_CAN_FC instruction.

-- Add chip ordering information.

Commands:

Added SWGP switch variable:

FCDA1; FCDA0

MIC3X1X V2.0

Commands:

Add VT command:

D, P2hh, PC, PO, PBR baud, PBRD command, ISOFI, FBA, FPA, FCPA, FCTRA, MFCA.

Modify the instructions:

SWGP (BZF, CRF, GR, PCB, IMCS, IAT, FC) , TOST (REP, REQ, REPQ).

Add ST command:

FA, FAC, PO, PC, FBC, FPC, FFCC, CSEGR, CSTM, CMM, PCB, PTRQ, PTO, PTOT FPA, FBA, FFCA, IFI.

MIC3X1X V1.2

Features:

- Serial receive buffer expanded to 1024 characters

Commands:

Add VT command:

SDST

Modify the instructions:

CFG_CAN, CFG_ISO, DISP_CAN_WM, DISP_ISO_WM, SET_CAN_FC, PROT, AT_PROTOCOL_ALL, UCS_ATI, UCS_ATDI, UCS_STDI, UCS_STI, UCS_MFR, UCS_STSN, SET_UDS, SDST, TP_END

MIC3X1X V1.1

Features:

- TP transmission protocol
- Serial receive buffer expanded to 512 characters

Commands:

Add ST command:

CSEGT h, CFCPC, CFCPA transmitter, receiver

Add VT command:

FCST, TP_RTS, TP_DT, TP_END, TP_ABORT

MIC3X1X V1.1

Features:

- TP transmission protocol
- Serial receive buffer expanded to 512 characters

Commands:

Add ST command:

CSEGT h, CFCPC, CFCPA transmitter, receiver

Add VT command:

FCST, TP_RTS, TP_DT, TP_END, TP_ABORT

MIC3X1X V1.0

Revision A (August 28, 2019) Initial release of this document.

Contact Information

Shenzhen Jinxusolu Technology Co., Ltd.

ADD： Room 202, Block C, Huafeng Creative World, Baoan District, Shenzhen, Guangdong, China

Tel: 0755-29955053

Fax: 0755-29610995

E-mail: sales@jinxusolu.com

techsupport@jinxusolu.com

Web: http://www.jinxusolu.com/