nodeG5 - CAN-J1939 iotasset config guide

Firmware version: fw_nodeG5_v2.1
Guide release date: 23OCT2023

Filename :	iotasset.txt
Location :	\user

1. Introduction
   The file 'iotasset.txt' contains the assets configuration that is required by the CAN-J1939 master program to acquire data from CAN-J1939 devices. Acquired data is then inserted into local database for downstream IoT cloud clients.

2. IoTasset 'field,value' general formation
   Each IoTasset is defined via a BLOCK of 'field,value' lines (CSV format).
   There are 4 CAN-J1939 field that must be present for each IoTasset.

CAN-J1939 field	**Description**
TYPE	Define the type of CAN bus communication
CANPGN	Define the Parameter Group Number (PGN)
CANSA	Define the Source Address (SA)
CANDATA	Define the J1939 raw data parsing and data type conversion

There are 2 IoTdata field that must be present for each IoTasset.

IoTdata field	**Description**
Key	Define the data tagname
IOTMODE	Define the data handling mode

Backslash (\) and double quote (") char usage is not allowed.
Hash (#) char is used for comments.

For parsing many types of assets, the asset blocks need to be located between the start and end of block markers.

CANBUS PORT	**CAN bus BLOCK MARKER**	**Description**
CAN_PORT_E	CANE_START	Block markers for CANbus assets on CAN_PORT_E.
	CANE_STOP	
CAN_PORT_C	CANC_START	Block markers for CANbus assets on CAN_PORT_C.
	CANC_STOP	

CAN_PORT_E : On-board CAN 2.0B port.
CAN_PORT_C : Expansion CAN 2.0B port
(refer to nodeG5 user manual for connector pin-out details)

3. IoTasset ‘field,value’ setup information

TYPE, m

Argument	Value	Description
m	J1939	CAN-J1939 protocol

CANPGN, n

Argument	Format	Value	Description
n	4 digit hex	eg. FEEE/F004[#1]	n = Parameter Group Number

#1 Refer to your CAN device manual for supported PGN lists.

CANSA, s

Argument	Format	Value	Description
s	2 digit hex	eg. 00/09/F7[#2]	s = Source Address

#2 Refer to your CAN device for parameter ‘CAN Node-ID’.

CANDATA, t, u, v [, x, y] datatype: INTEGER, STRING, FLOAT, CANRAW
CANDATA, B.b, c, v [, x, y] datatype: BITS

Argument	Value	Description
t, u	t=Byte start u=Byte length	Position of starting byte (dec:1-8) Length of byte (dec:1-8)
B.b, c	B.b= (Byte_start). (bit_start) c=bits length	Position of starting Byte.bit (dec:1.1-8.8) Length of bits (dec:1-8) [#3]
v	Data Type	Data type as conversion from CAN bus raw data
x	Multiplier	Value = Value*Multiplier + Adder [#4]
y	Adder	Value = Value*Multiplier + Adder [#4]

#3 Bits parsing can only be applied on single byte of CAN data and not across multiple bytes.
#4 Optional: for Data Type INTEGER & FLOAT, **both** x & y arguments required when applied.

Key, tag

Argument	Value	Description
tag	string	Unique name for this data value eg temperature, voltage, pressure, rpm

IOTMODE, c

Argument	Value	Description
c	0	Send to cloud immediately
	1	Store to local database for local IoT client processing

4. Data Type definitions for CAN-J1939

DATA TYPE BITS

v [Data Type]	c [Data Length (bits)]	Description
BITS	1-8	1-8 bits to unsigned integer[#5]

#5 Binary value parsed will be converted to decimal value, eg 1110_2 will be reported as 14_{10}. Bits parsing can only be applied on single byte of CAN data and not across multiple bytes.

DATA TYPE INTEGER

v [Data Type]	u [Data Length (bytes)]	Description
UINT8	1	8-bit data to **8-bit unsigned integer**
SINT8		8-bit data to **8-bit signed integer**
UINT12HL	2	8-bit data pair to **12-bit unsigned integer**, big endian
UINT12LH		8-bit data pair to **12-bit unsigned integer**, little endian
SINT12HL		8-bit data pair to **12-bit signed integer**, big endian
SINT12LH		8-bit data pair to **12-bit signed integer**, little endian
UINT16HL	2	8-bit data pair to **16-bit unsigned integer**, big endian
UINT16LH		8-bit data pair to **16-bit unsigned integer**, little endian
SINT16HL		8-bit data pair to **16-bit signed integer**, big endian
SINT16LH		8-bit data pair to **16-bit signed integer**, little endian
UINT32HLhl	4	8-bit data quad to **32-bit unsigned integer**, big endian
UINT32hlHL		8-bit data quad to **32-bit unsigned integer**, Word – little endian, Byte – big endian
UINT32LHlh		8-bit data quad to **32-bit unsigned integer**, Word – big endian, Byte – little endian
UINT32lhLH		8-bit data quad to **32-bit unsigned integer**, little endian
SINT32HLhl		8-bit data quad to **32-bit signed integer**, big endian
SINT32hlHL		8-bit data quad to **32-bit signed integer**, Word – little endian, Byte – big endian
SINT32LHlh		8-bit data quad to **32-bit signed integer**, Word – big endian, Byte – little endian
SINT32lhLH		8-bit data quad to **32-bit signed integer**, little endian

DATA TYPE STRING

v [Data Type]	u [Data Length (bytes)]	Description
STRING8	8	Set of eight 8-bit data to **8 ASCII characters** (abcdefgh)
STRING8R		Set of eight 8-bit data to **8 ASCII characters**, reversed (hgfedcba)
STRING4	4	Set of four 8-bit data to **4 ASCII characters** (abcd)
STRING4R		Set of four 8-bit data to **4 ASCII characters**, reversed (dcba)

DATA TYPE FLOAT

v [Data Type]	u [Data Length (bytes)]	Description
FLOAT32ABCD	4	Set of four 8-bit data to IEEE-754 single precision floating point number. Byte orientation=ABCD,DCBA,BADC,CDAB A,B,C,D=canbyte1,canbyte2,canbyte3,canbyte4
FLOAT32DCBA		
FLOAT32BADC		
FLOAT32CDAB		

DATA TYPE CANRAW

v [Data Type]	u [Data Length (bytes)]	Description
CANRAW	8	String of 16 hexadecimal char

5. Example for IOT asset configuration

```
#iotasset example for CAN-J1939 protocol

CANE_START                              #start of CAN block for CAN_PORT_E

TYPE, J1939                             #CAN type=J1939
CANPGN, FEEE                            #PGN=0xFEEE
CANSA, F7                               #SA=0xF7
CANDATA, 1, 1, UINT8, 1,-40             #byte start=1, byte length=1, value=value*1-40
Key, EngineTemp
IOTMODE,1

TYPE, J1939
CANPGN, F004                            #PGN=0xF004
CANSA, F7
CANDATA, 4, 2, UINT16HL, 0.25, 0        #value=value*0.25 + 0
Key, EngineRPM
IOTMODE,1

TYPE, J1939
CANPGN, FE6C                            #PGN=0xFE6C
CANSA, F7
CANDATA, 7, 2, UINT16HL
Key, VehicleSpeed
IOTMODE,1


TYPE, J1939
CANPGN, F003                            #PGN=0xF003
CANSA, F7
CANDATA, 6.3, 2, BITS                   #Byte start=6 ,bit start=3, length=2
Key, SpeedLimitStatus
IOTMODE,1

CANE_STOP                               #end of CAN block

CANC_START                              #start of CAN block for CAN_PORT_C


CANC_STOP                               #end of CAN block
```

6. Methods to upload ‘iotasset.txt’ file to nodeG5

-Upload the iotasset.txt file from your computer using the ‘Upload iotasset.txt’ button in the ‘IoT Hardware’ tab.

-Put the iotasset.txt file in \user folder of USB drive.
Plug the USB drive into any USB-A port and click the ‘Upload to nodeG5’ button in the ‘Management’ tab.

-Use SCP/Putty console or WinSCP.

<EOF>