EFI TECHNOLOGY PGU09 / PGT10 / PGV01 (EURO 4)

For ASW-NEXT

Short User Manual



October 2023

Important notes:

If engine starts but its speed is limited to 2000 rpm - possibly ECU is VIRGIN

When ECU is in VIRGIN state, engine starts with any transponder of correct type but engine speed is limited to 2000 rpm. Engine does not start without transponder or with transponder of incorrect type.

Please check the number of stored keys by pressing button <u>Read Identification</u>. ECU is VIRGIN if KEYS PROGR.: [*NO KEYS PROGRAMMED*]

Learn keys using key programming procedure for VIRGIN ECU or write at least 2 keys manually.

Now software of ASW-NEXT supports:

- *Key data extraction from ECU without erasing existing key values*
- *Read / Write ignition advance and fuel tables for chip tuning*
- IMMO-OFF / IMMO-ON

From V1.21 ASW-NEXT software will output troubleshooting information in case of error message Set REPROG...FAILED during write key or program maps.

This error takes place when engine is running at the moment of attempt to set REPROG mode or when supply voltage is lower than 10V or higher than 14.4V

You can suspect internal hardware fault of EFI TECHNOLOGY control unit if ignition voltage displayed in troubleshooting information significantly differ from the real voltage of power supply. Minor difference (+/-0.6V) is allowed.

1. How to connect on-the-bench?





2. How to connect in the scooter?

- ✓ Connect RED and BLACK crocodile clips of ASW-NEXT interface to "+" and "-" of scooter battery
- ✓ Connect GREEN wire of ASW-NEXT interface ("K") to pin 3 in the diagnostic connector this is a K-LINE.

Prepare yourself a piece of wire to make a bridge between green wire of ASW-NEXT and pin 3 of the diagnostic connector.



3. Software functions

Read Identification

• Read / Write keys

To read existing and to store new desired values of transponders. Up to 4 keys can be programmed during this procedure.

Reset to VIRGIN

Erases all keys. Control unit is ready to self-learn new transponder values after Reset-to-VIRGIN.

4. Key programming procedures

Let's oveview several possible situations:

All keys were lost

Connect wires from interface of ASW-NEXT to CDI unit, apply +12V power to connection and press "**Read /Set Keys**" button. Diagnostic connection to CDI unit will be established and transponder values will be read:

	2	4	ОК
	*With header /BE FA/	*With header /7D 5F/	Set / Remove key
MASTER	BE FA 00 00 00 04 3C 41	7D 5F 00 00 00 20 3C 82	BE FA 00 00 00 04 3C 55
SLAVE_1	BE FA 00 00 00 04 3C 55	7D 5F 00 00 00 20 3C AA	
SLAVE_2	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	
SLAVE_3	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	🐾 Set key 🛛 🎏 Remove key
SLAVE_4			
SLAVE_5			
SLAVE_6			Save to text file
SLAVE_7			
Notes	v stored * = Hea	der of TEMIC-11 depends	Apply changes

Write value of any active (marked with green) key to transponder **T5** as **TEMIC-11** using any suitable transponder programmer. Choose data with header "**BE FA**" or "**7D 5F**" depending on type of your transponder programmer.

> You want to set / delete key(s) manually in CDI unit

Press "**Read /Set Keys**" button. Diagnostic connection to CDI unit will be established and transponder values will be read. Select position to set key to. Read key you want to store to CDI unit with any transponder programmer.



Software will automatically re-allign bits and now it looks like this:

Number of st	ored keys	Max keys to store	Key storage CRC status
	1	4	ОК
	*With header /BE FA/	*With header /7D 5F/	Set / Remove key
MASTER	BE FA 00 00 00 04 3C 41	7D 5F 00 00 00 20 3C 82	BE FA 00 00 00 04 3C 41
SLAVE_1	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	
SLAVE_2	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	
SLAVE_3	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	🐾 Set key 😤 Remove key
SLAVE_4			
SLAVE_5			
SLAVE_6			Save to text file
SLAVE_7			
Notes			
🧭 = Ker	y stored * = Hea on tr y position empty that	ansponder programmer you use.	🖌 Apply changes
	For e	example, Zed-Bull decodes	

Important! CDI unit will treat itself as not completely programmed with only one key stored. To avoid this you must to store one more key, even if single key is needed. Just enter any valid value for a second key position, different from value of **MASTER** key.

nsponder k	Keys TEMIC-11 / T5-11		
Number of st	tored keys	Max keys to store	Key storage CRC status
	2	4	ОК
	*With header /BE FA/	*With header /7D 5F/	Set / Remove key
MASTER	BE FA 00 00 00 04 3C 41	7D 5F 00 00 00 20 3C 82	BE FA 00 00 00 04 3C 55
SLAVE_1	BE FA 00 00 00 04 3C 55	7D 5F 00 00 00 20 3C AA	
SLAVE_2	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	
SLAVE_3	BE FA BE FA BE FA BE FA	7D 5F 7D 5F 7D 5F 7D 5F	🐾 Set key 🛛 🎽 Remove key
SLAVE_4 SLAVE_5 SLAVE_6 SLAVE_7			Save to text file
Notes = Ke = Ke = Ke	y stored * = Hea on t y position empty that For y can't be stored here tran	der of TEMIC-11 depends ransponder programmer you use. example, Zed-Bull decodes sponders with header "7D SF"	Apply changes Close without changes

- Use "Save to text file" to access transponder data later if no physical transponder is prepared yet. Keep in mind that you can't read key values later from control unit
- Press "Apply changes" button to write key programming values to control unit
- > You want to reset PGU09 / PGT10 / PGV01 to VIRGIN state
- Fill in all key positions with **BE FA BE FA BE FA BE FA**
- Press "Apply changes" to program blank key values to memory

or

Press button "Reset to VIRGIN"

> You want to learn keys to the VIRGIN PGU09 / PGT10 / PGV01

• Make yourself two valid **T5-11** transponders or take two ready suitable keys of **TEMIC-11** / **TYPE-11**.

To make **T5-11** transponders please program to one of them, for example: **BE FA 00 00 00 11 22 33 (7D 5F 00 00 00 88 44 CC)*** and **BE FA 00 00 00 11 22 34 (7D 5F 00 00 00 88 44 2C)*** to another. Both transponders must be programmed with different values.

Both transponders must be programmed with different values.

- Turn ignition with first key for 1.5 sec. Key will be stored as a **MASTER**
- Turn ignition with second key for 1.5 sec. Key will be stored as a SERVICE
- Turn ignition with first key again for 1.5 sec. to close programming procedure.

IMPORTANT NOTE: Do not exceed time of 1.5 seconds when changing key in ignition lock between every step. Turn ignition off while LED is still on!

*- Depends on type of transponder programmer (does it use *BE FA* or *7D 5F* header)

> To check how many keys programmed to the PGU09 / PGT10 / PGV01

• Press "**Read Identification**" button to read HW / SW info along with number of keys programmed to the control unit

Read / Write MAPS

Software of ASW-NEXT is now able to read and reprogram ignition advance and fuel tables. You may find this useful to improve engine performance or to remove speed limiter.

NOTE: Software of ASW-NEXT by itself does not modify map files to remove any limitations. Please use appropriate software to edit map files, e.g. WinOLS.

NOTE: Software of ASW-NEXT will recalculate CRCs in map file, modified by third-party software before to program it to engine control unit.

MAP file

MAP is saved to binary file. File size for PGU09 / PGT10 / PGV01 MAP file is 96kb. Ignition advance and fuel tables in binary MAP file are located at their physical adresses in the memory space of STM8.

Upper 8 lines (128 bytes) can be modified by the user without any limitations. This area is reserved to store various text notes about MAP file.

NOTE: Do not edit data at 0080-008F. MAP file will be treated as corrupted once data at line 0080 was modified.

-Hex Editor -																	
No File																	
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	OF	0123456789ABCDEF
0x000000	50	47	4E	30	34	20	52	45	56	2E	47	00	00	00	00	00	PGN04 REV.G ^
0x000010	43	4D	32	35	35	33	31	31	00	00	00	00	00	00	00	00	CM255311
0x000020	43	42	4C	30	31	30	30	39	00	00	00	00	00	00	00	00	CBL01009
0x000030	00	00	00	00			_	<u>voo</u>	Br)R) op (19	39	2 4	NIC		ES
0x000040	00	00	00	00	60	79.67	-20	-90	00	60	-06		-90	<u></u>	96	66	
0x000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0x000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0x000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0 x 000080	03	6D	01、	B9	00	1C	1E	D7	6A	5D	F9	46	F1	63	59	9B	.mj].F.cY.
0x000090	00	00	00	80	00	00	00	00	00	00	00	00	00	00	00	00	• • • • • • • • • • • • • • • • • • • •
0X0000X0	00	00	00	00	00	20	00	00	00	00	00	00	00	00	00	00	
0x0000B0	00	00	00	00	<u>òq</u>	00	60	00	00	00	00	00	00	00	00	00	
0x0000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0x0000D0	00	00	00	00	00		70	NP	79	206	- M	IFT	00	00	00	00	
0x0000E0	00	00	00	00	00			16.6		0		чbu	00	00	00	00	
0x0000F0	00	00	00	00	00	00	-99	nhe	200	90	A	00	00	00	00	00	
0x000100	00	00	00	00	00	00		JUC		10		00	00	00	00	00	
0x000110	00	00	00	do	00	00	00	00	00	00	00	00	00	90	00	00	
0x000120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	· · · · · · · · · · · · · · · · · · ·

• IMMO-OFF / IMMO-ON

This function configures immobilizer feature presence. Works by K-Line.

Current immobilizer programing state:	WITH IMMOBILIZER
CRC check result:	CRC OK
Set immobilizer programming state:	WITHOUT IMMOBILIZER 🛷