This tutorial will explain several questions about the 9S12_912 series.

1.

Q:Why does the car diagnose a fault when some modules have gone through an ICP operation ?

A:Some of the modules on cars are very smart,especially newer cars.When the module is powered up, the program in the module will check whether the voltage and other working conditions at each point are normal.If it is not in the normal state, an error code is written to the EEPROM.However, ICP operation is to remove the module from the car and only power the microcontroller. The level of other ports must be abnormal, so this problem will occur.
So using ICP mode is risky, especially for operating smart module.

2.

Q:Why is it that some modules can be read and some modules cannot be read but the microcontroller is the same ?

A:The operating mode out of reset is determined by the states of the MODC, MODB, MODA pins during reset. The MODC, MODB, MODA bits in the MODE register show the current operating mode.

Mode Selection:

1). Normal modes: Some registers and bits are protected against accident changes.
2).Special modes: Allow greater to protected control registers and bits for special purposes such as testing.

Table MODE SELECTION

MODC	MODB	MODA	Mode Description
0	0	0	Special Single Chip, BDM allowed and ACTIVE BDM is allowed in all other modes but a serial command is required to make BDM active
0	0	1	Emulation Expanded Narrow, BDM allowed
0	1	0	Special Test (Expanded Wide), BDM allowed
0	1	1	Emulation Expanded Wide, BDM allowed
1	0	0	Normal Single Chip, BDM allowed
1	0	1	Normal Expanded Narrow, BDM allowed
1	1	0	Peripheral, BDM allowed but bus operations would cause bus conflicts (must not be used)
1	1	1	Normal Expanded Wide, BDM allowed

Therefore, please measure the MODE pin before operation to ensure that the chip is in normal MODE !

3.

Q:Why is the module function still abnormal after writing EEPROM and FLASH data when the master is HC912 series module ?

A:The EEPROM module contains an extra word called SHADOW word which is loaded at reset into the EEMCR, EEDIVH and EEDIVL registers.

EEMCR—EEPROM Module Configuration $00F0

bit 7	bit 6	bit 5	bit 4	bit3	bit 2	bit 1	bit 0
NOBDML	NOSHW	RES	RES	1	EESWAI	PROTCK	DMI

NOBDML—Background Debug Mode Lockout Disable
0 = The BDM lockout is enabled.
1 = The BDM lockout is disabled.

Loaded from SHADOW word at reset. Read anytime.
Write anytime in special modes (SMODN=0).

To unlock BDM with storage rest data contents of register please do next:

1) Read register data at address $000F0

2) Set MOST significant bit to logic "1" !!!

For example: If data at address $000F0 contain $79 make logical OR with $80
$79 OR $80 = $F9;

3) Next write this value to EEPROM

4) BDM will be enable after next reset ONLY !!!

NOTE: MC68HC912 without post fix has EEPROM shadow byte only.But it located at the same address $000F0.

4.

Q:Why does the chip encrypt after the FLASH data is written and reset ?

A:This FSEC register holds all bits associated with the device security. This register is unbanked.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
KEYEN	NV6	NV5	NV4	NV3	NV2	SEC1	SEC0

KEYEN—Enable backdoor key to security.
1 = backdoor to Flash is enabled.
0 = backdoor to Flash is disabled.
NV[6:2]-Non Volatile Flag Bits.
These 5 bits are available to the user as non-volatile flags.
SEC[1:0]-Memory Security Bits.

The SEC[1:0] bits define the security state of the device as shown in Security states table:

Security states table

SEC [1:0]	Description
00	Secured
01	Secured
10	Unsecured
11	Secured

The Flash security state is defined by the SEC bits of the FSEC register.During reset, the Flash module initializes the FSEC register using data read from the security byte of the Flash configuration field at global address 0x7F_FF0F.

Flash and security byte address

Flash size	Security byte address
32K	0x7F0F
64K	0xFF0F
128K	0x1FF0F
256K	0x3FF0F
384K	0x5FF0F
512K	0x7FF0F
768K	0xBFF0F
1024K	0xFFF0F

If the byte you want to write FLASH data in the above table for the address is XXXXXX00/XXXXXX01/XXXXXX11,then after you have written this data and reset ,the microcontroller will encrypt.