

CUSTOMER NOTIFICATION

SUD-DT-03-0447-E (1/4)
October 8, 2003
Yoshiro Harada, Senior System Integrator Microcomputer Group 2nd Solutions Division Solutions Operations Unit NEC Electronics Corporation

CP(K), O

**IE-789860-NS-EM1
(Control Code: A, B, C, D)**

Operating Precautions

Be sure to read this document before using the product.

1. Product Version 2
2. Product History 2
3. Details of Bugs and Added Specifications 3
4. Cautions 4

Notes on Using IE-789860-NS-EM1

1. Product Version

Product name: IE-789860-NS-EM1

Control Code ^{Note}	Remark
A	I/O EVA chip μ PD78E9860 1.0
B	I/O EVA chip μ PD78E9860 1.1
C	I/O EVA chip μ PD78E9860 1.3
D	I/O EVA chip μ PD78E9860A/61A 1.0

Note The “control code” is the second digit from the left in the 10-digit serial number starting with E (if it has not been upgraded). If the product has been upgraded, a label indicating the new version is attached to the product and the x in V-UP LEVEL x on this label indicates the control code.

2. Product History

No.	Bugs and Changes/Additions to Specifications	Control Code ^{Note}			
		A	B	C	D
1	Bug when a non-maskable interrupt via the key return signal is used	×	√	√	√
2	Bug in key return signal operation	×	×	√	√
3	Addition of support for the μ PD789052, 789062 Subseries	–	–	–	√
4	Modification of specification of 8-bit timers 30 and 40	–	–	–	√

×: Applicable, √: Not applicable (change of specification), –: Not relevant

3. Details of Bugs and Additions to Specifications

No.1 Bug when a non-maskable interrupt via the key return signal is used

[Description]

Because the interrupt request signal of the emulation chip is not cleared when a non-maskable interrupt via the key return signal is used, other interrupts cannot be acknowledged.

[Workaround]

Refer to the attachment.

This bug has been corrected in IE-789860-NS-EM1 control code B.

No.2 Bug in key return signal operation

[Description]

The key return interrupt (INTKR1) should be generated by the input of the falling edge of P40/KR10 to P43/KR13, but is inadvertently generated by a low-level input.

[Workaround]

There is no workaround.

This bug has been corrected in IE-789860-NS-EM1 control code C.

No.3 Addition of support for the μ PD789052, 789062 Subseries

[Description]

The μ PD789052, 789062 Subseries is supported in IE-789860-NS-EM1 control code D or later.

[Caution]

Use the device file DF789062 (E1.00e (July 5, 2002) or later).

No.4 Modification of specification of 8-bit timers 30 and 40

[Description]

The carrier generator output control register 40 (TCA40) has been changed from write-only (W) to read/write (R/W) in IE-789860-NS-EM1 control code D or later due to the specification change in the target device.

[Caution]

Use the device file DF789861 (E1.10c (May 31, 2002) or later) or DF789062 (E1.00e (July 5, 2002) or later).

4. Cautions

- (1) An RC oscillator cannot be emulated by the IE-789860-NS-EM1. Only the oscillator functions described in the user's manual can be emulated.
- (2) When emulation of the low-voltage detector and power-on-clear circuit detection voltage is performed, it is affected by voltage fluctuation and noise. Therefore, the detected voltage must be checked in the EEPROM product.
- (3) When a program that illegally accesses EEPROM is executed in the IE-789860-NS-EM1, an error message is displayed and a break occurs. The conditions for illegally accessing the EEPROM and the displayed error message are described below.

Table 4-1. Illegal Access Condition

Error message: Unspecified Illegal	
EEPROM illegal access conditions	
<1>	Write instruction to EEPROM is executed when EWC = 0.
<2>	Write instruction to EEPROM is executed while the clock selected by EEPROM is stopped.
<3>	Write instruction to EEPROM is executed while EEPROM is being written to.
<4>	Read instruction from EEPROM is executed while EEPROM is being written to.
<5>	Instruction is fetched from EEPROM while EEPROM is being written to.
<6>	EWC is set to 0 while EEPROM is being written to.
<7>	Main clock is stopped while EEPROM is being written to.
<8>	Count clock selection of the write time setting timer is changed while EEPROM is being written to.
<9>	RESET is applied while EEPROM is being written to.

- (4) The IE-789860-NS-EM1 includes a POC switching circuit, which is used to control the power-on-clear circuit function via software. Set this circuit to on/off using an SFR.
- (5) The oscillation stabilization wait time after STOP mode release by $\overline{\text{RESET}}$ input or reset release using POC is fixed to $2^7/f_x$ in the IE-789860-NS-EM1. Only the mask option functions described in the user's manual can be emulated.

Workaround for No.1 in Bugs and Changes/Additions to Specifications:

The interrupt request signal of the emulation chip is cleared when the EI instruction has been executed. Therefore, be sure to execute the EI instruction in the vector table if the non-maskable interrupt of the key return signal is used. At this time, the interrupt servicing of the key return signal is executed two times, and therefore, processing that returns as soon as servicing the interrupt is started the second time is necessary. Here is an example of software.

Example:

(Main routine)

```

•
•
MOV      B,#0      ; Clears interrupt counter
STOP
•
•

```

(Key return interrupt vector routine)

```

VINTKR:
  INC      B      ; Increments interrupt counter
  EI      ; Clears interrupt request flag for emulation
  MOV      A,B
  CMP      A,#02H ; First interrupt?
  BZ      $KR_END0 ; Return processing if interrupt is second interrupt
  •
  •
  •
  •
  •
  } Original processing
  BR      $KR_END1
KR_END0:
  MOV      B,#0
KR_END1
  RETI

```

<Flow of operation in this example>

- STOP instruction execution
- Occurrence of non-maskable interrupt because of key return
- Branch to vector of key return
- Incrementing counter (B ← 1)
- EI instruction execution (kept pending while non-maskable interrupt is serviced)
- Execution of original processing because interrupt is first interrupt (B = 1)
- Return to main routine (pending EI instruction is executed at this point and execution branches to vector of key return again)
- Increments counter (B ← 2)
- Clears counter and branches to return processing because interrupt is second interrupt (B = 2)