



VT6120/VT6121/VT6122

EEPROM-less Porting Guide

Revision 1.10
Aug 20, 2003

VIA Networking Technologies, INC.

Copyright Notice:

Copyright © 2003, VIA Networking Technologies, Incorporated. All Rights Reserved.

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written permission of VIA Networking Technologies, Incorporated.

VT6120, VT6121, VT6122 may only be used to identify a product of VIA Networking Technologies.



is a registered trademark of VIA Technologies, Incorporated.

All trademarks are the properties of their respective owners.

Disclaimer Notice:

No license is granted, implied or otherwise, under any patent or patent rights of VIA Networking Technologies Inc. VIA Networking Technologies Inc. makes no warranties, implied or otherwise, in regard to this document and to the products described in this document. The information provided by this document is believed to be accurate and reliable as of the publication date of this document. However, VIA Networking Technologies Inc. assumes no responsibility for any errors in this document. Furthermore, VIA Networking Technologies Inc. assumes no responsibility for the use or misuse of the information in this document and for any patent infringements that may arise from the use of this document. The information and product specifications within this document are subject to change at any time, without notice and without obligation to notify any person of such change.

Offices:

USA Office:

940 Mission Court
Fremont, CA 94539
USA
Tel: (510) 683-3300
Fax: (510) 683-3301 -or- (510) 687-4654
Web: <http://www.viatech.com>

Taipei Office:

8th Floor, No. 533
Chung-Cheng Road, Hsin-Tien
Taipei, Taiwan ROC
Tel: (886-2) 2218-5452
Fax: (886-2) 2218-5453
Web: <http://www.via.com.tw>

Revision History

Document Release	Date	Revision	Initials
0.8	2003/08/05	First draft for internal review.	Ryan
0.9	2003/08/11	Reformat & add EEPROM checksum calculation.	Ryan
1.0	2003/08/20	Modify by Pear's opinion & add introduction section.	Ryan
1.10	2003/08/20	Modify introduction section & doc format.	Ryan

VIA Networking
Technologies Inc.
Confidential
NDA Required

TABLE OF CONTENTS

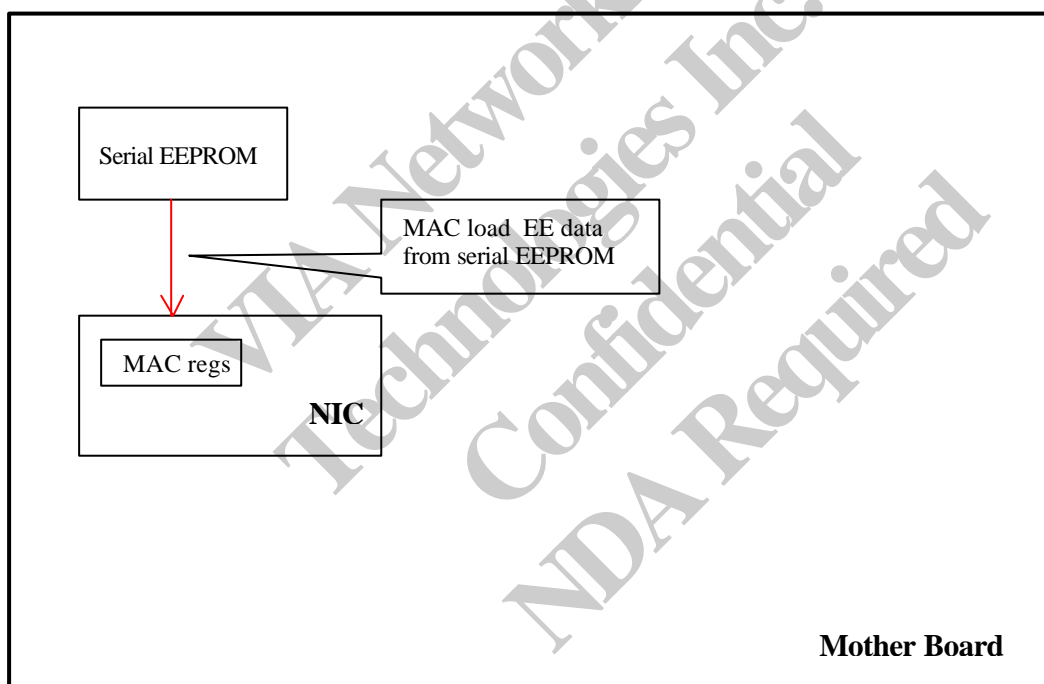
REVISION HISTORY.....	2
TABLE OF CONTENTS.....	3
1. INTRODUCTION	4
2. SYSTEM BEHAVIOR.....	6
3. REGISTER DESCRIPTION	7
4. BIOS PORTING GUIDE.....	8
4.1 WHEN AC POWER UP :.....	8
4.2 WHEN PCI DEVICE ENTER D0 VIA PCIRST# :	8
4.3 BIOS HAND-SHAKING WITH MAC SAMPLE CODES :	8
4.4 EEPROM CHECKSUM CALCULATION & SAMPLE CODES :	9
5. EEPROM SHADOW ACCESS SAMPLE CODES.....	11
6. BIOS DMI STRUCTURE EEPROM DEFAULT DATA	13

1. Introduction

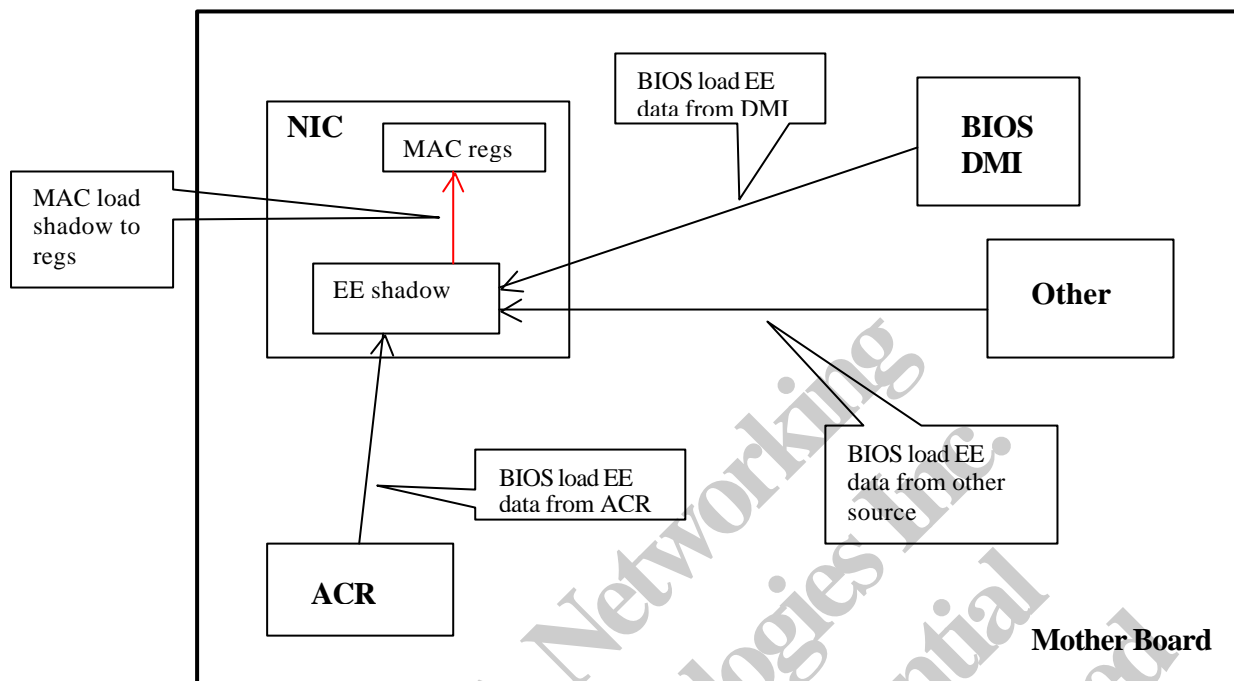
The VT6120/VT6121/VT6122 EEPROM-less function is used mainly in LAN-On-Motherboard(LOM) solutions. This function can save a serial EEPROM and uses other storage for EEPROM data such as ACR card or BIOS DMI structure. To control the data path of MAC EEPROM loading, manufacturer can do it by using jumper strapping. Please refer to VT6120/VT6121/VT6122 hardware spec for detail explanation.

The system configuration is as the following figures:

Normal way:



EEPROM-less way:



2. System Behavior

After AC power up, the original eeprom contents with important ID information is put in ACR configuration ROM or BIOS DMI structure. System BIOS reads eeprom contents either from ACR card or DMI structure and writes the information via backdoor port in PCI config space (0x58h) in order as the original eeprom layout into the 16 words shadow in the MAC controller suspend well, then set the "programmed" flag SEEPR. By the time SEEPR status is set, HC will reload eeprom automatically to fill internal configuration register then set SEELD on.

- While PCI power up in D0/D1/D2/D3, MAC controller will load the related registers from the shadow eeprom just like it loads from the external one.
- This makes mother board manufacturing the same, and leave the eeprom contents to the ACR card or BIOS DMI structure.
- In normal on-board LAN solution, we can use original external eeprom as usual, or use a external I2C eeprom for south bridge to load the configuration then write into MAC shadow, which is system BIOS dependant.

3. Register Description

31							19		16	15	0						
0x58	Reserved						WADDR		Write word data								
0x5C	RSVD	27	x	25	24	RSVD	RADDR		Read word data								

Quadlet access only, read/write address from 0 to 15

PCI register 0x58 is a write port to let BIOS read the desired Ethernet MAC information either from ACR configuration ROM or BIOS DMI structure then write them into the MAC eeprom shadow via the 0x58 port.

PCI register 0x5C is for fast diagnosis only, write is only accepted in RADDR, ERDBG(bit 27) and SEEPR(bit 24).

PCI Reg 0x58:

- WADDR[3:0] : eeprom shadow write address, read as "0", word address.
- WrData[15:0] : write data input to eeprom shadow, read as "0".

PCI Reg 0x5C:

- RADDR[3:0] : read /write as the read address of the eeprom shadow, word address.
- RdData[15:0] : read data out indexed by RADDR.
- ERDBG : Bit[27], enable read process from 0x5C, R/W.
- SEEPR : Bit[24], turn on this bit means eeprom shadow has been updated by BIOS, R/W, sticky bit, reset by SUSPRSTZ.
- SEELD : Bit[25], when system BIOS has shadowed the eeprom, it will turn on a status flag SEEPR, then HC will reload eeprom from the shadow eeprom to destination registers. When the reload process is complete, SEELD will be set. That is, if SEEPR = 0, HC will not do power-up load while PCIRST# is inserted. Sticky bit, HC updated only, cleared while SEEPR is reset.

Note : while SEEPR cleared by software, SEELD will be cleared also, but HC won't intend to load EEPROM again. That is, even BIOS updates the shadowed eeprom and set SEEPR, the other control bit, SEELD, won't be set again except an EEPROM reload command issued or PCIRST# event occurred to enable HC load eeprom sequence.

4. Bios Porting Guide

4.1 When AC power up :

1. BIOS checks ACR configuration ROM or DMI structure and begin to read the segment of the Ethernet MAC's eeprom contents.
2. Issue config word access on 0x58(always quadlet access) to write words to EEPROM shadow.
3. After writing contents(16 words) into eeprom shadow in MAC's suspend well, set SEEPR flag to indicate the shadow process is completed, then begin to polling the SEELD status (until SEELD = 1).

4.2 When PCI device enter D0 via PCIRST# :

1. Check SEEPR status, if (SEEPR=0), go to AC power-up shadow routine.
2. (SEEPR,SEELD) :
 - (1,1) : normal case;
 - (1,0) : abnormal PCI shut down while BIOS issues SEEPR flag to ask HC to reload eeprom.
 - (0,1) : impossible.
 - (0,0) : the eeprom is not shadowed, PCIRST# inserted power-up load will not occur.

4.3 BIOS hand-shaking with MAC sample codes:

```

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4
in   eax, dx
or   eax, 08000000h
out  dx, eax                                ;Enable read from 0x5C

```

Read_RX5C_Again:

```

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4

```

```

in    eax, dx                                ;Read from 0x5C
shr   eax, 24
and   al, 03h

cmp   al, 01h
je    Read_RX5C_Again

cmp   al, 03h
je    NO_MAC_Shadow_Write_Need

```

```
;Write eeprom shadow
```

```

.....
.....
.....

```

```

mov   eax, 80000000h+(LAN_SLOTID shl 8)+5ch ;R02
mov   dx, 0cf8h
out   dx, eax
add   dx, 4
in    eax, dx
or    eax, 01000000h                        ;BIOS Set SEEPR to inform MAC
out   dx, eax

```

```
Wait_Shadow_status:
```

```

in    eax, dx
test  eax, 02000000h
jz    Wait_Shadow_status                    ;Wait MAC set SEELD

```

```
NO_MAC_Shadow_Write_Need:
```

```

mov   eax, 80000000h+(LAN_SLOTID shl 8)+5ch ;R02
mov   dx, 0cf8h
out   dx, eax
add   dx, 4
in    eax, dx
and   eax, not 08000000h
out   dx, eax                            ;Disable read from 0x5C

```

4.4 EEPROM checksum calculation & sample codes:

If BIOS needs to modify EEPROM content such as MAC address, BIOS should also update EEPROM checksum data. The EEPROM checksum data is located in EEPROM 0x0F word high byte(byte address 31). The checksum calculation sample codes are as follows(in C language):

```

//-----
BYTE MISbyGenCheckSumB (PBYTE pbyBuffer, UINT uByteCount, UINT uSkipIndex)
{
    BYTE  byCkSum = 0;
    UINT  uu;

```

```
for (uu = 0; uu < uByteCount; uu++, pbyBuffer++) {  
    if (uu != uSkipIndex)  
        byCkSum += *pbyBuffer;  
}  
  
return (BYTE)(~byCkSum + 1);  
}  
  
//-----  
BYTE RomRegs[32];  
  
RomRegs[31] = MISbyGenChecksumB(RomRegs, 32, 31);
```

VIA Networking
Technologies Inc.
Confidential
NDA Required

5. EEPROM Shadow Access Sample Codes

; sample assembly routine for read eeprom shadow

```

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4
in   eax, dx
or   eax, 08000000h
out  dx, eax                                ;Enable read from 0x5C

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4
in   eax, dx
and  eax, FFF0FFFFh
mov  ecx, RADDR                                ;Set ecx with shadow read address
shl  ecx, 16
or   eax, ecx
out  dx, eax                                ;Set RADDR
shr  ecx, 16

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4
in   eax, dx                                ;Read eeprom shadow
and  eax, 0000FFFFh                        ;RdData[0..15](ax contains read data)

mov  eax, 80000000h+(LAN_SLOTID shl 8)+5ch    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4
in   eax, dx
and  eax, not 08000000h
out  dx, eax                                ;Disable read from 0x5C

```

; sample assembly routine for write eeprom shadow

```

mov  eax, 80000000h+(LAN_SLOTID shl 8)+58h    ;R02
mov  dx, 0cf8h
out  dx, eax

add  dx, 4

```

```
xor    eax, eax
mov    ax, word ptr ds:[si]           ;Set write data
shl    ecx, 16
or     eax, ecx                       ;Set WADDR(cx contains WADDR)
out    dx, eax                       ;Write data to eeprom shadow
shr    ecx, 16
```

VIA Networking
Technologies Inc.
Confidential
NDA Required

6. BIOS DMI Structure EEPROM Default Data

The following table is provided as an EEPROM data example for MB vendors. This EEPROM data is put in BIOS DMI structure. MB vendors can choose to put EEPROM data in other sources. However, we suggest MB vendors that they put the EEPROM data in DMI as this spec so that all our tools could be used without modification. The values provided in this table are default values. MB vendors can customize the EEPROM data to their own values.

In our example EEPROM DMI data, if it's necessary to support multiple EEPROM-less NICs, then we need to put multiple DMI structures in BIOS and use different handles in DMI structures to differentiate them.

VIA_EtherMAC_info label byte

db	D8H	;VT6120/VT6121/VT6122 MAC DMI type
db	45H	;Length
dw	0000H	;Handle
dd	00000000H	;EtherNet Address & Phy ID
dw	0000H	;EtherNet Address & Phy ID
db	07H	;EtherNet Address & Phy ID
db	00H	;VIA reserved data
dd	11060110H	;SubVendorID SubSystemID
dd	11063119H	;VendorID & SystemID
dd	0000101FH	;VIA User defined data,
dd	08030100H	;VIA User defined data,
dd	10131001H	;VIA User defined data,
dd	73733040H	;VIA User defined data,
db	20h dup (0)	;For Lan driver usage
db	01	;String No.
db	'VNTEK3216-NIC',0	;Signature
db	0	;Terminal character

About DMI structure format and access method, please refer to the following specs:

1. System Management BIOS Reference Specification.
2. Plug and Play BIOS Specification.