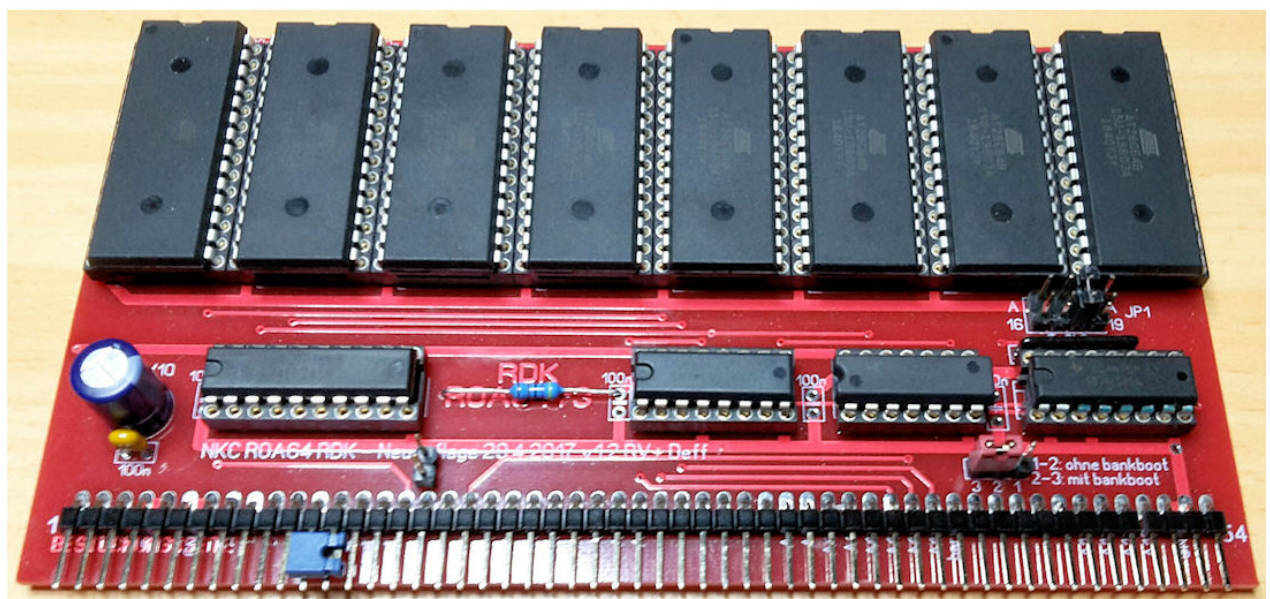


Spezifikation

EEPROM als EPROM Ersatz,
z.B. für eine ROA64 mit Grundprogramm



Version 1.0

Idee:

Sascha Neuschl
 Pirolweg 21
 48167 Münster
 Email: scn69@gmx.de

Dokumentenhistorie

Version	Autor(en)	Änderung	Datum
1.0	Neuschl, Sascha	Erste Version	12.02.2020

Inhaltsverzeichnis

1	Vorwort.....	4
1.1	Idee.....	4
1.2	Ansatz.....	4
1.3	Aktueller Stand	4
2	Beschreibung des Konzepts	4
3	Kopieren des Grundprogramms von der ROA64 mit EPROMs auf die ROA64 mit EEPROMs	6
3.1	Einsetzen der ROA mit den EEPROMs in den NKC-Bus	6
3.2	Laden, Übersetzen und Ausführen des folgenden Programms	7
3.3	Ersetzen der ROA64 mit den EPROMs durch die ROA mit den EEPROMS.....	10
4	Anhang.....	11
4.1	EEPROM AT28C64	11

1 Vorwort

1.1 Idee

Eigentlich war es mal die Idee, den NKC schneller zu machen. Und dazu muss zunächst mal der Zugriff auf die Speicher mit dem Grundprogramm schnell sein. Das hier betrachtete **EEPROM AT28C64B** besitzt eine **Lesezugriffszeit von 150 ns**. Dazu kommt aber noch, dass man sich das Brennen von EPROMs sparen kann.

1.2 Ansatz

Eine **ROA64** sollte mit **8 EEPROMs AT2864B (8K X 8)** bestückt und das **Grundprogramm** von der entsprechenden ROA64 dort hineinkopiert werden.

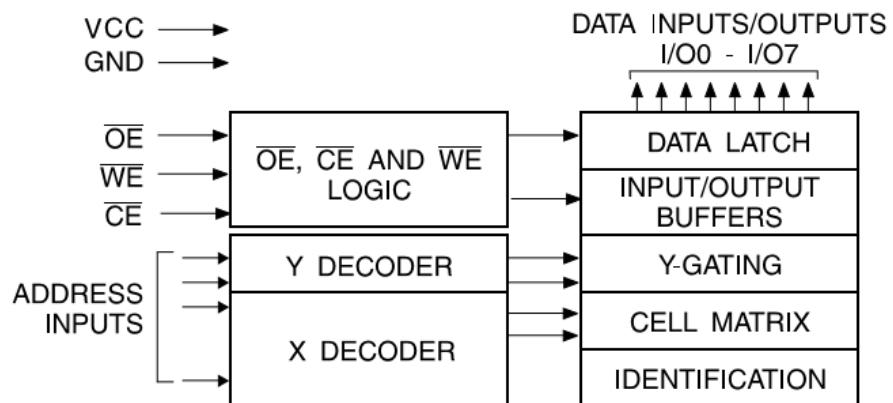
1.3 Aktueller Stand

Ich besitze eine ROA mit dem Grundprogramm 7.10v5 auf 8 EEPROMs, die brav ihre Dienste verrichtet. Das Testprogramm zum Schreiben in ein EEPROM liegt in Version 1.0 vor..

2 Beschreibung des Konzepts

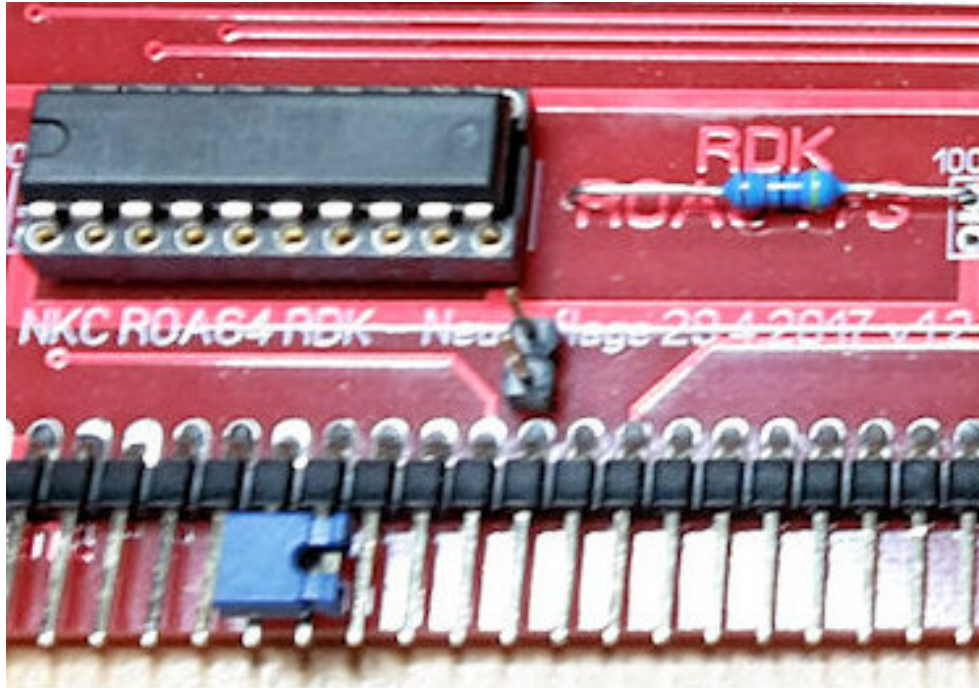
Alles was man braucht, ist 1 ROA64 mit dem Grundprogramm auf EPROMs und 1 ROA64, die mit den **EEPROMs AT28C64B** bestückt ist. Dass EEPROM ist PIN-kompatibel zum EPROM.

Block Diagram



Die **ROA64 mit den EEPROMs** benötigt eine **Erweiterung** um ein „write enable“, damit Daten auf den Speichern nicht zufällig gelöscht werden können. Dazu wird das **-WR-Signal** vom NKC-Bus in Richtung Schaltung unterbrochen und auf einen **Jumper** geführt. Auf der Seite des Jumpers in Richtung Schaltung wird ein **Pullup-Widerstand** (4,7 Kiloohm) gegen +5 Volt geschaltet.

Speech 2 – eine neue Sprachbaugruppe für den NDR-Klein-Computer



Des Weiteren wird ein Programm benötigt, das die Daten von der einen ROA64 liest und auf die andere ROA64 schreibt.

Folgende aktive Komponenten werden eingesetzt:

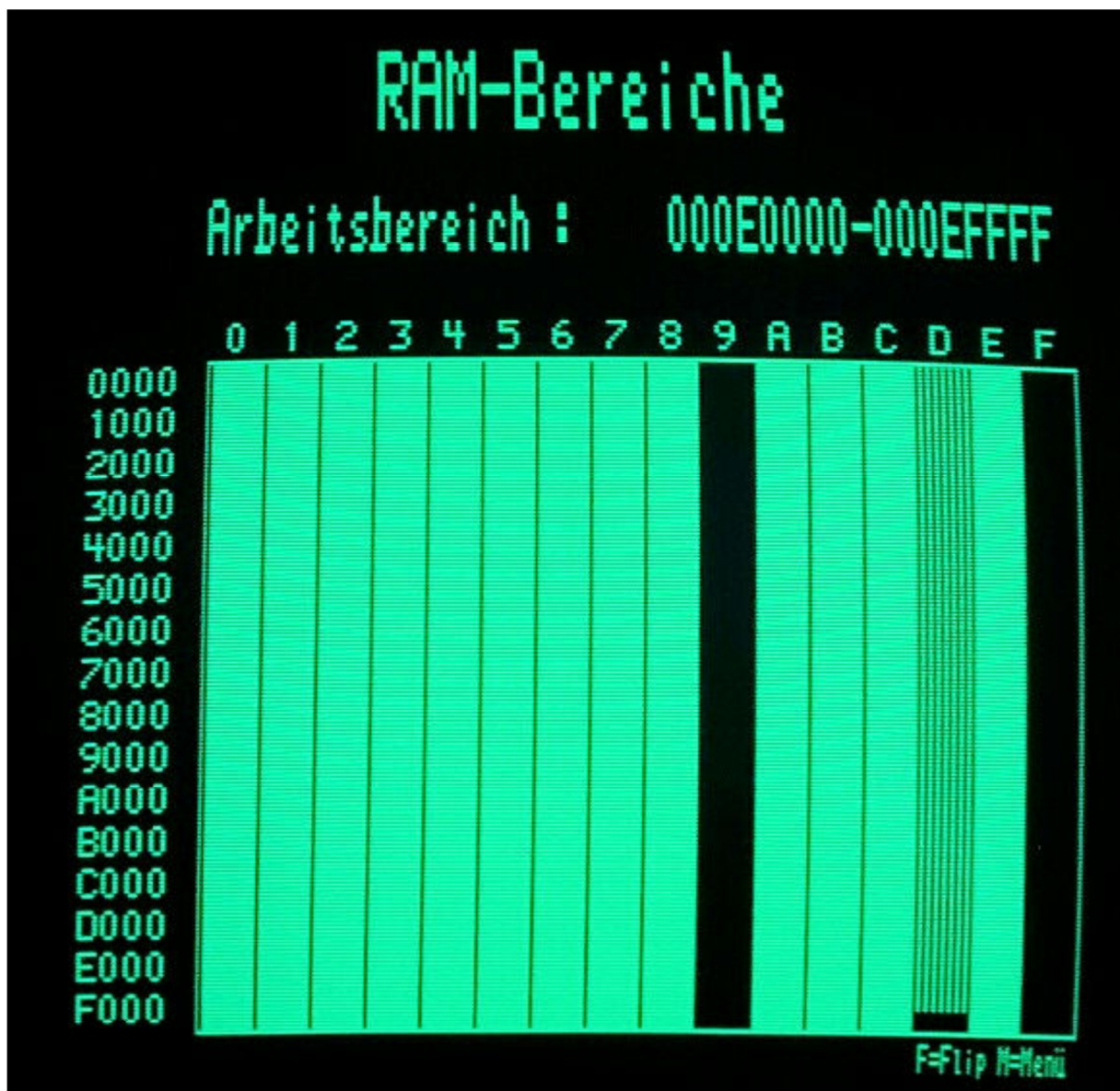
Funktion	Baustein	Beschreibung	Datenblatt
Speicherbaustein, normal im read only Modus	AT26C64B	EEPROM	AT28C64B.pdf.pdf

3 Kopieren des Grundprogramms von der ROA64 mit EPROMs auf die ROA64 mit EEPROMs

Hier wird der Ablauf beschrieben, wie man eine funktionsfähige ROA64 mit EEPROMs mit Grundprogramm funktionsfähig bekommt.

3.1 Einsetzen der ROA mit den EEPROMs in den NKC-Bus

Zunächst wird die ROA64 mit den EEPROMs auf einen freien 64K-Bereich (hier \$0009000) eingestellt und auf den NKC-Bus gesteckt. „write enable“ ist „enabled“ . also der Jumper gesteckt. In „Speicherbereiche“ des Grundprogramms sieht man die Karte jetzt nicht!



3.2 Laden, Übersetzen und Ausführen des folgenden Programms

Wenn das Programm in den Editor geladen ist, dann sind folgende **Konstanten einzustellen**:

- **Quelle** = Start-Adresse der ROA mit Grundprogramm auf EPROMs (hier \$000D000)
- **Schluss** = End-Adresse+1 der ROA mit Grundprogramm auf EPROMs (hier \$000E0000)
- **Ziel** = Start-Adresse der ROA mit den EEPROMs (hier \$00090000)

Damit man die EEPROMs erfolgreich schreiben kann, muss man nach einem „page write“ (**64 Bytes**) einen **Lesevorgang** durchführen und das **Bit 6 prüfen**. Solange es „toggelt“ - also sich von „0“ auf „1“ ändert usw. - , dann ist der „page write“ nicht beendet und man **muss warten**.

```

*****
* Programm zum Schreiben von Daten in das EEPROM AT28C64      *
* z.B. als EPROM-Ersatz.                                     *
* Hier wird das Grundprogramm auf einer ROA64 auf Adresse     *
* $D0000 auf eine andere ROA64 auf Adresse $90000 kopiert.  *
*****
*                      Version 1.0 - 10.02.2020 - by sEn      *
*****

```

```
; * Konstanten:
```

```

quelle          equ $D0000
schluss         equ $E0000
ziel            equ $90000

```

```
start:
```

```

clr d0
clr d1
clr d2
clr d3
clr d4

```

```

lea quelle, A0
lea ziel, A1

```

Speech 2 – eine neue Sprachbaugruppe für den NDR-Klein-Computer

```

move.l #schluss, d0

;* Gesamtschleife von Adresse 'Quelle' bis 'Schluss'
loop:
    ;* page write beim AT28C64B
    move #64-1, d4
    page:
        move.b (A0)+, (A1)+
        move.l A0, d1
        cmp.l d1, d0
        beq.s stop
    dbra d4, page

    ;* check toggling bit 6 für Ende eines write cycle
    toggle:
        move.b (A1), d2
        and.b #%01000000, d2
        move.b (A1), d3
        and.b #%01000000, d3
        cmp.b d2, d3
        bne.s toggle    ;Rücksprung solange das Bit toggelt

bra loop
stop:
rts

end

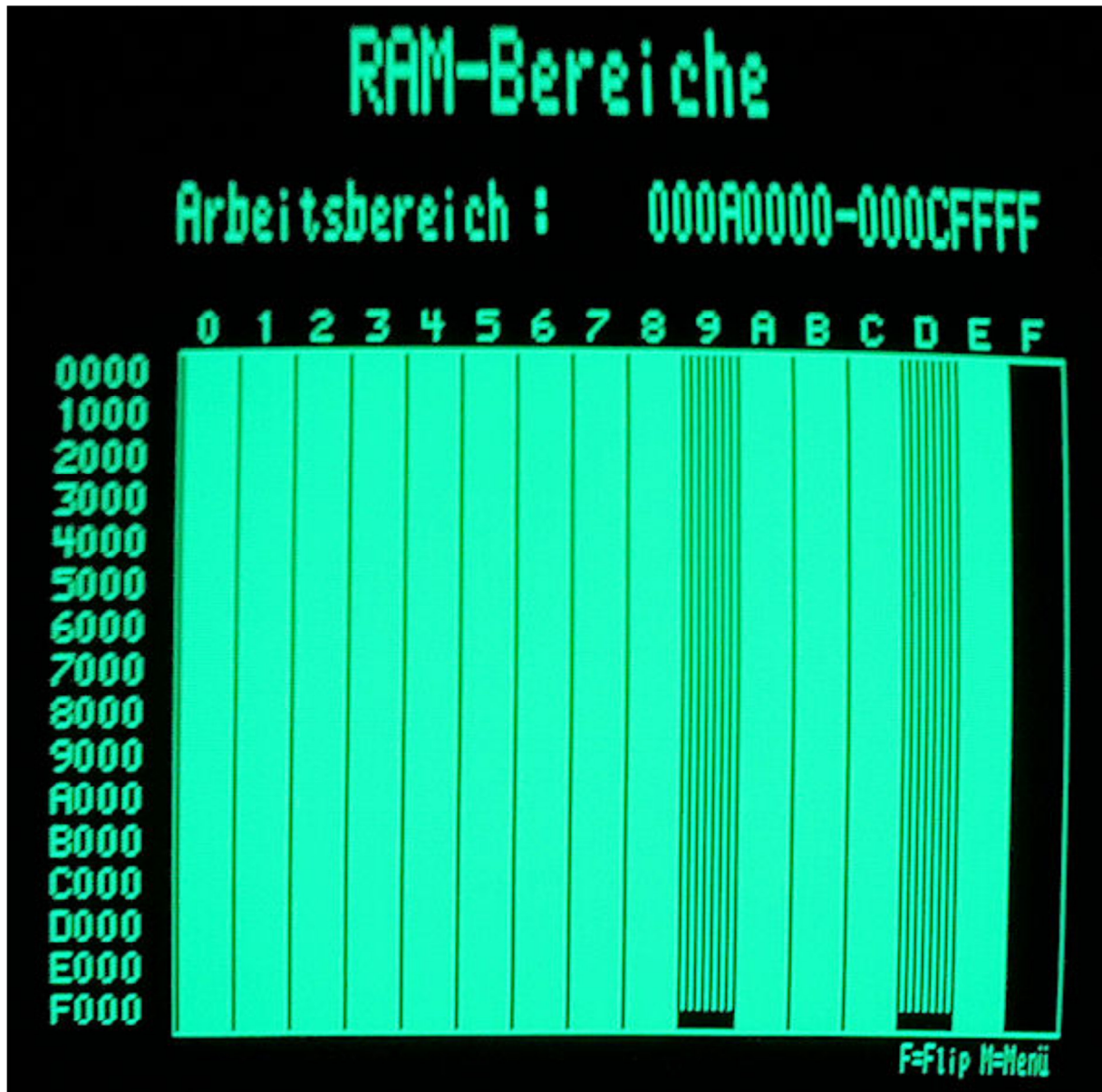
```


Speech 2 – eine neue Sprachbaugruppe für den NDR-Klein-Computer

Danach das Programm assemblieren und starten.

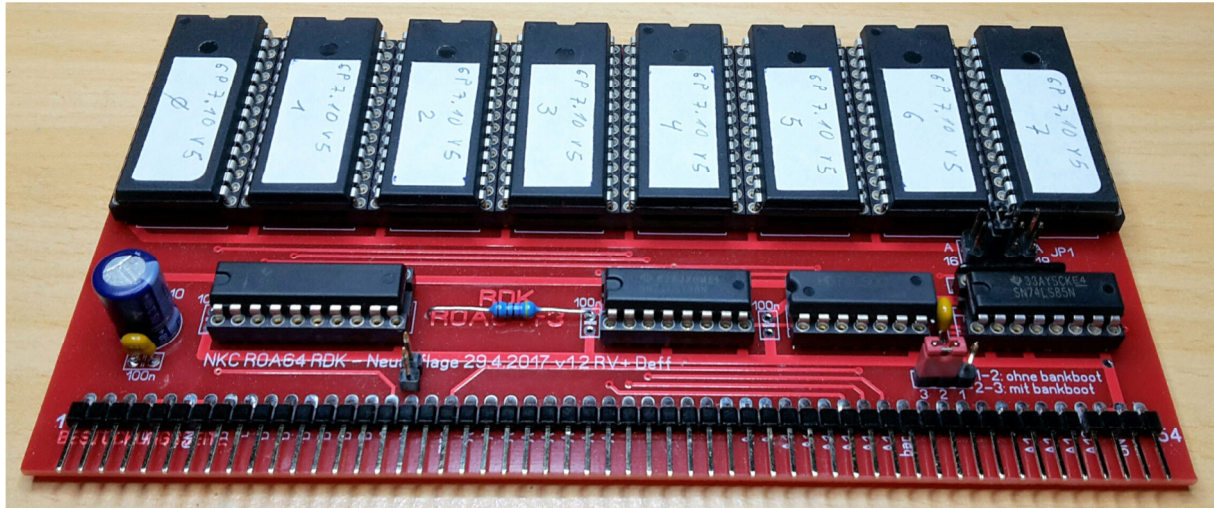
Jetzt ist das Grundprogramm auf die ROA64 mit den EEPROMs kopiert.

„write enable“ auf „disable“ setzen – also den Jumper entfernen! Und jetzt stellt sich die Karte in „Speicherbereiche“ des Grundprogramms wie eine Karte dar, auf der EPROMs stecken.



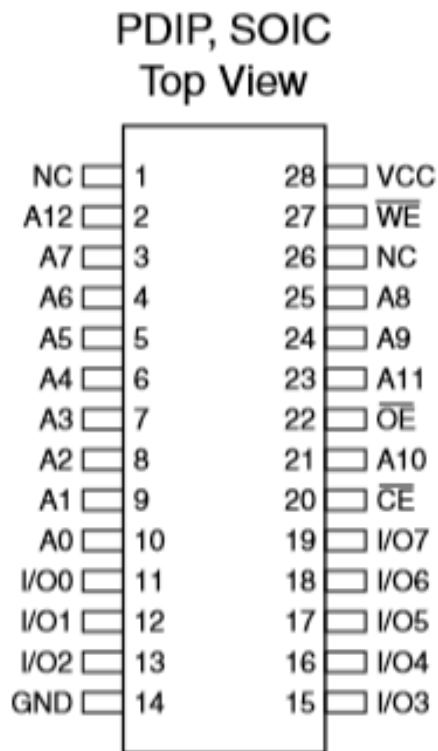
3.3 Ersetzen der ROA64 mit den EPROMs durch die ROA mit den EEPROMs

Nun braucht man nur noch die ROA mit den EPROMs vom NKC-Bus entfernen und die ROA64 mit den EEPROMs auf die Adresse der ROA64 mit den EPROMs einzustellen.



4 Anhang

4.1 EEPROM AT28C64



Pin Configurations

Pin Name	Function
A0 - A12	Addresses
\overline{CE}	Chip Enable
\overline{OE}	Output Enable
\overline{WE}	Write Enable
I/O0 - I/O7	Data Inputs/Outputs
NC	No Connect
DC	Don't Connect