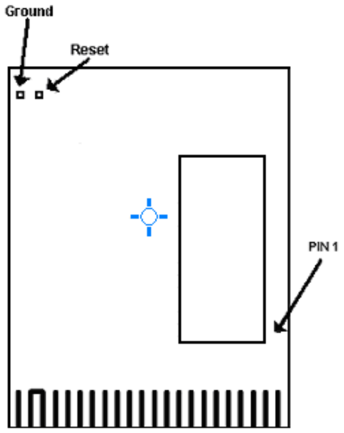
DIRECTIONS FOR USE

Thank you for purchasing an Alphaworks product, we hope that you find this PCB and supporting documentation to be of the highest quality.

The (C-1010 PCB) cartridge board is designed for use with 2764 series EPROM’s. The EPROM may be soldered directly to the board or sockets may also be used. See diagram blow for correct placement and orientation of the EPROM.

The EPROM site is always addressed to 32768 ($8000). When the cartridge board is plugged into the computer this EPROM takes the place of 8K of RAM at that address. See the memory map at the bottom of page 265 of the Programmer’s Reference Guide for further details.



PLASTIC CARTRIDGE CASE

The (C-1010 PCB) cartridge board was designed to fit inside a standard Commodore Cartridge Case, such as the ones used to house classic Commodore programs like International Soccer. To install your cartridge board in the plastic case, lay the PCB on the bottom half of the case (with the solder pins facing down). Place the cartridge board EPROM side up. Fit the top half of the cartridge case and screw together.

IMPORTANT

It is extremely important that you insert the PCB the correct way up on your Commodore 64 as to avoid problems. On the underside (bottom) of the PCB it contains the words -

© 2003 Alphaworks Pty Ltd
C64-PCB REV 8K-R3

CARTRIDGE CONFIGURATION

No additional IC's are required to build an 8 Kbytes cartridge. The cartridge board maps 8K to $8000-$9FFF. Maximum size of file is 33 BLOCKS.

HOW TO PROGRAM AN AUTOSTART CARTRIDGE

The system provides for "auto-start" of the program in a Commodore 64 Expansion Cartridge. The cartridge program is started if the first nine bytes of the cartridge ROM starting at location 32768 ($8000) contain specific data. The first two bytes must hold the Cold Start vector to be used by the cartridge program. The next two bytes at 32770 ($8002) must be the Warm Start vector used by the cartridge program. The next three bytes must be the letters, CBM, with bit 7 set in each letter. The last two bytes must be the digits "80" in PET ASCII.

AUTOSTART ‘CBM8O’ METHOD

The most common cartridge autostart method is known as the CMB80 method. It can be used by both 8K and 16K cartridges that reside at $8000. Although this method is sometimes called the cartridge autostart option’, it can be used equally well by RAM-based programs, and often is. One of the first things the KERNAL RESET routine does is check locations $8004-08 for the string of characters CBM80. If these exact characters are NOT found there, the KERNAL RESET process continues normally.

If the **“CBM80”** is found, the RESET routine is interrupted, and the processor immediately jumps to whatever location is specified by the **CARTRIDGE COLD—START VECTOR**. This vector is expected to be found at locations $8000-0l. You must place a pointer here, in standard b-byte / hi-byte order, directing the processor to the beginning of your cartridge code. From that point on, your cartridge must handle all the initialization itself for any functions it will use, such as the I/O devices or KERNAL or BASIC routines. Fortunately, you still have the KERNAL initialization routines available for use. Unless you know exactly what you are doing, your cartridge should use these routines to initialize the functions it needs.

Figure 1 presents a generic cartridge initialization routine. These routine duplicates most of the normal RESET process. In fact, its taken right from the main parts of the KERNAL ($FCEF-FE) and BASIC ($E394—9F) RESET routines. This generic routine will be adequate for 99% of all cartridges.

Figure1: **<u>CBM80 Cartridge Initialization</u>**

```
8000   09 80                         Cartridge cold-start vector = $8009
8002   25 80                         “ “ warm “ “  ‘ “ = 8025
8004   C3 C2 CD 38 30                CBM80 - Autostart key

                                     KERNAL RESET Routine
8009   8E 16  D0     STX $D016       Turn on VIC for PAL / NTSC check
800C   20 A3  FD     JSR $FDA3       IOINIT - Init CIA chips
800F   20 50  FD     JSR $FD50       RANTAS - Clear/test system RAM
8012   20 15  FD     JSR $FD15       RESTOR - Init KERNAL RAM vectors
8015   20 5B  FF     JSR $FF5B       CINT - Init VIC and screen editor
8018   58            CLI             Re-enable IRQ interrupts

                                     BASIC RESET Routine
8019   20 53 E4      JSR $E453       Init BASIC RAM vectors
8DlC   20 BF E3      JSR $E3BF       Main BASIC RAM Init routine
8O1F   20 22 E4      JSR $E422       Power-up message / NEW command
8022   A2 FB         LDX #$FB
8024   9A            TXS             Reduce stack pointer for BASIC

8025                                 START YOUR PROGRAM HERE
```

The cartridge cold-start vector and autostart key (CBM80) have already been discussed. The warm-start vector at $8002-03 is a feature that allows you to re—enter your program after a full initialization has already been done. Once a cold—start has been done, it usually doesn't need to be done again. Pressing the RESTORE key calls the NMI routine (NON-MASKABLE INTERRUPT), which will see the CBM80 and jump to the location indicated by the warm-start vector. Therefore, many programs restart themselves when you press the RESTORE key. In our initialization routine we have pointed the warm-start vector to the start of your program; you could also point it to $8009 to perform a full cold-start on RESTORE. If you want to disable the RESTORE key entirely, point the warm—start vector to $FEBC (return from NMI).

We have included the BASIC RESET process in this cartridge initialization routine too. The normal BASIC RESET routine dead-ends with a jump to the BASIC direct mode interpreter, also known as "READY" mode. This prints the "READY." prompt and then sits there waiting for, you to type a BASIC command. You won't usually want to exit into READY mode at this point since BASIC will take over and your cartridge will lose control. If you do want to exit to BASIC now or later, you may do so with JMP $E386. By the way, the routine called at $801F (JSR $E422) prints the normal power-up screen and does a NEW command. If you want to skip the power—up message, just call the NEW command directly using JSR $A644 instead of JSR $E422.