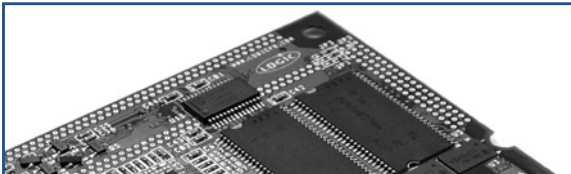




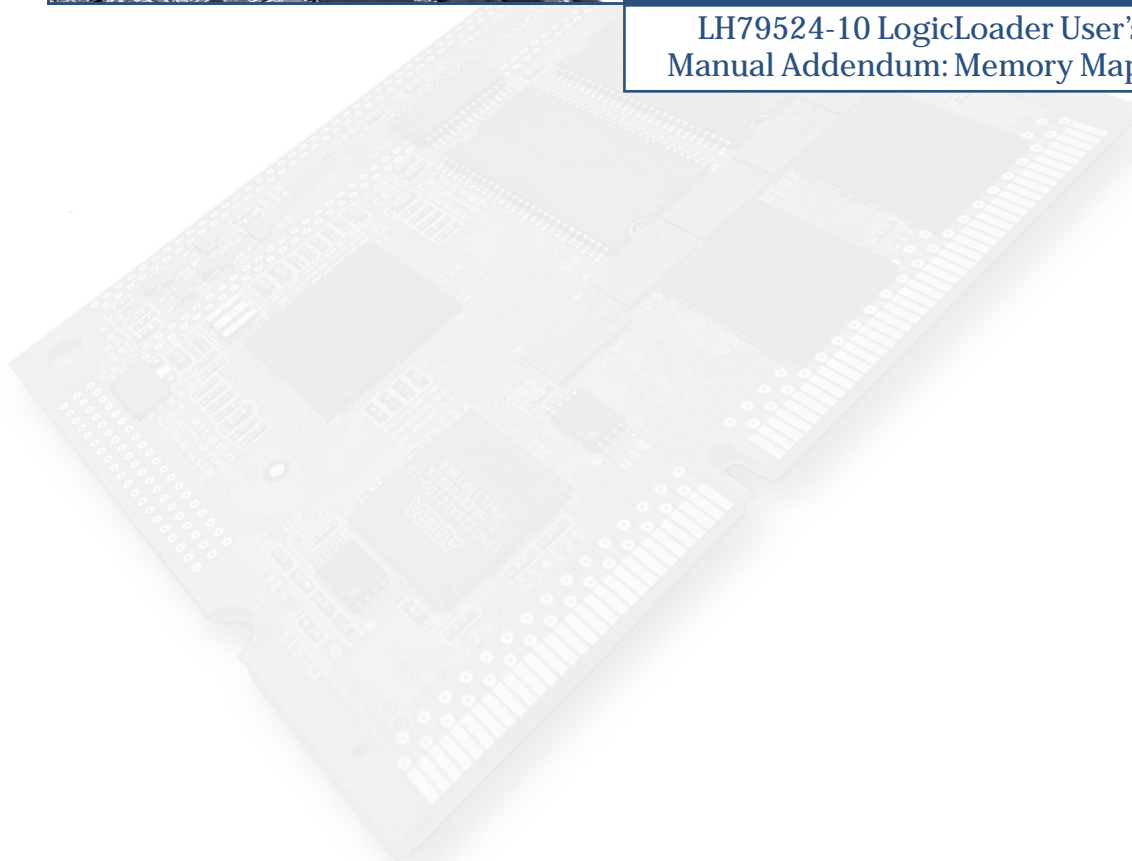
LOGIC PRODUCT DEVELOPMENT WWW.LOGICPD.COM



ZoomTM

Card Engine

LH79524-10 LogicLoader User's
Manual Addendum: Memory Maps



REVISION HISTORY

REV	EDITOR	REVISION DESCRIPTION	APPROVAL	DATE
A	Bruce Rovner	Release	ME	02/09/05

1 LH79524-10 LogicLoader Addendum: Memory Map Diagrams

1.1 LH79524-10 Hardware Memory Map

The LH79524 Card Engine is designed to accommodate SDRAM of different sizes. LoLo sets up the MMU to remap physical memory to logical memory. Type "info cpu" at the "losh" prompt to see how LoLo remaps physical memory to logical memory. (In order to view the information in entirety, lengthen the number of lines your software terminal emulator displays in TeraTerm by selecting **Setup/ Window**, and then adjust **Scroll Buffer** to the maximum 10000 lines.)

Figure 1.1: Type "info cpu" to See the Remap by the MMU

```

TeraTerm - COM1 VT
File Edit Setup Control Window Help
*****
LogicLoader

r Copyright 2002-2003, Logic Product Development, Inc.
All Rights Reserved.
Version 1.2.0
*****

Available commands:
load - download a binary image of type 'elf', or 'srec'
burn - burns the already-loaded image into flash device 'device'
erase - erases 'device' from start_address for length bytes
jump - jump to a loaded image, or [address]
exec - disable cache & ints, then jump to a loaded OS, or to [addr]
source - execute a series of losh commands stored in <filename>
w - write memory [of specified width] at addr
x - examine memory with [width][format] at an addr for a [len]
date - display the number of seconds since boot
info - print information about: version, arch, mem, net, cpu
help - print help for a single command or a group of commands.

losh> info cpu
Version : 1.2.0
vaddr -> paddr W - writable B - bufferable C - cacheable
sect 00000000 -> 00000000 WBC dom:4
sect 20000000 -> 20000000 WBC dom:4
sect 20100000 -> 20100000 WBC dom:4
sect 20200000 -> 20200000 WBC dom:4
sect 20300000 -> 20300000 WBC dom:4
sect 20400000 -> 20400000 WBC dom:4
sect 20500000 -> 20500000 WBC dom:4
sect 20600000 -> 20600000 WBC dom:4
sect 20700000 -> 20700000 WBC dom:4
sect 20800000 -> 20800000 WBC dom:4
sect 20900000 -> 20900000 WBC dom:4
sect 20a00000 -> 20a00000 WBC dom:4
sect 20b00000 -> 20b00000 WBC dom:4
sect 20c00000 -> 20c00000 WBC dom:4
sect 20d00000 -> 20d00000 WBC dom:4
sect 20e00000 -> 20e00000 WBC dom:4

```

Note: the figures you see may differ from those presented in this example.

The resulting logical hardware memory map can be seen in the following diagram.

Figure 1:2: LH79524-10 Hardware Memory Map

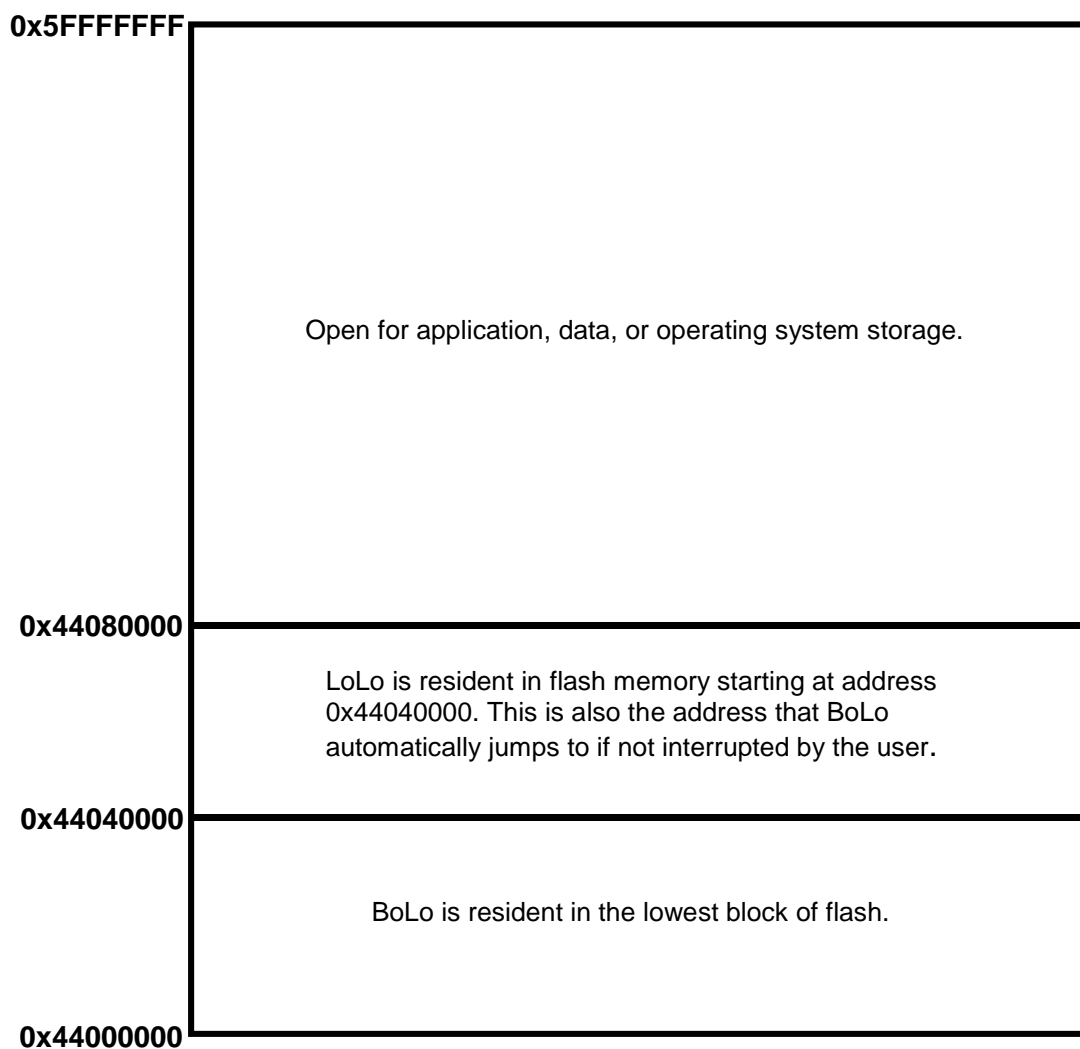
LH79524 Memory Map During Execution of BoLo and LoLo

0xFFFFFFFF	ADVANCED HIGH-PERFORMANCE BUS PERIPHERALS
0xFFFFF000	ADVANCED PERIPHERAL BUS PERIPHERALS
0xFFFC0000 0xFFFC0000	RESERVED (Invalid Access)
0xA0000000	BOOT ROM
0x80000000	INTERNAL STATIC MEMORY (16 KB on-chip SRAM)
0x60000000	Reserved (Invalid Access)
0x50000000	Fast Peripherals Chip Select 3 (CS3)
0x4C000000	Slow Peripherals Chip Select 2 (CS2)
0x48000000	Flash Memory
0x44000000	Empty
0x40000000	System SDRAM
0x20000000	System SDRAM (mirrored)
0x00000000	

1.2 Location of BoLo and LoLo in Flash Memory on the LH79524-10

The Zoom Starter Development Kits come with both BoLo and LoLo programmed into the Card Engine's resident flash array. See Figure 1.3 below.

Figure 1.3: Flash Memory Layout



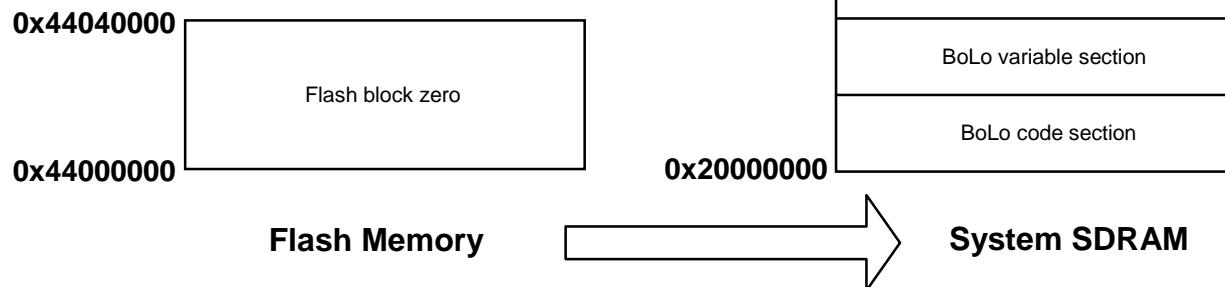
1.3 Run-time RAM location of BoLo and LoLo on the LH79524

Both BoLo and LoLo execute out of RAM. The diagrams shown on the following page, Figure 1:4: LH79524-10 RAM Execution Environment for BoLo and LoLo, depict the memory used by these two programs.

Figure 1.4: LH79524-10 RAM Execution Environment for BoLo and LoLo**Run-time Location of BoLo**

At reset, BoLo relocates itself from flash memory to system SDRAM. BoLo then spends the remainder of its run-time executing out of system SDRAM.

The size of BoLo's code and variable sections shown here are estimates. The size depends on the exact features built into the BoLo image and may change with new releases. The location of BoLo's stack is dynamically determined at run-time based on the size of the code and variable section. Therefore, the location of the stack is also given as an estimate.

**Run-time Location of LoLo**

If not interrupted by the user, or the presence of the RAM cookie, BoLo jumps to the address 0x44040000, where it expects to find LoLo. However, users may choose to overwrite LoLo with their own program code.

LoLo exhibits the same behavior as BoLo. LoLo relocates itself from flash memory to system SDRAM and then spends the remainder of its run-time executing in SDRAM. LoLo completely overwrites BoLo while relocating itself.

The size of LoLo's code and variable sections shown here are estimates. The true size depends on which features have been included in the final image. The location of the stack is determined at run-time based on the code size. Therefore, this data is given as an estimate.

