MKVI Download Board - Technical Overview

Introduction

This Technical Note describes the functionality of the MKVI DLB (Download Board) and the interface of the MKVI gaming platform to a universal download server. The MKVI receives downloaded games from an authenticated server and stores them as EPROM-type information.

MKVI supports game replacement and expansion via a memory expansion board interface. The DLB uses the interface to replace the MKVI EPROMs (Erasable Programmable Read-Only Memory) with SDRAM (Synchronous Dynamic Random Access Memory). Although the MKVI download board does not write to SDRAM memory, it includes a secondary CPU, Ethernet, CompactFlash®, two independent banks of SDRAM, EEPROM (Electrically Erasable Programmable Read-Only Memory), Ethernet disable switch input, and a GAT (Guidance Automation Toolkit) port. The DLB also supports secondary CPU (Central Processing Unit) interface, which serves two purposes:

1. To avoid ad network processing from MKVI and any impact on the games.
2. To significantly increase security by eliminating possible weaknesses in the TCP/IP (Transmission Control Protocol/Internet Protocol) stack or network handling software (a recurring problem for networked devices).

Game Storage

The MKVI download board is equipped with two independent 64MB banks of SDRAM:

- **Local Buffer:** A stack/working memory bank and temporary storage for downloaded games. This buffer is local to the secondary CPU.
- **Shared Buffer:** A shared memory bank between the secondary and MKVI CPU's. Access is controlled by the MKVI CPU, and only one buffer may have access at a time. When the MKVI CPU is granted access to the shared buffer, the data appears as if it were in EPROMs, allowing a large degree of backward compatibility with existing software.

The MKVI base software stores the new activation message in the EEPROM on the DLB in a pending activation area. When the idle and other conditions are met, the game sets a marker in the DLB EEPROM signifying there is a pending activation. When the game restarts, if there is a pending activation, if the pending activation marker is set, it requests the package in the pending activation. On successful package load, it copies the pending activation into the current activation and clears the pending activation marker.

Downloading and Authentication

The MKVI grants the secondary CPU access to the shared buffer, which facilitates the download from the server to the local buffer. Once the download is complete, the game is copied to the shared buffer. When a game has been copied from the local buffer into the shared buffer, the DLB secondary CPU is free to download another game into the local buffer or CompactFlash.

The MKVI primary CPU accesses the shared buffer, and authenticates the data. Once the primary CPU accesses the shared buffer, the secondary CPU cannot access or modify the shared buffer, and the game is locked and authenticated. This authentication process provides superior game security. In the event of network corruption or other disturbances, the CPU will not allow the DLB to run unauthenticated code.

Authentication is achieved using DSA (Digital Signature Algorithm) cryptographic signatures. Prior to successful authentication, the MKVI has only limited access to the game data via a one word window with the address being provided by automatically by the FPGA in an automatically incrementing sequence every time a word of data is read. The window resides at a fixed address in MKVI memory, but allows MKVI to scan out the buffer contents for DSA signature calculation without the possibility of accidentally executing any software in the shared buffer. When the game has been successfully authenticated, the MKVI DLB enables the shared buffer within the MKVI memory as if it were EPROM.

Inter-Processor Communication

A FIFO (First In First Out) communication protocol between the MKVI and secondary CPU negotiates the game download and buffer swapping. The communication protocol is designed to maintain machine security.

Secondary CPU

The secondary CPU runs from an on-board Boot EPROM.

CompactFlash®

A CompactFlash memory card can be used to store downloaded games in the event that the connection to the download server is lost. Similar to the downloading process described earlier, a game is copied from the CompactFlash card into the shared buffer, and then authenticated.

The EGM continues to play the last game downloaded. For games of up to 32MB, a CompactFlash card of at least 64MB is required. This allows one game to be stored and a new game to be written. If connection is lost before the second game is written, the first game will remain intact and valid. A CompactFlash card with greater storage capacity will support multiple games.The MKVI DLB is equipped with a 50-pin, CompactFlash connector to accommodate Type I and Type II cards. Pin connections support true IDE mode only.

GAT3 Port

The GAT3 protocol is implemented by the MKVI, with a RS232 interface and standard 9-pin, D-type connector on the DLB. The UART (Universal Asynchronous Receiver-Transmitter) is controlled by the MKVI primary CPU. The GAT port is accessed through the logic cage door opening.

EPROM

The DLB is equipped with three EPROM sockets:

- One Boot EPROM, which runs the download board CPU
- Two system EPROMs located on the DLB, not the MKVI main board

EEPROM

The DLB is equipped with one 1MB EEPROM socket. It stores download logs as required by regulation, and is accessed via the MKVI secondary CPU.

Remote Access Control

The MKVI download board is designed with a 2-way Micro-fit connector to support a magnetic card reader. This feature signals the DLB when a card is being inserted into the reader.

This feature may not be required in the production board, in which case the switch will not be fitted.

Mechanical Aspects

The storage expansion board attaches (piggybacks) to the MKVI main board by the retentive force of the DIN41612 connector and three spacers. The holes for each of the spacers correspond to holes in the main board. A second hole in close proximity of each allows for additional boards to be stacked with alternating spacer positions.

Power Supply

The MKVI download board is powered by 3.3V DC, which is supplied by the main board.

Board Expansion

Up to two boards can be fitted to the MKVI-XP expansion connector. The MKVI-XP will check for games at both locations upon initialization. The MKVI download board is configured as board location No.2 and does not require a switch.

Local Access Switch

A two-pole, piano-slide, local access switch is located on the front of the MKVI download board. Pole No.1 (when **ON**) prevents external access to DLB memory, and disables the Ethernet port.

This feature may not be required in the production board, in which case the switch will not be fitted.

Board LEDs

Board LEDs (Light-Emitting Diode) illuminate to indicate that the MKVI-XP has selected the download board.

An illuminated green LED mounted on the RJ45 Ethernet connector indicates an operational Ethernet link. An illuminated yellow LED indicates transmit/receive activity.

System Interface

Expansion Interface

A pass-through, expansion interface is provided to mate with the 96-way DIN41612 MKVI expansion interface connector of the main board. The connector extends the expansion interface of the MKVI download board to support additional stacked boards.

The expansion interface connector is female on the topside of the board and male on the underside. This combination is achieved using a female press-fit style connector with 13mm gold plated tails mounted on the topside of the board. A standard shroud fitted onto the tails forms a male connector on the underside of the board.

System Code

System code is currently supplied by EPROMs fitted to the DLB; future development of the board will support system code provided directly by SDRAM.

Game Code

Game code is supplied through the SDRAM based on the signal timings. It requires that “wait states” shall not be asserted during SDRAM access as it may adversely affect timings within the gaming machine. Therefore, SDRAM refresh cycles cannot require the assertion of the NMEBWAIT line.