MO5 FPGA Implementation Technical Manual

Introduction

This document details the VHDL implementation of the Thomson MO5 computer on an Intel/Altera FPGA platform. The MO5 was a popular French home computer from the 1980s, and this FPGA implementation faithfully reproduces its functionality using modern hardware.

System Overview

The Thomson MO5 FPGA implementation is designed for the DE1 FPGA development board featuring an Intel/Altera FPGA. The system reproduces the original MO5 architecture including:

- 6809-compatible CPU

- 32KB RAM

- ROM system with cartridge support

- PIA interface for keyboard, joystick and peripherals

- Video output supporting color modes

- PS/2 keyboard interface with multiple layout support

- Sound generation

- SD card interface for storage

Hardware Architecture

Top Module (DE1_MO5)

The top module `DE1_MO5` interfaces with the DE1 board hardware and connects all system components. It manages:

- Clock generation and distribution

- Reset logic

- Memory mapping between CPU, RAM, ROM and peripherals

- I/O interfacing (keyboard, video, audio, storage)

CPU Implementation

The system uses a 6809-compatible CPU core (`MO5_CPU`), which is a wrapper around John Kent's CPU09 core. Key features:

- Full 6809 instruction set compatible

- Address space of 64KB

- Support for IRQ and FIRQ interrupt modes

- Switchable between 1MHz and 10MHz operation

Memory System

The memory system consists of:

1. **RAM (MO5_RAM)**:
   - Maps the original 32KB memory model to the SRAM on the FPGA board
   - Handles memory banking for the video memory and main memory

2. **ROM (MO5_ROM)**:
   - Maps the original system ROM (monitor) and cartridge ROM to flash memory
   - Provides cartridge selection mechanism

3. **RAM Initializer**:
   - Initializes the RAM at system startup
   - Ensures proper state during boot sequence

Peripheral Interface Adapters (MO5_PIA)

The system includes a MC6821-compatible PIA implementation that handles:

- Keyboard matrix scanning
- Border color control
- Sound bit output
- Light pen support
- System timing synchronization

Video Subsystem (MO5_VIDEO)

The video system implements the MO5 graphics capabilities with enhanced VGA output:

- Supports 1024x768 VGA output (configurable to other resolutions)
- Implements the original 320x200 resolution with border
- Separate memory maps for character shapes and colors
- 16-color palette matching the original MO5 colors
- Hardware-accelerated pixel generation

Keyboard Interface (MO5_KBD)

The keyboard subsystem converts modern PS/2 keyboard input to the original MO5 keyboard matrix:

- PS/2 protocol decoder
- Scan code translation

- Support for multiple keyboard layouts:
  - QWERTY
  - AZERTY (French)
  - Direct 1-to-1 mapping
- Special key combinations for system control

Sound System (MO5_SOUND)

The sound system implements the 1-bit audio of the original MO5 with improvements:

- Converts 1-bit digital sound to analog via WM8731 codec
- I2C interface for codec configuration
- 48kHz sampling rate
- 16-bit audio output

SD Card Interface (MO5_SDDRIVE)

The system includes an SD card interface for storage:

- SPI protocol implementation for SD card communication
- ROM-based driver
- Activity LEDs for read/write operations

Detailed Signal Documentation

Clock System

The `MO5_CLOCK` module generates four critical clocks:

1. **CPU Clock**: 1MHz or 10MHz, selectable via switch
2. **VGA Clock**: 25MHz for standard VGA, higher for enhanced resolutions
3. **SYNLT Clock**: 50Hz for system timing
4. **Sound Clock**: 48kHz for audio sampling

Memory Map

The MO5 memory map is preserved:

- **$0000-$1FFF**: Video memory (accessed based on 'forme' signal)
- **$2000-$9FFF**: Main RAM
- **$A000-$A7BF**: System RAM
- **$A7C0-$A7C3**: PIA registers
- **$A7BF**: SD card interface

- **$B000-$EFFF**: Cartridge ROM
- **$F000-$FFFF**: System ROM (monitor)

Control Signals

- **reset_n**: System reset (active low)
- **forme**: Memory access mode selection (video shape vs. color data)
- **synlt_clock**: 50Hz system timing signal
- **cpu_reset_n**: CPU reset signal

Keyboard Layout Implementation

The keyboard system supports three modes selected by switches:

1. **QWERTY Mode (00)**: Standard US keyboard mapping
2. **AZERTY Mode (01)**: French keyboard mapping
3. **Direct Mode (10/11)**: Direct mapping for custom configurations

The conversion from PS/2 scan codes to MO5 keyboard matrix is handled in three stages:

1. **PS/2 Decoder**: Decodes the raw PS/2 protocol
2. **Scan Code Assembler**: Processes make/break codes and extended keys
3. **MO5 Keyboard Decoder**: Maps processed scan codes to MO5 keyboard matrix

Special key combinations:

- Ctrl+Alt+Del: System reset
- Function keys: Mapped to numeric keys

Video System Details

The video system generates VGA output from the MO5's original display data using multiple components:

1. **VGA Controller**: Generates timing signals for the selected resolution
2. **Coordinate Translator**: Maps VGA coordinates to MO5 video memory
3. **Shape and Color Memory**: Dual-port RAMs for pixel and color data
4. **Pixel Selector**: Extracts individual pixels from memory
5. **Color Selector**: Determines the final pixel color
6. **Palette**: Converts MO5 color codes to RGB values

Display modes:

- **Pixel Mode 00**: Blanking area (black)

- **Pixel Mode 01**: Border area (border color)

- **Pixel Mode 10/11**: Active display area (foreground/background color)

SD Card Interface

The SD card interface provides mass storage capability:

- Simple SPI protocol implementation

- Command/data transfer capabilities

- Status indication via LEDs

Implementation Notes

Board Compatibility

This implementation is specifically designed for the DE1 FPGA development board with:

- Altera/Intel FPGA

- 8MB SDRAM

- 4MB Flash memory

- I2C audio CODEC

- PS/2 keyboard port

- SD card slot

- VGA output

Resource Utilization

The implementation requires:

- Logic elements: ~8,000

- Memory bits: ~300,000

- PLLs: 1-2 depending on configuration

- I/O pins: ~70

Building and Configuration

Build Requirements

- Intel Quartus Prime (or Altera Quartus II)

- ModelSim for simulation (optional)

- VHDL-capable synthesis tools

Configuration Options

The system provides several configuration options:

1. **Video Resolution**: Configurable in the VGA_CTRL generic parameters
2. **Keyboard Layout**: Selectable via SW[1:0] switches
3. **CPU Speed**: Selectable via SW[2] switch

Operation Guide

1. **Power on** the DE1 board with the MO5 FPGA configuration
2. The system will initialize RAM and reset the CPU
3. **Keyboard Mode Selection**:
   - Set SW[1:0] to select keyboard layout (00=QWERTY, 01=AZERTY, 10/11=Direct)
4. **CPU Speed Selection**:
   - Set SW[2] to select CPU speed (0=1MHz, 1=10MHz)
5. **Reset**:
   - Press KEY[0] for hardware reset
   - Use Ctrl+Alt+Del for software reset

Technical Limitations and Enhancements

Limitations

- No cassette interface implementation
- Limited cartridge selection mechanism
- Some timing differences from the original hardware

Enhancements

- Higher resolution VGA output
- Switchable CPU speed
- Multiple keyboard layout support
- SD card storage instead of cassette
- Enhanced audio output

Debugging Features

- 7-segment displays showing CPU address
- LED indicators for system status
- KEY buttons for manual control

Credits

This implementation relies on several open-source components:

- CPU09 core by John Kent

- VGA controller architecture

- PS/2 keyboard decoder logic

- I2C controller for audio codec

Appendix: Signal Descriptions

External Interfaces

Interface	Description
VGA	Video output (HS, VS, R, G, B)
PS/2	Keyboard input (CLK, DAT)
AUDIO	I2C audio codec interface
SD CARD	SPI-based storage interface

Key Internal Signals

Signal	Width	Description
address	16 bits	CPU address bus
data_in	8 bits	CPU data input
data_out	8 bits	CPU data output
rw	1 bit	Read/write control
vma	1 bit	Valid memory address
irq_n	1 bit	Interrupt request
firq_n	1 bit	Fast interrupt request
reset_n	1 bit	System reset
forme	1 bit	Video memory mode selection