Documentation for CPU Simulators

Brent Seidel
Phoenix, AZ

April 1, 2024

Contents

Chapter 1

Introduction

This project includes simulators for some existing processors. It can be embedded into other code as a library, or used stand-alone with a command line interpreter. The intent of these simulators are to provide instruction level simulation and not hardware level. Generally, no attempt has been made to count clock cycles - instructions may not not even take the same relative amount of time to execute.

Interfaces are provided to allow the simulator to be controlled by and display data on a simulated control panel (see the Pi-Mainframe (`https://github.com/BrentSeidel/Pi-Mainframe`) project. These may be stubbed out or ignored if not needed.

Chapter 2

Simulators

Several simulators are available for use. Each simulator may also have variation. So, one simulator may provide variations for different processors in a family of processors.

Each simulator is based on an object that derives from the `simulator` object defined in the `BBS.Sim_CPU` package. A generic simulator interface is defined with some procedures or functions that must be defined by a specific simulator and some that may be defined, if needed. There are also a number of utility functions that are not expected to be overridden by a specific simulator.

2.1 Example

The example simulator provides and example of using the simulator object interface. Its primary purpose is to blink the lights in interesting ways in the Pi-Mainframe (`https://github.com/BrentSeidel/Pi-Mainframe`) project. There are a number of different patterns selectable.

2.2 8080 Family

2.3 68000 Family

Chapter 3

I/O Devices

3.1 Clock

3.2 Serial Ports

3.3 Disk Interfaces

Chapter 4

Command Line Interface

4.1 Commands

4.2 Lisp Programming