
Easier68k Documentation

Easier68k Contributors

Feb 21, 2019

Table Of Contents:

1	easier68k package	3
1.1	Subpackages	3
1.1.1	easier68k.assembler package	3
1.1.2	easier68k.core package	3
1.1.3	easier68k.simulator package	5
	Python Module Index	7

Easier68k is a Python-based assembler and simulator of the Motorola 68000 microprocessor. It is based [Prof. Charles Kelly's EASy68k](#), with the goal of adding better cross-platform support without the use of Wine or emulation (among other features).

Warning: Easier68k is still very incomplete, and does not yet have the same feature set of EASy68k implemented. As a result, it's not quite ready for any wide-spread use (although why are you writing 68k in the first place?).

A sample use-case of Easier68k can be found at github.com/Chris-Johnston/Easier68k-SampleProject.

If you would like to contribute to Easier68k's development, refer to [our Issue list on GitHub](#) as well as [the contributing guide](#).

CHAPTER 1

easier68k package

This is the root package for all Easier68k modules and subpackages.

1.1 Subpackages

1.1.1 easier68k.assembler package

The assembler modules are responsible for parsing and converting instructions into machine code.

Submodules

`easier68k.assembler.assembler` module

1.1.2 easier68k.core package

The *easier68k.core* package contains types and logic that are used by both the assembler and the simulator.

Subpackages

`easier68k.core.enum` package

Submodules

`easier68k.core.enum.condition` module

`easier68k.core.enum.condition_status_code` module

`easier68k.core.enum.ea_mode` module

`easier68k.core.enum.ea_mode_bin` module

`easier68k.core.enum.op_size` module

`easier68k.core.enum.register` module

`easier68k.core.enum.srecordtype` module

`easier68k.core.enum.system_status_code` module

`easier68k.core.enum.trap_task` module

`easier68k.core.enum.trap_vector` module

`easier68k.core.models` package

Submodules

`easier68k.core.models.assembly_parameter` module

`easier68k.core.models.list_file` module

`easier68k.core.models.memory_value` module

`easier68k.core.models.trap_vector` module

`easier68k.core.opcodes` package

This page lists all of the opcodes that have been implemented in Easier68k.

Submodules

`easier68k.core.opcodes.add` module

`easier68k.core.opcodes.adda` module

`easier68k.core.opcodes.cmp` module

`easier68k.core.opcodes.cmpi` module

`easier68k.core.opcodes.dc` module

`easier68k.core.opcodes.eor` module

`easier68k.core.opcodes.jsr` module

`easier68k.core.opcodes.lea` module

`easier68k.core.opcodes.move` module

`easier68k.core.opcodes.movea` module

`easier68k.core.opcodes.neg` module

`easier68k.core.opcodes.opcode` module

`easier68k.core.opcodes.opcode_or` module

`easier68k.core.opcodes.ori` module

`easier68k.core.opcodes.rts` module

`easier68k.core.opcodes.simhalt` module

`easier68k.core.opcodes.sub` module

`easier68k.core.opcodes.subq` module

`easier68k.core.opcodes.trap` module

`easier68k.core.util` package

Submodules

`easier68k.core.util.conversions` module

`easier68k.core.util.find_module` module

`easier68k.core.util.input` module

`easier68k.core.util.opcode_util` module

`easier68k.core.util.parsing` module

`easier68k.core.util.split_bits` module

1.1.3 `easier68k.simulator` package

Submodules

easier68k.simulator.clock module

Used by m68k to handle stepping through actions

```
class easier68k.simulator.clock.Clock  
    Constructor
```

easier68k.simulator.m68k module

easier68k.simulator.memory module

e

`easier68k.simulator.clock`, 6

C

Clock (class in easier68k.simulator.clock), 6

E

easier68k.simulator.clock (module), 6