# The PHINIX+ System Architecture Documentation
## Part 1: The CPU

# Contents

# 1 Introduction

This document is the official specification for the PHINIX+ Central Processing Unit. It is intended to explain in detail the capabilities and the layout of the processor in an abstract manner in order to remain agnostic of the possible implementations of it. While this document doesn't try to make any assertions of a "correct" sort of implementation, the architecture was built with the intention to exploit pipelining to gain in performance.

## 1.1 Ancestral History

PHINIX+ is a "constructed" acronym which stands for *Pipelined High-speed INteger Instruction eXecutor*. The "+" in the name is to signify and advancement from a previously designed processor, PHINIX, from whom most ideas where directly taken and improved upon. PHINIX used 16bit word-addressing which turned out to be unwieldy and not deliver in terms of memory capacity.PHINIX+ expands to 32 bits while also adding byte-addressing to simplify integration with the existing computing paradigms, all based around 8 bit units.

## 1.2 Influence Sources

PHINIX+ mainly derives from the *Reduced Instruction Set Computing* (RISC) paradigm. However that does not mean it follows the established norm for a RISC processor, opting instead for a more expansive set of instructions, mainly concerning the improvement of flags management and bit math. The core principles of RISC like the load-store paradigm and the general usage nature of the provided registers do exist in PHINIX+ but not without being improved upon.

One of the most apparent features a programmer wishing to use PHINIX+ encounters is the dual register file. This is a feature influenced directly by the Motorola 68000 series of processors. Though that processor was in no way following RISC, the the adoption of the dual register file was due to similar reasons. As a result PHINIX+ has been lovingly nicknamed the *Actually-RISC™ M68k*[1].

---

[1]Disclaimer: Not actually a trademark.