Baby040 Specification

A Personal Computer Platform

Anthony Guerrero

Document Revision 0.0.1

1 System Overview

1.1 Design Goal

The primary objective of the Baby040 is to construct a personal computer platform that aligns with the capabilities and aesthetics of computing systems from the 1990s. The system is not merely a reflection of retro hardware but a modern interpretation emphasizing complete transparancy, user understanding, and modifiability. The design principles prioritize open-source documentation, source code, and hardware description language (HDL) to empower users and enthusiasts to understand, modify, and expand upon the system.

1.2 Key Features

- **Open Architecture:** An open-source ethos is at the core of the project. All documentation, designs, and code associated with this system will be freely accessible and modifiable by the end-users and community.
- **MC68040 Processor:** Serving as the heart of the system, this 32-bit microprocessor offers a blend of performance and functionality suitable for a wide range of computing tasks. It's Instruction Set Architecture (ISA) and built-in MMU lends itself well to implementing UNIX-like operating systems.
- **Flexible Expansion:** The platform is designed with expansion and modularity in mind, featuring an "Expansion Bus" architecture to facilitate the addition of new components. The bus dynamically assigns memory space to bus members, bypassing address conflicts and setup jumpers.
- **Integrated FPGA:** Utilizing the iCE40-HX4K FPGA, with full access to the CPU bus, most glue logic and peripheral controllers can be implemented and modified after the design is finalized.
- **512MB Dynamic RAM:** With four 72-pin SIMM slots, the platform supports up to 512MB of main system memory. (3.3v only!)
- **Standard Form Factor:** The PCB will be sized to fit in a pre-ATX form factor called "Baby AT". This allows for eight expansion slots, a DIN-5 sized cutout for a keyboard, and simple power management.

1.3 System Philosophy

Beyond the technical specifications, this project embodies a philosophy of transparency, education, and user empowerment. The system is not just a computing platform but a testament to the ethos that technology should be understandable, modifiable, and, above all, open. Whether for education, nostalgia, or innovation, this platform seeks to be a canvas for users to explore the intricacies of computing, reminiscent of an era when personal computing was just blossoming.

2 Hardware

2.1 CPU

The Motorola MC68040 was chosen because I found it in a box. The processor operates in the non-multiplexed address mode and with control circuitry that asserts the . Data Latch Enable (DLE) is not used.

2.2 SRAM

There are 512KB of SRAM for boot ROM, et al...

2.3 DRAM

Four 72-pin SIMM slots provide the bulk RAM for the system. Each SIMM's $\overline{\text{RAS}}$ lines are tied together and sent to the memory controller, $\overline{\text{CAS}}$ lines are routed separately. The memory controller only supports 3.3v SIMMs (TODO)
The SIMMs I have (Keystron MK16DS3232LP-50) have (16) 16M × 4-bit DRAMs per module. (Place chip to $\overline{\text{CAS}}$, $\overline{\text{RAS}}$, DQ here)

2.4 I2C Bus

I2C is used on-board for communication with the Real

3 Memory Map

Pay no attention to the memory map diagram at Figure 1, it's probably inaccurate.

When $\overline{\text{RESET}}$ is asserted from a reboot, power-up, etc., the address decoder maps addresses `$00000000-$000003FF` to `$8000FC00-$8000FFFF`. This allows the processor to read the exception vector table from ROM, but writes to these addresses will be routed to RAM. Once the system RAM test passes, the boot program copies the table to system RAM and writes to a configuration register which remaps all accesses from `$00000000` to RAM.

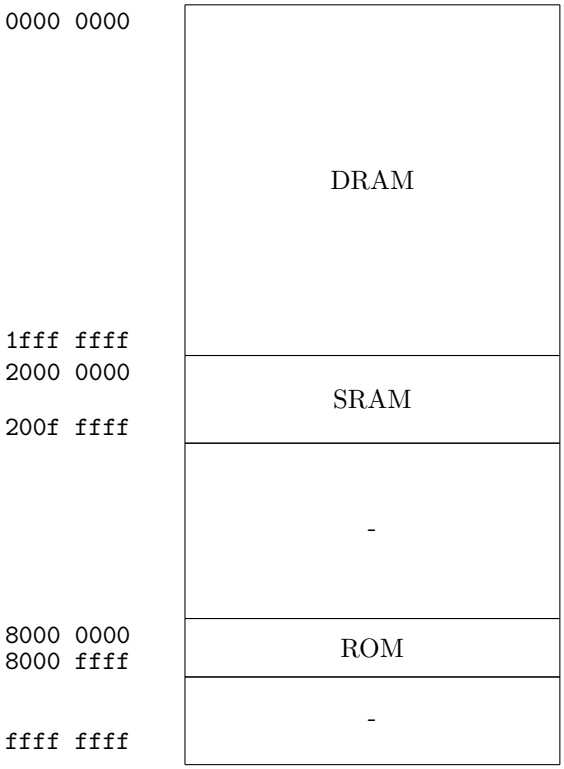
4 On-board Bus

Standard 68040 bus, DRAM data is buffered through FPGA, MC68150 bus resizer for expansion bus, I2C provided by PCA9564...

5 Expansion Bus

The system provides a Zorro III-inspired expansion slot bus, the main feature of which is dynamic memory mapping of add-in devices.

Figure 1: System Memory Map



5.1 Signal Description

5.2 Electrical Specifications

5.3 Mechanical Specification

6 Interrupt System

7 Special Thanks

Ben Eater for creating engaging videos on computer engineering, inspiring me to build a computer of my own.
Lawrence Manning MAXI030
Stephen Moody Y Ddraig(030)
Dave Haynie Author of "The Zorro III Bus Specification"

Figure 2: Expansion Bus Connector Pinout

Pin	Name	Pin	Name	Pin	Name
a1	NC[1]	b1	NC	c1	NC
a2	NC	b2	NC	c2	NC
a3	NC	b3	NC	c3	NC
a4	NC	b4	NC	c4	NC
a5	NC	b5	NC	c5	NC
a6	NC	b6	NC	c6	NC
a7	NC	b7	NC	c7	NC
a8	NC	b8	NC	c8	NC
a9	NC	b9	NC	c9	NC
a10	NC	b10	NC	c10	NC
a11	NC	b11	NC	c11	NC
a12	NC	b12	NC	c12	NC
a13	NC	b13	NC	c13	NC
a14	NC	b14	NC	c14	NC
a15	NC	b15	NC	c15	NC
a16	NC	b16	NC	c16	NC
a17	NC	b17	NC	c17	NC
a18	NC	b18	NC	c18	NC
a19	NC	b19	NC	c19	NC
a20	NC	b20	NC	c20	NC
a21	NC	b21	NC	c21	NC
a22	NC	b22	NC	c22	NC
a23	NC	b23	NC	c23	NC
a24	NC	b24	NC	c24	NC
a25	NC	b25	NC	c25	NC
a26	NC	b26	NC	c26	NC
a27	NC	b27	NC	c27	NC
a28	NC	b28	NC	c28	NC
a29	NC	b29	NC	c29	NC
a30	NC	b30	NC	c30	NC
a31	NC	b31	NC	c31	NC
a32	NC	b32	NC	c32	NC

[1] This means it's not connected.