

Chip Replacement with IP and FPGAs: 68000 Processor Example

June 2011

Newton Abdalla, CAST, Inc.
Yuzo Ishikura, Spinnaker Systems

The value of semiconductor intellectual property (IP) in saving time and cost while developing complex new Systems-on-Chip (SoC) designs is well understood. But IP also has a valuable role when old systems need updating.

Many processor or controller-based systems produced 10 or 20 years ago are still going strong. But several of the popular processor chips from that era have become impossible to purchase—they have reached “End of Life”—so replacing worn out system boards or manufacturing new copies becomes difficult.

Designers tasked with extending the life cycle of such systems have two choices:

- Replace the obsolete processor chip with a currently-available processor chip, writing new code that retains compatibility with the rest of the system, or
- Develop your own plug-in, instruction set compatible chip replacement for the existing processor, using an IP core implemented in an FPGA.

Using a new processor chip is a good approach in some situations. But we have found that exact replacement using IP in an FPGA is the best approach for many customer projects.

The Case for Replacement with IP

Replacing an obsolete processor or microcontroller chip with one that’s still available seems attractive and is in fact the best choice for certain projects (more on this later). The biggest challenge here lies in coding the new processor to work with the rest of the existing system, as explained in this EE Times article:

The customer initially might have considered replacing the original 68HC11 hardware with a completely different processor, but this approach would have required replacing the application software. That would have been a daunting task, because the software was written in tight relation to 68HC11 instructions and internal peripherals. Consequently, switching to a new processor would have required considerable effort and time just for the software redesign.¹

The challenge of new programming is especially burdensome for products that require certification or other formal approval. As examples, consider that any

change in the software of an air flight control system in the US requires recertification by the Federal Aviation Administration (FAA), or in a medical device by the Food and Drug Administration (FDA). Note this comment in a discussion of processor replacements:

The original customer for this design makes air data computers, and projects demand to continue well beyond when the "obsolete part stock" quantities of the Z8000 will be around. Since the software for this system has to be FAA certified, changing even one line of code is horrendously expensive. (Monte Dalrymple²)

A variation on this approach is to take advantage of the extra capacity of a modern processor to run an emulation of the legacy device. This reuses the existing application software, but introduces new boot code and timing challenges. The effort to resolve these challenges can exceed the appropriate lifecycle extension budget, consuming any profit to be made in extending the product's life.

In contrast, simply replacing the obsolete processor part with a new FPGA device that's fully hardware and software compatible is usually significantly easier and less expensive.

Several CAST customers have found this approach to work well in chip replacement applications. Some examples:

- Sunplus Technology extended the market life of early Sega video game consoles by replacing their discontinued 68000 chip with a C68000 core in an FPGA.
- HAPA AG used the same C68000 core to continue the 15-year-old stepper motor control system in their pharmaceutical printers (see photos). This move also gave them physical space and an effective platform to make improvements to later generations.
- The advantages of still running the considerable software for Intel® 80C188EC processors (e.g., Windows 3.1) were retained by ThyssenKrupp Elevator using the C80188EC core and by Defense Science and Technology (DSTO) in Australia with an 80186EB core.
- Fabless semiconductor provider Innovasic Semiconductor cost-effectively developed new niche markets for 8051 and 80186 discrete chips implementing CAST controller IP.³
- A system manufacturer in Japan intends to keep critical 68000-based traffic operation control systems functioning for several more years by replacing the end-of-lifed chips with FPGAs implementing the C68000 core.

Another benefit of the IP replacement approach lies in the new foundation for product improvement using an FPGA provides. Once software compatibility is

proven, an FPGA with room for additional IP cores can present a great opportunity to improve that design. As written in a *Chip Design* article:

*Finally, end-of-life conditions may force a designer to consider replacement hardware. Modern FPGAs are often ideal replacements for end-of-life FPGAs, CPLDs, and ASSPs like PCI controllers and physical-layer interfaces. A single-chip FPGA solution on the printed-circuit board is attractive, as it eliminates the need for an additional configuration device. Another issue is the European Union's Restriction of Hazardous Substances (RoHS) directive, which is forcing system engineers to adopt modern lead-free devices.*⁴

Faster performance, better use of silicon, high-level programming languages, and modern debug tools help make it easier than ever to squeeze new features or quicker operation into an existing product or control system.

Choosing the Best Replacement Approach

Consider these issues when deciding between replacing an obsolete chip with a new discrete processor chip or with an IP core for the original processor.

- End User Product Life — How many more years do you expect to ship the product? The longer this life, the greater the chance your new processor chip will also become obsolete, and the more using IP makes sense.
- Software Code Language and Volume — How much assembler code must be rewritten to run with a new processor chip? If the system software is small and mostly written in C, for example, rewriting it for a new processor chip may be easier than using processor IP.
- Licensee's IC Units per Year — If the product's unit volume is very small (e.g., 10 or fewer each year), then the possible future revenue stream is unlikely to justify the expense of redesigning with a new processor.
- Number of Peripheral Circuits — Peripherals obtained with the original processor may also be nearing (or past) the end of their availability. If this is the case, and especially if there are many of them, then starting with a new processor and its modern peripherals probably makes the most sense.
- End-User Equipment Cost — Possible future revenue will rarely justify the costs of switching to a new processor for very inexpensive products, unless their expected annual unit volume is very high.
- Processor-Specific Chip and Programming Experience — If the original programmers of the legacy processor are no longer available then continuing to maintain it will be difficult, possibly even more difficult than switching to a newer processor.

- Experience using IP and FPGAs — If your design team has little experience using IP cores and FPGAs, then using a discrete processor chip is likely the better approach.

These factors should be weighed differently from application to application. For example, a long product lifetime of ten or more years shifts the decision towards using IP and an FPGA even if other factors suggest a new discrete processor chip makes sense (because a new discrete processor chip is also likely to go obsolete in that time).

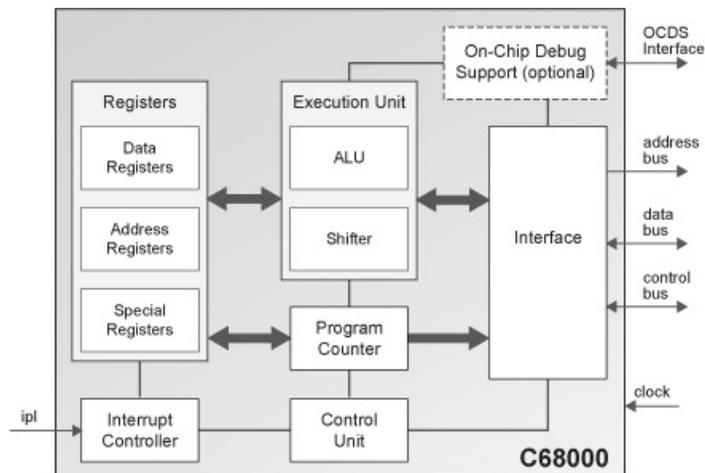
Using a Replacement Processor Core: the MC68000

The M68000 series of CISC processors began with the MC68000 released by Motorola in 1979. The MC68000 used 32-bit internal and 16-bit external interfaces, making it more advanced than the 8- and 16-bit processors popular at the time. Its approach of “forward compatibility” and early 32-bit instruction set made it competitive even against later 32-bit processors.

It was originally produced with a 3.5 micrometer HMOS technology. Smaller feature sizes and faster speeds followed, reaching 16.67 MHz with the 12F version in the late 1980s. Equivalent chips were “second sourced” by several other chip manufacturers, including Hitachi (HD68000), Rockwell (R68000), and Thomson (EF68000 / TS68000). A better CMOS version, called the 68HC000, began appearing from Hitachi, Motorola, and others in 1985. Later family members included the 68010, 68020, and 68030.

The 68000 was a dominant processor in its heyday. Noteworthy products based on it include the early Unix workstations from Sun and Apollo, pioneering personal computers including the Amiga, Atari ST, and Apple’s Lisa and Macintosh, and the first laser printers from Apple and HP. Second sourcing helped spur this 68000 adoption; the Amiga 500 computer, for example, was built using Motorola, Hitachi, Sygnetics, and Rockwell chips interchangeably.

Introduced in the spring of 2000, CAST’s C68000 IP core was one of the first such products available.⁵ The C68000 implemented in an FPGA works identically to the 68000 chip. It uses the same 16/32-bit architecture, runs 55 instructions, has 14 address modes, and includes interfaces to M68000 family peripherals.



In an improvement over the original, the core also supports IEEE1149.1 with a JTAG port. This lets designers use third-party commercial compiler/debugger tools.

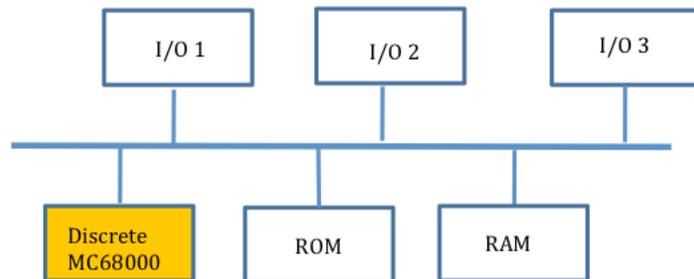
Planning A Replacement Chip Project

Our experience with dozens of customers successfully replacing obsolete chips with IP cores such as the C68000 suggests a two-step process is most effective:

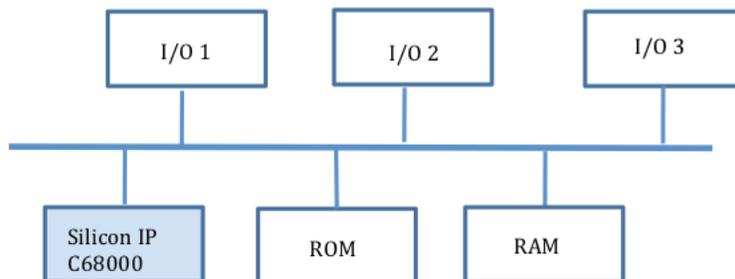
1. Verify that the core is software-compatible.
2. Merge other functions from the board into the FPGA.

A cost-effective way to conduct these steps is by using an FPGA in a commercial or custom platform board.

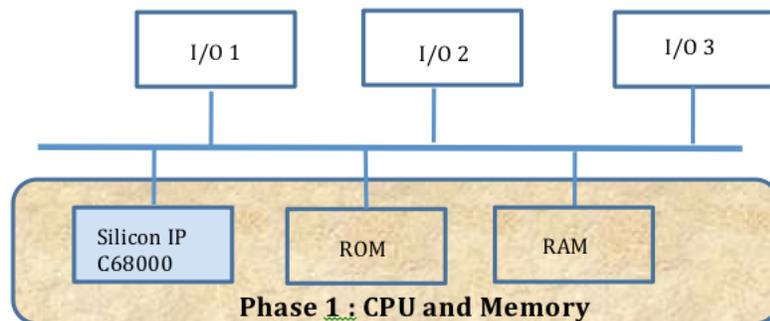
For example, consider this 68000-based system.



Conceptually, the IP core just replaces the discrete chip.

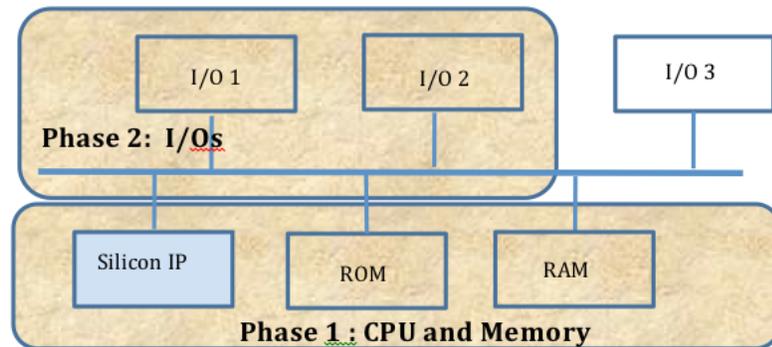


In practice, you'll also burn the system's ROM image into the FPGA, and controlling its RAM through the FPGA.



The system I/Os should be initialized by the bootstrap code in ROM, and the system start functioning normally, verifying that the software executes correctly with the IP core.

With software compatibility proven, a next step is to integrate additional functions to take advantage of any remaining capacity in the FPGA. You might, for example, find that I/O 1 and I/O 2 perform well in the FPGA.



You can evaluate the feasibility of this approach by combining your system hardware with a low-cost commercial FPGA evaluation board. Because the voltage is different between the new FPGA and the original MC68000 chip, you'll need to implement an adapter board with a level-shifter to plug in the FPGA. Additional pragmatic issues to be careful of include potential cross talk between the FPGA and the original hardware.

Other Chip Replacement Opportunities

We have focused here on MC68000 chip replacement, but similar benefits can be found in replacing several other obsoleted chips. For example, CAST currently carries these IP cores that are suitable for replacing older microcontrollers and other discrete chips:

- 8051 and 80251 Processors
- 6502 and 65C02 Processors
- 80186XL and 80186EC Microcontrollers
- 80188EC Microcontroller
- Z80 CPU
- 68000 16- and 32-bit versions
- 8254 Timer/Counter
- several UARTs
- 8237 and 82380 DMA Controllers
- 32025 Digital Signal Processors
- 387 Math Coprocessor

Family variations for any of these can also be effective solutions. When updating a 68020-based system, for example, you might buy and modify CAST's RTL 68000 core if you have considerable 68000 experience, or contract with CAST for a 68929 version if not.

Conclusions

We have looked at some of the considerations behind replacing an obsolete processor chip with an IP core in an FPGA rather than using a new discrete chip processor. This approach is conceptually simple, but is likely to present real-world challenges for most designs. For many specific systems and projects though, extending product longevity with IP is still the smartest and most economical approach

References

- ¹ Using FPGAs to avoid microprocessor obsolescence, John Swan and Tomek Krzyzak, EE Times, 3/5/2008
<http://www.eetimes.com/design/programmable-logic/4015159/Using-FPGAs-to-avoid-microprocessor-obsolescence?pageNumber=1>
- ² EDAboard discussion: <http://www.edaboard.co.uk/obsolete-processors-resurrected-in-fpgas-t414369.html>
- ³ Innovasic Website, <http://www.innovasic.com/products/ic/>
- ⁴ Legacy FPGA Designs Can Be Migrated to Achieve Better Performance, Troy Scott, Chip Design Magazine, June/July 2006
<http://chipdesignmag.com/display.php?articleId=566&issueId=17>
- ⁵ CAST Unveils New Cores at DATE 2000
March 27, 2000, http://www.cast-inc.com/company/archives/news/2000/032700_date2k-cores.html