Intellegently Expanding Microprocessor Connectivity Using Low-cost FPGAs

A Lattice Semiconductor White Paper

March 2013

Lattice Semiconductor
5555 Northeast Moore Ct.
Hillsboro, Oregon 97124 USA
Telephone: (503) 268-8000
www.latticesemi.com

Microprocessor Connectivity Challenges

Whether they be CPUs, micromicroprocessors or microcontrollers, microprocessors are an indispensable component in modern electronic system design. However, as systems become more complex and host a wider array of features and user interfaces, system architects using mid-range microprocessors in particular typically face three key challenges connecting the microprocessor, or microprocessors, they are using to the rest of their system:

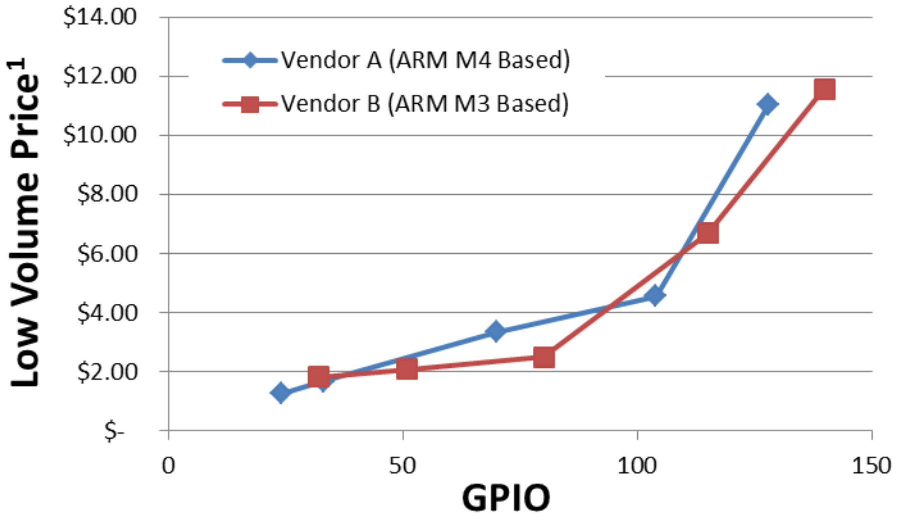
- Implementing more than 150 general purpose I/Os (GPIO)
- Finding cost effective solutions in the 100 to 150 GPIO range
- Matching available I/O peripherals with system needs

These challenges can easily be overcome with the use of FPGAs (field programmable gate arrays), which over the past 10 years have seen a significant reduction in cost and power consumption, making them ideal for a wide range of high-volume, low-cost applications including mobile.

Review of Microprocessor Connectivity

A review of mid-range microprocessors reveals three common connectivity characteristics. First, the maximum GPIO for most families is limited to between 100 and 150. Second, the pricing for per GPIO is not linear with a step increase in cost per GPIO typically occurring around 100 GPIO. Third, while most microprocessors provide some connectivity peripherals such as UART, SPI and I2C, the number of peripherals and available combinations is limited.

Figure 1 illustrates the typical range of GPIO options and associated low volume pricing, from a US distributor, for two common mid-range microprocessors, one ARM M3 based and the other ARM M4 based. Both families illustrate a knee in pricing at approximately 100 GPIO and limits to totals GPIO.

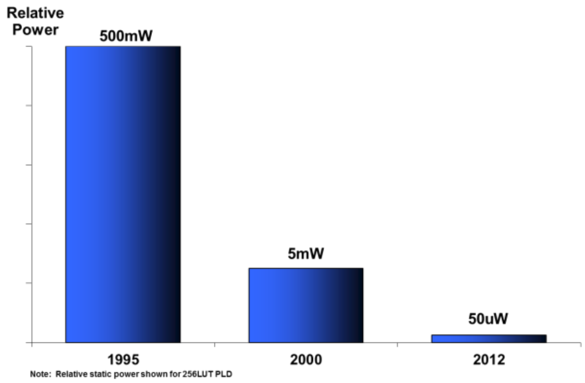


1. Based on US distributors web pricing, March 2013

Figure 1 – Mid-range Microcontroller IO Options

Review of FPGA Technology

For the last twenty years FPGA technology has provided a method to rapidly implement connectivity expansion, but the cost and power consumption of these devices has limited the applicability of this approach. However, the power and cost reductions associated with FPGAs dramatically illustrate Moore's law. In 1995, the cost of a device equivalent to one of today's 256 Look-up Table (LUT) capacity FPGAs was over $50. Today, the latest devices of this capacity are available for as little as $0.50. Improvements in power have been equally impressive. For the devices in the example given, the typical static power was 0.5W in 1995. This has been reduced to sub 50uW. These trends are illustrated in Figure 2.



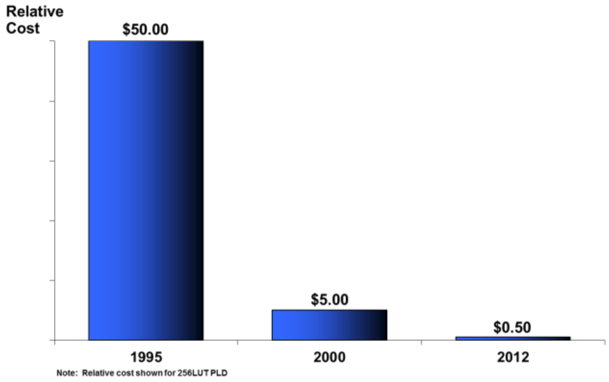
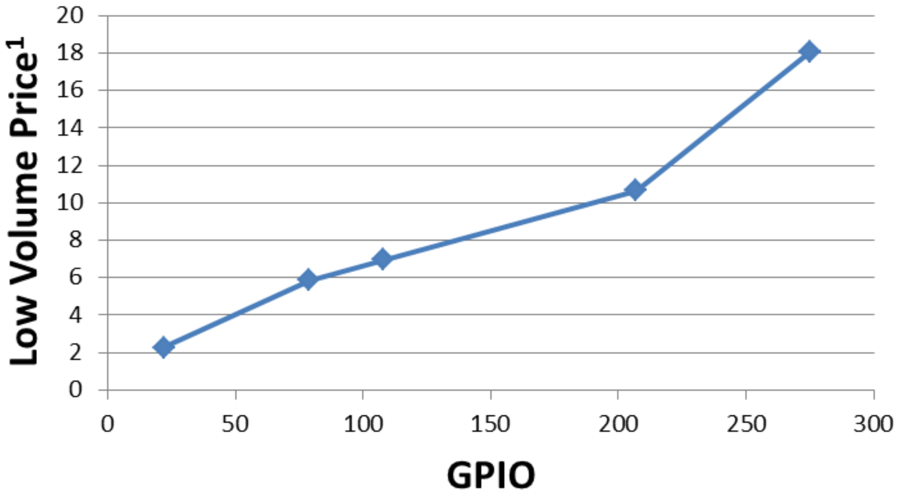


Figure 2 – FPGA Cost versus Power Trends

Low cost FPGAs provide a wide range of I/O counts, for example, the Lattice Ultra Low Density MachXO2 devices provide I/O counts from 22 to over 250. Pricing is fairly linear from over the I/O range as illustrated in Figure 3, which plots low volume pricing from a US based distributor.



1. Based on US distributor's web pricing, March 2013

Figure 3 – Low-cost FPGA I/O Characteristics

In addition to logic and I/O, the MachXO2 provides a number of other blocks that can be helpful in the creation of functions to expand micromicroprocessor connectivity. These include Embedded Block RAMs (EBRs), Phase Locked Loops (PLLs) and an Embedded Function Block (EBR) that provides hardened blocks such as, SPI, 2*I2C, timer counter and user flash memory.

Interface Implementation Approaches

The workhorses of microprocessor connectivity are UARTs, SPI and I2C. The vast majority of mid-range microprocessors are supplied with a limited number of these workhorse interfaces. Figure 4, indicates the number of these interfaces that are provided with the devices highlighted in the previous section.

Interface	Vendor A (ARM M4 Based)	Vendor B (ARM M3 Based)
I2C	1-2	1-2
SPI	1-3	1-2
UART	1-6	1-3

Figure 4 – Avaiable Interfaces on Microcontrollers

In the case were a design requires more interfaces than the microprocessor supplies, there are two common approaches a designer can take:

- Microprocessor control of GPIO (referred to as bit banging)
- Hardware implementation in an FPGA

The bit banging approach has three significant drawbacks. First, speed of interface operation can be limited to the low MHz given the difficulty for the microprocessor to rapidly control the state of the I/O pins. Second, microprocessor power consumption increases due to the additional workload and the reduction in time that the microprocessor can be in lower power sleep modes. Third, due to the limitations in timing accuracy inherent with a microprocessor implementing asynchronous UART type interfaces becomes impossible at all but the slowest of speeds.

A comprehensive FPGA implementation of any of these interfaces is possible in approximately 250 LUTs and optimized versions can be implemented in approximately 100 LUTs. Depending on the capacity of the FPGA device chosen, tens of additional interfaces in the appropriate mix can easily be implemented. Because of the hardware nature of FPGAs implementation speeds of many tens of MHz are easily attainable. The FPGA can also implement buffering and smart interrupts which can allow the microprocessor to spend more time in sleep mode, reducing overall system power.

Easy Implementation with Reference Designs

FPGA suppliers make the implenetation of microprocessor connectivity expansion realteively simple through the provision of reference examples that can be used as starting points for implementation. Four representative examples from Lattice Semiconductor are:

- RD1124 -- I2C Slave Peripheral Using Embedded Function Block
- RD1125 -- SPI Slave Peripheral Using Embedded Function Block
- RD1042 – Wishbone UART
- I2C Slave to SPI Master Bridge

Intelligent Connectivity Expansion Examples

By intelligently partitioning connectivity between microprocessors and FPGAs, designers can provide the connectivity required in their systems while at the same time reducing their overall implementation costs. This is illustrated in Figure 5 which uses the pricing data provided in previous sections to investigate the cost of implementation strategies that use a variety of FPGA and microcontroller resources. It can be seen that by maximizing the use of FPGAs for connectivity expansion costs can be reduced by between 24% and 34%.

Low-cost Low-power FPGAs Allow Rethinking of System Partitioning