

SHOULD YOU USE AN ASIC OR A STRUCTURED ASIC IN YOUR CONSUMER DESIGN? DON'T BOTHER WITH EITHER, SAY THE FPGA MAKERS; THERE'S A NEW CLASS OF LOW-COST PROGRAMMABLE PARTS COMPETING FOR THOSE SOCKETS.

Low-cost FPGAs move into consumer designs

In the design of products intended for high-volume manufacture, programmable logic has conventionally had a limited role. Designers frequently use lower-density chips to absorb small amounts of “glue” logic and carry out simpler stand-alone functions. But for designs using logic at the hundreds-of-thousands-of-gates level or above, and that will also enter volume production, ASIC has been the preferred option. The advantages of the programmable route are obvious and frequently repeated by the sector’s vendors. Benefits include lower development costs, no ASIC-style NREs (non-recurring engineering charges) or mask costs, flexibility to accommodate design changes right up to—and during—manufacture, and so on. The problem? They cost too much. The unit cost of programmable devices has simply been too high to be affordable in the average “consumer” product’s bill-of-materials.

The reason is no secret: whereas with an ASIC, only the logic that the circuit requires is built-to-order, in an FPGA it must be fashioned from a general-purpose structure. This structure has to have flexibility to allow it to be shaped as required—which makes it less area-efficient as an ASIC. Then, there is the overhead added to provide the programmability. Pass transistors to select the logic paths, memory cells to retain the logic configuration, and interconnect to provide all the possible paths through the chip - all add to the silicon area of the device, and force up the cost per chip.

Now, however, the latest generations of programmable parts are eroding this cost barrier, and the PLD vendors are stepping up their efforts to capture some of those mass-market applications. Already, they are claiming significant penetration of the “consumer” market.

Before looking into those questions, it’s worth noting that there is a further barrier to the adoption of programmable technology that is beginning to be breached. Sectors such as automotive have traditionally shied away from FPGAs because—aside from the cost, which is always a dominant factor in automotive build-costs—engineers have simply not trusted the chips. Perhaps because of the demanding environmental specifications that prevail, the automotive world has tended to be wary of programmables. Now that, too, seems to be changing. FPGAs have now built up a sufficient track record to establish their reliability, even with ultra-cautious automotive system designers. We probably will not see them in safety-critical or power-train functions any time soon, but the cost-plus-credibility equation is beginning to shift to the point where automotive designers will consider them.

SHRINKING SILICON IS KEY

What has led to this change? As with so much else in the semiconductor world, progress in process technology is the biggest single factor. Moving to process geometries of 0.13 micron and 90 nm has reduced the silicon area needed to build large FPGA logic arrays to the point that the chips become affordable. Of course, ASIC technologies (and other formats such as structured arrays) also scale in proportion and maintain their relative advantage. The difference in the current process generations is that the absolute size of the silicon die has shrunk to the point where other factors such as package cost account for a greater proportion of the final device price, and the differential due to die size matters less. An FPGA die might still be ten or more times the size of a functionally-equivalent ASIC die, but the

silicon cost has fallen as a proportion of the device cost, eroding the differential. Simultaneously, standard-cell ASIC engineering costs are going in the wrong direction - upwards.

Building on those facts, most of the programmable-device vendors have introduced low-cost versions of their complex parts, intended to capture those medium-volume board sites. None of them will claim to have the economics to challenge ASICs for the true mass-market products made in millions, but those are relatively scarce projects. For a range of projects that will see production volumes of tens of thousands up to a few hundred thousand, the programmable option looks increasingly attractive.

How to choose? The decision can come down to simple arithmetic. On one hand, figure the total cost of designing an ASIC: that is, the costs of taking the design through the (relatively expensive) tool chain and design flow. Add factors such as the cost of a mask set to produce the ASIC, and factor in the cost of all the silicon you will buy—estimate total production quantity times the projected unit cost of the ASIC, and the sum is more or less complete. The corresponding calculation for the FPGA features (usually) a lower-cost tool chain, lower development cost, no mask cost, and a similar calculation of lifetime volume times a (higher) device cost. To that, you must add the cost of a separate memory to store the device's configuration data (for an SRAM-based part) and/or the additional costs of programming the configuration data into each individual production unit.

What begins as a straightforward spreadsheet exercise is complicated by factors such as the growing presence in the market of the structured-ASIC offerings from numerous vendors, which will generate its own set of figures for a lifetime use calculation. Also, you need to figure in the very price curve that is leading to this analysis in the first place. If your project will be in production for more than about a year, you can reasonably expect to be paying less for the parts later in the production run than at the start.

LOW-COST CHIPS SQUARE UP

To capture the medium-volume applications, most of the programmable-logic vendors now have a low-cost series of devices that is evolving in parallel with their flagship product range. Xilinx

Figure 1



90-nm silicon technology underpins the Spartan 3E FPGA series - Xilinx intends the chips for designs dominated by logic gate count

matches its Virtex line with the lower-cost Spartan series; Altera pairs Stratix with its Cyclone parts; and so on. In general, what is on offer is a part with slightly lower logic flexibility, and with a somewhat reduced feature set. All of the parts offer clock speeds in the hundreds of MHz, generally more than adequate for consumer-type applications. The vendors have done a market analysis and aimed to reduce the chips' feature sets to cut back on silicon area while still addressing the majority of applications. In some cases, I/O options will be reduced. The list of voltage levels and signalling protocols addressed by the I/O cells on a Virtex or Stratix part, for example, is formidable, and that versatility is gained at the cost of, once again, silicon area.

The area occupied by an I/O cell does not scale down linearly as silicon process generations roll on—while core voltages decrease, chip designers must maintain a certain minimum feature size in the I/O to allow the cell to continue to handle older bus voltages, such as 3.3V-levels. Trimming the I/O feature set to address the bulk, but not all, of the likely applications is another route for the chip designer to save area and cost. There can also be a trade-off for the chip designer. For a given amount of logic resource, the "right" number of I/Os (based on market needs) placed around the perimeter of a chip may enclose an area greater than is needed to build that number of logic gates in the most advanced technology.

What you do not get with these low-cost device series is older technology. It is somewhat paradoxical, but the concept

only works if the latest, leading-edge processes are used to build the chips.

The most recent introduction from Xilinx is the Spartan 3E series of parts (Figure 1), for which Xilinx has used its 90-nm technology, and which is intended for "gate-centric" designs and optimised for lowest cost vs. gate count. The Spartan 3 series, previously announced, is the other side of the I/O trade-off mentioned above, and is optimised for lowest cost per I/O. There will be five devices in the family, and they are aimed at digital consumer applications, ranging from 100 k gates to 1.6 million gates. Projected pricing looking forward to late 2006, and in half-million quantities, allows Xilinx to quote "100k gates for under \$2" and "1.2 million gates for under \$9". The first device into sampling (now) is the smallest part, the XC3S100E, which has 66 I/Os, 87k of block and distributed RAM, four 325 MHz embedded multipliers and two digital clock managers. If a design needs an embedded microprocessor, an 8-bit PicoBlaze core will use 10% of the logic cells, or an 32-bit MicroBlaze core will take up 49%. The four hardware multipliers can yield 1.3 Gmac/sec performance without using other logic resources.

Spartan3E devices have an I/O feature set that Xilinx says it has selected for the consumer electronics market, including support for 18 common I/O standards comprising PCI 64/66, PCI-X 100, RS485, and mini-LVDS as well as interfaces to commodity-priced DDR memories. The intention is to reduce the need for other discrete devices. There is support for commodity serial (SPI) and byte-wide, parallel flash memory for configuration.

As an SRAM-based technology, the product designer must provide a configuration memory, but Xilinx claims that many of the consumer applications it sees have sufficient unused flash available to host the configuration files for (effectively) no incremental cost.

At Xilinx, European Marketing Director Alan Matthews sees Spartan devices making an impact in designs that run in the 100,000 production-volume region. Among the applications that the FPGA manufacturers quote as break-throughs into volume markets, one that they frequently cite is that of flat-panel television, and Xilinx is no exception. In the video signal path of an LCD or plasma television, there is a need for signal processing to pre-condition the video signal to match the characteristics of the display medium. Appropriate digital filtering can have a marked effect on the perceived quality of the picture. This is ideal territory for the FPGA. The designer needs a direct implementation of a filtering or signal processing algorithm, and the implementation needs to run at “hardware” speeds. Furthermore, it is an application where small adjustments to the filtering algorithm may be required to upgrade performance, or even to accommodate the characteristics of different display panels. All of these factors suit the FPGA - you can make refinements to the signal processing chain late into development and while the product is in production, without the impact of an ASIC re-design.

FITTING INTO THE BOM

Whether or not the large-screen LCD or plasma television is a true case of the “consumerisation” of programmable logic is open to question. One could view it as an example where the bill-of-materials budget has widened to embrace the FPGA, as much as the FPGA having dropped in price to meet consumer-product demands. As the price of high-density programmable parts has declined through the hundreds-of-euros level, to tens-of-euros and below, they have become feasible for a wider selection of designs. At the same time, the advent of large-screen flat-panel formats has driven the selling price of a television sharply higher than was traditionally the case. So, there is more headroom in the bill-of-materials for a non-traditional part, especially one that contributes time-to-market advantage.

Xilinx’ Matthews acknowledges that low-cost flat-panel TVs will be built in millions and will likely still shift to an ASIC solution. For Xilinx, Matthews says that Europe accounts for most of the company’s penetration into the automotive market with FPGAs. Applications are typically focussed around telematics: although automotive is conventionally a very high volume market, some of the options that can be ordered on high-end models fit neatly into the around-100,000 bracket that FPGAs can address. With many incremental changes from one model year to the next, and on even shorter timescales, the FPGA also allows incremental changes to be made to on-board automotive system designs without ASIC re-spins. A full automotive temperature range part is planned.

THE NAÏVE FPGA USER

Matthews confirms that the primary issue in converting the consumer/automotive designer to programmables is cost—followed by a lack of knowledge of the parts and their design flow. Especially in automotive, where programmables have previously been avoided, an unusual situation can arise. The programmable vendors can find themselves dealing with a very competent design team—perhaps one that has undertaken complex ASIC designs—that is at the same time completely new to FPGAs. Introduction via a standard development board, pre-configured IP and a standard, language-based tool flow would be a standard offering from any of the sector’s vendors.

Altera’s Cyclone product line is currently on the Cyclone 2 generation, although most of the design wins now in production are on the first series. The first of the Cyclone 2 parts, the EP2C35, is now being delivered. It too is built in 90-nm technology, and the family spans 4,608 to 68,416 logic elements, offers up to 150 18×18 multipliers and up to 1.1 Mbit of on-chip memory. Altera also offers an embedded microcontroller core, its own Nios design; a Nios core can cost as little as 35 cents measured by the logic it occupies.

Technical Marketing Manager at Altera, Pat Mead sees a polarisation among consumer designers who are embarking on using FPGAs; they have previously either worked with standard-cell ASICs to provide high density logic, or they have focussed on ASSPs and processor-based solutions. Mead notes that one factor contributing to the willingness of consumer-product designers to consider FPGAs is that of component obsolescence. The IP in an FPGA can be configured into a new generation of hardware if required: an obsolete ASSP or μ C is a different problem. Altera cites a number of design wins in medium/high volume products, some of which are firmly in the consumer space.

An example is Hirschmann’s TV antenna unit for analogue and digital reception in cars (**Figure 2**). The unit uses a Cyclone part to implement the DSP functions that the TV receiver chain requires; it employs a Nios processor core to manage a triple-diversity antenna system. Altera also cites the flat-panel TV as

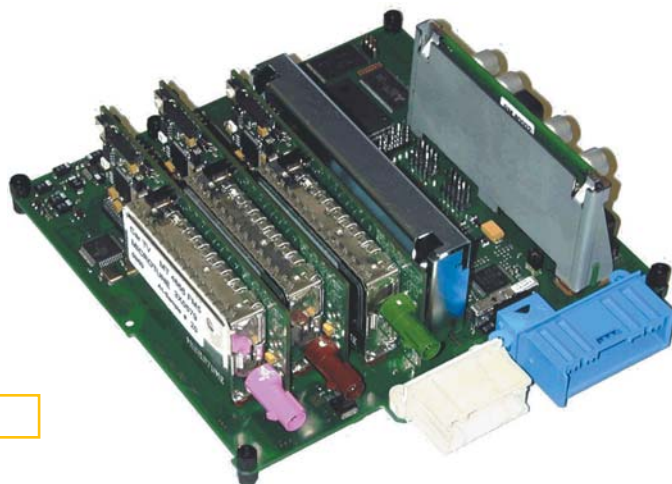


Figure 2

Altera’s Cyclone FPGAs are in production in this diversity TV receiver by Hirschmann, providing both signal-path DSP and control logic

Figure 3



Actel believes that the flash technology on which its ProASIC3/E devices are built will provide a winning advantage in silicon area efficiency

a design example, their quoted win being for Thomson. Mead says that the parts will comfortably go into a bill-of-materials that is down to 100 € or even less.

FLASH FPGAS

The most recent release from Actel takes a different approach, using flash-memory-based parts as the basis for a low-cost FPGA offering. The company quotes ProASIC3/E (Figure 3) as offering 600k to 3 million gates, with up to 504 kb of SRAM and 604 I/Os. Operation is said to be at up to 350 MHz. The lowest-priced device, offering 30k “system gates” sells in volume at \$1.50. Actel says that flash has a natural advantage for low-cost FPGAs in that the configuration memory cell on the device is much smaller than with SRAM; with internal programming, the devices are live at power-up without the need to load a configuration file from an external device.

Actel believes that in the consumer space, design security is a major issue. There is a risk that the configuration file for a key device can be stolen, and along with it, the IP it embodies. In an SRAM-based device, the data is vulnerable as it is stored off-chip in a configuration memory. The conventional way to deal with this, now virtually standard on complex SRAM-based parts, is to encrypt that data stream. However, that does not protect against theft by design over-run. In today’s commercial environment, companies contract-out consumer product manufacture to a plant that may be on the other side of the world. There may

even be a chain of sub-contract relationships separating design owner from manufacturing—the fear is that a few thousand “grey” units of the product will be covertly produced to the same design, effectively stealing market share and profit. Actel’s solution to this is to use a flash ROM on board the chip as a security measure. Either Actel or its customer can place an AES-encryption key securely in this memory, and the devices are shipped to manufacture. The “untrusted” manufacturer loads an unencrypted bit stream into the part; only those chips with the correct 128-bit key will configure and operate. In-field upgrades are also possible using this method of file distribution. Keys can be programmed on a batch, or even an individual device, basis.

For consumer products, designers can employ the same security features to give hardware-based control of subscription services, only enabling certain functions when subscriptions are verified. “Flash will dominate the value-based FPGA market sector”, declares Actel’s Director of flash marketing, Martin Mason.

Also introducing a flash-based offering for volume market use is Lattice, with the LatticeXP. These devices employ a flash array for non-volatile storage of the configuration data, but they also have an SRAM-based structure in the logic array itself. Data is copied from flash to SRAM at power-up in under 1 msec; the paths by which this takes place are not accessible from the chip’s external connections, providing protection of the bitstream. External circuitry can also configure the SRAM array directly, adding possible variations in the way the chips are used. The architecture is based on 4-input look-up tables, and the device family spans 3k to 20k LUTs, 25% of which have distributed memory associated with them. A 10k LUT device will be priced

under \$15 at 250k quantity, in 2006. That device will sample shortly and will include 216 kb of embedded block RAM. As with other devices in this sector, the devices support the most popular interface standards to address the largest possible applications base consistent with controlling silicon cost. They also support DDR DRAM memory access, for low overall system costs.

IN-SYSTEM PROGRAMMING

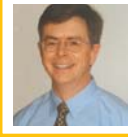
Lattice has always been a proponent of in-system programming techniques. XP parts support programming via a microprocessor interface, known as sysCONFIG; or via JTAG. An additional cost that designers must allow for is the time taken to carry out in-system programming, to load the configuration file into the completed product. This can

be done as part of in-circuit production board test, but time on a full-featured board tester can cost 1 € per second: several seconds spent programming a chip will materially alter the spreadsheet computation outlined above. Suppliers such as Asset Intertech (www.asset-intertech.com) offer a route to

use of a low-cost PC-based programming system that can dramatically cut this cost.

A key development in this area is the IEEE 1532 Standard for Boundary-Scan-based In System Configuration of Programmable Devices, which is now at the final ballot stage (<http://grouper.ieee.org/groups/1532/>). 1532 defines a standard method for accessing and configuring programmable devices that already support IEEE 1149 (“JTAG”). 1532 is concerned with all aspects of configuring programmable devices during the manufacturing process, but, in particular, it proposes a method doing so concurrently. This could potentially save significant time in the manufacturing cycle, and further promote the adoption of programmables in mass-market applications. □

You can reach Editor
Graham Prophet at
+44-118-935-1650, fax
+44-118-935-1670,
e-mail prophet@reed-business.com



FOR MORE INFORMATION...

For a comprehensive index of companies active in the field of programmable logic, see EDN’s annual Programmable Logic Directory at <http://www.edn.com/article/CA421504.html>.