FPGA Configuration via USB

Fred Dart, CEO of Future Devices Technology International (www.ftdichip.com)

With the electronics sector getting ever more cutthroat, and profit margins being squeezed, original equipment manufacturers (OEMs) now face heightening challenges which threaten their very livelihoods. This is forcing them to look at more innovative ways of developing new products that will give them the edge over rival outfits, bolstering their revenues and allowing them to grow their market share. In many scenarios application specific integrated circuits (ASICs) have already been superseded by more versatile field programmable gate arrays (FPGAs). The following article examines how best to go about configuring these devices.

Traditionally OEM electronic circuits always had at their centre some form of ASIC. Such devices have been the stalwart of the industry for many decades now, offering high running speeds, low unit costs, and strong power efficiency. As a result they present a favourable value proposition to many electronic designs, especially when production volumes are very large and price points need to be very competitive. However, there are other characteristics that define these devices which can be less attractive under certain circumstances and as a consequence another type of semiconductor technology can at times be called for.

The standard cell approach for ASIC design is highly suited to electronic systems where the market is already assured and there is stability in the end products functionality (games consoles, cell phone handsets and automotive electronics systems all therefore tend to make use of custom ASIC devices), but if either of these factors is missing then an element of risk is brought into the equation. The main underlying reason for this is that in order to benefit from the cutting edge performance and cost effectiveness of ASIC solutions the OEM must first make a major upfront investment. ASIC vendors will charge for non-recurring engineering (NRE) costs, such as creating photolithographic masks, carrying out debug activities, etc.

For OEMs releasing pioneering products in to a sector that has yet to mature, this is unlikely to be the best strategy. Here they will be more comfortable testing the

market first. Likewise cash-strapped start up companies who lack the available budget required will be unable to follow the standard cell route. Should the product need to be upgraded over the course of its lifecycle then once again the large expense involved in creating an ASIC is very hard to justify. Finally, if the OEM has a series of products which all stem from a single platform, in an attempt to maximize development resources, ASICs do not make commercial sense.

These issues are pushing an increasing proportion of OEMs towards programmable logic. Though still only representing a little over 30% of the total ASIC market, which is worth a staggering $14 billion (according to 2010 figures by Semico Research), it is important to note that the number of programmable logic design starts now outstrips ASIC designs starts..

By employing an FPGA as opposed to a fixed function ASIC, engineers have the ability to upgrade the design via firmware during the product cycle, resulting in greater flexibility. This proves to be a very attractive trait for OEMs in the set top box or telecoms infrastructure spheres, where standards and protocols change frequently. It is something that can also be considered advantageous when prototyping a design that then goes on to be hardwired at a later stage.

FPGAs give engineering teams the opportunity to validate their design concept and iron out any difficulties before major financial outlays are called for. Furthermore, with these devices OEMS can alter a generic platform to create different models (standard, advanced, deluxe, etc). Another plus point of FPGAs is that they enable reconfiguration of the target application so that different operating modes can be selected. Finally they offer OEMs much faster time to market than ASICs, so if there is only a tight window of opportunity to exploit, then FPGAs could be the best option.

FPGA Configuration Techniques

Though implementing FPGA-based system designs, as opposed to ASIC solutions, is becoming more attractive to many OEM design teams it is not without its difficulties. A major obstacle is that the configuration data normally has to be stored on an additional Flash memory IC or a discrete EEPROM for downloading every time the FPGA is powered up, as its on-chip SRAM memory is volatile and can’t hold data when it is not powered. This increases the bill of materials, adds additional complexity to the design process and potentially extends development time – impacting on all of the reasons for originally choosing programmable logic over standard cell chips.

An alternative approach which makes use of ‘on-the-fly’ configuration of the programmable logic IC should also be considered. Normally this requires the use of reasonably expensive pieces of hardware, but by carrying out configuration directly through the universal serial bus (USB) the cost involved can be kept to a minimum.

FPGA Configuration via USB

FPGA configuration via USB is both more subtle and highly cost-effective, as well as being faster to implement. This is likely to prove particularly beneficial in applications where a number of different configurations are being implemented

(e.g. an Ethernet switch) as it is possible to hold different pieces of firmware on the one piece of hardware, as opposed to needing a separate EEPROM for each configuration.

The Morph-IC-II development module from FTDI (as shown in Figure 1) is targeted at system designs where hardware is reconfigured (or “morphed”) over USB. It combines the company’s high-performance FT2232H USB-to-multi-purpose UART/FIFO IC with an Altera Cyclone-II FPGA device. The low cost module has a total of 4,608 embedded programmable logic elements, equating to 80,000 logic gates. This gives design engineers plenty of headroom for implementing their design.

The module communicates with the host PC through the FT2232H, which supports the USB 2.0 Hi-Speed communication standard (480Mbps data rate). It allows engineers to program and interact with the FPGA through Altera’s Quartus II software offering, with a sample application being provided to transfer created files to it from the PC. A set of 4 connectors on the module give access to all the FPGA IOs plus a JTAG header.

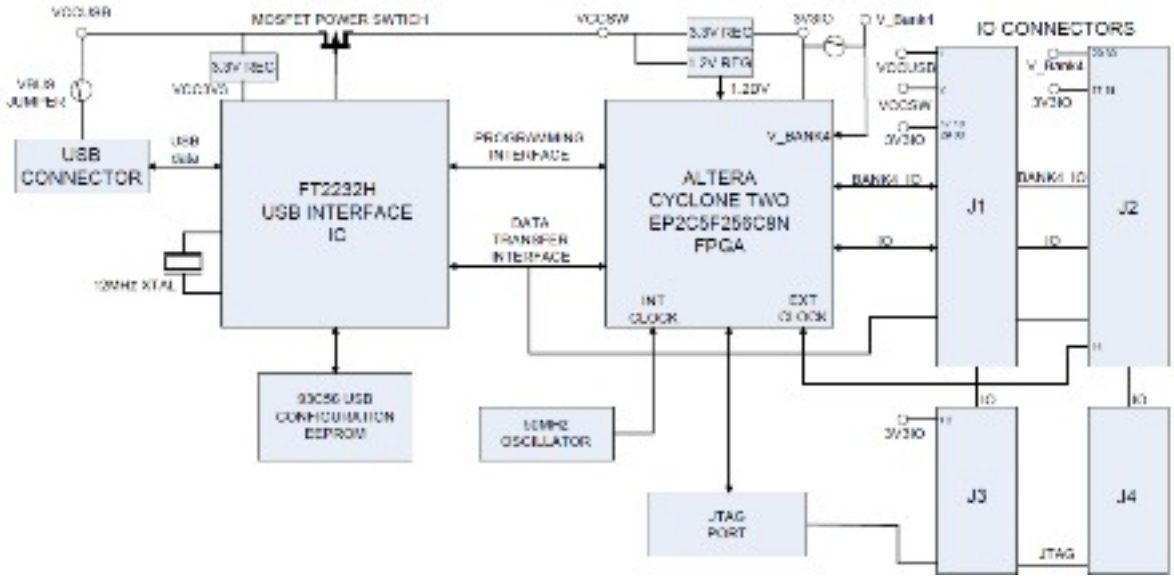

Figure 1: Hardware configuration of Morph-IC-II module

Through the FT2232H it is possible to create a communications bridge between the PC and FPGA. This bridge is split up into 2 channels - a programming interface channel which configures the FPGA through Altera’s Passive Serial interface to take care of the application and an 8 bit parallel FIFO communications interface channel which is used to transfer data, either synchronously or asynchronously.

The Morph-IC-II module utilises the functionality of the proprietary Multi-Protocol Synchronous Serial Engine (MPSSE) architecture to adapt to the Altera Passive Serial (PS) interface. The architecture was developed to allow different synchronous serial protocols to be configured on any available data channel. Once the FPGA has been configured, the MPSSE can be configured to operate as general purpose IO (GPIO) pins. As data formatting and clock synchronisation can be configured through the MPSSE in a variety of different ways it can address a broad number of design requirements. The MorphLD FPGA loader utility, that forms part of the Morph-

IC-II package, can be used to load raw binary files generated when Quartus-II compiles a HDL project.

Hardware reconfiguration via USB means that the FPGA specified only requires enough logic gates to deal with the most complex discrete function it will need to implement, it does not need the capability to cover all potential functions. This maximises silicon utilisation so that a lower cost device can be used. By being multi protocol in nature, the MPSSE architecture allows communication with many different types of synchronous serial devices (such as those based on SPI, I2C and JTAG, as well as GPIO signals that have pre-defined functions) to a USB port. Using the Morph-IC-II the reconfiguration process can be done rapidly, normally taking less than 0.1s to complete.

In conclusion, despite its far greater monetization, the standard cell market is at best static while the popularity of FPGAs in system designs continues to grow at a strong rate. From consumer to broadcast applications and point of sales systems to home automation equipment, a large proportion of electronic designs now rely on the flexibility enabled by using programmable logic rather than a hardwired methodology. This approach is suitable for OEM development teams prototyping their design concepts before they are hardwired into an ASIC, carrying out one off FPGA design projects (such as in research institutes or academic establishments), addressing low/medium volume markets, address markets where the development time needs to be kept very short, or in order to serve as a stepping stone to move on to an ASIC once higher volume business has been secured. Using USB to configure the FPGA, as described in this article, rather than having to resort to other means shows itself to be both economically viable and easy to accomplish.

About FTDI
Future Technology Devices International (FTDI) specialises in the design and supply of silicon and software solutions for the Universal Serial Bus (USB). FTDI offers a simple route to USB migration by combining easy-to-implement IC devices with proven, ready-to-use, royalty-free USB firmware and driver software. The company’s single and multi-channel USB peripheral devices come with an easy-to-use UART or FIFO interface. These popular devices can be used in legacy USB-to-RS232/RS422 converter applications or to quickly interface an MCU, PLD, or FPGA to USB. A wide range of evaluation kits and modules are available to evaluate FTDI’s silicon prior to design-in.

Vinculum is FTDI’s brand name for a range of USB Host/Slave controller ICs that provide easy implementation of USB Host controller functionality within products and use FTDI's tried and tested firmware to significantly reduce development costs and time to market.

FTDI is a fab-less semiconductor company headquartered in Glasgow, UK, and has regional sales offices in Oregon, USA, Shanghai, China and Taipei, Taiwan. More information is available at http://www.ftdichip.com [1]

Source URL (retrieved on *12/09/2013 - 7:05am*):
http://www.ecnmag.com/blogs/2011/07/fpga-configuration-usb

Links:
[1] http://www.ftdichip.com