Innovative Integration
... real time solutions!

DSP
Data Acquisition
Embedded Control

FMC Interoperability Considerations

March 8, 2013

Use of FMC modules with hosts manufactured supplied by multiple vendors results in portability and interoperability considerations

Introduction

I/O mezzanines promise flexibility and a reduced system footprint. Most mezzanines use standard bus-based interfaces or a proprietary format. Aimed at FPGA centric solutions, the FMC (FPGA Mezzanine Card- ANSI VITA 57.1), is different. In the same way as one might strip out unnecessary weight in a car aimed for racing, the FMC is a performance solution that strips away unnecessary generic interfaces for direct FPGA driven I/O. But that requires knowledge to achieve the desired performance and to ensure the host and FMC module will work well together. This paper outlines some of the considerations in order to assess and the ensure that the host and module will integrate.

FMCs compared to the XMC Mezzanine Format

The advantage of the XMC mezzanine format is that the electrical interface is self-contained. Each module typically contains a processor or FPGA directly controlling onboard I/O resources. An XMC module is a mechanical variation of the ubiquitous PCI Express (PCIe) card, present in most desktop PCs. Such cards may be mixed and matched with relative ease under most modern operating systems. At the electrical level, assuming correct voltage matching (and Innovative cards are voltage tolerant), XMC cards will readily operate in any XMC slot. Software presents the usual issues in that an appropriate driver is required for the operating system with some dependencies on the specific platform, as defined by its BSP, BARs (Base Address Registers) and interrupt routing. However, FMC modules provide I/O in isolation which presents integration challenges, but significant advantages as well.

What are FMCs?

New generations of FPGAs present developers with a level of processing performance and potential I/O bandwidth that cannot easily be matched by conventional CPU configurations. The FPGA mezzanine card directly addresses the challenges of FPGA I/O by solving the dual problem of how to maximize I/O bandwidth while still being able to change the I/O functionality. The FMC format offers an elegant solution because they focus exclusively on I/O devices, such as ADCs, DACs or transceivers.

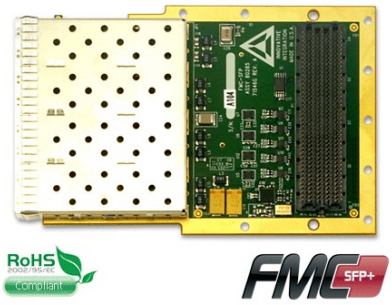

Illustration 1: Typical FMC Module

FMC modules have no on-board processors, bus interfaces or bridges, such as PCIe as required by the XMC module format. FMC modules directly connect the I/O devices to an FPGA located on the host carrier.The direct linking of physical I/O devices on one card (the FMC mezzanine) to the FPGA on another (the host), means that immediate interoperability between host cards with no software or firmware code

changes is highly improbable. However, the advantages of FMCs for high performance applications are considerable, including high bandwidth, low latency interfaces, lower power and more I/O real estate. At around half the length of a PMC, FMCs are small, but not at the expense of functionality because FMCs don't need bridges, memories, etc.The speed and number of connections that an FMC format module uses, together with direct FPGA to I/O devices means that the FMC format is particularly suited to applications benefiting from multi-Gbyte per second I/O with low latency. Examples of such applications are direct RF I/O, radar, Signals Intelligence (SigInt), satellite communications and Electronic Counter-Measures (ECM).

Variable Number of Connections

The FMC specification does not define a generic interface. Instead it defines a maximum number of FPGA connections; the maximum number of connections that the host FPGA has to connect to the analog to digital converter chips, for example. And because only the maximum number of connections is defined, by definition not all FMCs will have the same number of connections. This is also true for the FMC host; not all hosts will provide the same number of FPGA connections. This is a practical tradeoff because the FMC specification allows for up to 160 connections, and some designs simply cannot afford this volume of FPGA connections to be tied up to the FMC site – and two FMC sites connected to the same FPGA could be double the resource. Even for large FPGA packages, this is a huge number of links.

Even though the number of FMC-to-host connections is not defined, but left up to the individual host and FMC requirements, the order in which the links are established is not. Connections are assigned to the FMC starting at a common point and are added in a defined manner with no gaps in the sequence. For example, two FMC hosts which provide 140 connections each, will use the same 140 pins. If, in another example one host uses 135 connections and another uses 155, both hosts will use occupy the same 135 pins but the larger host will just have more. This ensures there is control between hosts such that if an FMC requires a certain number of connections and a hosts advertises the appropriate number of connections, then the connectivity will be provided without the risk of some connections being missing.

Vendor Tools

The FPGA tools environment is also critical. When a single vendor supplies both the host FPGA processor and the FMC card will be supplied with cohesive software/HDL optimized for the low level connectivity of the combination. Sometimes it may be preferable to use an FMC and a host card from different vendors. The code provided by one vendor will either be in the form of HDL abstracted to a generic FPGA host, (which will need further host specific integration) or a fully integrated host-specific FPGA combination. Either way, mixing product from two vendors is likely to need some integration development work. At its simplest, this may be re-associating the exact FPGA pinout definitions because no two vendors are likely to have chosen exactly the same FPGA pins to the FMC connector – even though the FMC connector may have the same number of connections.

It's not very common, but as FMCs excel in high speed I/O, some FMC hosts do not have a fast enough FPGA or architecture to support the I/O and bandwidth to subsequently move the data to another processor or off-board. The first issue is related to FPGA speed grades. If the FPGA is large, then sometimes only the slowest speed grade is available. This might provide challenges to meet HDL timing closure. In this case it is an advantage to match the FMC and host from a common vendor to ensure that the combination will inter-operate. The back-end architecture is

simply down to the application requirement, but as some FMCs can generate as much as 10Gbytes of data, the overall architecture may be important.

HPC or LPC?

The maximum number of standard connections to an FMC is 160 signals. But that is for High Pin Count (HPC) variants. Low Pin Count (LPC) is a sub-set allowing up to 80 signals. HPC and LPC FMCs are mechanically interchangeable, but with different sized connectors.

Mechanical

Mechanical incompatibilities with FMCs are rare (such as the double width module), but other issues such as those supporting different environmental levels may require further scrutiny. The FMC specification allows for different profiles depending whether or not front panel I/O is required, and additional areas for conduction-cooling. Three regions are defined within the FMC specification (see Figure 2). The regions are not areas simply reserved for a particular purpose, but are optionally omitted.

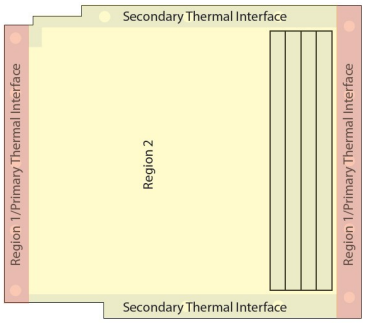

Illustration 2: FMC Outline

Region 1 is for I/O or as a thermal interface for a rugged solution. Region 3 is an optional rugged thermal/mechanical interface. The majority of FMCs will use regions 1 and 2. Conduction-cooled cards will usually require all three regions. FMCs may also use choose to populate the area shown as secondary thermal interface (see Figure 2). However, such cards cannot be fitted to hosts assuming this area to be a thermal interface, as components in this area will cause mechanical interference. Consequently, it may be advisable for even commercial-only (non-rugged) cards to not use the area defined for secondary thermal interface, due to possible interoperability issues. However, some FMC hosts may provide removable ribs in recognition of this possibility. Region 1 use for conduction-cooled hosts, if required for I/O, is likely to require mechanical modification for the host in order to provide I/O through the front panel/stiffening rib. In consideration of regions 1, 2 and 3 together with the secondary thermal interface, all FMCs will be mechanically compatible with all commercial FMC hosts. However, care needs to be taken with rugged FMC hosts, especially conduction-cooled hosts.

Considerations

As FMCs continue to increase in popularity – thanks to their high bandwidth, low latency interfaces, lower power and increased I/O real-estate – it's important to be familiar with the differences between different classes of FMC cards so that interoperability can be maximized and the benefits of these flexible, compact I/O cards optimized. Following is a checklist of key point that should be considered for interoperability of FMCs and their hosts:

Electrical/Mechanical

- What is the FMC stacking height? This will usually be 8.5mm, but could be as much as 10mm

- Consider the type of FMC and host in primary & secondary thermal interfaces and whether regions 1 & 3 are required.
- Is the requirement conduction-cooled? The front panel I/O nature of FMC would require conduction-cooled hosts to be adapted accordingly.
- Is the FMC HPC or LPC?
- Does the host support HPC or LPC?
- Assuming the FMC HPC/LPC match, does the host provide sufficient connectivity to fully support the FMC?
- Is the host FPGA fast enough to support the FMC?

Software/HDL

- Is a fully integrated solution required?
- Is example HDL code sufficient?

There are often interoperability issues with mezzanines and hosts, and is not particular to FMCs. However, the nature FMCs means that it worth doing some additional checks. FMCs are very powerful and can form the basis of an elegant solution, but it is essential to perform due diligence.

When mating an third-party FMC to a host, intimate familiarity with the host electrical and mechanical design as well as it's development tool-chain are essential. Consequently, it is typically most efficient for the host carrier vendor to add support for an FMC as opposed to having an FMC vendor perform attempt this work. Feel free to contact Innovative Integration to discuss specific FMC module/host carrier combinations. Our engineering team is available to perform or assist in the engineering tasks involved in mating third-party FMCs to our host carriers billed on a NRE time and material basis. Often, the custom firmware, software and qualification testing needed to add support for a third-party FMC module to an Innovative host requires 140-200 engineering hours.

Conclusions

The FMC format provides a very powerful solution to high-end I/O requirements. However, use of the FMC does require more questions to be asked for successful implementation than merely, “does a driver exist for the right operating system?”. FMCs offer a raw, and yet elegant, low level interface unencumbered by the more usual bus interfaces. This is the key to their power. Some additional due diligence means that power can be harnessed to great effect – and include electrical, mechanical and software/HDL considerations.

Illustration 3: Innovative SBC-K7 FMC embedded PC/FMC carrier