A Versatile USB-based Control System for Instrumentation in a Magnetic Field

D. Foreman[1]**, G. Kolansky**[1]**, and D. I. Hoult**[1]
[1]National Research Council Institute for Biodiagnostics, Winnipeg, MB, Canada

Introduction

We have recently built a Cartesian feedback (CF) magnetic resonance transceiver that functions in a high magnetic field at the entrance to the magnet bore. This advanced instrument is digitally controlled and has a large number of parallel control lines. Passing these numerous lines through the "penetration panel" of the MRI suite's shielding to a computer that is, perforce, outside the magnetic field poses significant problems of complexity, cross-talk, interference and speed reduction, and these problems will worsen as more CF transceivers are added to operate phased-array transmitting and receiving coils. Serial data transmission is the obvious solution, for the number of lines passing through the penetration panel is then greatly reduced; further, fibre-optic or coaxial cables are inherently radiation-free and the risk of interference to the MRI system is minimised. However, Universal Serial Bus (USB) extender hardware relies heavily on many ferrite-based components. Their replacement with non-ferrite equivalents was not feasible. We have therefore developed a ferrite-free, high speed serial data transport system. To ensure that this system is compatible with numerous current and future instruments (not just our transceivers), and to simplify high-level application development, we have also designed a programming language that is able to define an instrument's control hardware and automatically generate an application programming interface (API).

Methods

Syntax: A simple syntax (or programming language) was conceived for defining in a text file the topology of a computer-instrument interface. Using the free software tools "bison", "flex", and "gcc" (www.gnu.org), a compiler was implemented to parse this language. The latter allows a programmer to define a map of the data "flow" starting at the computer and ending at the instrument's digital and analogue control lines. This map consists of "endpoints" – wires or groups of wires that control some aspect of the instrument and for which control functions will be created in the API, or "hubs" where control information is translated from one form to another (analogue to digital, serial to parallel, etc.) either by software manipulation, or by physical components (usually integrated circuit chips) that are part of the overall "glue" hardware that attaches the instrument to the computer. For each hub, a template must be created that records the hub's input- and output-data formats, and the number of channels of data it utilises. Our library of templates includes several digital-to-analogue converters (DAC's), analogue-to-digital converters (ADC's), and controllers that serialize and deserialize arrays of digital control lines. New templates are created by embedding C code in a device definition file. While defining new templates is possible, if a programmer only intends to use pre-defined devices, no knowledge of the C programming language is required.

Our compiler parses the map definition to generate a C/C++ API for the instrumentation control system. In addition, the compiler can use the generated API to create both an interface for controlling the equipment using *Mathematica* software (Wolfram Research Inc., Champaign, IL, USA), or a graphical user interface for manual control. Documentation for the function calls available in the API – with usage instructions supplied by the programmer – and for the pin-to-pin connections of the physical interface components are also available from the compiler.

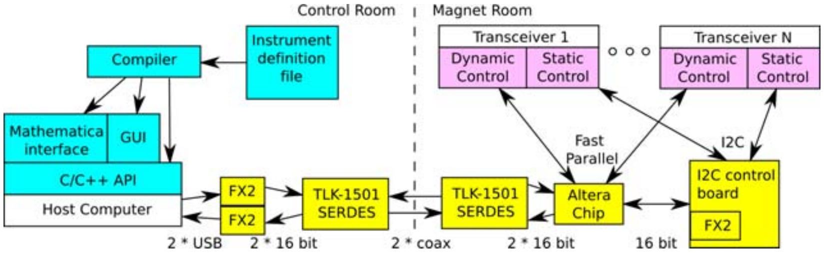

Figure 1. The components of our instrument control system. Software components are in cyan, serial transport is shown in yellow and control elements are in pink.

Bus system: To make this language applicable to instrumentation inside the magnet room, we have developed a USB-based high speed data transport system (Figure 1) that circumvents the problem of ferrites. It connects that instrumentation to a host computer in a control room beyond the magnet's fringe field. The system uses a serializer/deserializer (SERDES) capable of transferring simultaneously 96Mbytes per second of data in both directions – exceeding the practical speed limit of a USB port, 40Mbytes per second. The USB interface is provided by two Cypress Semiconductor CY7C68013 FX2 micro-controller evaluation boards (Cypress Semiconductor Inc., San Jose, CA, USA) selected for their integrated USB interface, ease of (re-)programming, and the availability of a wide array of free software development tools. The FX2 chip is a slow processor used to configure a high-speed data pipeline running between a USB port and a single 16-line-wide parallel interface that moves data at 48MHz, allowing a "burst" transmission rate of 96MByte/s, thereby matching the SERDES' data rate. To simplify the connection to the host computer, two FX2 evaluation boards were used – one for each direction of data transfer, limiting the required modifications to the SERDES board to a few logic gates.

The SERDES units are Texas Instruments TLK-1501 evaluation boards (Texas Instruments Inc., Dallas, TX, USA). They translate 16 parallel uni-directional data lines into one – a coaxial or fibre-optic cable, allowing us to pass just 2 cables into the magnet room, one for each direction of communication. These units were selected because: 1. they operate at the maximum transfer rate of the FX2 boards, thus allowing us to drive them from the clock provided on an FX2 evaluation board, and 2. they perform internal "8b/10b" data encoding, which improves the transmission characteristics of the serial data. This relieved us of the burden of developing our own rapid 8b/10b encoder/decoder system. Additionally, the units have the ability to pre-empt the data stream with limited control information. In the event that one end of the bus is unable to accept a transmission, we depend on this control information to "throttle" the sender to prevent data loss.

On the instrument end of the link, following the SERDES unit we need a controller to process the data stream. However, it would be complicated to use a pair of FX2 boards due to their need for synchronization and communication. (In the control room, they merely act as pipes.) Instead, we programmed an Altera programmable chip (currently "MAX" but "Cyclone" in future; Altera Corporation, San Jose, CA, USA) to connect both the transmit and receive paths of the SERDES to a single FX2 evaluation board. A circuit board with an array of "I2C" bus controllers was affixed to the headers of the FX2 board, allowing us to dedicate a single I2C bus to each of our CF transceivers. An I2C bus is a 2-wire data transport system for connecting as many as 127 devices, and the devices we use are capable of signalling rates up to 1MHz. In our CF transceivers, we used mostly I2C chips, thereby allowing 54 device functions over the 2-wire bus. The text file (see "Syntax" above), describing everything in the figure in yellow and pink, contains 224 lines, and includes usage information for each of the 54 control functions. While this 2-wire bus is convenient for static instrument control, a separate dynamic system is required for MR data generation and acquisition. Then, the Altera chip assumes control of the SERDES and distributes its data stream to our CF instrument's "pulse programming" interface.

Only small modifications were required to the evaluation boards to allow them to be placed in the magnetic field. The SERDES evaluation board had 3 ferrite inductors removed and replaced with larger ferrite-free equivalents, while the FX2 evaluation board had several unused magnetic connectors and switches removed. As a result, the equipment placed in the magnet room does not contain significant amounts of ferromagnetic material. The communications equipment has been successfully tested at the bore of a 7 T magnet with a bit error rate of approximately 1 in 42 billion. It occupies a space of about 25 x 20 x 10 cm and costs about $3000 Cdn. In the future, an error correction scheme, based on a Hamming code with parity, will be implemented.

Conclusion

We have implemented and tested a configurable, ferrite-free system for control of an instrument in a high magnetic field. The system's scripting language simplifies the process of computer control of instrumentation, and allows changes to be made quickly when equipment is altered, which is of benefit for rapid prototyping. The serial nature of the high speed data transport system enables control of complex instrumentation with only 2 coaxial or fibre-optic cables passing through the magnet room wall, thereby requiring minimal space on a penetration panel.