	IST-2002-002114 – ENACTIVE NETWORK OF EXCELLENCE **Milestone 1.1.1.**
Reference	**EI_MRD111_INPG_050630**
Title	**INPG contribution to Milestone 1.1.1.**
Object	**The gesture bridge: a prototype implementation**
Authors	**Damien Couroussé, Jean-Loup Florens, ACROE**
Participants	**Daniela Urma, INPG**
Dissemination	**RD11 partners**
Nb of Pages	**12**
Date of the version	**28/06/2005**
Confidentiality	**ENACTIVE Internal Restricted Use**

1 Introduction

1.1 The gesture bridge

A multi-sensory simulator is mainly composed of specialized peripherals that are dedicated to the interaction according to human modalities (gesture, audition and vision) and of a real-time physical model simulating system. The sound and graphical peripherals may be technically similar to the corresponding multi-media or general-purpose ones. On the contrary, standard peripherals for gesture interaction no longer exist and specific hardware is necessary to satisfy the needs of gesture interaction.

We define the *gesture bridge* as the component of a real time system for multi-sensory simulation playing the role of a communication interface between the simulator (i.e. the machine calculating the simulated model) and the gesture peripherals. In this definition the gesture peripheral includes a haptic interface with only hardware level of signal conditioning.

Hence, the *gesture bridge* is a very particular component of real time systems for gesture interaction, because of its role, which is:

- To transmit data about the state of the gesture peripherals to the simulator.
- To transmit back data information processed by the simulator back to the gesture peripherals.
- To execute low-level signal treatments whose main role is to make the adequate fitting between the two different signal structures, on one side at the gesture interface and on the other side in the physical model simulator.

Thus, if we get into the details, information exchanged between the gesture bridge and the simulator will be in a digital format whereas in most of the cases analog signals will be transmitted between the gesture bridge and the gesture peripherals (see next section). That is, the core function of the gesture bridge is to convert digital information to analog signal, and vice versa. This function has to remain completely transparent from the user's point of view, and has to remain as transparent as possible from the designer's point of view.

More precisely, since we assume that the simulator is built on a relatively standard layer (i.e. mono-or multiprocessor architecture with a real time OS or real time software dispositions), and that the use of a gesture peripheral should not interfere with the internal hardware structure of the simulator, the hardware connection must use a standard path. The usual peripheral connection systems such as USB, Fire-Wire or Ethernet are not completely adapted nor validated for haptic interfacing; specific boards for the PCI bus are usually chosen instead. PCI board solution, although more expensive because of the costs generated by the material used and by the involvement required in development, still seems to be one of the most adapted for a high performance simulator.

We consider the gesture bridge component as one of the main bottlenecks of most of the real time platforms for multi-sensory simulation, because:

- Of the *latency*[1] introduced in the data transmission, which is for an important part due to the conversion time. This point has a particular impact on the frequency at which the simulation loop can be calculated.
- Of the *limited number of available communication channels*. Limiting the number of available communication channels can be very constraining for the morphology or the type of gesture peripherals chosen.
  This point is a general problem of communication ports in computer hardware. However, it has a stronger effect when using gesture peripherals, because temporal multiplexing cannot solve the addition of a communication channel. This addition has therefore an impact on the overall hardware cost.
- Of the *constrained format* of the data exchanged between the simulator and the gesture peripherals. This point has an impact on the information transmitted, and therefore on the quality of the gesture interaction and on the accuracy of the simulation.

1.2 The necessity of the analog stage in the Gesture Bridge for high quality haptics

Considering the gesture bridge, the input and output communication ports between the simulator and the gesture device are sensible points of an interactive multisensory simulator. Depending on what kind of actuating and sensing techniques are chosen, it might be possible to avoid the analog stage: based on techniques inherited from the fields of robotics and automatics, numerous haptic devices have been developed trying to skip this analog stage between the gesture peripheral and the simulator, using sensors with digital signal outputs or actuators directly driven from digital input. We will however show in this part that avoiding the analog signal stage is not evident, and furthermore that analog stage in the gesture bridge should be inevitable for high quality gesture interaction.

It is indeed a common use in robotics or automatics to have digital techniques for sensing and actuation instead of using an analog stage. But there are major differences between the common uses in robotics and these for the haptics field.

• Resolution

The required level of resolution is much more important in haptics: whereas 12 bits resolution or less in robotics could lead most of the time to a suitable solution in terms of controllability and stability, in the case of haptics 16 bits of resolution are at least necessary, because of the sensing capacities of the human and of the dynamics of the signals. Indeed, a lack of resolution may generate very low-level oscillations perceptible by humans (i.e. haptic noise), and conversely, such low-level oscillations are not perturbing in robotics.

Usually in haptics, velocity measurements are obtained from the temporal derivation of the position signal, in order to save the cost of velocity sensors. In this case, the quantification noise is widely increased by the derivation operator, and thus greatly affects the velocity signal.

High resolution is very important too when the application requires a great *dynamic scale*: we may cite the example of the percussion, where it is necessary to enable as well movements of great amplitude, and precise movements in a close workspace (for instance staying almost in

[1] We define latency as the time elapsed between the moment where the data is available on the gesture device, and the moment where it is available in the simulator memory, or vice versa.

contact with the surface). The violin bowing is another example where movements of great amplitude are performed, and where the specific oscillatory phenomenon involved must be represented with a high spatial resolution. Where a same signal has to represent the superposition of these two movements, a high signal dynamic is necessary. Hence, a high resolution is needed.

• Bandwidth

As a general rule, haptic needs high bandwidth. Model computation of non-linearities such as hard contact, friction rugosity, etc., or of light objects, which means low time constants, induces high frequency for the simulation.

In the case of multi sensory simulation, the more natural representation of elementary sounding objects like tapped objects, induces in most cases high bandwidth requirements for the haptic channel. Most of the time, haptic-sound applications make use of artefacts to cope with the differences between the bandwidths of sound and gesture signals. But, in an enactive approach, it is necessary to realise minimal sound-gesture metaphors (without such artefacts), and in this case sound and gesture outputs have to be computed within the same model. Therefore, the bandwidths of haptic and gesture signals could be the same.

Considering these points, solutions using digital inputs and/or outputs that are judged to be proper in robotics are no more suitable for haptics, and we might need for high quality haptics up to a resolution of 16 bits and a bandwidth of 3 kHz (that is, a sampling frequency of about 15 kHz because of the limitations of physical modelling), for both input and output communication ports of the gesture bridge.

1.2.1 Sensing and A/D converters

Digital sensors are mainly optical encoders, either based on counting (incremental encoders) or based on the reading of an absolute measure (absolute encoders). These solutions for sensing can be interesting because of direct digital output fed to the simulator, and because such sensors are often embedded in actuators such as DC motors too. This is mainly why these kind of encoders are mainly used in haptics implementations.

- Incremental digital position sensors

Digital position sensors are widely available today. It the standard resolution for this kind of sensors remains 8 to 10 bits resolution, it is possible now to go up to 12 bits or more, using quadrature interpolation for instance (see [Enactive, 2004]). Considering that 16 bits resolution is available for this technology, if $1/10^{e}$ of the whole range of the sensor is covered at a sampling frequency of 10 kHz, the running frequency of an incremental digital encoder should be: $2^{16} / 10 \times 10^{4} \approx 65{,}5$ Mhz. This is not feasible at the current state of the technology.

- Absolute digital position sensors

Absolute digital encoders, despite there are more expensive, have an advantage on incremental encoders in terms of running frequency, since at high frequency, there is not the same abrupt loss of signal.

However, absolute encoding supposes that all the 16 bits resolution can be read at each position. Here, the mechanical properties of the sensing support (mechanical film for instance) and its stability along time do not allow for such a high resolution.

- Measuring other variables

Position sensor remains unavoidable in haptic systems; even in admittance mode, it is necessary to measure positions for position servo-control. Sensing other variables (such as velocity, acceleration, etc.) might be very useful, but at our knowledge no interesting techniques allow for digital output signals without an intermediary analog stage.

- Sampling frequency: analog vs. digital stages

In any case, in addition to the lack of resolution when combined with high bandwidth, digital sensors do not avoid the inherent latency introduced by sampling: on this point, the solution composed of analog sensors plus A/D converters is equivalent.

We therefore assume that, considering the current state of the technology, sensors providing analog outputs plus A/D converters will still lead to a solution of higher gesture quality compared to the use of digital sensors alone.

1.2.2 Actuation and D/A converters

A part of haptic systems architectures makes an intensive use of Pulse Width Modulation (PWM), or other related modulation techniques, allowing for the saving of D/A converters. Such techniques are widely used in systems whose design comes from the robotics field. Robotics made an intensive use of such modulation techniques; the control input of brushed DC motors can easily be entered as PWM. It is well known that PWM amplifiers are very interesting due to the great energetic efficiency; however, PWM modulation is known for generating noise, providing a “scratchy” feeling in the haptic rendering: the noise generated by the subharmonics of the square based output signal make this modulation technique inadequate for high quality gestural rendering.

Related to haptics, [Bouzit et al., 2002] for instance, in their implementation of the Rutgers Master II-ND Force Feedback Glove, used the PWM technique to control the cylinder pressure for the actuation part. They had the PWM running at some hundreds of Hertz (300 Hz in [Bouzit et al., 2002a], 500 Hz in [Bouzit et al., 2002b]), which might seem low, but the opening and closing response time of the solenoid valves is 2ms, so the noise generated by such modulation technique is here partially or totally filtered by the mechanical properties of the actuators. Thus, thanks to the low reactivity of the mechanical pneumatic actuation, the haptic performance won’t be lowered by the use of PWM.

[Ellis et al., 1996], in their implementation of a planar haptic interface preferred linear current amplifiers to pulse-width modulation amplifiers to power the low-inertia brushed DC motors used. They explain that these PWM amplifiers are more compact and do not demand such important heat dissipation as linear amplifiers do, but PWM amplifiers excite audible and palpable subharmonics in the mechanical structure, even if the switching frequency is above 20 kHz, which are felt by the user of the system during manipulation.

[Çavusoglu et al., 2002] perform a critical study of the mechanical and electrical properties of the Phantom haptic interface. One of the main limitations of the Phantom pointed out here, is that its motor drive electronics uses a motor drive based on PWM: “*the high-frequency switching associated with the PWM signal makes it inadequate for high-performance application*”. As like Ellis & al., they explain that the switching generates high-frequency range signal components, which could lead to instability as compared to the use of linear amplifiers.

Another solution might be the implementation of a PWM running at a very high frequency, for instance 100 kHz. In this case again, the problem of the combination of high bandwidth plus high resolution arises: a PWM running at 100 kHz with a resolution of 16 bits means to generate transients at $10^5 \times 2^{16} \approx 6{,}5.10^{11}$ Hz, which is not possible given the current state of the technology.

1.3 First conclusion

As a first conclusion, we can say that the use of analog actuators and sensors still allows higher performances, compared to the equivalent solutions (in terms of bandwidth and resolution) in the digital domain.

This is mainly the reason why we cannot avoid the stage of A/D and D/A conversion in the gesture bridge for high quality gesture interaction.

2 Overview of the existing signal conversion techniques

2.1 Analog-to-Digital converters

Given the current State-of-the-Art in the technology for analog-to-digital conversion, two main categories of converters are found: low latency and low resolution converters, and high resolution but high latency converters.

2.1.1 Low latency and low resolution converters

These kinds of converters are of wide use in application fields where an important amount of data are to be transferred, or where the bandwidth rate is important. The main concerned field is video applications.

For this category, the conversion latencies easily fall below 5 μs, but the conversion resolution can't usually exceed 10 or 12 bits.

- Multiplexed encoding

The typical A/D conversion chain contains an analog multiplexer that choose from up to 16 analog input signals; an amplifier then buffers the analog signal and may also apply gain to boost its voltage level; a sample-and-hold circuit holds the input signal constant during the conversion, and at last, an A/D converter changes the analog signal into a digital code. This widespread solution uses only one A/D conversion component for all the input channels, thanks the multiplexing of input channels; but its major drawback is the fact that all channels input value cannot be hold at the same time.

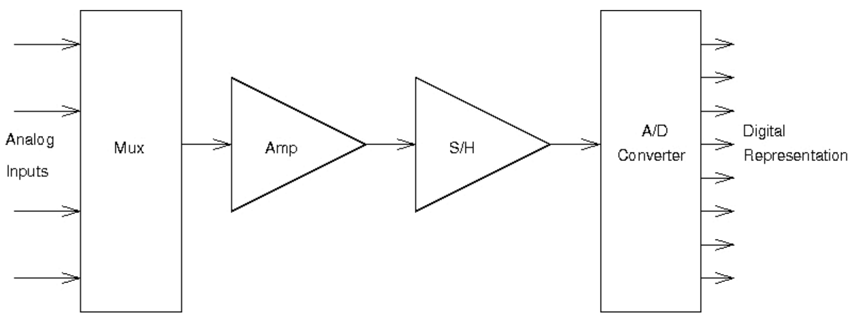

Figure 1 – Conventional architecture for A/D conversion

To cope with the lack of simultaneous sampling proposed by the previous solution, successive approximation converters have one supplementary sample and hold circuit for every channel.

Thus, the multiplexing still allows for the use of only one A/D conversion chip, but the A/D samples correspond to the same conversion time.

However, this architecture still presents a drawback of great importance for haptics, where there is a real need for an important number of independent conversion channels. Here, the increase in the number of simultaneous conversion channels makes the conversion time raise as well.

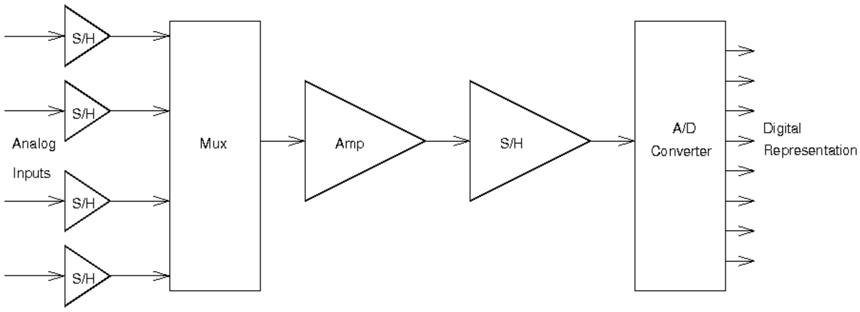

Figure 2 –Simultaneous Sample and Hold A/D systems

- Parallel encoding / flash converters

The major drawback of multiplexed encoding is that the more input channels you have, the more time you need for one sample conversion on the whole channels. Parallel encoding provides a faster conversion method: the input signal is fed to 2^{n-1} (where *n* equals the number of bits of resolution) analog comparators whose reference inputs are derived from equidistant taps off a linear resistor fadder.

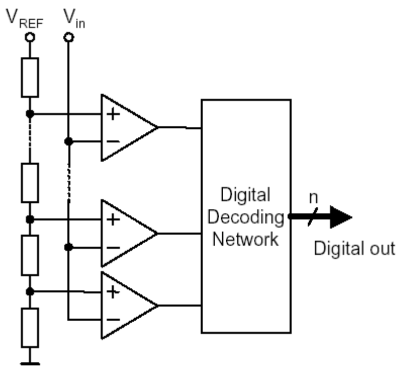

Figure 3 –Diagram of the parallel encoding architecture

This conversion architecture provides the fastest way for D/A data conversion.

However, this implementation becomes quickly impracticable, as soon as the requirements for resolution exceed 6 or 8 bits, due to the high number analog comparators needed, and to the lowest possible variability in the components characteristics. Furthermore, the power consumption and the area consumption increase dramatically with the number of resolution bits.

- Two step flash converters

This architecture takes the principles of the flash architecture back. It allows to increase the conversion resolution by adding a supplementary conversion stage, which is identical to the first one. Thus, a first conversion stage (*Coarse flash ADC* in the diagram) generates the MSB bits of the digital output; paralleley, the second conversion stage (*fine slash ADC* in the diagram) generates the LSB bits of the digital output.

The benefits of this architecture are that it becomes possible to use flash conversion architecture to have digital output resolution higher than 8 bits. Furthermore, at equivalent resolution performance, this architecture is not so costly in terms of area consumption.

The main drawback of this architecture is there is an increase of the conversion latency, compared to the one-stage flash architecture.

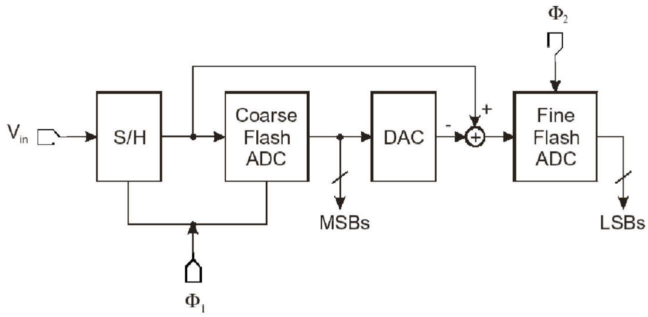

Figure 4 – Diagram of a 2-step flash conversion architecture

- Pipelined converters

One the same principles than the 2-step flash converters architecture, it is possible to use many conversion stages in a pipelined implementation. Benefits are a rapid possible increase of the resolution, even using low-resolution stages.

In this architecture, latency is equal to *n* clock tops, where *n* is the number of conversion stages used.

2.1.2High resolution and high latency converters

The most famous category of high resolution analog-to-digital converters is called Sigma-Delta.

Now most technologies for high-resolution conversion are derived from the Sigma-Delta conversion techniques.

The basic principle of Sigma-Delta converters involves the trade-off of amplitude resolution for sampling rate. In contrast to other converter technologies such flash converters, Sigma-Delta converters sample signals many times faster that the Nyquist sampling frequency (i.e. twice the bandwidth of the input signal) but only with one bit of amplitude resolution. Generally the sampling frequency is at least 64 times faster that the Nyquist frequency. They offer high resolution achieved principally by their high-speed sampling combined with feedback, noise shaping and digital filtering.

The Sigma-Delta conversion technology is mainly used for A/D conversion, where very few solutions allowed for high resolution conversion, but this conversion technique can be used as well for D/A conversion.

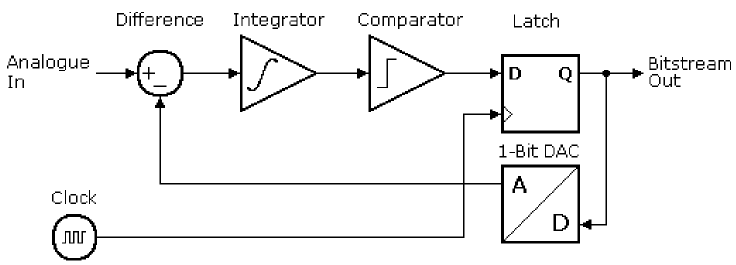

Figure 5 – Analog diagram of the Sigma-Delta modulation (first order)

The main advantages of this technique are:

- The possibility to have a high resolution of conversion: the Sigma-Delta converters are widely used for applications requiring a sampling resolution output on 24 bits or more.
- This conversion technique requires only one 1-bit converter, whatever the resolution conversion is (which depends most on the ration between the Nyquist frequency of the converted signal and the frequency at which the converter works).
- The conversion is per definition monotonic: a change of slope in the analog input will always correspond to the same change of slope in the digital output.
- The conversion is linear.

However, the main drawback of this conversion technique remains if the use of digital filtering to obtain an *n*-bit resolution signal starting from one over-sampled 1-bit signal: this digital filtering needs an important amount of incoming data (1-bit samples) before being able to produce one output data (one *n*-bit sample). Usually, the digital filters included in the Sigma-Delta converters need more than one hundred 1-bit samples before being able to generate one *n*-bit sample of output data. That is, *the conversion latency at the start of the sampling cycle takes an important amount of time*. Sigma-Delta converters can produce high resolution digital signals, but the latency is important as well.

Thus, this technology is widely used in applications requiring high resolution digital conversion, such as high quality video signal treatment, or high quality sound applications, where the latency is not an important factor because of an open-loop implementation.

2.2 Digital-to-Analog converters

Digital to analog conversion is not such a major problem as analog to digital conversion can be. Most of the analog output conversion circuits usually have one separate D/A converter and a data buffer for each channel, and allow for a digital input resolution of at least 16 bits, for an important number of independent channels.

When talking about digital to analog conversion, latency can be related to settling time. Settling time is the time needed for the converters to produce a stable analog output, given a rated accuracy, from the digital data at the converters entry. Full-scale settling time usually measures the worst case, that is, the time needed for the signal to go from one extremity of the output analog scale to its other extremity. Smaller than full-scale changes take less time to settle.

Most of the current digital to analog output conversion boards provide a settling time of 10 μs, given an accuracy performance below 0.01%, for up to 32 or 64 independent output channels.

2.3 Conclusion: a fine balance between latency and resolution

Today's implementations for applications involving high quality gesture require the following criteria:

- A high number of independent input (ADC) and output (DAC) channels between the host (the calculating machine) and the gesture peripheral (usually called the haptic device)
- A low conversion latency time. The multi-sensory simulation needs that the simulation works in a closed loop. Hence, the lower the conversion time will be, the more time

the simulator will have for the computation of the model, and the higher can be the simulation frequency.

- High-resolution converters: it is necessary to use at least 16 bits converters for the D/A and A/D channels.
- All the channels used have to be converted at the same time; that is, multiplexed conversion techniques are not available for haptics.

The two categories previously presented answer to very precise needs of some application fields. We have seen that very often a compromise has to be made on latency requirements when there is a need for higher resolution, and conversely a compromise has to be made on resolution, when there is a need for high latency.

Unfortunately, haptic applications meet these two requirements: low latency and high resolution.

Furthermore, we can observe that the boards commonly available for data acquisition only provide 4 or 8 independent channels. For haptics, 6 independent channels is the minimum, since it corresponds to one manipulated point with 6 degrees-of-freedom.

To conclude, we can present the requirements for a communication interface between the host and the gesture peripheral of a real time platform for multisensorial high quality simulation.

- 16 A/D and D/A independent channels
- All the channels are converted at the same time (no channel multiplexing)
- Time latency for conversion should be lower than 10 μs
- Conversion resolution should be at least of 16 bits (i.e. output of A/D converters, and input of D/A converters)

3 Work plan

The current implementation of the Telluris real time simulator at INPG allows for the simulation to be run at 3 kHz. The main objective of this multi sensory simulator is to provide a very high-quality simulation. Therefore, considering the points presented above, we do not have other solution than using D/A converters combined with linear amplifiers for the output signals towards the gesture peripheral, and D/A converters in combination with the electro-magnetic actuation of our gesture device (the TGR).

3.1 Study of the market for A/D-D/A conversion boards

Considering the acquisition and generation boards currently available on the market, we have bought the Toro board from Innovative Integration. This PCI board has the following characteristics:

- 16 independent channels for A/D conversion:
  - o 16 bits resolution
  - o Max bandwidth: 250 kHz
  - o No multiplexing of the input channels
- 16 independent channels for D/A conversion
  - o 16 bits resolution
  - o Max bandwidth: 200 kHz
- External synchronisation is possible

- Embedded CPU: TMDS320C6711 from Texas Instruments (32-bit floating point DSP – 150 Mhz/600 Mflop)
- External SDRAM memory: 32Mbytes
- 64 digital I/O communication channels (64Mbytes/sec bidirectional)

3.2 First implementation of the board

Our application will run under a Linux real time OS; drivers for a Linux host have to be developed. This work is currently in progress, and the functionalities already implemented are the following:

- Reset of the board (CPU and peripherals)
- Board configuration (PCI port check up, memory initialization)
- Asynchronous communication at low bandwidth between the host and the board, for configuration and command of the board.
- Loading of a program into the CPU memory; development of such programs using the Code composer Studio IDE for DSP development.
- Implementation of a bypass loop involving the DSP embedded onto the Toro board. Data are got from the 16 input channels (A/D converters), a small amount of data processing is performed, and data are sent back to the 16 output channels (D/A converters). Synchronisation is attached to an external clock signal. This loop runs with no failure and no loss of data up to 200 kHz.

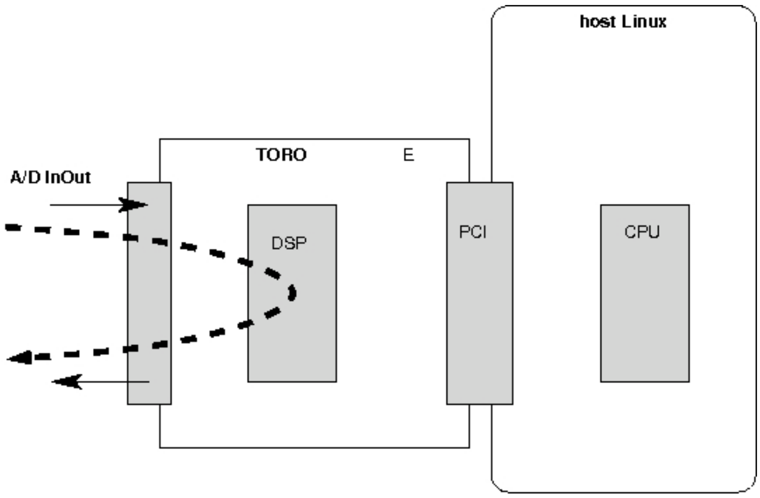

Figure 6 — Implementation of a bypass loop involving the DSP embedded on the Toro board

Concerning the driver design, the remaining work is the following:

- Implementation of functionalities for fast data transfers between FIFO of the converters and the host memory.
- Implementation of a bypass loop involving the host, and performance measurements of this bypass loop.
- Implementation of the driver into the RedHawk real time Linux OS (Concurrent Computers), and performance measurements.
- Implementation of the final gesture bridge prototype in an interactive multi sensory simulation.

3.3 Open development

Aside the technical developments necessary for a basic use of the conversion board, some other work can be envisaged, depending on the results obtained during the previous implementation phases:

- Implementation of various low-level treatments on the DSP, related to kinematics and dynamics, depending on the haptic interface configuration.
- Implementation of an embedded version of our real time simulator on the DSP of the board for higher simulation frequency.

4 Bibliography

[Bouzit et al., 2002a] Bouzit, M., Burdea, G. C., Popescu, G., and Boian, R. (2002). The Rutgers Master II–New Design Force Feedback Glove. *IEEE/ASME Transactions on Mechatronics*, 7(2):256–263.

[Bouzit et al., 2002b] Bouzit, M., Popescu, G., Burdea, G. C., and Boian, R. (2002). The Rutgers Master II-ND Force Feedback Glove. In *Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, page 145. IEEE Computer Society.

[Çavusoglu et al., 2002] Çavusoglu, M. C., Feygin, D., and Tendick, F. (2002). A critical study of the mechanical and electrical properties of the PHANToM haptic interface and improvements for high-performance control. *Presence*, 11(6):555–568.

[Ellis et al., 1996] Ellis, R. E., Ismaeil, O., and Lipsett, M. (1996). Design and evaluation of a high-performance haptic interface. Manuscript version for article in *Robotica*, 4:321–327.

[Enactive, 2004] Enactive (2004). Report on the state of the art on existing sensors/actuators technologies for haptic interfaces. Deliverable D3.1, Enactive Interfaces Network of Excellence.