Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Simulink-Rtwtgt Co-Design of Real Time Digital Interfaces

A. Bakhti, and L. Benbaouche

Abstract—**This paper describes the Simulink design of real time digital interfaces linked to I/O devices. This method is a hardware-software co-design of digital blocks, based on Simulink and real time windows target and is an alternative to S-function tool, which is a computer language description of a Simulink block. It gives a better level of abstraction of digital design and is straightforward applicable to I/O devices described by timing diagrams. This method seeks the use microcontroller resources such CPU, A/D, PWM in a control system designed in Simulink environment. A Simulink interface to a PIC 18Fxx8 microcontroller, operating at low level and involved in a two level hierarchical control system, is presented. The capability of the method as an alternative tool for S-function is detailed through an interfacing example of an A/D converter and a Simulink block. The procedure of real time simulation is described and the simulation result of the A/D conversion is presented and discussed.**

I. INTRODUCTION

Digital blocks are used to interface computer-based systems to input-output (I/O) devices such A/D converters, stepping motors, switches and so on. Additionally, when two or more microprocessors must exchange information, they need such digital blocks, generally designed using SSI, MSI and LSI technologies [1]. Simulink environment is a means for interface design of digital blocks: its external mode allows the connection of I/O devices to a Simulink block diagram which is a graphical user interface for real time simulation. The interfacing problem in real time systems is a hardware-software co-design [2]; the hardware and software must be designed together so that the implementation functions properly and respects the critical time of the system. The interfacing design allows I/O devices to exchange information with a Simulink model and can be done using system functions or S-functions [3], which are computer language description of Simulink blocks. An alternative is to use I/O drivers supported by real time windows target [4], together with Simulink blocks. All Simulink blocks can be simulated together and synthesized into working hardware using Xilinx's System Generator [5]. The S-function method, largely used, describes a Simulink block as a code written in C, C++, Matalab, Ada or Fortran. The S-function block can be customized by masking or writing a target language compiler file for the code generated by the real time function for the S-function block. We propose the second method which describes the interface as a Simulink block connected to an external device by the means of I/O drivers (supporting different digital interface boards), available in the library of real time windows target or rtwtgt. This method can be used in conjunction with the Simulink-based hybrid co-design tool [6] to interface complex systems. In control system design, resources such A/D, D/A, PWM are needed and are implemented in modern microcontrollers having up to 10 MIPS throughput. Mathworks [7] has developed drivers for microcontrollers like Motorola HC12, MPC555 and the Infineon C166 of Siemens. These drivers are dedicated for embedded systems. The build process uses the real time workshop embedded coder with a Microsoft C compiler. RTI, real time interface, manufactured by dSPACE [8], generates real time code from Simulink model and implements this code on dSPACE real time hardware. To connect the real time simulation to the physical world, dSPACE has developed S-functions I/O drivers for each dSPACE board. The RTI interface implements real time execution, and S-functions I/O drivers allows the connection to the external world. In this paper, we seek a flexible real time digital interface based on Simulink and rtwtgt, to link I/O devices to Simulink model of a control system; the computing capabilities is performed by means of parallel processing on a microcontroller network. Each network node consists of a PID controller linked to a host PC, implementing the high level control by means of the parallel slave port (PSP) of the microcontroller. The PSP is the interface between the microcontroller and the real time Simulink-based digital interface. The rest of the paper is organised as follows:

In section 2 we present rtwtgt principles and its power for dealing with real time systems. Section 3 describes a two level hierarchical control system interfaced to a PIC18Fxx8 microcontroller performing the low level control. Section 4 details the interface design of an A/D converter and a Simulink block. The fifth section presents the procedure of real time simulation.

A. Bakhti is with the Electronics Department, University of Setif, 19000, Setif, Algeria (e-mail: bakhti_dz@ yahoo.fr).

L. Benbaouche is with ESTP, 57 Boulevard Saint Germain, 75240 Paris Cedex 05, France (e-mail: benbaouche@estp.fr).

II. REAL TIME WINDOWS TARGET PRINCIPLES

It is a PC solution for the design and test of real time systems; The PC used as a host for the development process, serves also for targeting the real time application. A real time application under rtwtgt can be created as follows:

- Create the model with simulink blocks and simulate it in non real time mode.
- Add input and output devices to the simulink model using I/O driver blocks from the library provided by rtwtgt.
- Create a real time application using real time workshop, real time windows target and a C compiler to produce an executable that the kernel can run in real time.
- Execute the application in real time using Simulink in external mode.

III. TWO LEVEL HIERACHICAL CONTROL SYSTEM DESCRIBED BY SIMULINK

A. *Architecture of the Control System*

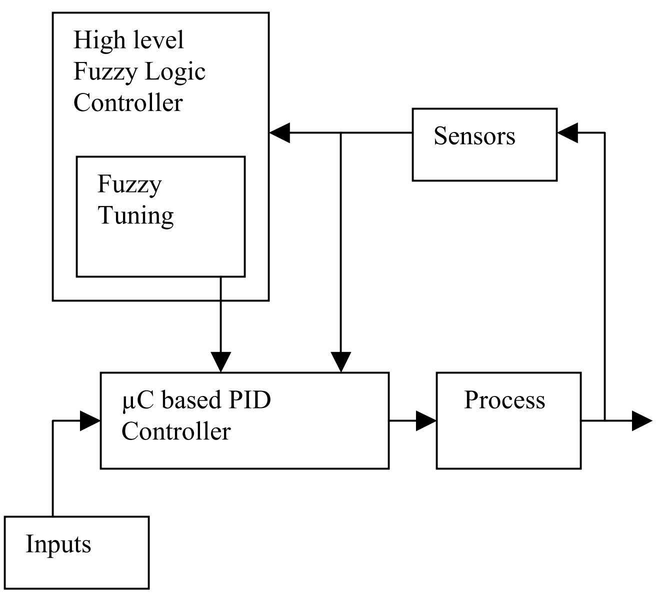

Fig.1. Architecture of the control system.

The architecture of the control system [9] is shown in figure 1 and consists of:

- A fuzzy logic controller operating at high level, performing tuning and adaptive control functions rather than in-loop direct control.
- A microcontroller based PID controller implementing the low level direct digital control by means of hardware blocs like A/D, PWM and TIMERS.

The control system is designed in Simulink in the host PC. So, it is necessary to interface the microcontroller to the Simulink block using the PSP port.

B. *The 8 bit Parallel Slave Port (PSP)*

The parallel slave port [10] is available on PIC 18Fxx8 microcontroller series. This port is asynchronously readable and writable by external devices. It can be interfaced to an 8 bit data bus as shown in figure 2.

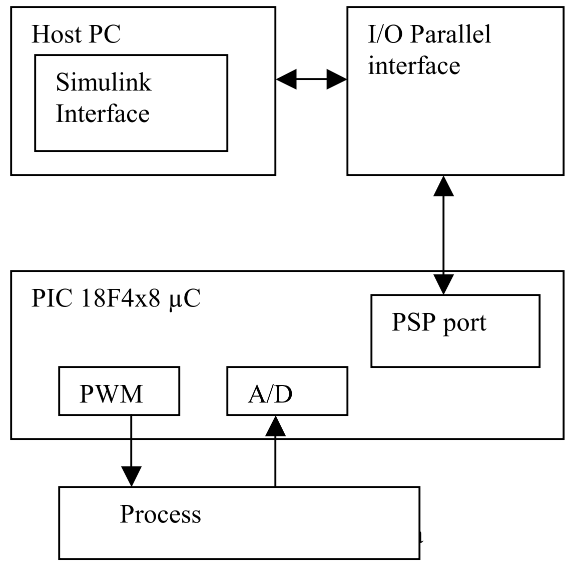

Fig.2. Host PC and microcontroller interface.

The host PC performs real time simulation of the control system simulink model. The I/O parallel interface is supported by a driver available in the rtwtgt library. The Simulink interface describes the communication process between the PSP port and the control system Simulink block.

C. *Real time Simulink- based digital interface*

The real time Simulink-based digital interface is shown in figure 3 and consists of:

- Simulink digital blocks such as switches, steps, mathematical operators.
- I/O drivers from rtwtgt library : CIO-DIO digital input and CIO-DIO digital output.
- A Simulink based Fuzzy Logic Controller.

The digital output allows the application of the input signals WR, CS and WR to the PSP port for reading and writing operations as follows:

- A write to the PSP port occurs when the chip select signal CS, and the write signal WR are first detected low.
- A read from the PSP occurs when both CS and the read signal RD are first detected low.

The digital output 2 applies control data to the PSP port.
The digital input allows the reading of the output of the process by means of the PSP port.

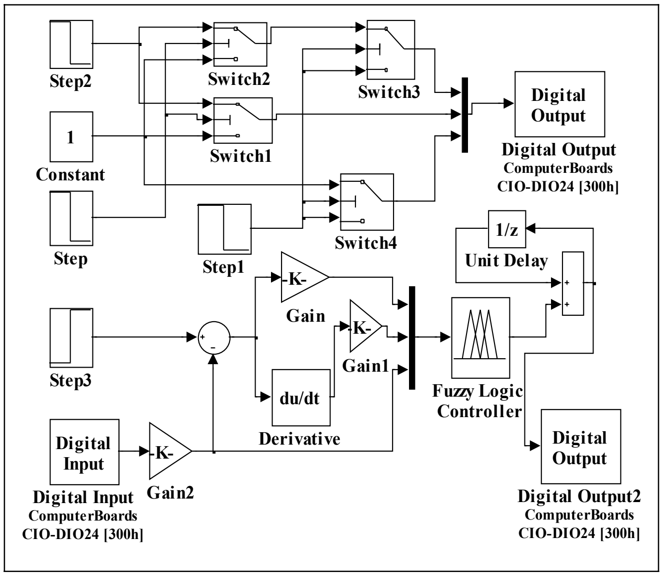

Fig.3. Simulink-Rtwgt Co-Design of a control system interface.

III. HARDWARE INTEFACE OF AN A/D CONVERTER TO A SIMULINK BLOC

A. Timing Diagram of the ADC0804 A/D Converter.

The interfacing process of the A/D is based on its timing diagram shown in figure 4 [11]. The normal operation proceeds as follows: On the high to low transition of the WR input, the internal latches and the shift register stages are reset, and the INTR output will be set high. As long as the CS input and WR input remain low, the A/D remains in a reset state. The conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low to high transition. After the requisite number of clock pulses to complete the conversion, the INTR pin will make a high to low transition. An RD operation (with CS low), reading valid data, will clear the INTR line high again.

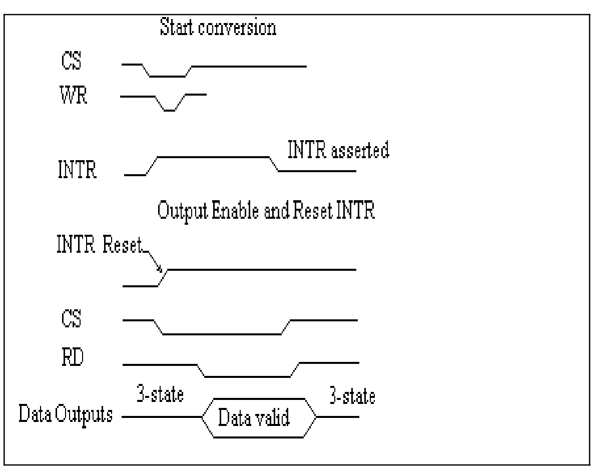

Fig.4. Timing Diagram of the A/D Converter

B. Simulink interface of the A/D Converter

Simulink interface aims to produce signals described in the timing diagram using hardware I/O blocks available in rtwtgt library. Figure 5 shows the simulink block used to produce these signals. Hardware blocks are referenced as CIO-DIO inputs and outputs and are composed of 24 lines which can program inputs or outputs. The digital output block applies CS, WR and RD signals to the A/D converter. The digital input is connected to the INTR output pin of the A/D converter. The masked subsystem block is responsible of reading the result of A/D conversion and consists of a digital input block triggered by the falling edge of the INTR signal. This signal serves also to synchronize the WR signal for the free running operation of the A/D.

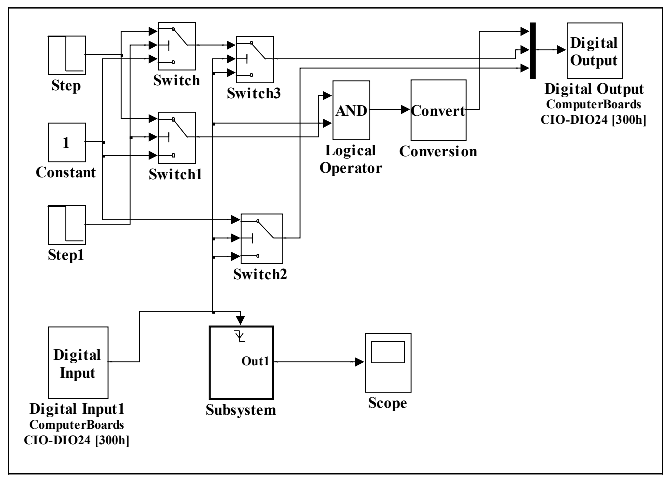

Fig.5. Simulink interface of the A/D Converter.

IV. REAL TIME SIMULATION

A. Code Generation

After entering the simulation parameters, the first step to real time simulation is the generation of the real time code by the build process that consists in the following:

- Real time workshop generates C code from the Simulink model.
- Watcom C compiles and links the C code into a real time application.
- The real time application, which is a binary file, is then executed with the real time windows target kernel.

B. Simulation Results

In order to test our real time Simulink-based interface, we have used it to perform the A/D conversion of a sinusoid delivered by an external function generator. The ADC converter operates in continuous mode. The simulation parameters are:

Fixed step size: 1e-3 s, single tasking
Sample time: 1e-3 s for all blocks
Sample rate: 1000 samples/s
Figure 6 shows the simulation result recorded on the Simulink scope.

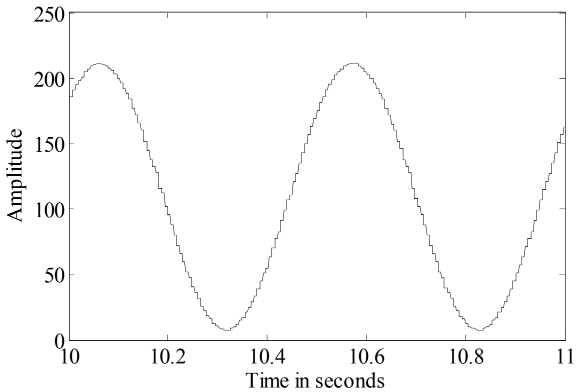

Fig.6. A/D conversion of a sinusoid.

VI. CONCLUSION

In this paper, we have proposed a real time interface design based on Simulink and real time windows target, as an alternative to S-function language description. This method is suitable to interface I/O devices described by timing diagrams. It can be used to interface complex systems using Xilinx's System Generator. The method principle is applied to interface a two level hierarchical control system and a microcontroller performing the low level control, on one hand. On the other hand, it was tested successfully under the real time windows target kernel for A/D conversion. In future work, we will implement the method to a multilevel hierarchical control system, emphasizing its multiprocessing capabilities.

REFERENCES

[1] J.P, Hayes, *Digital System Design and Microprocessors.* USA: McGraw Hill, 1984.
[2] W.H, Wolf, "Hardware-Software co-design of embedded systems", *in Proc .1994 IEEE*, Vol. 82, pp. 967-987.
[3] Writing S-functions, Copyright 1998-2001 by the MathWorks, Inc.
[4] Real Time Target User's Guide, Copyright 1998-2001 by the MathWorks, Inc.
[5] Xilinx's System Generator v7.1. Available://http:www.Xilinx.com
[6] LM, Reyneri "A Simulink-based hybrid co-design tool for rapid prototyping of FPGA's in signal processing systems", *Microprocessors and Microsystems,* Vol. 28, pp. 273-289, 2004.
[7] Matlab Version 7.0.0.19920 (R14), Copyright 1984-2004, The MathWorks, Inc.
[8] RTI and RTI-MP Implementation Guide, dSPACE, March 2004.
[9] De Silva, C.W. and MacFarlane, A.G.J, "knowledge based structure for robotics manipulators", in *1988 Proc. IFAC. Workshop on Artificial Intelligence in Real Time Control*, Swanesa, Wales, pp.143-148
[10] PIC 18FXX8 Data sheets, DS41159C, Microchip Technology, 2003.
[11] Component Data Catalogue, *ADC 0802-ADC0804, 8bit µP compatible A/D converter*, Intersil 1986, pp. 6-22, 6-36.