A Custom Communication System for I/O Control in Industrial Applications

J. A. Boluda, J. J. Pérez, S. Felici, E de Ves, F. Pardo.
Institut de Robótica
Universitat de València. València.
email: jboluda@glup.eleinf.uv.es

S. Pelaez, J. M. Insenser.
SIDSA-Tres Cantos. Madrid.
email: segis@sidsa.es

1 Abstract

This paper describes a control system for industrial applications with a custom protocol and communication system. The project has been partially supported by the Spanish initiative for the microelectronics promotion (GAME initiative) and also by a private company. Two ASICs have been developed using synthesis, high-level design and macrocell integration techniques. Finally the system has been firstly tested, and afterwards, fully incorporated to a real industrial application: For industrial reasons this application should be kept confidential, but we can describe the communication protocol and the design methodology.

2 System Description

The control system was developed to cover a real application [8], but it can be employed for many other different systems with similar characteristics. We can simplify our view of the machine and we can see the automatism like a set of inputs and outputs controlled by a central system. This central control follows a program and makes decisions taking into account the sensors (inputs) of the system.

The system may have around 200 Inputs-Outputs, depending on the options included inside the machine. The old control system had a programmable automata plus an additional set of I/O modules. The physical distance between the control module and part of the I/O, forces an excessive quantity of wires, that implicates: high cost, small reliability and slowness in the assembly.

Another fundamental characteristic is the great quantity of different options that could be incorporated to the machine, apart from the great quantity of different programs that it has to execute. These characteristics impose a microelectronic solution that is capable of combining flexibility and low cost. The implemented design is a master-slave system, with communication through a coaxial cable. Then, the central unit (ASICUC), the peripheral units (ASICPER) and the custom communications protocol are described.

3 The Communications: Protocol and Fault-tolerance

With the idea of improving the product quality increasing the reliability [5] and making the product more competitive, the elements have been connected through a bus of two wires that crosses completely the machine. The distance covered may be near of 30 meters, connecting the ASICUC with all the ASICPERs. We assign to each input and to each output an ASICPER. Each ASICPER has several I/O and they differ from other I/O only by their physical address configured with switches. The ASICUC is the master and the ASICPER are the slaves, in other words, each ASICPER answers to the ASICUC only when it is required.

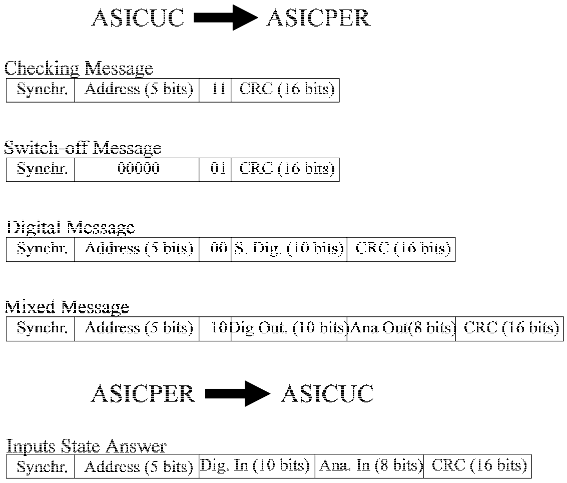

Figure 1. Description of messages.

The communication runs at 9600 baud, fast enough to make a polling of all the I/Os, as it is described subsequently. Taking into account the noisy environment where the system is involved, and with the aim of getting robust communications and avoiding electromagnetic interferences [1] a

Design of Integrated Circuits and Systems. DCIS'96. Sitges, Barcelona. November 1996.

redundant code has been implemented. A Manchester code has been utilized, code for displacement of frequencies (FSK) and also check redundant codes (CRC) [2]. The messages are constructed with the structure shown in figure 1.

The module of communications continually makes a sweep of the peripheral circuits. The way of carrying out the sweep is sending checking messages beginning by the ASICPER with address 1 to the last one. If the communication with one of the ASICs fails, it resends the check message five times more. If it faults five successive times the ASICUC actives the signal of failure and quits carrying out the sweep until the microcontroller module sends any message to the line, that usually will be the general machine switch-off.

4 The ASICUC and the ASICPER

The ASICUC is entrusted to the administration and control of the system. The versatility is incorporated by means of the use of a microcontroler that, along with an external ROM, permits the exchange of programs and machine options. The ASICUC is formed by the following modules (figure 2):

- 8051 Microcontroler: This macrocell has been developed by TreeSoft, in 1μ CMOS technology. It has absolute compatibility with the 8051 Intel's microcontroler. In this way, we can take advantage of the great quantity of development tools that exist for this microcontroler. The 8051 executes the external program kept in ROM, and it controls the I/O and the administration of all the system.

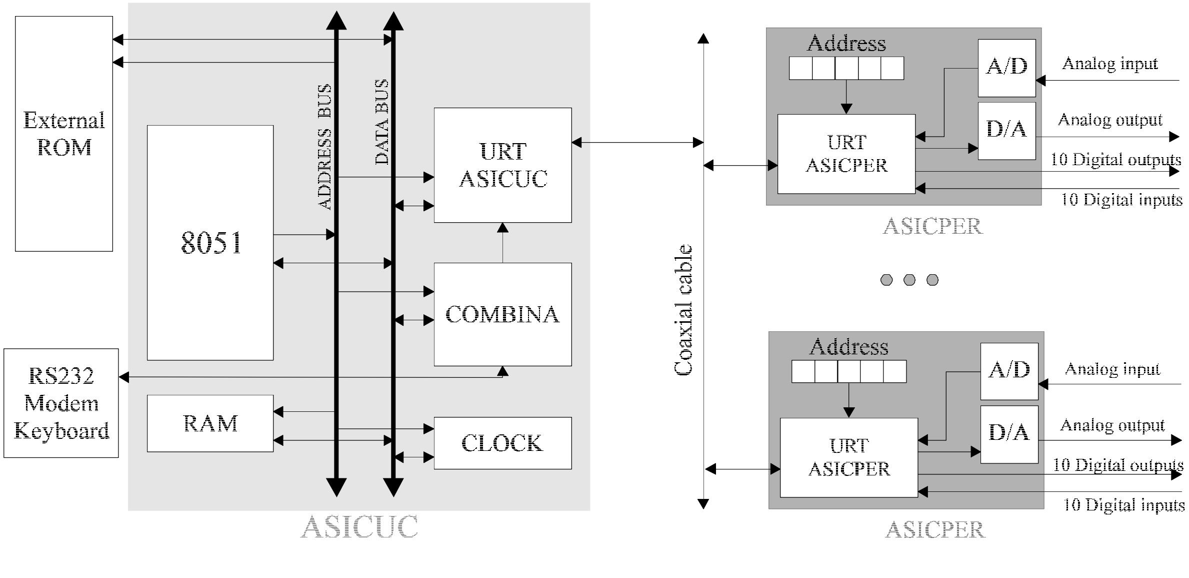

Figure 2. The ASICUC and ASICPER schema.

- URT module: It is the Receptor-Transmitter Unit of signals (interface) between the microcontroler and the I/O. It manages the request of messages of the 8051 macrocell and makes the transmission-reception in the previously explained manner.
- Internal RAM: Macrocell generated by the synthesizer of Cadence. The size is 256 bytes.
- Module "Combina": It generates all the bus control signals, the "enable" signals and the selection of the external and internal elements of the ASICUC.
- Real-time clock: It is utilized to have a control of when and for what is used the machine and to carry out statistics. It is supplied by an external battery.

The ASICPER, developed completely by SIDSA, concentrates 10 digital outputs, 10 digital inputs, 1 analogue output and 1 analogue input. Due to that, they utilize 5 bits in order to code the address of each input or output. The ideal number of I/O in each ASICPER depends on the final number and the distribution of the I/O, and the prize of the ASIC package.

All the modules, except for the macrocells, have been described in HDL Verilog. They have been behaviorally and post-synthesis simulated. Finally, a schematic has been realized in order to connect them, this schematic is the highest level in the design hierarchy.

5 Design Methodology

The design of the ASICUC and the ASICPER have been made with the Cadence DFWII silicon compilation tools. The description of the modules that compose the two ASICs have been implemented in Verilog XL. The election of this language, as the most adequate to perform the synthesis, is justified by its current advantage against VHDL [3], [4].

The Verilog XL is a behavioral HDL, with several restrictions in the synthesis, but in any case with a far more direct compilation than VHDL. The modules description has been made at RTL level [6], or transfer among register level, which guarantees the possibility of a successful synthesis. Also in the case of Synergy, that is the Cadence compiler for Verilog, one could generate several solutions optimizing the speed or the silicon area.

The synthesis should be made taking into account a set of rules to avoid non compilable parts in the circuit description [7]. In this way, complex mathematical operations or logical functions, not totally specified, should not be included. Finally, the library of synthesis is selected (ES2 CMOS 0.7µ) and it is possible to compile the modules

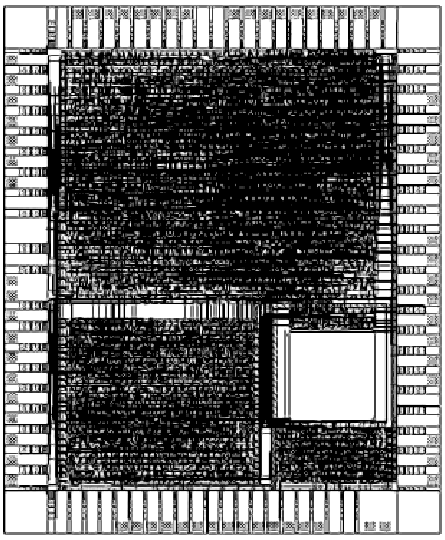

Figure 3: The ASICUC layout

6 Industrial Benefits

There are many benefits obtained by the company, due to the modernization of their machine control system by means of the microelectronic introduction. Some of them are:

- *Depreciation of costs in materials*: The elimination of a great quantity of cables (a cable for each Input or output) permits a reduction of around 20% in the cost of the machine control system.
- *Depreciation of costs in assembly time*: The cost in hours due to the assembly work of hundred of cables is drastically reduced. In this way, the labour time decreases and the product is cheaper.
- *Reliability*: The reduction of the number of connections and cables increases the system reliability. Fault-tolerance techniques and intelligent failure handling have also been introduced.
- *Features Improvement*: Renovating the control electronics, new features for the machine have been introduced, like the remote administration using modem, statistics generation, connection to a PC through a serial port, etc.
- *Other improvements*: Easy machine programmability, protection against copy, inclusion of new technologies in the company and, in general, product improvement.

7 Conclusions

An example of an ASIC application for the industry has been presented. The system has been described like a set of I/O managed by a central control system. A custom master-slave system of communications has been also presented. An analysis of the protocol and the structure of the messages has been also presented. The big quantity of programs and different options that the machine may incorporate, implicates a programmability that is included with the 8051 macrocell. A summary of the design methodology has been presented.

7 References

[1] F. Daura, “El ruido en sistemas digitales” *Mundo electrónico*. Nº 169-180, 1986-1988

[2] B. M. Cook and N. H. White, *Computer Peripherals*, Ed. Edward Arnorld. 1995.

[3] B. Fuchs “Verilog HDL est bien preferable a VHDL”. *Electronique*, Nº 35. Febrero 1994.

[4] J. Cooley. “Verilog won and VHDL lost? You be the judge!”. Netscape page at: http://www.chronologic.com/misc/reviews/verilog.review.html

[5] P. A. Lee and T. Anderson, *Fault Tolerance. Principles and Practice*. Ed. Springer-Verlag Wien. 1981.

[6] SYNERGY. Family of design synthesis technology. Verilog HDL Design Guide. © Cadence Design Systems 1992.

[7] D. D. Gajski, N. D. Dutt, and B. M. Pangrle, “Silicon Compilation. (Tutorial)”, *in Proc. 1986 Custom Integrated Circuits Conf.*, 1986, pp. 102-109.

[8] J. A. Boluda, S. Felici, J. J. Pérez, E. de Ves, F. Pardo. “Proyecto Puente de Lavado. Una Experiencia de Diseño en el Marco de la Colaboración Universidad Industria”. IBERCHIP96. Sao Paulo, Brasil, Febrero de 1996.