Real Time TCP/IP Control of Modular Production Systems with FPGAs

A. Castillo-Atoche[1], J. Vázquez-Castillo[2] & V. Sánchez-Huerta[2]

[1]Autonomous University of Yucatan
Department of Mechatronics
C.60 #491-A por 57, CP. 97000
Mérida, Yucatán, México

[2]University of Quintana Roo
Department of Sciences and Engineering
Boulevard Bahía s/n Esq. Ignacio Comonfort, CP. 77019
Chetumal, Quintana Roo, Mexico

ABSTRACT

The Modular Production System (MPS) can be used to reproduce nearly any function in complex manufacturing processes. In this work a different way of real time TCP/IP control of three different MPS controlled by embedded processors inside a FPGA is proposed. The digital design with FPGA is particularly advantageous due to its parallel processing made possible by its flexible structure, high integration and velocity. The TCP/IP control is carried out with a Rabbit Core microprocessor. This device contains the functions required to implement a web server allowing the remote control of the MPS. A user enters a public IP direction on a designed web page to enable/disable each independent manufacturing process.

RESUMEN

Los MPS son Sistemas Modulares de Producción que pueden utilizarse para reproducir casi cualquier función en procesos complejos de fabricación. La propuesta de éste artículo se basa en el procesamiento en tiempo real de tres sistemas modulares de producción controlados por procesadores embebidos en un solo circuito integrado VLSI, específicamente un FPGA. El diseño digital con FPGAs presenta una gran ventaja por su procesamiento en paralelo debido a su estructura flexible, alta velocidad e integración. Otra característica importante de esta implementación es que los sistemas de producción son habilitados/deshabilitados a distancia a través de una página web, mediante la implementación de un microprocesador Rabbit que contiene las funciones de servidor de red integradas. El usuario deberá ingresar una IP pública en la página Web diseñada para habilitar/deshabilitar independientemente cada proceso de manufactura.

KEYWORDS: MPS, FPGA, Picoblaze, TCP/IP.

1. INTRODUCTION

Since the Industrial Revolution, manufacturing methods have been characterized by process centered work organization in which material and components are transferred from machine to machine. However, manufacturing requirements have changed enormously in the last few decades and today the emphasis is on greater responsiveness, cost effectiveness, improved production quality and shorter product life-cycles. As a result, it is becoming increasingly evident that 'traditional' process centered production methods and work structures are proving inappropriate to meet the new demands. Put simply, there is now pressure to restructure manufacturing along new lines. To meet this new era of agile manufacturing, the Modular Production Systems (MPS) [1],[2],[3], seek to establish a radical new manufacturing

22

paradigm based upon standardizing both the production system 'elements' and the methods of production system design. Combined with powerful design tools, the belief is that it will be possible to rapidly configure entire fully integrated component production and assembly systems for producing a wide range of low to medium technology consumer goods.

In order to increase production flexibility and decrease production times, new system concepts based on modular, process-optimized cell systems are being developed - and not just the production of small parts. Control and process technology is being shifted away from the central PLC and into the production cell, thus becoming an integral part of the system module. Individual, recurring sub-modules can be completely pre-manufactured, tested and freely combined. Hierarchical control structures are replaced by modular distributed and simply structured production cells. Control programs for all processes and a dynamic material flow are standardized and pre-tested to guarantee overall interoperability and availability. The use of intelligent safety devices makes it possible to transfer over the network safety signals directly between sensor and actuator.

Through the gradual increase of Ethernet technology in the automation field, industrial communication could be standardized considerably to deal with all the requirements of future automation and production strategies. Nowadays, the new productions cells are based on Ethernet automation network with the following features:

Decentralized control structure
Standardized Ethernet real-time communication
HTTP for diagnosis, configuration and manual operation of all components
Standardized web-based HMI technology and backup/restore mechanism
Standardized plug&work mechanism
High-speed control and synchronization tasks

Following the Ethernet technology, the RCM3700 [4] is a compact module that incorporates the powerful Rabbit 3000® microprocessor, flash memory, onboard serial flash, static RAM, and digital I/O ports. See Figure 1. The RCM3700 Rabbit Core is the lowest priced Rabbit 3000 microprocessor based core module designed for Ethernet/Internet applications.
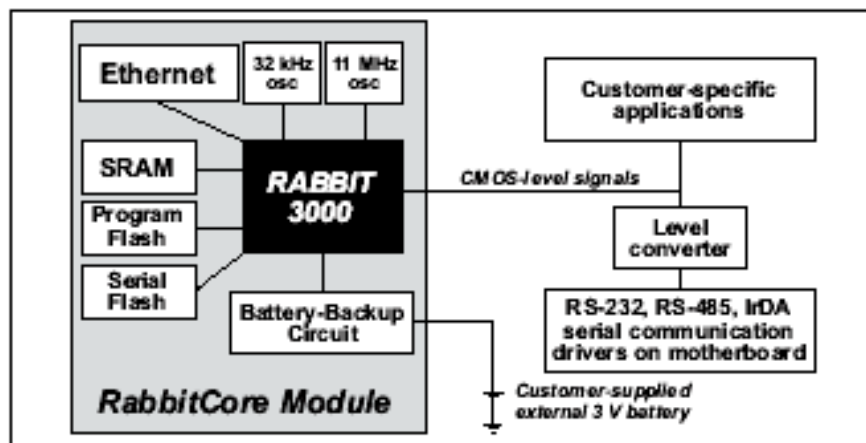
Figure 1. RabbitCore Module

Figure 1. RabbitCore Module

The increased complexity of control systems in MPS applications requires improved system design technology. Industrial applications are requiring the ability to customize common hardware and software platforms. New technology demands higher performance DSP capabilities, connectivity and video for today's industrial applications. Networking of industrial applications is also becoming more of an issue as demand for remote monitoring and wireless-enabled devices continue to grow. Programmable logic is an

23

ideal solution for these types of applications. Whether the design requires low-cost CPLDs (Complex Programmable Logic Devices) or high-performance FPGAs (Field Programmable Gate Array), embedded microcontrollers in FPGAs offers a whole host of solutions to enable the industrial design.

The Xilinx PicoBlaze microcontroller [5] is specifically designed for Xilinx FPGA devices. PicoBlaze delivers 44 to 100 million instructions per second (MIPS) depending on the target FPGA family and speed grade many times faster than commercially available microcontroller devices. PicoBlaze occupies 192 logic cells, which represents just 5% of a Spartan-3 XC3S200 device. Because the core only consumes a small fraction of the FPGA and CPLD resources, many engineers can use multiple PicoBlaze devices for tackling larger tasks or simply keeping tasks isolated and predictable.

In this paper, our aim was to provide a real time TCP/IP control of three modular production systems using the Picoblaze embedded microcontroller and the Rabbit 3000 microprocessor based core module designed for Ethernet/Internet applications [6],[7], these stations are:

MPS Distributing station
MPS Testing station
MPS Handling station

2. IMPLEMENTATION

Real time remote control of production modules are implemented with three different embedded processors inside the FPGA (PicoBlaze) and a finite state machine for the SPI control protocol with the Rabbit Core. The digital design with FPGAs is particularly advantageous due to its parallel processing made possible by its flexible structure, high integration and velocity. The SPI-Ethernet controllers enable/disable each of the MPS modules independently. The Rabbit Core Microcontroller RCM3720 transmits the control words to the FPGA [8],[9]. The FPGA decodes the incoming control words and carries out the respective operation by the embedded processors. The Rabbit core has a great feature, the Ethernet port is embedded in the core for network connectivity, with royalty-free TCP/IP software. Some alarms are displayed by the FPGA to verify the system behavior. In Figure 2, system architecture is presented.
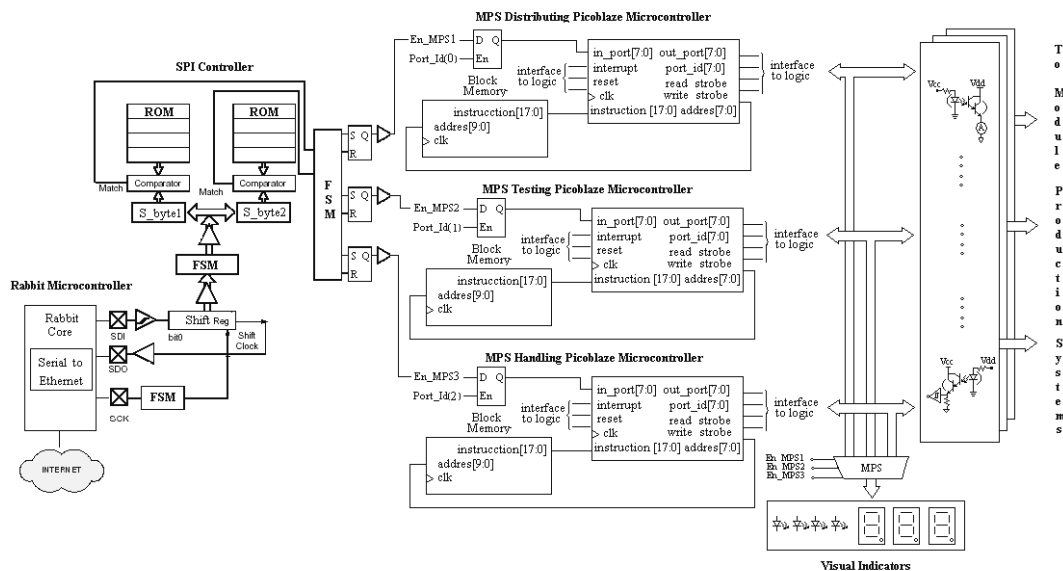

Figure 2. System Architecture of Remote control TCP&IP

TCP/IP (Transmission Control Protocol/Internet Protocol) is a set of protocols independent of the physical medium used to transmit data, but most data transmission for Internet communication begins and ends with Ethernet frames. All Ethernet interfaces have a unique 48-bit address that is supplied by the manufacturer. It is called the Ethernet address (also known as the MAC address, for Media Access Control).

To access the web, it is necessary to introduce a public IP, a gateway, subset and DNS. So, each production module will be controlled independently. See Figure 3.

Figure 3. Web Site Controller of MPS system

2.1 SPI CONTROLLER

This state machine receives the incoming control words corresponding to the production modules from the web site and they are transmitted from the rabbit microcontroller using the SPI protocol. The design of this state machine in the FPGA can be seen in Figure 4.

The FPGA was configured like the slave and the rabbit microcontroller like the master. The Rabbit Core generates the clock and the FPGA loads the incoming bits in a shift register. Once the first control word corresponding to the MPS module is captured by the state machine, the data is loaded in the S-Byte1 register. The second control word is loaded in the S_byte2 register. Both registers are compared to enable/disable the production module selected.
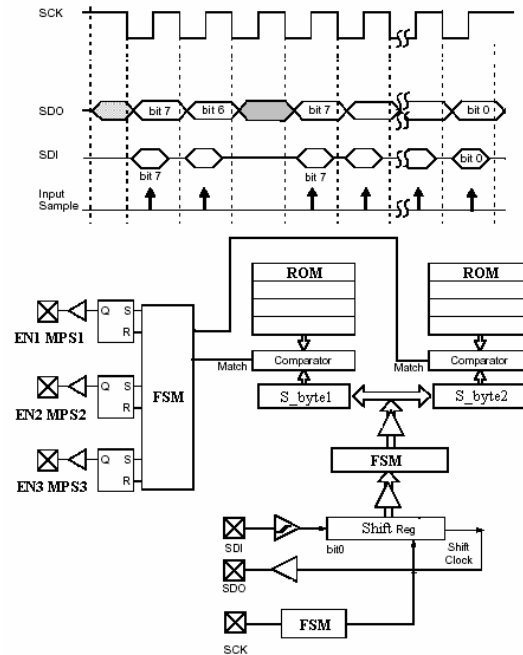
25

Figure 4. SPI Protocol Implementation

2.2 MPS DESIGN IMPLEMENTATION WITH PICOBLAZE

The Picoblaze microcontroller core is totally embedded within the target FPGA and requires no external resources. Its basic functionality is easily extended and enhanced by connecting additional logic to the microcontroller's input and output ports. The feature set includes 16 general-purpose 8-bit registers, specified as s0 to sF (these can be renamed in the assembler). All register operations are completely flexible, with no registers reserved for special tasks or given any priority over any other register. No accumulator exists as any register can be adopted for use as an accumulator. The Arithmetic Logic Unit (ALU) provides all the simple operations expected in an 8-bit processing unit.

All operations are performed using an operand provided by any register. The result is returned to the same register. For operations requiring a second operand, a second register can be specified or a constant 8-bit value can be supplied. The ability to specify any constant value with no penalty to the program size or performance enhances the simple instruction set. Each MPS station was controlled by an independent Picoblaze microcontroller.
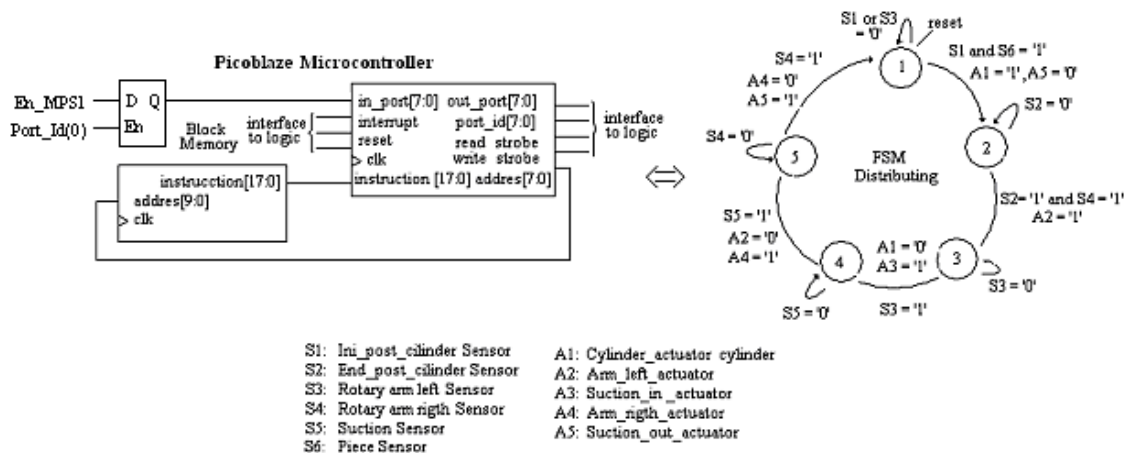

Figure 5. Finite State Machine Distributing Implementation with Picoblaze

26

2.2.1 MPS DISTRIBUTING STATION

The Distributing station separates workpieces. Up to eight workpieces are stored in the magazine tube of the stacking magazine. A double-acting cylinder pushes the workpieces out one at a time. The Changer module grips the separated workpiece via a suction gripper. The swivel arm of the changer, which is driven by a rotary actuator, moves the workpiece to the transfer point of the downstream station. The Distributing station utilizes various actuators, all of which are industrial components. The rotary actuator of the swivel arm can be set to various angles between 90° and 270°. The end positions are sensed by means of micro switches. A double-acting linear cylinder pushes workpieces out of the stacking magazine. The end positions are sensed using proximity sensors. See Figure 5.

2.2.2 MPS TESTING STATION

The Testing station detects the various properties of the workpieces inserted into it. It differentiates workpieces with the aid of an optical and a capacitive sensor. A retro-reflective sensor monitors whether the operating space is free before the workpiece is raised via a linear cylinder. A linear cylinder guides the correct workpieces via the upper air slide to the neighboring station. Faulty workpieces are rejected via the lower air slide. The Testing station employs all basic types of industrial sensors in typical applications: optical and capacitive proximity sensors and optical retro-reflective sensors. These are supplemented by various cylinder limit switches (inductive, magnetic). See Figure 6.
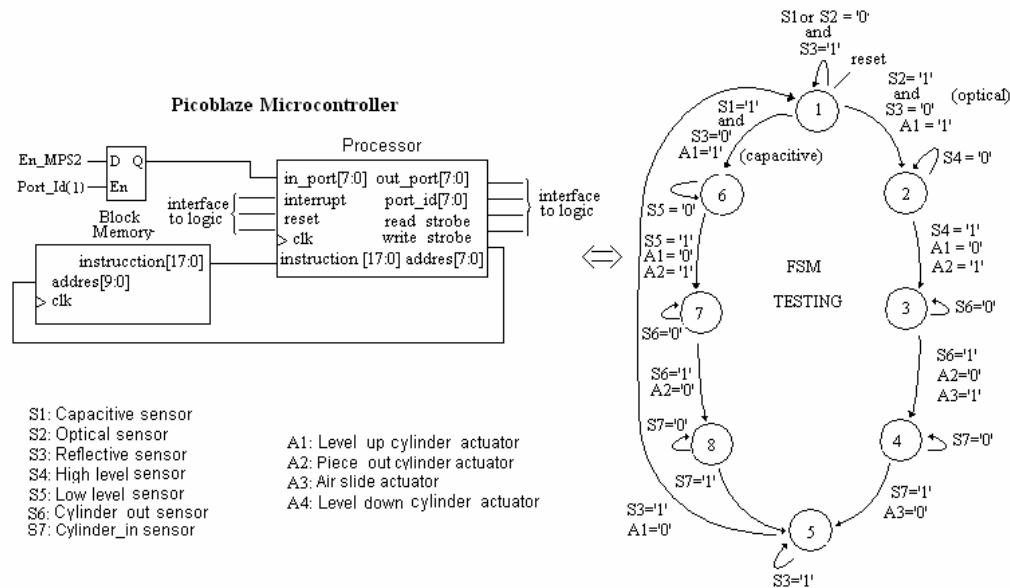

Figure 6. Finite State Machine Testing Implementation with Picoblaze

2.2.3 MPS HANDLING STATION

The Handling station is equipped with a flexible two-axis handling device. Workpieces inserted into the holder are detected by an optical diffuse sensor. The handling device picks up the workpieces from there with the aid of a pneumatic gripper. The gripper is equipped with an optical sensor which differentiates between "black" and "non-black" workpieces. The workpieces can be placed on different slides according to this criterion. Other sorting criteria can be defined if the station is combined with other stations. Workpieces can also be transferred to a downstream station.
The Handling station utilizes industrial handling components. A pneumatic linear axis with flexible end-position adjustment and cushioning allows fast positioning, including to intermediate positions. A flat linear cylinder with end-position detection serves as the lifting cylinder for the Z axis. See Figure 7.
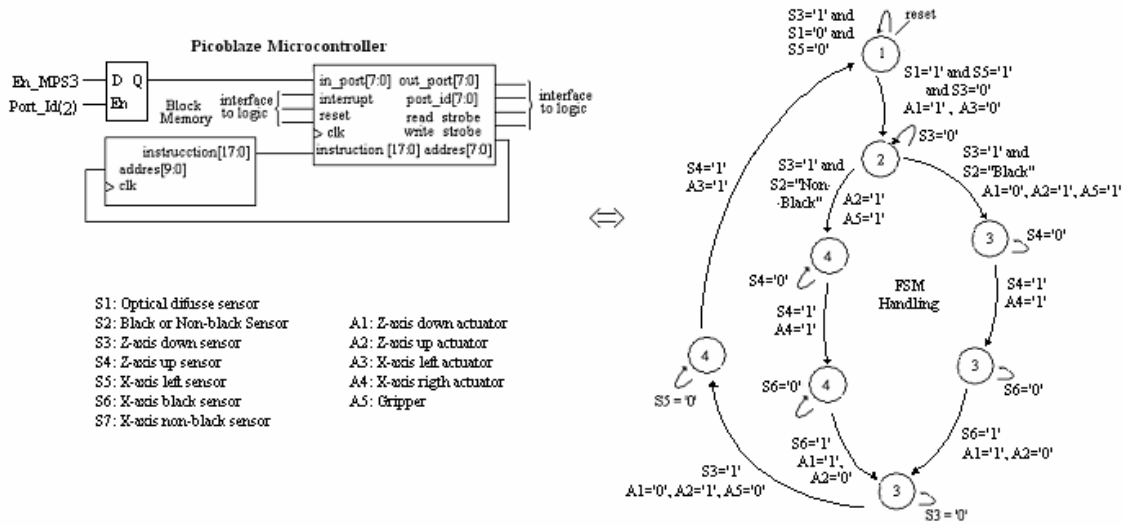
27

Figure7.Handling Finite State Machine Implementation with Picoblaze

2.2.4 MPS INTERFACE

This block serves as an interface between the MPS stations and the FPGA. The interface carries out the communication among the elements of the system, mainly with the sensors and the actuators, in order to carry out diverse functions such as the attention of the flags generated by the device. These interruptions may be in a hierarchical order and they indicate diverse important events inside the processing. The MPS interface design can be seen in Figure 8.
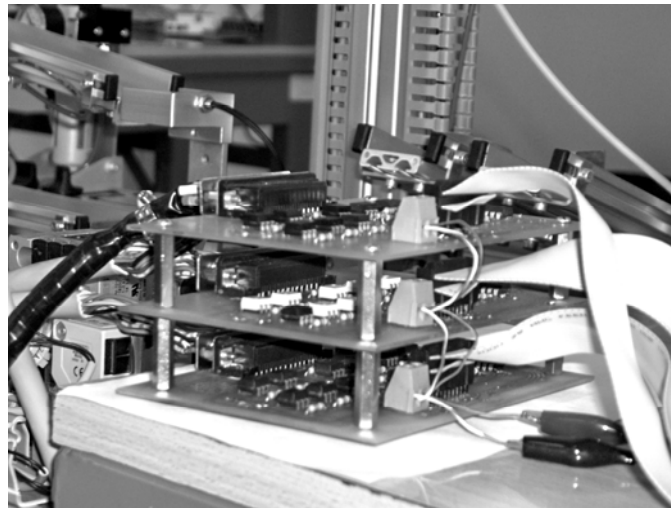

Figure 8. MPS Interface Design

28

3. RESULTS AND FINAL CONSIDERATIONS

Every module designed in VHDL [10],[11] satisfies the main features of each MPS system. These modules are:
MPS Distributing.- This module was designed with an embedded microcontroller and contains six different sensors and five actuators.
MPS Testing.- This module was designed with an embedded microcontroller and contains seven different sensors and four actuators.
MPS Handling.- This module was designed with an embedded microcontroller and contains six different sensors and six actuators.
SPI Controller.- This module was implemented with a finite state machine that carries out the serial to Ethernet transmission and enables/disable each MPS system independently.
Figure 9 shows the control system that was implemented with an FPGA Spartan 3 XC2S200FT256-4, the embedded web server was carried out with the Rabbit Core RCM3720 and the interface system was designed in the laboratories of the University of Yucatan.
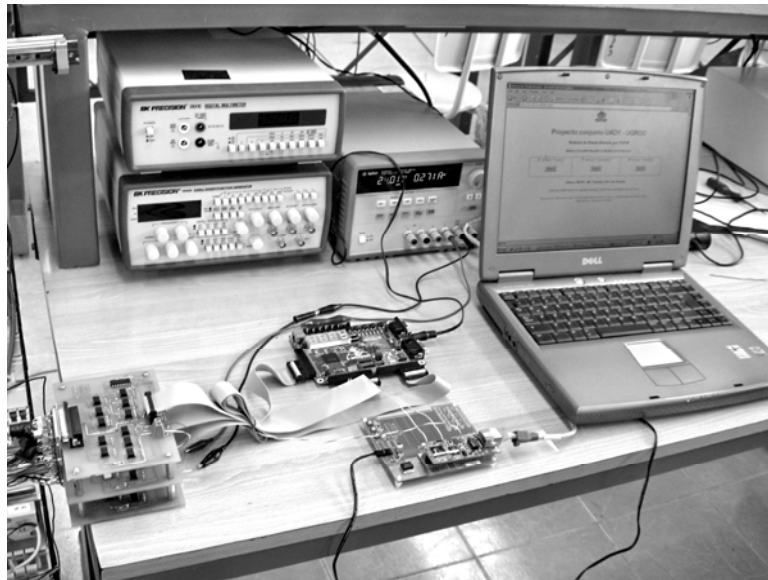

Figure 9. Remote TCP/IP Control System with Embedded Microcontrollers

The final design was synthesized with the ISE-FOUNDATION tool. The synthesis process maps the VHDL code into hardware logic gates for a specific technology library. The results obtained in the synthesis were targeted in a FPGA Spartan 3 XC2S200. In order to carry out a verification of the design, the developed device has been simulated with ModelSim. Functional verification was also carried out. In Table 1, the synthesis results of the complete Remote TCP/IP controller of production modules are presented.

Number of RAMB16s	3 out of 12 25%
Number of Slices	354 out of 1920 18%
Maximum Pin Delay	8.769ns
Maximum Frequency	114.038MHz

Table 1. Synthesis Results in the FPGA XC2S200

Figure 10 shows the FPGA resources implemented in the design. The hardware designed in the FPGA clearly represents the three embedded microcontrollers and the SPI protocol controller, everything inside the programmable integrated circuit.
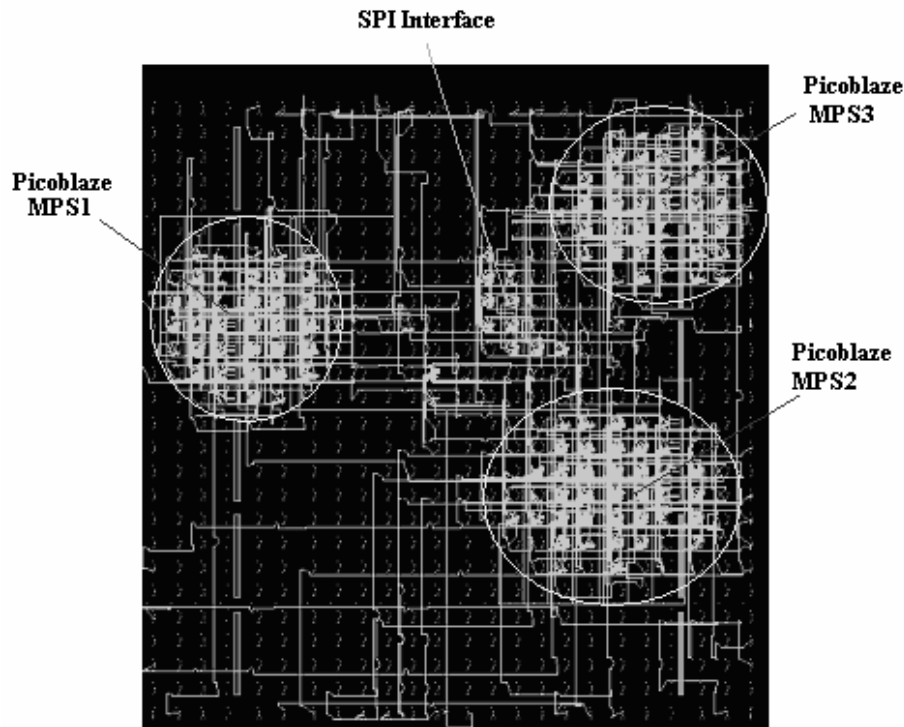

Figure 10. Resources implemented in the FPGA

4. Conclusions

A Real Time TCP/IP Control of Modular Production Systems with FPGAs was implemented. We worked on a system of embedded microcontrollers that carry out the complex finite state machine of each Production module. Figure 10 shows the structure in the FPGA with three Picoblaze microcontrollers and the SPI controller. Each Picoblaze design can work independently or in sequence. The interface between the FPGA and the RCM3700 rabbit Core was also carried out. The SPI controller was designed and the rabbit microcontroller was connected with the Ethernet interface (RJ-45 cable) to internet. With this system it is possible to control the productions systems with any PC with internet connection from anywhere. At present, we are working on the improvement of other network interfaces such as GPRS/GSM.

6. REFERENCES

[1]   User's Manual of Testing MPS System, Festo, Inc., 2005.

[2]   Le Thuyen, Renner Frank, Glesner M., "Hardware in-the-loop Simulation –A Rapid Prototyping Approach for Designing Mechatronics Systems", Proceedings, 8th IEEE International Workshop on 24-26 June 1997. Page(s) 116-121.

[3]   Rogers, G.G., and Bottaci L. "Modular Production Systems: A New Paradigm for Manufacturing", International Journal of Intelligent Manufacturing, Volume 8, Issue 2, April 1997, Chapman & Hall, London, England, pp 147-156.

[4]   RabbitCore RCM3700 User's Manual, Z-World, Inc., U.S.A. 2003 – 2004.

[5]   Xilinx System Generator v7.1 User Guide - PicoBlaze Microcontroller, Xilinx, Inc., U.S.A. 2005

[6]   Grega Wojciech, "Hardware-in-the-loop simulation and its application in control education", 29th ASEE/IEEE Frontiers in Education Conference, November 10 - 13, 1999 San Juan, Puerto Rico.

[7]   Masashi, O., Tatsuya, F., Masaguki T. and Shinji Y. "Development of Autonomously Mobile Robot with Control System Connected by TCP/IP Network" 27th Annual Conference of the IEEE Industrial Electronics Society, 2001.

[8]   Clive Maxfield, "The design warrior's guide to FPGA's", Elsevier 2005.

[9]   Hyder, K. and Perrin B."Embedded Systems Design using the Rabbit 3000 Microprocessor: Interfacing, Networking, and Application Development", Elsevier 2005.

[10]   Heinkel, U. "The VHDL Reference: A Practical Guide to Computer-Aided Integrated Circuit Design including VHDL-AMS", Chichester, New York: Wiley, 2000.

[11]   Yalamanchili, S. "Introductory VHDL: From Simulation to Synthesis", Prentice Hall, New Jersey 2001.