Solutions for Increasing the Number of PC Parallel Port Control and Selecting Lines

Mircea Popa

Abstract: *The paper approaches the problem of control and selecting possibilities offered by the PC parallel port. Although it is the most frequently used PC port for controlling external applications, the parallel port has severe limitations in generating chip select and control signals. Their number can be increased by using the parallel port data lines, extra hardware and specific routines. Higher speed can be obtained if the EPP and ECP modes are used. Techniques for increasing the number of chip select and control signals generated by the PC parallel port are presented and analyzed.*

Key words: *PC Parallel Port, IEEE 1284, Enhanced Parallel Port (EPP), Extended Capability Port (ECP)*

INTRODUCTION

A lot of applications exist in which the PC controls and monitors the external world. There are 4 ways for connecting an external application to a PC: the PCI and ISA slots, the serial port, the USB port and the parallel port. The first solution has the advantages of the speed from the motherboard and a great number of signals but it requires to open the PC and to connect boards directly to the slots from the motherboard. The disadvantage is that the operation may create hardware damages.

The serial port is the PC oldest external port, it is well known, simple but it offers a small number of lines and the voltage levels are different from those used in decision logic. The serial port is recommended only for transferring data at long distances.

The USB port offers high speed and a complex protocol which permits to link many devices to the same USB port. But it is a serial port too and there are only two data lines on which the logical levels are represented by differential voltage. Consequently, there are not lines for control purposes.

The parallel port is the best choice and the prove is the fact that it is the most used PC port for controlling and monitoring the external world. It has more lines than the two serial ports above mentioned and it offers more speed than the serial port. It is very well known and easy to program. If it is compared to the other solutions its disadvantages are: less speed than the USB port and the PCI and ISA slots and less lines than the PCI and ISA slots.

Although the parallel port offers several lines, it has severe limitations in generating chip select and control signals. There are a lot of applications which need more chip select and control signals. This paper presents some techniques for increasing the number of such signals, generated by the PC parallel port. The solutions are compared theoretically and practically.

The next section briefly describes the PC parallel port, the third section presents techniques for extending the chip select and control lines for standard and bidirectional modes, the fourth section presents similar techniques for EPP and ECP modes and the last section illustrates the conclusions.

THE PC PARALLEL PORT – A BRIEF PRESENTATION

A lot of works describe the PC parallel port and how can it be used in control and monitoring applications. Good examples are [2] and [5]. The parallel port consists in the parallel interface and a 25 pin, D type, connector situated on the PC back side. The parallel interface is made of all the circuits and the command, data and state registers which ensure the parallel communication between peripheral equipments or external applications and the PCI or ISA buses. It can be implemented on an extension board, at the older PCs or on the motherboard, at the newer PCs.

The connector has 17 signal lines and 8 ground lines. The signal lines are divided in: 8 data lines, 4 control lines and 5 status lines. The data lines were unidirectional at the original parallel port but became bidirectional at the following variants. The control lines are for dialog and control purposes and the status lines are for dialog and status indication purposes.

The parallel interface includes registers whose bits are assigned to the external signals of the connector. The registers are addressed by the processor as contiguous blocks of 3 registers starting from a base address. The most used base address is 378h, other alternatives are 278h and 3bch. All the parallel ports have 3 registers called Data Port, Status Port and Control Port and the newer parallel ports have more registers too.

A continuous evolution of the PC parallel port took place. At the beginning the parallel port had a unique data transfer mode, called Centronix or Standard, unidirectional and slow. The present parallel ports maintain this mode and the transfer rate is 50 – 150 Kbit/s. Two data transfer modes were added later: the Nibble mode and the Bidirectional mode. The Nibble mode was the first one which permitted reverse transfer of a byte but it required that the byte is divided in two nibbles (4 bits) and only one nibble is transferred at a time. The Bidirectional mode ensures the transfer of a whole byte in any direction at a time. All the above described modes are controlled only by program, meaning that they are slow. The last step was done by adding two fast bidirectional modes, EPP ("Enhanced Parallel Port") and ECP ("Extended Capability Port") which ensure a transfer rate up to 2 Mbit/s. From the 1994 year, the PC parallel port is standardized by the IEEE 1284 standard which describes the structure and the operation modes of a modern parallel port.

The PC parallel port has limitations in generating chip select and control signals and those limitations come from the existence of only 4 control signals. It means 16 chip select and 0 control signals or 8 chip select and 1 control signals or 4 chip select and 2 control signals or 0 chip select and 4 control signals. A lot of applications need more such signals and the increase of their number became necessary.

There are papers which approached this problem. The reference [1] presents a solution which uses a FPGA for connecting a real time vision system to the PC parallel port and in [3] a solution for a data acquisition system based on a DSP circuit connected to the parallel port through the EPP protocol is presented. Another implementation for the EPP protocol is described in [7] which offers good performances but is rather complex. The reference [4] presents a solution for generating individual commands but only for a small number of external elements and [6] describes a solution for designing physical interface for the IEEE 1284 standard.

The techniques which will be described in the following sections increase the number of chip select and control signals generated by the PC parallel port using extra hardware and specific routines. The solutions are cheaper and easier to implement than the ones from the mentioned papers and does not need specialized circuits as FPGAs or DSPs.

EXTENDING THE CHIP SELECT AND CONTROL LINES FOR STANDARD AND BIDIRECTIONAL MODES

The first solution uses the data lines which will be latched in registers and their outputs will be the chip select and control signals. Two original control lines from the parallel port one for loading the data in registers and the other for enabling the registers outputs will be necessary. Fig. 1 presents such an example. The register outputs are control inputs for an 8255 circuit which increases, at its turn, the number of data lines. The control of the 8255 circuit is not possible with only the original parallel port control lines. The specific routine implements the following sequence:

- a configuration at the Data Port is generated which constitutes the chip select and control signals for the 8255 circuit;
- the configuration is loaded in register using a line from the Control Port; the registers outputs are not yet enabled;

- another configuration at the Data Port is generated which is the data or command word for the 8255 circuit;
- the chip select and control signals for the 8255 circuit are enabled through another line from the Control Port.

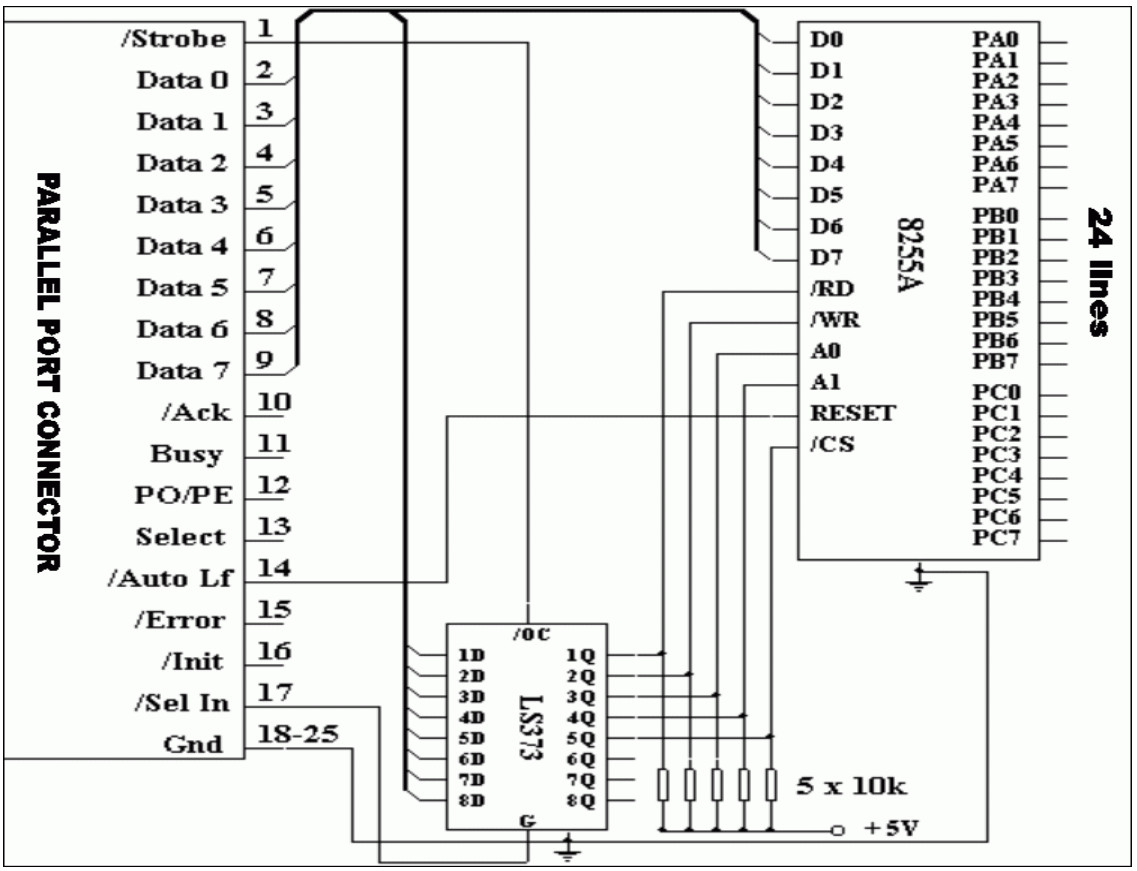

Fig. 1 Generating chip select and control signals with a register

The described solution has the following advantages:

- the chip select and control signals can be simultaneously generated only by enabling the outputs of a register;
- more combinations of chip select and control signals can be generated: from 16 chip select and 0 control lines to 0 chip select and 16 control lines, passing through middle combinations such as 8 chip select and 8 control lines.

The disadvantages are:

- a moderated increase was obtained; at the original port combinations from 16 chip select and 0 control lines to 0 chip select and 4 control lines are possible and with this solution the domain was extended to 0 chip select and 16 control lines;
- the loading and the enabling of the chip select and control signals is done only by program; in the example from fig. 1, 8 instructions (assembly language) for loading and 4 instructions for enabling are necessary, resulting speed limitations; for instance considering a 433 MHz Celeron processor and taking into account the instructions for transferring the data too (assembly language), the chip select and control signals period was measured as 8 μs.

Another solution consists in using registers followed by decoders. Fig. 2 presents such an example.

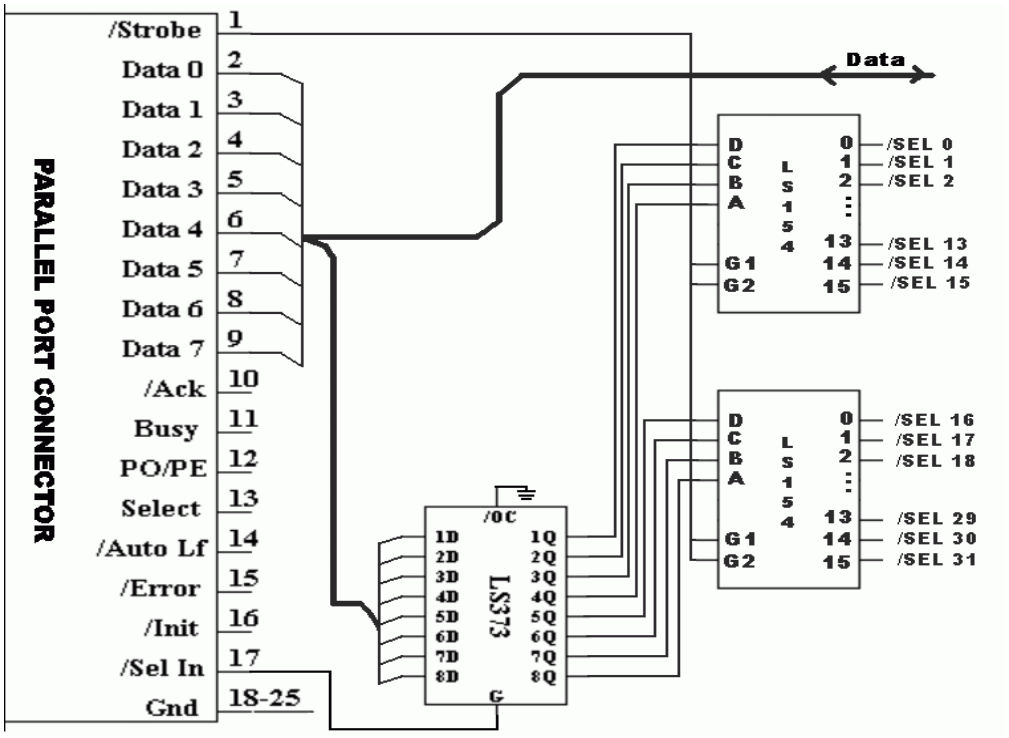

Fig. 2 Generating chip select and control signals with registers and decoders

The following sequence is implemented by the specific routine:

- a configuration at the Data Port is generated, it is decoded and chip select signals will be obtained;
- the configuration is loaded in register using a line from the Control Port; the registers outputs are not yet enabled;
- another configuration at the Data Port is generated which is the data or command word;
- the chip select signals are enabled through another line from the Control Port.

EXTENDING THE CHIP SELECT AND CONTROL LINES FOR EPP AND ECP MODES

The EPP and ECP modes are the most performant ones. They are bidirectional and ensure high data transfer rate, up to 2 Mbytes/s. Their speed comes from the fact that the dialog for transferring data is achieved by hardware and not by software as in the other transfer modes.

Fig. 3 gives a solution for extending the number of data and address lines generated in the EPP mode. The transfer is unidirectional but the solution can easily be modified for a bidirectional transfer.

A unique OUT instruction is needed for transferring a data or an address. Since to load the data and the port address is necessary in each case, the routine which transfers a data and address was measured to have 2,8 μs, the technical conditions being the same as in the second section. The number of chip select and control lines is from 256 chip select and 0 control lines to 0 chip select and 8 control lines. The domain is obtained from the 8 bit register which loads the addresses.

Fig. 4 shows a solution for generating the data and addresses in the ECP mode. The ECP mode specific channel concept is used for generating the addresses. The channel addressing differs from the usual port addressing. It assumes that a unique physical equipment, with a unique parallel port, includes more logical equipments. An example is a fax/ modem/ printer multifunctional equipment. The channel addressing permits to transfer data with the fax even if the printer is busy because they are two distinct logical

equipments. In the ordinary port addressing there is no distinction between the physical and logical equipments, consequently it is not possible to transfer data with one of them if the other is busy.

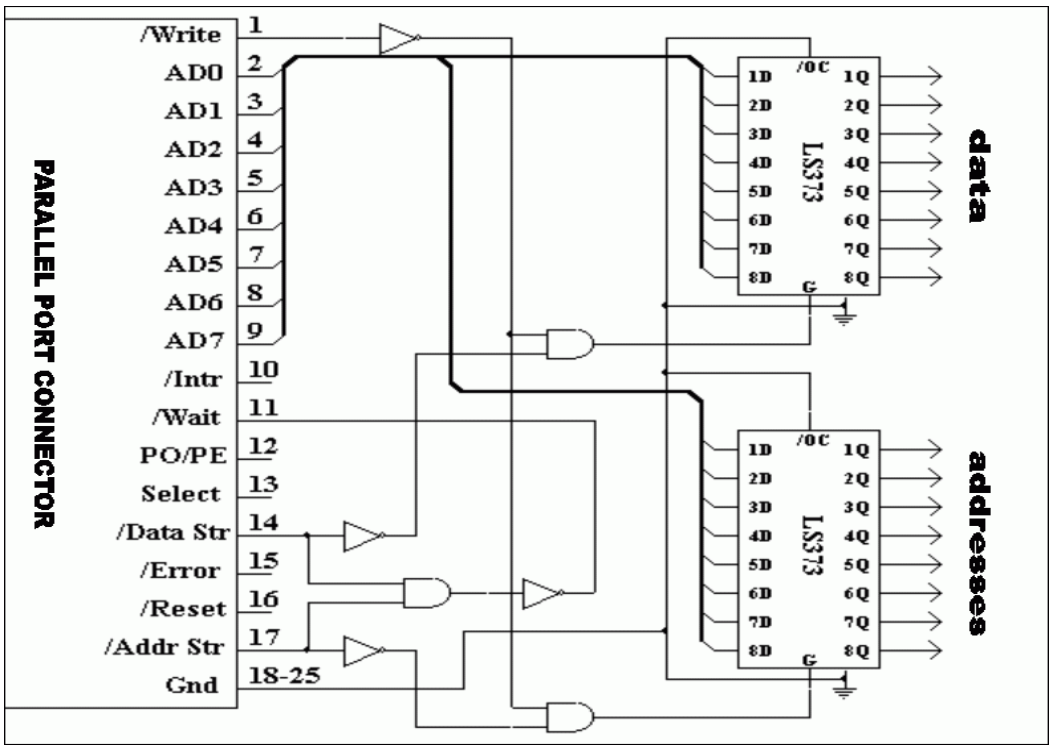

Fig. 3. Generating data and address lines in the EPP mode

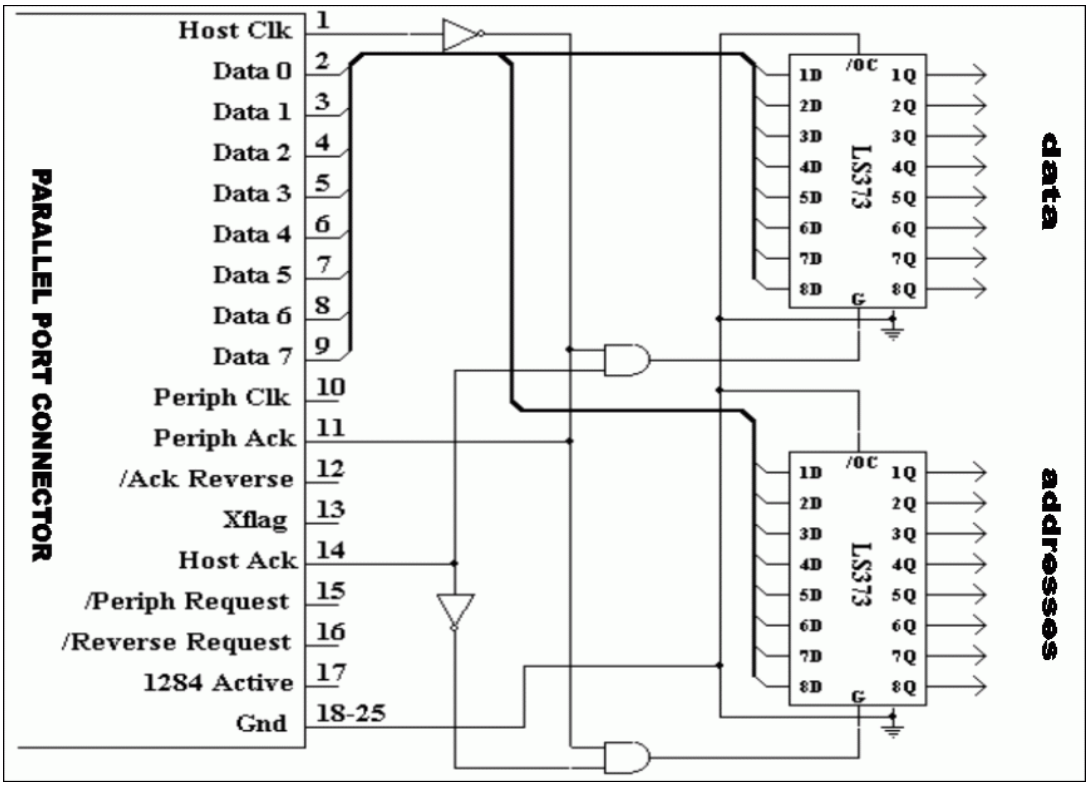

Fig. 4. Generating data and address lines in the ECP mode

As in the EPP mode, a unique OUT instruction is needed for transferring an address or a data. Two more instructions will be necessary for each transfer, namely one for loading the data and one for loading the port address. In the same experimental conditions

as the previous ones, the routine which transfers a data and an address have 2,65 µs. The number of chip select and control lines is from 128 chip select and 0 control lines to 0 chip select and 7 control lines. As in the EPP mode, an 8 bit register is used for loading the addresses but the most significant bit is always 1. This is necessary because the configuration used represents a channel address and the configurations with 0 in the most significant bit have a different meaning, namely they are compression codes.

Using the EPP and ECP modes for extending the number of chip select and control lines offers some advantages, compared to the standard and bidirectional modes, such as: a higher transfer rate, a higher number of chip select and control lines and fewer instructions.

CONCLUSIONS

The paper described some solutions for increasing the number of chip select and control lines obtained from the PC parallel port. Two types of solutions were presented: for the slower and for the faster data transfer modes.

The solutions for the standard and bidirectional modes use registers or registers plus decoders. They are slower because the entire transfer is done by program. A moderate increase was obtained.

The EPP and ECP modes are faster because the transfer is hardware controlled. The experiments done in the same technical conditions, meaning the same hardware, the same operating system and the same instruction subset showed a speed at least 3 times higher in the EPP and ECP modes. Also, an important increase of the chip select and control lines was obtained.

Using the same solution types more chip select and control lines can be obtained with extra hardware and specific routines.

REFERENCES

[1] Arribas, P. C., F. M. H. Macia. FPGA Board For Real Time Vision Development System. Proceedings of ICCDCS2002 Fourth International Caracas Conference on Devices, Circuits and Systems, Aruba, Dutch Caribbean, 17 – 19 April 2002, pp. 1 - 6

[2] Axelson, J. Parallel Port Complete. Lakeview Research, USA, 1997

[3] Bengtsson, M., T. Lofnes, V. Ziemann. A DSP controlled data acquisition system for CELSIUS. Nuclear Instruments and Methods in Physics Research A 441 (2000), pp. 76 – 80, Elsevier, Nl.

[4] Conrad, J. M., J. W. Mills. A PC – Based Controller for the Stiquito Robot. Circuit Cellar Ink, Issue 108, July 1999, pp. 18 – 22

[5] Gadre, D. V., D. V. Gadre. Programming the Parallel Port: Interfacing the PC for Data Acquisition & Process Control. CMP Books, USA, 1998

[6] Innes, C., S. Clark. IEEE 1284 transceivers aid in successful implementation of the IEEE 1284 standard. International IC – Taipei Conference Proceedings, May 2000

[7] Viscovini, R. C., A. Scalabrin, D. Pereira. An enhanced parallel port interface for controlling and monitoring experiments. Measurement Science and Technology, 11, February 2000, pp. 23 – 27, Electronic Journal

ABOUT THE AUTHOR

Prof. Mircea Popa, PhD, Computer and Software Engineering Department, Faculty of Automation and Computers, University "Politehnica" Timisoara, ROMANIA, Phone: +40 256 403263, E-mail: mpopa@cs.utt.ro, http://www.cs.utt.ro/~mpopa