

PARALLEL PORT INTERFACE

by **Alexis Artes**

The Parallel Port Interface is the second part of the Touch Screen Implementation. The interface has two parts: the software and the circuit.

SOFTWARE

The Visual Basic Programming Language(VB) is the software used for the interface. The VB is a Window's based graphical programming language. Being a Window's based language, programming is as easy as drag-and-drop. However, its strength is also its weakness in terms of I/O programming.

The I/O requests must first go through Windows. There is a need for a Dynamic Link Library that would be called by the program to access the I/O ports. For this program, the vbio.dll is used in the program. It is put into the windows/system directory.

In the actual program, the vbio.dll is first called to enable I/O port control. After enabling the port control, the parallel port must be first be set.

The Parallel port has 3 built-in ports: the data, the control and the status port. For Input, the status port is the traditional port that is used. The status port can accept 5 inputs. In this interface, the data port, which can accept 8 inputs, is used. The data port is selected for expandability purposes. It can be seen later that the circuit is also applicable even if the status port is used.

The Data port in default is for data output and cannot read input. However, setting bit-5 of the control port to high enables the bi-directional function of the Parallel port. It means, the data port can now read the data being sent through it. However, this is only available on true bi-directional ports.

After enabling data input through Data port, the port will then be read. If there is an input, it will be stored in a variable that will be used by the main program. This first input is the X-coordinate of the screen. After storing the x-coordinate, the bit one of the control port will then be set to high. This will set the data selector of the hardware to let the y-coordinate to be passed to the data port. Therefore, after the x-coordinate polled, the y-coordinate is polled. The port will be polled every 10 millisecond.

HARDWARE

The hardware part of the interface uses 13 chips: 1 (4 2-to-1) data selector, 2 (8-3) encoder, 2 inverter chips, and 8 opto-isolators.

The sensors of the screen are connected to the input of the 2 encoders. The trigger input of the encoder is low. However, the y-coordinates is always low if no there is no input. Therefore, an inverter is needed in between the sensor and the encoder.

The output of the encoders is then feed to the data selector. The data selector will only output the inverted output of one of the encoder. The data to be output by the selector is controlled by the status port bit 1. The output will pass through the optoisolators for protection. The final output will then be sent to the data port.

The main strength of the interface is its expandability. The present interface only utilizes 4 of the 8 lines of the Data Port making an 8 by 8 resolution, 16 bits resolution. Therefore, duplicating the circuit and utilizing the 4 left lines of the data port and the 4 lines of status port can easily expand the resolution. There will be a total of 12 lines making an 144 bit resolution.

→ the program listing is on the Software part.

→ the circuit is shown below.

