A Realized SONY PS2 1-to-4 Joystick Multiplexer Interface[*]

Chua-Chin Wang†, Rong-Sui Kao, Po-Ming Lee, Yo-Lih Huang
Dept. of Electrical Engineering
National Sun Yat-Sen University
70, Lian-Hai Road
Kaohsiung, Taiwan 80424
Tel : 886-7-5252000 ext. 4144
Fax : 886-7-5254199

Abstract: - In this work, an 1-to-4 joystick multiplexer interface which can be attached to SONY PS2 (playstation 2) is developed. The multiplexer interface can allow 4 persons to play simultaneously through one port at the original game console. Up to 8 players can be supported to use the game console at the same time if both ports of the PS2 console are plugged with our MUX interface. The FPGA implementation in the field testing shows there is neither delay nor jitter when the multiplexer is hooked up with the console. The multiple player entertainment effect can be drastically enhanced by the usage of such an interface.

Key-Words: - multiplexer interface, PS2, joystick, MUX, FPGA, SONY

1 Introduction

Since Sony started to market their impressive playstation (known as PS) in 1994, more than 73 millions of the PS consoles have been sold [1]. In April, 2000, Sony debuted its new generation game console, ``PS2". So far, more than 3 millions of PS2 consoles has been sold. However, due to some design considerations, there are only two joystick ports as well as the memory card ports on PS2. It makes no more than two players be able to enjoy the same console. In this paper, we design and implement an 1-to-4 joystick multiplex interface, and hook it up with the PS2 console successfully. Multi-player games can be fully enjoyed by up to eight players at the same time. The entertainment effect is highly increased, while the overall cost of the game accessaries is reduced.

2 Polling MUX Interface

In a multi-player mode, an additive module is required to receive the signals from 4 joysticks, and communicate with PS2 using these signals. The most important of all is that every player using this additive module should feel like he is directly using the console without any delay or jitter.

2.1 MUX Design

A simple thought is to design an 1-to-4 joystick multiplex interface (MUX), as shown in Fig 1, to extend the capability of the console. The function of the MUX is sequentially multiplexing the signals from every signal at the extended port. In other words, the signals from different joysticks are time-multiplexed by the MUX. The other side of MUX is in charge of communicating with the PS2 console. The game console will poll the 4 joysticks one by one to detect the response of each player. According to the physical measurement, the PS2 console senses the signals of the 4 joysticks roughly 30 times/sec. Thus, if the speed of time multiplexing of the MUX can reach this mark, the inputs from all of the players who are plugged in the MUX are ensured to communicate with the console in real time. They will never feel the delay nor jitter effect to lose the fidelity of the game itself.

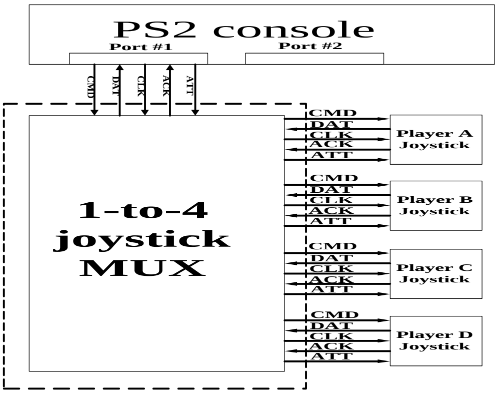

Fig 1: The design of the 1-to-4 joystick MUX

[*] This research was partially supported by Nation Science Council under grant NSC 89-2215-E-110-014.

† Contact author

2.2 Polling and Command Signals

There are a total of five signals to execute the communication protocol between the console and one joystick. Table 1 summarizes the description of these signals.

signal	description
CMD	command or data to the joyst the memory card from the co
DAT	data from the joystick or the memory card
CLK	clock
ACK#	acknowledge signal from the joystick or the memory card
ATT#	select the joystick or the memory card

Table 1 signal description

Please be noted that the "#" indicates the logic low active in the above table.

2.3 Polling Operation Sequence

The entire polling operation sequence is deemed as a large FSM (finite state machine) which comprises 30 states, i.e., state ONE to state THIRTY. The details are given as follows.

Initialization : Referring to Fig. 2 and Table2, it is the scenario after power on or reset. DAT is set up to FF in state 1st. As soon as ATT is detected to be low, the data on CMD is sensed at the rising edge of CLK. When CMD = 21, the FSM moves to state 2nd, and echoes a low ACK to notify the console that the data are correct. Then, the sequence is simply as that shown in Table 2. After the entire sequence is done, the FSM is back to state 1st and monitors whether ATT is low.

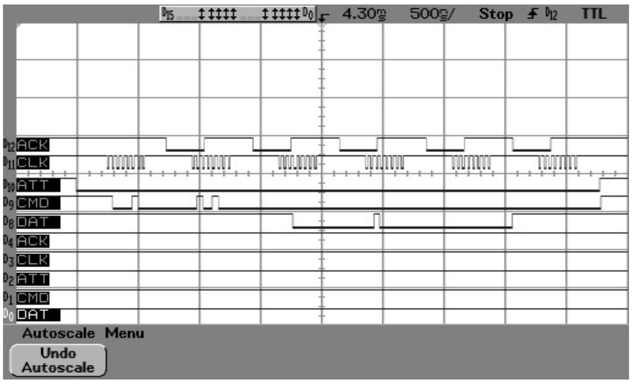

Fig. 2 : initialization sequence

CMD	21	12	00	00	00	00
DAT	FF	FF	07	04	00	FF
state	1st	2nd	3rd	4th	5th	6th

Table 2 : The state transition sequence in the initialization

joystick D : The operation is very much similar to that of the initialization sequence. As soon as the CMD and DAT are received correctly, the state 12th terminates and the right to use the joystick port of the console is transferred to joystick D. When the operation is done, the FSM moves back to state 1st in which DAT is set to FF and ATT is monitored. Fig. 3 and Table3 reveals the sequence of this part.

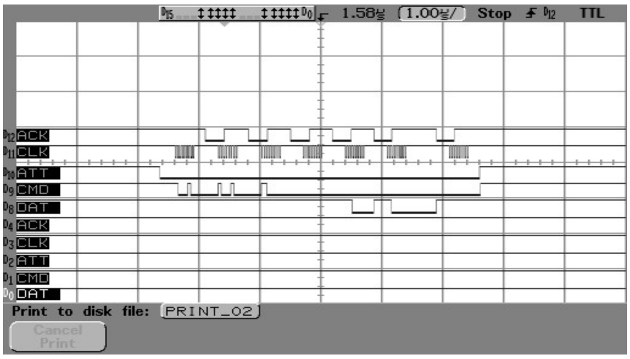

Fig. 3 : the operation sequence of joystick D

CMD	21	21	03	00	00	00	00
DAT	FF	FF	FF	FF	07	03	FF
state	1st	7th	8th	9th	10th	11th	12th

Table 3 : The state transition sequence for joystick D

joystick C : Again, the operation of joystick C is also very much similar to that of the initialization sequence. As soon as the CMD and DAT are received and verified correctly, the state 18th terminates and the right to use the joystick port of the console is transferred to joystick C. When the operation is done, the FSM moves back to state 1st in which DAT is set to FF and ATT is monitored. Fig. 4 and Table 4 reveals the sequence of this part.

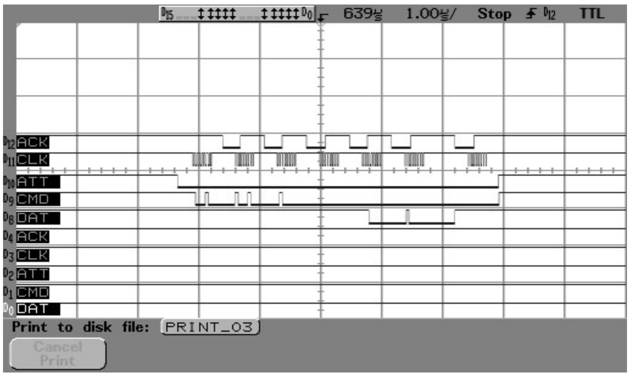

Fig. 4 : the operation sequence of joystick C

CMD	21	21	02	00	00	00	00
DAT	FF	FF	FF	FF	07	02	FF
state	1st	2nd	14th	15th	16th	17th	18th

Table 4 : The state transition sequence for joystick C

joystick B : The operation of joystick B is also deemed as another bran of the whole FSM. When the CMD and DAT are received and verified correctly, the state 24th terminates and the authority of the joystick port of the console is transferred to joystick B. As soon as the operation is done, the FSM moves back to state 1st in which DAT is again set to FF and ATT is being monitored. Fig. 5 and Table 5 reveals the sequence of this part.

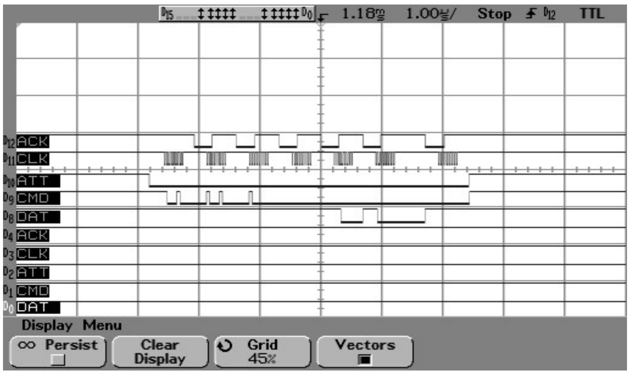

Fig 5 : the operation sequence of joystick B

CMD	21	21	01	00	00	00	00
DAT	FF	FF	FF	FF	07	01	FF
state	1st	19th	20th	21th	22th	23th	24th

Table 5 : The state transition sequence for joystick B

joystick A : At last, the operation of joystick A is shown Fig. 6 and Table 6. When the CMD and DAT are received and verified correctly, the state 30th terminates and the authority of the joystick port of the console is transferred to joystick A. As soon as the operation is done, the FSM moves back to state 1st in which DAT is again set to FF and ATT is to be monitored.

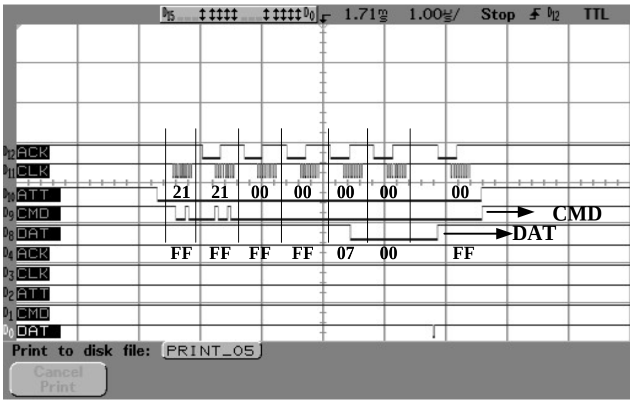

Fig. 6 : the operation sequence of joystick A

CMD	21	21	00	00	00	00	00
DAT	FF	FF	FF	FF	07	00	FF
state	1st	25th	26th	27th	28th	29th	30th

Table 6 : The state transition sequence for joystick A

2.4 Architecture of MUX

In order to carry out the design of Fig. 1, several circuit modules are physically implemented by Verilog RTL hardware description language as shown in Fig. 7. The PS2 console communicates with two modules in our design, which are “memory card signal encoder and decoder” (MEM) and “joystick signal encoder and decoder” (Joy). The former module is charge of access the data stored on each of the memory cards plugged in the MUX. This feature makes each player can own his (or her) game progress and data. Meanwhile, the Joy executes the polling action to detect the existence of joystick A, B, C, and D. Since the polling scheme is adopted, the joystick can be “hot plugged,” which means any player can attend and leave at will without any problem. That is, the console does not have to be shutdown.

“Auto-Reset” in Fig. 2 reset every block in the MUX as soon as ATT is raised high, which indicates that none of the joysticks nor the memory cards is selected. The FSM of the MUX should then be reset to its initial state.

“MUX-Ack” is responsible of counting the number of clocks. Since CMD is an 8-bit data, the MUX has to receive the eight bits consecutively. When the count is 8, ACK is pulled low and delivered to the console to signal that the CMD is successfully received. “Joy-selector” is basically a decoder to determine which port is accessed by the console.

3 Implementation and Testing

To verify our design, a physical testing site is built. Referring to Fig. 8, the black box is the PS2 game console of SONY. An Altera demoboard with FLEX 10K100A RC240-1 FPGA chip is employed to realize the MUX. At the low-right corner, signal grabbers are installed to detect whether any signal loss or attenuation is found. The entire design of the MUX is implemented by Verilog and programmed into the on-board EEPROM of demoboard.

Fig. 9 shows the scenario when the MUX is correctly hooked up with the console. It is found that the console polls the ports by the sequence of D, C, B, A. As soon as we hot plug a joystick at port A, the signals change to be those shown in Fig. 10. It can be easily found that the signals from joystick A after the label "A" are activated. They indicate that instructions and commands of the joystick A are successfully delivered to the console while the console echoes the acknowledge and grant signals.

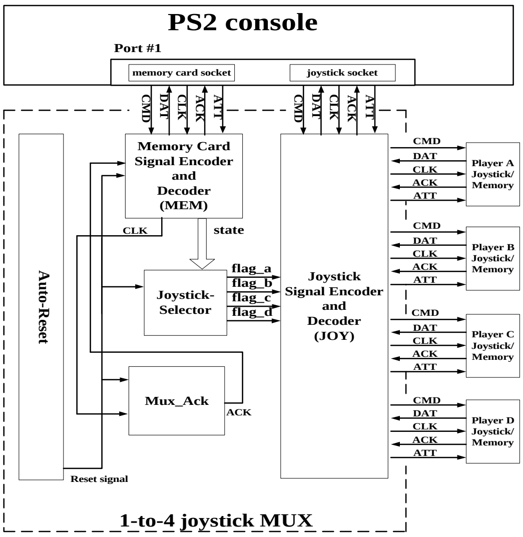
PS2 console
Port #1
memory card socket
joystick socket
CMD
DAT
CLK
ACK
ATT
Auto-Reset
Memory Card Signal Encoder and Decoder (MEM)
CLK
state
Joystick-Selector
flag_a
flag_b
flag_c
flag_d
Joystick Signal Encoder and Decoder (JOY)
Mux_Ack
ACK
Reset signal
Player A Joystick/ Memory
Player B Joystick/ Memory
Player C Joystick/ Memory
Player D Joystick/ Memory
1-to-4 joystick MUX

Fig 7: The schematic of the proposed MUX

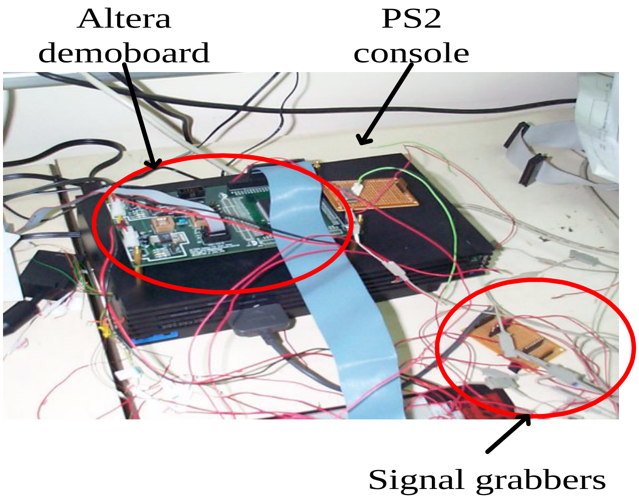

Fig 8: The connection between Altera demobroad and PS2 console

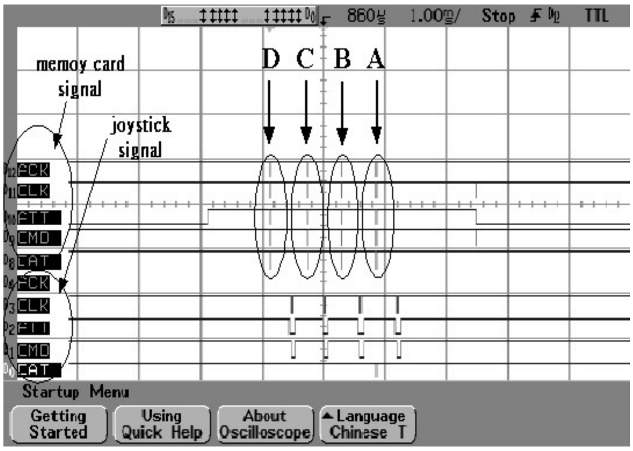

Fig. 9 : signals of MUX without any joystick

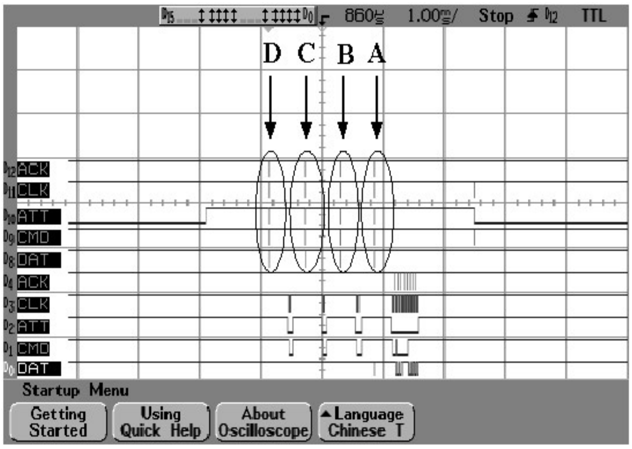

Fig. 10 : signals of the MUX when joystick "A" is plugged

4 Conclusion

In this paper, we propose an 1-to-4 joystick multiplex interface for the top seller game console, PS2. A single Altera FLEX 10K100A RC240-1 FPGA demoboard is adequate to implement the entire prototypical multiplex interface. In the field tests, the prototype can hooked up to 4 joysticks and work correctly with the PS2 console. Neither delay nor jitter is found. By the similarly thought, we may add wireless solutions, e.g., IrDA [2] or RF [3], in the future to provide wireless joystick interfaces.

References:

[1] http://www.us.playsation.com

[2] F.-W. Gee, *Wireless Transmission Technology by IrDA*, Reading: Course Notes, July 1999.

[3] Texas Instruments, *TRF6900 -- single-chip RF transceiver*, May, 2000.