A Study of BUS Architecture Design for Controller of Nuclear Power Plant Using FPGA

Dongil Lee [a*], Donghwa Yun [a], Sungjae Hwang [a] Myeongyun Kim [a] Dongyun Lee [a]
[a]*PONUTech Co., Ltd., Research Institute., 145, Anam-ro, Seongbuk-gu, Seoul, Korea*
[*]*Corresponding author: ldi1838@ponu-tech.com*

Abstract

Generally, the controller in NPP (Nuclear Power Plant) has parallel bus structure which is well-used in the past. Recently, CPU (Central Processing Unit) operating speed and communication rate have been more technically improved than before. However, whole system is been a degradation of performance by electronic and structural limitation of parallel bus.

Transmission quantity and speed have a limit and need arbiter in order to do arbitration because several boards shared parallel bus. Arbiter is a high complexity in implementing so it increases component per chip. If a parallel bus uses, it will occurs some problems what are reflection noise, power/ground noise (or ground bounce) as SSN (Simultaneous Switching Noise) and crosstalk noise like magnetic coupling [1].

In this paper, in order to solve a problem of parallel bus in controller of NPP (Nuclear Power Plant), proposes the bus architecture design using FPGA (Field Programmable Gate Array) based on LVDS (Low Voltage Differential Signaling).

1. Introduction

PONUTech Co., Ltd. has studied developing and upgrading controller of NPP. One of this is research and development of using FPGA. This paper proposes a bus architecture which solved a bottleneck to affect all system. Bus controller is implemented by FPGA and serial link is made by physically based on the LVDS.

2. Bus Design

This section describes about the architecture of bus.

2.1 Bus Architecture

The controller consists of divided by function power supply, backplane, CPU, I/O (Input/Output), communication and etc. boards. Identify number of all boards are numbered from position of slots, each boards have own channel of bus.

All bus channels are connected to bus controller board. (Generally built in CPU board) Controlling of all bus is controlled by bus controller board, and has independent of each other bus structure.

Proposing basic bus architecture is shown in Fig. 1.

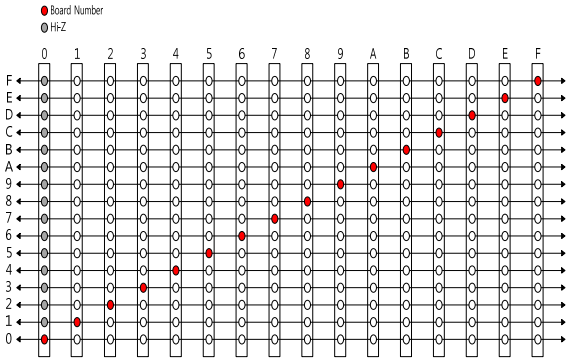

Fig. 1. Basic model of bus architecture

2.2 Transmission method

Data of all channels is synchronous transmitted by controller. Transmission way has full-duplex and half-duplex. Each channel can select separate speed and transmission method.

If the CPU module is designed to multiplex in system, the bus controller can contain several CPU data in each channel.

2.3 Other Feature

In case of the Hot-swap event, bus controller and each board run to recovery mode transmission for the recovery.

In Addition, if the channel needs to transmit heavy data, use the special mode to transmit full-duplex.

Used transmission data on all channels is constructed by package. Data is designed by 32 bits unit for easy to use DSP.

3. Implementation

In this section describe about implementation of bus.

3.1 Serial Path

A bus is implemented from parallel path, used for memory interface or inter-processor communication based on the single signaling, to serial path based on LVDS.

Structural difference between the SE (Single-Ended) used in parallel bus and the DM (Differential Mode) used in serial bus is shown in Fig. 2 [2].

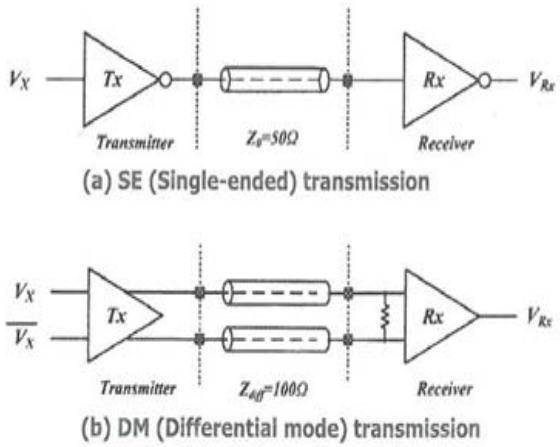

Fig. 2. Single-ended transmission vs differential mode transmission

When the occurred common-mode noise effect and ground bounce, difference of the result from the SE and the DM is shown in Fig. 3. SE is distinguished between 0 and 1 by reference voltage but DM is distinguished by to the difference in voltage level. So you can know that is not affected by noises [2].

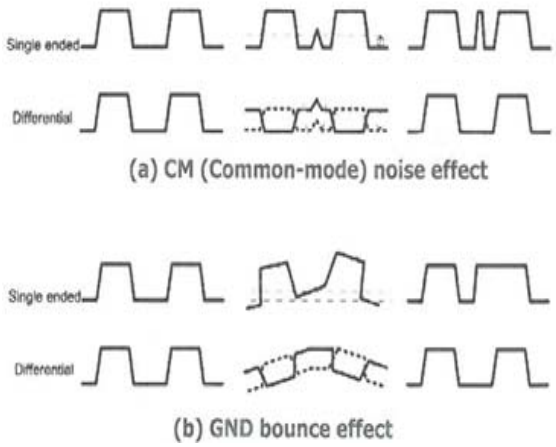

Fig. 3. Common-mode noise effect vs GND bounce effect

3.2 Bus controller

Bus controller is implemented by a FPGA having master/slave modes. The bus-controller of master-mode control all channel in bus.

3.2 Board based on FPGA

Target board was consisted of using total three FPGAs as shown in Fig. 4 One system FPGA and two bus-controller FPGAs (proportional to the number of the buses). System FPGA was designed about an operating should be done in board. And two bus controllers are controlling the each bus group packed with several channel.

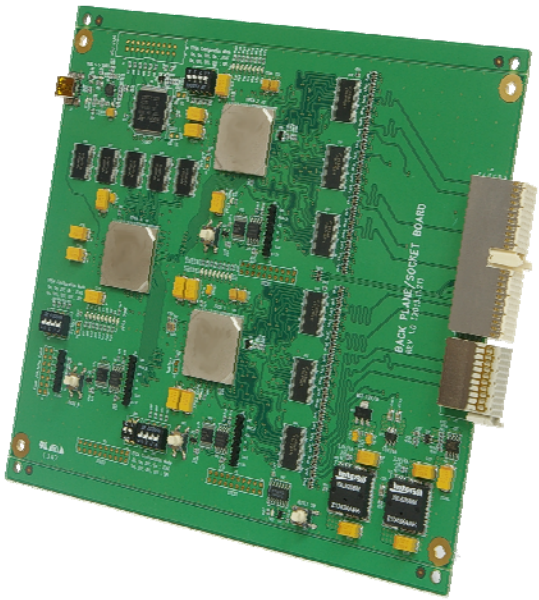

Fig. 4. Target board based on FPGA

3.3 System designing based on FPGA

To organize the system of bus communication between channels needs controller, transmitter and receiver boards. The system consists of one bus controller board and two boards to do communication like I/O data package based on FPGA. Fig. 5 is shape that each board has been mounted in a backplane.

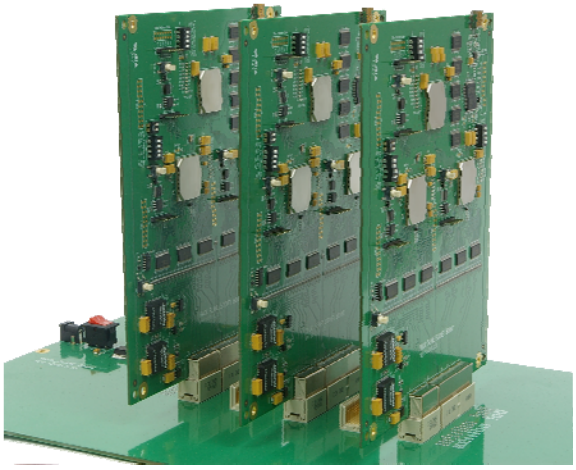

Fig. 5. One of the systems based on FPGA

4. Result

When the system is composed by serial bus, 625,000 bits can transmit on the base of 25ms response time, 25 MHz clock rate. If it is organized with triple CPU, it can transmit about more than 125,000 bits except state frame. Table I shows a comparison between 32 bit parallel-bus is used and serial bus is used.

Table I: Parallel Bus vs Serial Bus

Item \ Bus	Parallel bus	Serial bus
Bit rate	2.5Mbps	25Mbps
1 Kbyte time	102,400ns	327,680ns
1 KB x 16 CH time	1,638,400ns	327,680ns
Bit per line	1 line	2 lines
Control line per 1 CH	3	4 (with clock)
32 bits data per line	56[1)]	2
16 CH per line	56	67[2)]

1) data line (32), address line (23), control line (3)
2) data line (32), clock line (32), control line (3)

5. Conclusions

To solve the bottle-neck of the controller consisting of parallel bus applied serial bus. When the using the proposed bus, you can see to transmit many of data to do not affect to performance degradation. The bus-stability is maintained in the high-capacity high-speed communication result from that the bus is maximized of electrical advantage of a LVDS. The package and the communication way do not use commercial methods for the data security.

In the future, bus architecture will do verification of safety-class. And then it will apply to safety controller and safety system in NPP.

REFERENCES

[1] J. Y. Sim, High speed interface circuit design, POSTECH, 2010.
[2] K. R. Cho, LVDS interface circuit, Chungbuk National University, p5, p58, 2013.