

Simultaneous data transfers over peripheral bus using CDMA technique

Tatjana Nikolić, Goran Djordjević, and Mile Stojčev

Abstract - The need for an efficient interconnect architecture has been caused by continued increase of the required communication bandwidth and concurrency of small-scale digital systems. The issue of applying the code division multiple access (CDMA) technique for data transfer over peripheral bus are discussed in this paper. The proposed technique represents an efficient interconnection solution for implementation in embedded systems based on low pin-count processing elements. Eight different system configurations, in respect to the number of transmitters and receivers, are realized at register-transfer level (RTL) using VHDL. The simulation results show that the communication bandwidth is scalable as the number of transmitter-receiver pairs increase.

I. INTRODUCTION

As chip manufacturing technology improves, larger embedded systems (ESs) with more components can be constructed. Nowadays, complex embedded systems (ESs) consist of heterogeneous mix of processing-elements (PEs) - such as microcontrollers (MCUs), digital signal processors (DSPs), etc. -, and functional units (FUs) - such as accelerators and input-output peripherals - that are tailored towards meeting several system requirements, including overall system performance, power consumption, and reliability. The complex ES performance consists of two factors: computation performance and communication performance. Thanks to the development in the CMOS technology, the computation performance of integrated circuits has been increased dramatically. As computation performance rises, the communication bandwidth requirements also increase with the same rate. Inter-chip communication architecture is increasingly being regarded as one of the major obstacle for complex ES designs which should be able to transport heterogeneous traffic efficiently while still maintaining the required performance. Moreover, the number of input/output (I/O) pins and the total I/O bandwidth of a single chip have been scaling much more slowly. Therefore, the communication between multiple on- or off-chip semiconductor intellectual property (IP) blocks is becoming a dominant cost, performance, and power factor in modern embedded systems [1].

Techniques for sharing communication resources (e.g. pads, pins, and metal wires) have always been a crucial

choice for embedded systems. However, there is no standard solution for building fast, flexible, and efficient communication network which is able to connect a large number of components that have heterogeneous requirements. Most of the interconnect networks in complex ESs rely on a simple, shared-medium interconnect, i.e. buses, which apply time-division multiple access (TDMA) in order to reuse expensive inter-chip wires. However, buses suffer from resource contention issues – as the number of processing elements increases, performance degrades due to excessive conflicts [2].

Code division multiple access (CDMA) has recently been proposed as a new interconnect mechanism for next generation ESs. Compared to a conventional TDMA-based bus, a CDMA-based bus has better features concerning channel's isolation and channel's continuity in time domain since channels are divided by the spreading codes [2]. CDMA technology relies on the principle of codeword orthogonality, such that when multiple codeword are summed, they do not interfere completely with each other at every point in time, and can be separated without loss of information [3].

This paper focuses on using CDMA technique for achieving wire-efficient “point-to-multipoint” communication in ES based on low pin-count PEs. The term “point-to-multipoint” refers to a communication system in which a single transmitting unit (e.g. PE) that is located at one particular point sends separate bit sequences to multiple receiving units (e.g. other PEs, or other off-PE peripheral units) that are located at various other points. That is, a first bit sequence B_1 is sent to a first receiving unit, a second bit sequence, B_2 , to a second receiving unit, etc., and all these bit sequences are sent at the same time. In an ES, such bit sequences can represent output signals of various peripheral units located within the PE (timers, UARTs, PWMs, etc.). Although the transfer of peripheral output signals typically comprises a small fraction of the total communication bandwidth in an ES, it could be extremely demanding in terms of interconnection complexity if the PE sends each of the signals via a separate pin (wire) to the corresponding receiving units. First, when the number of signals (bit sequences) is large, the available PE's pins could easily be exhausted. Second, when the receiving units are remotely located from the PE, too much connecting wires are required. By using a “point-to-multipoint” CDMA scheme, the PE will send all of bit sequences via small and constant number of pins that

results in considerable saving in both PE's pin-count and interconnection wiring.

The rest of the paper is organized as follows. Section II identifies problem of interconnecting low pin-count devices, that have large number of integrated peripheral modules (functional units - FUs), and suggests the CDMA as a potential solution. Section III describes the basic principal of CDMA technique and its application for data transmission in complex ESs. Section IV presents detailed design of the proposed system for simultaneous data transfer over peripheral bus. Section V gives experimental results, and Section VI discusses conclusions.

II. PROBLEM DEFINITION

Designers of complex ESs permanently encounter the problem of how to increase the PE's functionality from one side, and how to decrease the pin number from the other. In general, all PEs have their I/O pin-outs defined by the chip makers. Especially complicated are the low pin-count devices, where it is common to see three or more functions multiplexed onto one pin (Fig. 1). Multiplexing leads to problems when allocating the PE's resources in ESs, where the needed peripherals such as universal asynchronous receiver transmitters (UARTs), parallel port, serial peripheral interface (SPI), timers, pulse width modulators (PWMs), etc., are found multiplexed on the same pins. In situations such as those shown in Fig. 1, by assigning one FU to a pin, prevents the others to use the same pin at the same time. Designers have the following options to deal with this problem: a) to use higher pin count PEs where the I/O lines are demultiplexed; b) to use additional devices, such as multiplexers and extra glue logic; c) to implement peripheral functions in software, what may result in lesser performance from the PE, or higher power dissipation due to faster PE operation [4].

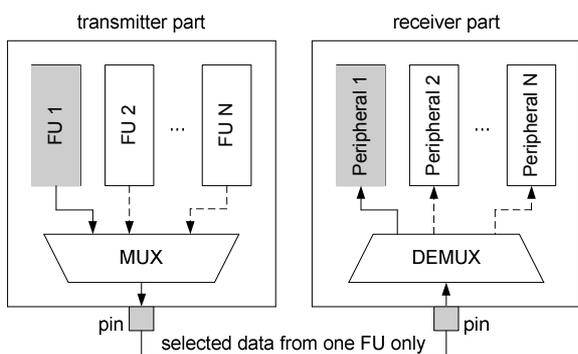


Fig. 1. N peripheral functions multiplexed onto one pin

Previously mentioned options cannot solve the problem completely because they require to build an additional hardware or to use additional processor time. An alternative solution is to implement CDMA technique which permits multiple resources to use the communication media (pins) concurrently by separating data from different

sources in the code domain [5], [6]. This solution uses CDMA multiplexer at the transmitter part and CDMA demultiplexer at the receiver part, as it pictured in Fig. 2, permitting all peripherals to use the same pin simultaneously.

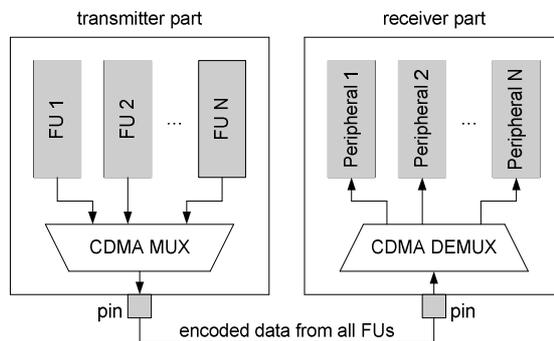


Fig. 2. N peripheral functions multiplexed onto one pin using CDMA technique

The CDMA MUX encodes each output bit sequence with a particular spreading code. The composite signal as a sum of all spreading bit sequences is simultaneously send serially, through a single output-pin connection, to all receivers. At the receiver part, a corresponding spreading code generator, to each peripheral unit, is appended. The CDMA DEMUX recover original bit sequences by multiplying the composite CDMA signal with identical spreading codes that were used at the transmitter part. This means that spreading codes are used in pairs. For each spreading code of the transmitter part, a corresponding image exists at the receiver part.

III. CDMA INTERCONNECT

CDMA is a spread-spectrum technique that applies a set of orthogonal codes to encode the data from different sources before transmission in a shared communication media. The encoded data from different senders are added together for transmission without interfering with each other because of the orthogonal property of spreading codes. The orthogonal property means that the normalized autocorrelation value and the cross-correlation value of spreading codes are 1 and 0, respectively. Autocorrelation of spreading codes refers to the sum of the products of a spreading code with itself, while cross-correlation refers to the sum of the products of two different spreading codes. Prior to transmitting the data onto the channel, each sender modulates each transmitted bit by XORing it with a S -bit spreading code sequence, which expands the transmission time from a single bit into S -chip periods. Then, the data chips which come from different senders are added together arithmetically according to their bit positions in the S -bit sequences. Therefore, after the add operations, we will get S sum values of S -bit encoded data. Because of the orthogonal property, at the receiving end, the data can be decoded from the received sum signals by multiplying (i.e.

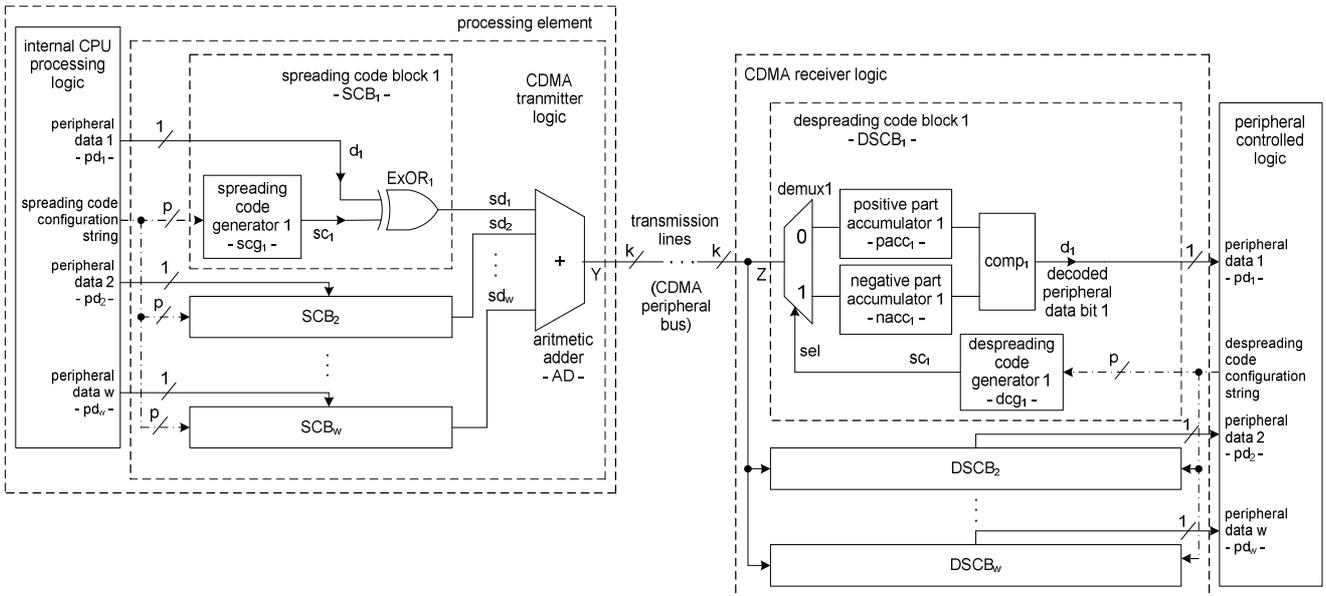


Fig. 3. Global system structure

XORing) the received signals with the spreading code used in the processor of encoding [6].

Modulated signal, which corresponds to one sum value at the output of the arithmetic adder, can be transferred as: a) multi-level signal over one line [7]; b) binary data in parallel form over k lines; and c) binary data in serial form over one line. The first transfer technique is very efficient in respect to the interconnection resources consumption. In general, the multi-level logic has the following drawbacks: i) voltage margins between two adjacent logic levels are narrow; ii) power supply voltage is high; iii) voltage translators are needed; iv) rise and falling pulse times are not balanced; vi) the swathing time from one level to another is relatively large; and vii) the multi-level logic is not well suited for implementation with standard CMOS process technology. The second transfer mode is optimal in respect to the transmission rate, can be implemented with standard CMOS technology, but requires multiple-lines for a chip-value transmission. The third option is wire-efficient, but requires multiple clock cycles per one chip-value transmission. In this paper we adopt binary data transmission in parallel form, as the compromise solution.

IV. DESIGN SOLUTION

We consider the ES with multiple PEs where each PE, in addition to a CPU, comprises a number of FUs and peripheral units that need to exchange data, status and control information with other PEs and/or other off-PE system components via a limited number of PE's input and output pins. In order to provide opportunity for simultaneous and uninterrupted communication for all peripheral units, we use PE's pins in a way to form a chip-parallel CDMA peripheral bus. Note that k -wire wide

CDMA bus can provide simultaneous transfer of $N=2^k$ output signals. However, the considerable saving in pin (wire) count comes at expense of transfer rate since the maximal frequency of an output signal can not exceeds $f_c/2^k$, where f_c is operating frequency of the CDMA bus. For example, 3-wire wide CDMA bus, which operates at frequency of $f_c=128\text{MHz}$, can transfer up to 8 output signals with maximal bit frequency of 16MHz. Having in mind that the peripheral units typically output signals with frequency much lower then the system frequency, the bit transfer rate offered by the CDMA bus is acceptable in most cases.

The global structure of the system intended for simultaneous data transfer over peripheral bus using CDMA technique is sketched in Fig. 3. This scheme is used for connection of several transmitter and receiver blocks over shared serial bus. At the transmitting end, peripheral data ($pd_i, i=1, \dots, w$) originating from different CPU's sources (UARTs, timers, parallel ports, PWMs, etc.) are individually encoded using a set of orthogonal spreading codes (sc_1, \dots, sc_w). Each spreading code block (SCB_i) encodes peripheral data, d_i , with a corresponding spreading code sequence, sc_i .

The SCB_i consists of: a) a spreading code generator, scg_i , that, for each data bit, outputs an entire serial p -bit spreading code bit sequence, sc_i . During the initialization phase of the system each scg_i is configured with a corresponding configuration string called pn -sequence; and b) a XOR gate, $ExOR_i$, that generates original or complemented spreading sequence, sd_i , depending on the current data bit value, d_i , (0 or 1). Each bit of the p -bit encoded data generated by the ExOR gate is called a data chip.

The encoded data from different SCB s are summed by the arithmetic adder building block, AD , and after that transmitted over $k = \log_2 w$ transmission lines, where w

TABLE I
DATA TRANSACTION SPECIFICATION

		CDMA ₁₁	CDMA ₂₂	CDMA ₃₃	CDMA ₄₄	CDMA ₅₅	CDMA ₆₆	CDMA ₇₇	CDMA ₈₈
gate count	Tx	375	762	1137	1518	1914	2295	2679	3060
	Rx	589							
delay (ns)	Tx	6.175	6.541	6.675	6.464	6.606	6.263	6.522	6.451
	Rx	6.751							
com. bandwidth (Mbps)		18.5	37	55.5	74	92.5	111	129.5	148

Notice: Target device: xc2s300e-6fg456; Tx - transmitter; Rx - receiver

corresponds to the number of *SCBs*. At the *AD*'s output, we will obtain p sum values of the p -bit encoded data.

Because of the orthogonal property, at the receiving end, the data can be decoded from the received sum signals by multiplying the received signals with the spreading code used for encoding. The receiver block, *DSCB_i*, accumulates the received sum values into two separate parts, a positive part, *pacc_i*, and a negative part, *nacc_i*, according to the bit value of the spreading code, *sc_i*, used for decoding. After accumulating the sum values, the original data bit can be decoded by comparing the values of two accumulators. Namely, if the value of the positive accumulator is larger than the value in the negative accumulator, the decoded data bit is "1"; otherwise, the decoded data bit is "0". Values stored in *pacc_i* and *nacc_i*, at the end of *pn*-sequence, are compared by the comparator block, *comp_i*. Finally, the *comp_i* outputs drive peripheral control logic.

V. EXPERIMENTAL RESULTS

Simultaneous data transfers over peripheral bus that applies the CDMA technique is realized at register-transfer level in VHDL. CDMA transmitter logic and CDMA receiver logic (showed in Fig. 3) are described as independent blocks with configurable number of output and input channels. In all our experiments we use the same 3-wire CDMA bus, which is capable of transferring data from maximally 8 different sources, and we vary the number of active transmitter-receivers pairs. Eight different system configurations have been implemented in the Spartan 2 FPGA using the software package Xilinx ISE 9.1 for synthesis, placement, and routing. In all cases an 8-bit spreading code was used, and all *pn*-sequences were orthogonal.

In Table I data concerning gate count, maximal frequency of operation (indirectly delay), and the communication bandwidth of the proposal for each configuration are given. CDMA_{*ii*} denotes a configuration with i transmitters and i receivers.

According to the obtained results given in Table I we can conclude that the communication bandwidth linearly increases as the number of transmitter-receiver pairs increases. Time resolution of the decoded signals is determined by the chip-interval.

V. CONCLUSION

An efficient system for simultaneous data transfers over peripheral bus based on CDMA technique is described. In respect to traditional bus solutions the number of parallel peripheral data transfer lines is decreased from w to $k=\log_2 w$. The proposed scheme is implemented on FPGA device from Xilinx Spartan 2 series. The obtained results shows that this architecture is scalable, both in respect to the communication bandwidth and spreading codes. This means that the number of transmitters and receivers can be increased without any noticeable performance losses.

REFERENCES

- [1] J. Kim, I. Verbauehede, M.-C. F. Chang, "Design of an Interconnect Architecture and Signaling Technology for Parallelism in Communication", *IEEE Trans. VLSI Syst.* Vol. 15, No. 8, 2007, pp. 881-894
- [2] B. C. Lai, P. Schaumont, and I. Verbauehede, "CT-bus: A Heterogeneous CDMA/TDMA Bus for Future SOC" in *Proc. 38th Annu. AsilomarConf. Signals, Systems, and Computers.*, Vol. 2, Issue 7-10, Nov. 2004, pp. 1868 - 1872
- [3] K. Sh. Zigangirov, *Theory of Code Division Multiple Access Communication*, John Wiley & sons, Inc., IEEE Press, Piscataway, NJ, 2004
- [4] A. Athavale, C. Christensen, *High-Speed Serial I/O Made Simple*, Xilinx Connectivity Solutions, San Jose, April 2005
- [5] R. H. Bell, Jr., K. Y. Chang, L. John, and E. E. Swartzlander, Jr., "CDMA as a multiprocessor interconnect strategy" in *Conf. Record 35th Asilomar Conf. Signals, Syst. Comput.*, 2001, pp. 1246-1250.
- [6] X. Wang, T. Ahonen, and J. Nurmi, "Applying CDMA Technique to Network-on-Chip", *IEEE Transactions On Very Large Scale Integration (VLSI) Systems*, Vol. 15, No. 10, October 2007, pp. 1091-1100
- [7] T. D. Richardson, C. Nicopoulos, D. Park, N. Vijaykrishnan, Y. Xie, C. R. Das, V. Degalahal, "A Hybrid SoC Interconnect with Dynamic TDMA-Based Transaction-Less Buses and On-Chip Networks", *19th International Conference on VLSI Design*, January 2006, pp. 657-664