# LARGE CAPACITY FIFO-ORGANIZED SDRAM-BASED MEMORY BLOCK

**Boyko Baev Petrov**

Department of Electronics, Technical University - Sofia, branch Plovdiv, Sankt Petersburg 61, 4000, Plovdiv, Bulgaria, +359 32 629760, e-mail: abpetrov@iname.com

**Petar Yordanov Atanasov**

AP-Electronics Ltd, Plovdiv, Bulgaria, +359 885-353654, e-mail: traskbg@abv.bg

**Anna Aleksandrova Petrova**

AP-Electronics Ltd, Plovdiv, Bulgaria, +359 887 400132, e-mail: anna@persecteam.com

*This paper presents a method end realization of large SDRAM-based FIFO-organized memory block useful for large buffer samples digital processing algorithms up to 30MHz and more sample frequency. The consider task is actual in many applications such us predictive signal compression, correlation analysis, data delays, TV and video mixing and synchronization, large data buffer signal generators in communications and networking test-devices.*

**Keywords**: SDRAM, FPGA, FIFO, large capacity, memory.

## 1. INTRODUCTION

Memories are one of the invariable parts of all digital units and systems. At now, the nesseserities of more large, more fast and more cheaper blocks of energy depended memory are increased simultaneously with increase of clock frequencies and data block volume in digital signal processing units and communication systems. The modern memory market is rich with various energy depended memory structures such as SRAM, DRAM, SDRAM, DDRAM, DDR2 SDRAM, RLDRAM [1]. One of the possible decisions for many embedded applications is SDRAM structure because of presence of well compromise of capacity, clock speed and price. The proof of this is fact that more of the modern embedded processor cores included SDRAM control logic such as ARM-cores [2], SHARC [3], Blackfin [4] and TigerSHARC [5] - DSP processors of Analog Devices and other MPU and DSP cores manufactures. For many digital signal and communication applications consecutive programmable logic is not suitable because of necessarily of higher sample frequency. In these cases digital logic are implemented by programmable parallel structures such as CPLD and FPGA devices.

The use of SDRAM in digital systems based on FPGA is embarrassed of missing of embedded SDRAM controller. Many of FPGA manufactures facilitate SDRAM based FPGA applications by SDRAM controller core reference designs [6], [7], [8], [9], [10]. In these cores, external programmable device single data bus oriented MPU or DSP is expected (fig. 1. [8]).

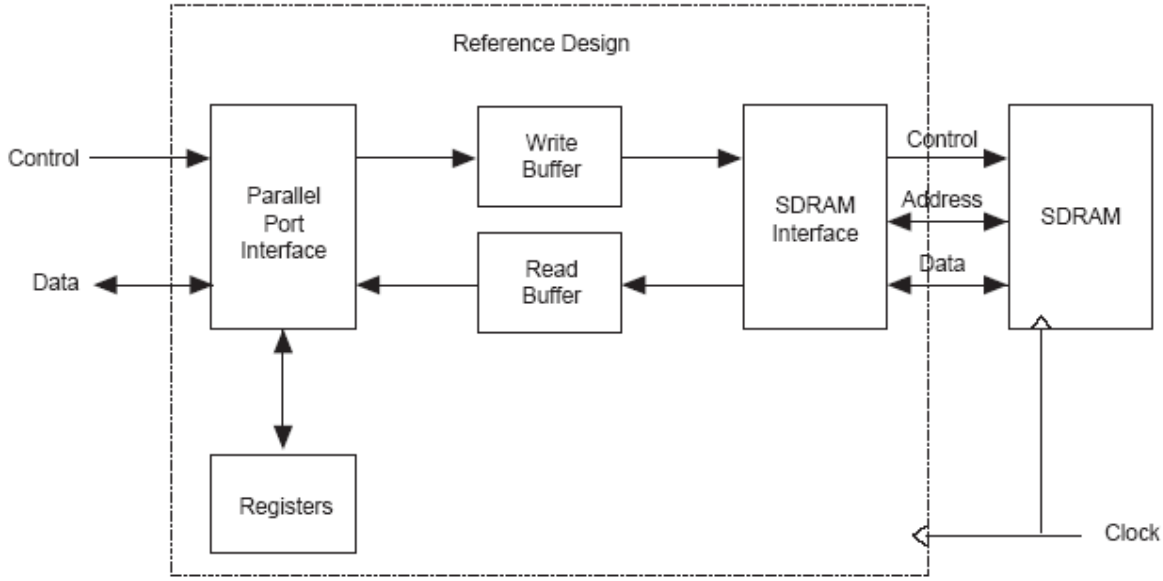


**Fig.1. SDRAM control logic in single data bus applications**

In case of two data buses applications - one for input data stream and one for output data stream with independent clocking and handshaking this paper describe for.

## 2. PROBLEM STATEMENT

Figure 2. shows an accepted approach.

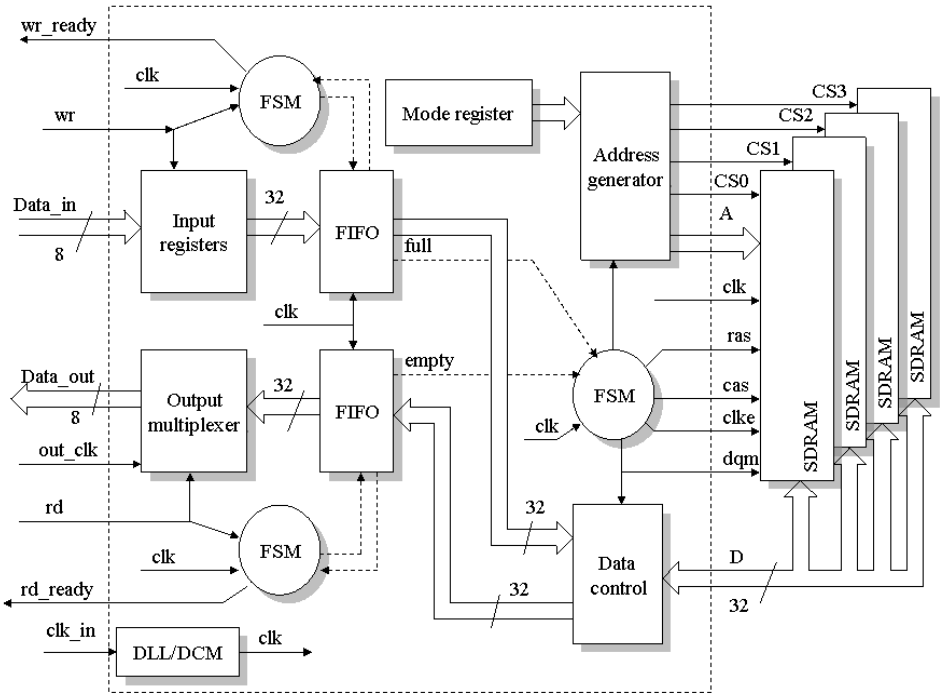

**Fig.2. SDRAM control logic based for two independent data bus applications**

## 2.1. Architecture description

The main approach can be described in these simple principles:

- architecture contains: input module, output module, SDRAM data control module, SDRAM address control module, SDRAM finite state machine module (Verilog terminology);
- for time synchronization are provide two internal FIFO modules between input/output block and SDRAM data control module;
- composite modules may be associated according clock signal in three equal-clock blocks;
- every block of modules has single clock signal and finite state control machine;
- external access is possible after writing of start addresses for input and output streams respectively;
- external access is parallel and individual handshaking;
- SDRAM logic watches about FIFOs flags condition and keep it in: Input_FIFO - not full, Output_FIFO - not empty;

Each block is described below.

### 2.1.1. Input block

As it can be seen, each four 8-bit writes to four register respectively leads to single 32-bit write cycle to input_FIFO by data_ready internal signal. The start condition sets by external reset to writing to the LSB data register. Next writing may be done after wr_ready signal is released. FSM controls this behavior.

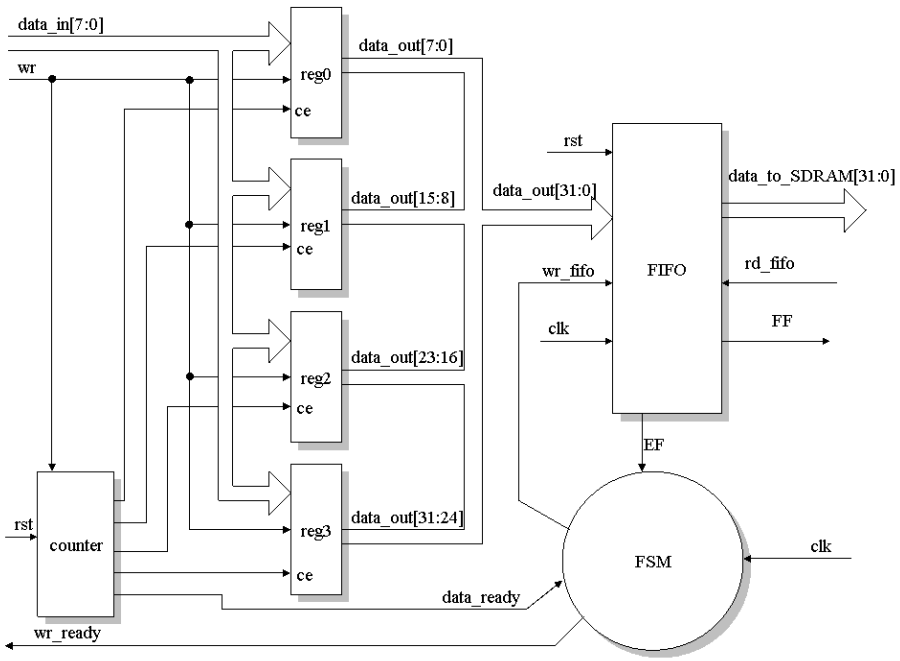


**Fig. 3. Input block architecture**

### 2.1.2. Output block

The output block behavior is simultaneous to input block accept for counter clocking. It is done by independent external clock input. FSM controls this process by internal new_data signal and inform external logic by rd_ready signal.

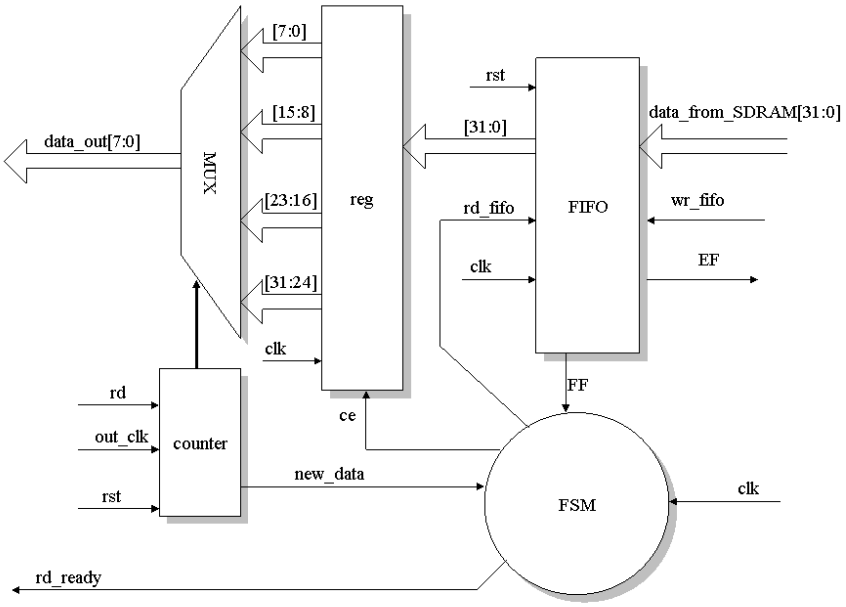


**Fig. 4. Output block architecture**

### 2.1.3. SDRAM logic block

The architecture of this logic block is based on [6] accept a part of address module and small deference in SDRAM FSM module. Figure 5. shows address logic module.

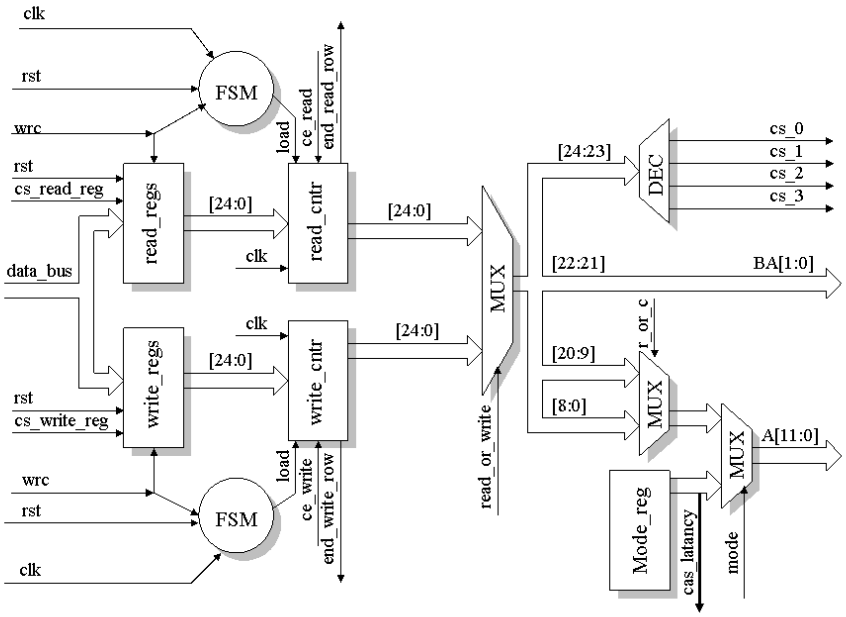


**Fig.5. Address control logic architecture**

The repaired SDRAM FSM control logic instance is:

**module sd_fsm** (sd_clk, rst, rdFIFO_f, wrFIFO_e, row_column, read_write, we_rdFIFO, re_wrFIFO, rd_addr_cntr, wr_addr_cntr, mrs, auto_cs, sd_ras, sd_cas, sd_we, start_cntr_end, ar_cntr_end, refresh_cntr_end, ar_cycles_cntr_end, ras_cntr_end, end_of_rd_row, end_of_wr_row, ar_cntr_load, ar_cycles_cntr_ce, ras_cntr_load, refresh_cntr_load);

The SDRAM data control module includes two parallel register and multiplexer who can be chosen between two or three clocks delay data depend of cas-latency output of SDRAM address control module.

### 2.2. Implementation

The implementation of design is made for 128Mbytes SDRAM composed by 4 x SAMSUNG K4S281632E-TC75 on XC2s200pq208-5 as shown below:

| HDL Synthesis Report | Timing Summary | Design Summary |
|---|---|---|
| Macro Statistics<br># FSMs : 5<br># RAMs : 2<br>16x32-bit dual-port block RAM : 2<br># Registers : 106<br>2-bit register : 2<br>1-bit register : 102<br>32-bit register : 1<br>12-bit register : 1<br># Counters : 18<br>2-bit up counter : 4<br>4-bit up counter : 5<br>4-bit updown counter : 2<br>9-bit up counter : 2<br>16-bit up counter : 2<br>4-bit down counter : 2<br>10-bit down counter : 1<br># Multiplexers : 5<br>2-to-1 multiplexer : 4<br>12-bit 4-to-1 multiplexer : 1<br># Tristates : 1<br>32-bit tristate buffer : 1 | Speed Grade: -5<br><br>Minimum period: 10.007ns (Maximum Frequency: 99.930MHz)<br>Minimum input arrival time before clock: 9.960ns<br>Maximum output required time after clock: 12.227ns<br>Maximum combinational path delay: No path found | Number of errors: 0<br>Number of warnings: 0<br>Number of Slices: 223 out of 2,352 9%<br>Number of Slices containing unrelated logic: 0 out of 223 0%<br>Number of Slice Flip Flops: 270 out of 4,704 5%<br>Total Number 4 input LUTs: 283 out of 4,704 6%<br>Number used as LUTs: 267<br>Number used as a route-thru: 16<br>Number of bonded IOBs: 131 out of 140 93%<br>IOB Flip Flops: 22<br>Number of Block RAMs: 4 out of 14 28%<br>Number of GCLKs: 3 out of 4 75%<br>Number of GCLKIOBs: 3 out of 4 75%<br>Total equivalent gate count for design: 70,161 |

After detail analysis it is determined that the main sources of delays are address counters.

### 2.3. Test Board

This design is tested on experimental FPGA board, shown on fig. 6

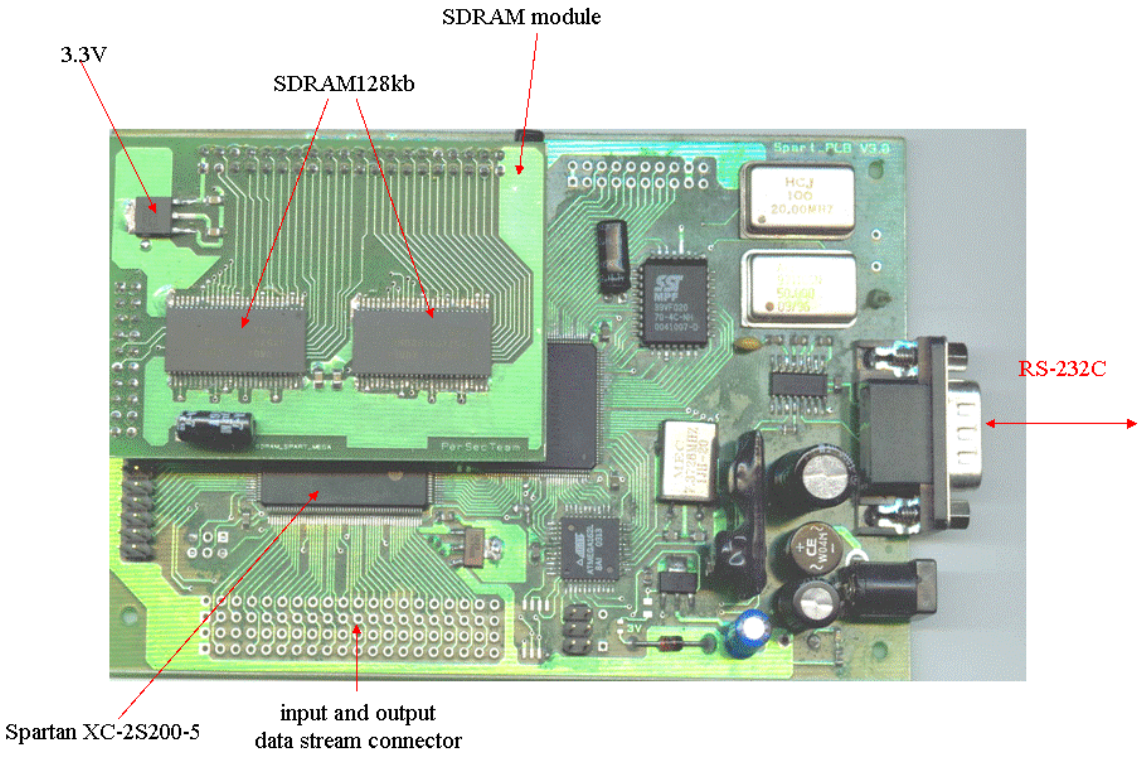


**Fig.6. FPGA board with addition of SDRAM daughter board**

## 3. SUMMARY

The result code can be implemented in other FPGA such us Spartan 3 and Virtex for better frequency result and can be easy modified to control of larger and DDRAM-based memory blocks.

## 4. REFERENCES

[1] http://www.micron.com
[2] http://www.arm.com/documentation/ARMProcessor_Cores/index.html
[3] http://www.analog.com/processors/processors/sharc/
[4] http://www.analog.com/processors/processors/blackfin/
[5] http://www.analog.com/processors/processors/tigersharc/
[6] http://www.xilinx.com/bvdocs/appnotes/xapp134.pdf
[7] http://www.xilinx.com/bvdocs/ipcenter/data_sheet/NW_Logic_sdram_sdr.pdf
[8] http://www.altera.com/literature/an/an334.pdf
[9] http://www.altera.com/literature/ds/ds_legacy_sdram_ctrl.pdf
[10] http://www.cmosexod.com/ip/sdram/sdram32.zip