An Investigation on the Possibilities of ADSP-2104 Shared Access to the Synchronised SRAM

Eng. Boyko Baev Petrov - Assistant Professor
Department of Electronics and Electronic Technologies
Technical University - Plovdiv Branch
E-mail address: bpetrov@tu-plovdiv.bg

Designing data acquisition systems for analogue or digital signals, obtained as a result of parameters transformation of a defined physical process, following questions have to be answered :

- is it necessary that the system works in real-time mode (considering the scale of time in which the physical process is developed)?
- are the accuracy and the speed of the process' sensors transducing the physical parameters convenient to the design specifications?
- what is the mathematical base for signal processing needed to obtain physical event information or operation and who will receive the result?

The development of data acquisition system is going in the way of the sensor capabilities enlargement (range, speed, accuracy) as so as of hardware improvement of signal processing methods (digital filtration, spectral and correlation analysis etc.)

During the last years a powerful development of a new kind of programmable circuits, especially designed for an easy and quick well-known digital information processing methods is observed. There are digital signal processors (DSP). These circuits for digital signal processing provide:

- increased binary number format (16, 24 bits) with fixed or floating point (with different number of preceding and succeeding bits)
- possibility of multiplication and division by internal schematic structures.
- support of methods for data addressing, convenient for description of a special signal processing algorithms (improved index addressing with auto-modification of the index register in participation of a program generated parameter-modificator, allowing the increasing or decreasing of operand address).
- separated buses for operands and operation code extraction more over on the bus of operation code it is allowed to extract also an operand.
- integrated software structures such as IF.. Then.. and For.. TO.. DO with a detached cycle counter and a flag for its tangent.
- a possibility to work at clock frequency of 20-100 MHz and etc.

All these features broaden the areas of application of real-time operating systems in the direction of high-speed, broadband and random signals. One of the main module in the structure of each digital system for analogue signal processing is the analogue-to-digital converter (ADC), which parameters define in a big measure the features of the system as a whole. When the ADCs operation at frequency of 10-60 MHz i.e. compatible to the clock frequency of a SDP, it is possible to organise the signal processing on quasi-real time principle. This is a process of storage a discrete and quantified analogue signal in a temporary memory. This process is developed in the real scale of time - typical for physical event. The information for this event is stored and acquired in the real (for the user of the result of acquisition) scale of time.

Designing such a kind of systems, it is important to answer the question what method should be implemented for the organisation of the ADC and the processing unit access to the common data massive. It means - how the current input values from the ADC will be written to, and how the processing unit will storage the calculated values and the final results (after read/modification/write operations).

In order to sustain the quasi-real time principle both of the processes should evaluate without mutual waiting state . This can be obtained only if the ADC and processing unit are synchronised. Then it is of big importance to create a possibility for execution of at least two access cycles to an arbitrary elementary memory cell - one for the ADC, and the second - for the DSP during a common synchronisation period. Taking into account that the DSP dispose with a feature to form the current cell access address (via the address bus), it will be necessarily looking from the ADC side to use an address counter for consequently coming addresses in order to form structures of the type "queue" or "cycle" .

The requirements for the dynamic features of the memory in use are changing to the direction of reduction of its dynamic parameters (such as access time, address and data hold-on time, time of setting-up etc.) If random access memories are used, the read/write cycles depend only on the signals OE (output enable) and WE (write enable). To make the input and/or output devices access synchronised additional registers for temporary storage of input and output data (as so as data for the current address) have to used. The write operation for these additional registers is synchronised with the common clock sequence.

A class of random access memories is available on the market whose the read/write operation is synchronous to an additional clock sequence (so called synchronised SRAMS). The frequency of the synchronising signal (defining a read/write cycle) is going up to 50-100 MHz.

The main goal of this presentation is the synthesis of a controller, allowing phase-alternative simultaneous access of two devices - a ADC and a DSP- to one and the same massive of synchronised memory. The scheme of interconnection is supposed to be as the shown in fig.1.

A particularity of this scheme is the use of a common clock, in synchronisation of which the analogue-to-digital conversion of the input signal - by ADC, and its processing - by DSP is done. The task of the controller for access management to a synchronised memory consist in a direct synthesis of:

- the output signals SWR and SRD actual at the current moment of phase change;
- the read/write signals in the temporary registers, connecting data transmitters and receivers from and to the memory;
- the signals for buffers enable, controlling the local memory, address and control buses.

For the generation of these signals can be used only:

- signals, obtained after synchronously division of the clock sequence;
- signals, controlling the read/write operations to an external data memory, generated synchronously with a defined for DSP clock sequence.

Choosing the components for controller design it is important to provide a logical function realisation on six input variables at least. The value of the logical function could be present to a fixed moment of phase time (synchronously to the clock sequence). These requirement are accomplished by programmable logic devices (PLD). For the proposed scheme an AMD-type PALCE16V8HD-5 chip have been chosen [3]. As a data buffer between the DSP and the synchronised memory the Motorola chip 74F543 have been used.

Description of the input signals:

- **CLK_PLD**- input signal for clock pin of the PLD. All of the output signals are synchronising with the **CLK_PLD's** rising edge. Generated by inverting of the **CLK_SYNC_SRAM** signal.
- **CLK_OUT**- signal, generated by ADSP-2104, synchronously with **RD, WR, CS_SRAM.** For master access control signal, the **CLK_OUT** has been chosen. When **CLK_OUT** is in high level, ADSP-2104 cannot operate with external devices.
- **RD** - external memory read enable signal
- **WR** - external memory write enable signal
- **CS_SRAM**- external memory chip select signal, generated by the address decoding of the DSP's address bus. In this application the generation is from external logic for more readability of the next logic equations.
- **BG**- external bus output of the DSP's data, address control buses.

Description of the output signals:

- **OE_GEN_BUF** - consequently address generator output enable signal. Low level on this line must be in time of data writing to the sync SRAM.

- **OE_ADDR_BUF** - address buffer output enable signal. Low level on this line must be in every DSP access to the sync SRAM.

- **SWR** - synchronised write command signal. Low level in time of **CLK_SYNC_SRAM** rising edge starts to write in the sync SRAM.

- **SRD** - synchronised read command signal. Low level in time of **CLK_SYNC_SRAM** rising edge starts read from a sync SRAM

- **OE_INP_BUF**- output enable signal of the input buffer (from ADC)

- **OE_DSP_BUF_1**- output enable signal of the DSP address buffer in time of reading from sync SRAM

- **CLK_DSP_REG**- clock signal for DSP address storage

- **OE_DSP_BUF_2**- output enable signal of the DSP address buffer in time of writing to the sync SRAM

Based on the previously points, the logical equations for output signal generation by selected PAL device (corresponding with the syntax of AMD firm-made PLD-compiler) are:

```
OE_GEN_BUF  =   / (CLK_OUT);
OE_ADDR_BUF=/((/CLK_OUT*/RD* BG * /CS_SRAM) + (/CLK_OUT
* / WR * BG* /CS_SRAM));
OE_INP_BUF    = /(CLK_OUT);
OE_DSP_BUF_1 = /(/CLK_OUT * /RD * BG * /CS_SRAM);
OE_DSP_BUF_2 = /(/CLK_OUT * /WR * BG */CS_SRAM);
CLK_DSP_REG   = (/RD) + (/WR);
SWR    = /((/CLK_OUT) + (/WR*BG*/CS_SRAM));
SRD   = /(/RD * BG * /CS_SRAM);
```

After a hardware experimentation of these logical equations, a non-critical data exchange and data proceeding between ADC and ADSP-2104, via common SRAM have been observed.

References

1. ADSP-2100 Family-User'Manual-Third Edition(9/95)-Analog Devices.
2. SRAM Data Book - Micron Technology - 1992.
3. PAL® Device Data Book - Advanced Micro Devices - 1992.

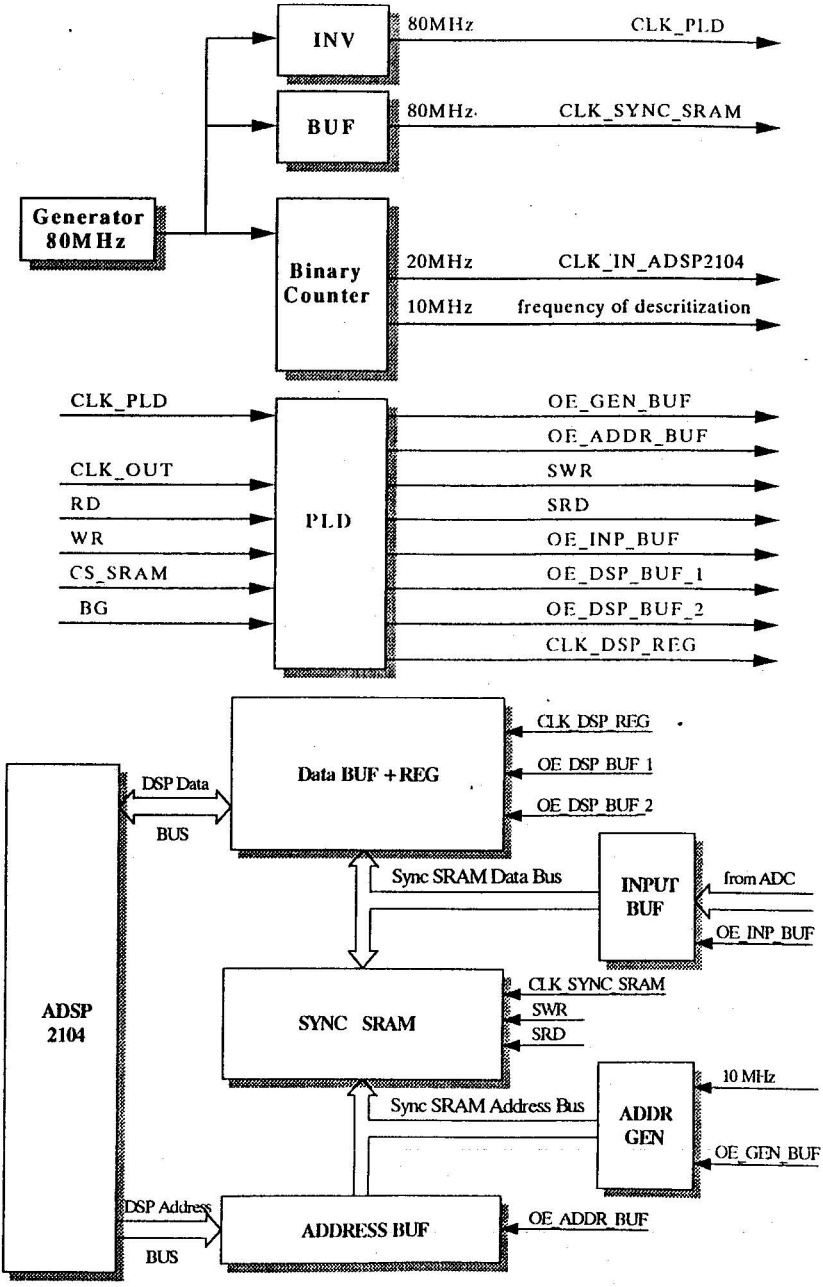

Fig.1. Block Diagram of the Control Logic for a simultaneose access to the sync SRAM