DIGITAL SIGNAL PROCESSING IN A THREE-PHASE POWER METER IC

Borisav Jovanović, Milunka Damnjanović, *Faculty of Electronic Engineering Niš*

Abstract – *Being a part of Three-Phase Power Meter IC, digital signal processing (DSP) block calculates root-mean square values of current and voltage, active power , reactive power , apparent power , and power-factor cos(φ), based on instantaneous values of current and voltage. It is considered here from the functional and architectural point of view. Cadence tool was used for the DSP verification and implementation.*

1. INTRODUCTION

Modern power meter devices relays on single chip referred to as *integrated power meter* (IPM). Digital signal processing block is one of the crucial parts of IPM This paper considers an original design of the digital signal processing block embedded into three-phase IPM.

Integrated power meter is mixed signal IC consisting of analog and digital signal processing blocks. The signal processing chain is given in Fig.1.

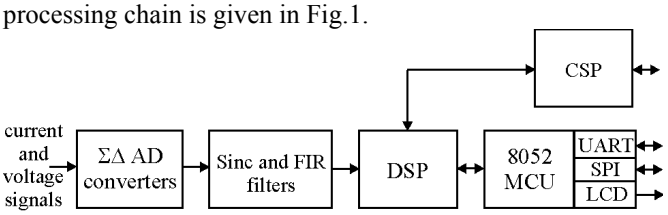

Figure 1. *Integrated Power meter processing chain*

The analog part of IPM contains Sigma-Delta AD converters for current and voltage channels, Band-Gap voltage reference and PLL circuits. The digital part is composed of digital filters, digital signal processing block (subject of this paper) and microcontroller unit with embedded peripherals.

The digital filters decimate oversampled signals from the on-chip AD converters for both voltage and current signal channels in three phases [1]. The final data rate of these voltage and current samples is 4096 Hz.

The DSP performs precise computations necessary to measure power-line signal parameters. It operates at 4.194MHz in four operating modes: reset, initialization, normal and testing. In the normal operating mode it calculates all mentioned power-line parameters.

The microcontroller unit (8052 MCU shown in Fig.1) is compatible with 8052 microprocessors. It includes several communication peripherals: UART, serial port interface (SPI) and LCD driver circuit.

This paper is organized in five sections and References. The following section gives an overview of DSP function. The third section considers the architecture of DSP. The subsequent section gives its interface. The fifth section describes the verification process and the final concluding remarks.

1. DSP MODULE FUNCTION

Based on instantaneous values of current and voltage, DSP calculates root-mean square values of current (Irms), and voltage (Vrms), active power (P), reactive power (Q), apparent power (S), and power-factor cos(φ) in every second. It measures the instantaneous value of the power-network frequency with accuracy of 0.01 Hz. A short survey of used relations would be as follows. Instantaneous value of current as function of time can be represented in the form:

$$i(t) = \sqrt{2} Ieff \cos(2\pi ft + \varphi) \tag{1}$$

After the discretisation in time, it becomes:

$$i(nT) = \sqrt{2} Ieff \cos(2\pi \frac{f}{f_{sempl}} n + \varphi) \tag{2}$$

where f = 50Hz, f_{sempl} = 4096 Hz.

Root-mean-square, Irms, is calculated once per second according to the expression:

$$Irms = \sqrt{\frac{\sum_{n=1}^{N} i(nT)^2}{N}} \tag{3}$$

where N=4096.

Similar expression, like for Irms, is used for Vrms calculation.

The relation

$$S^2 = P^2 + Q^2 \tag{4}$$

between apparent power (S), active power (P) and reactive power (Q), suggests us that it is enough to calculate two of three values and then use the relation (4) to find the third. If the instantaneous values of current and voltage are as follows,

$$i(t) = \sqrt{2} Ieff \cos(2\pi ft + \varphi 1) \tag{5}$$

$$v(t) = \sqrt{2} Veff \cos(2\pi ft + \varphi 2) \tag{6}$$

the instantaneous power is

$$p(t) = i(t) \cdot v(t) \tag{7}$$

After the discretization of the instantaneous power, the active power is calculated according to:

$$P = \frac{\sum_{n=1}^{N} p(nT)}{N} \tag{8}$$

Apparent power and power-factor *Cos(φ)* are calculated according to (9) and (10):

$$S = Ieff * Vef \tag{9}$$

$$\cos(\varphi) = P / S \tag{10}$$

Possible sources of error in active power calculation are the phase difference between voltage and current values and the fact that power-network frequency is slightly changed round the nominal (50Hz), so there is not an integer number of voltage half-periods in a second. Error eliminating is necessary, so after the multipication of the current and voltage values, the values $i^2(t)$, $u^2(t)$, p(t) i q(t) are filtered, accumulated 4096 times per second and the achieved total is devided with 4096 every second.

2. ARCHITECTURE OF DSP BLOCK

DSP has four operating modes: reset, initialization, normal operation and testing mode. DSP calculates root mean square values for voltage and current, mean values for active and reactive power, apparent power, active and reactive energy, power factor and frequency. The current input dynamic range is from 10 mA RMS to 100 A RMS, and voltage is up to 300V RMS.

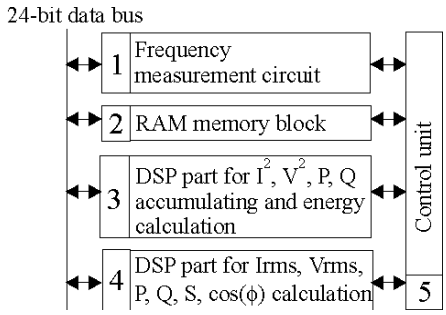

Figure 2. *DSP block structure*

DSP consists of several sub-blocks (Fig. 2):

- Frequency measurement circuit [2]
- RAM memory block
- Part for I^2, V^2, P, Q accumulating and energy calculation
- Part for current and voltage RMS, active, reactive and apparent power and power factor calculation);
- Control unit [3] that manages all other parts of DSP.

There is a single 24-bit data bus connecting these sub-blocks of DSP. DSP performs over 16000 multiplication operations and over 50000 addition operations in a second. Arithmetical units for multiplication, division and square-rooting, embedded in DSP, use gated clock architecture reducing the power consumption, and therefore, DSP is optimized for low power design.

The RAM memory block is divided into three memory modules. One module is used for storing intermediate and final measurement results for one energy system phase. Each memory block consists of 64 24-bit registers.

Instantanious, offset and root-mean-square signals for current, voltage, active and reactive power signals are all represented in RAM block by 24-bit signed two'complement binary values and have the following data format:

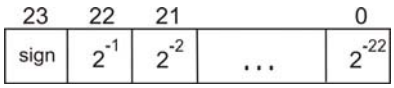

Figure 3. *Data format for current, voltage and power signals*

Signal values are in range from –1 do 1 and are normalized to some full-scale value. For voltage full-scale value is $\sqrt{2}\,300\text{V}$, for current signals it is $\sqrt{2}\,100\text{A}$, for power is 2*100A*300V =60kW.

Cos(φ) (power factor) and frequency value are also represented by 24-bit signed two'complement binary values and have following format:

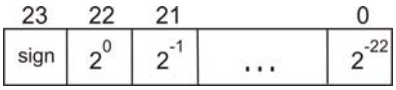

Figure 4. *Data format for power factor and frequency*

Measured frequency is normalized to actual value of 50Hz.

The control path of DSP unit is implemented as a finite state machine and it generates a number of control signals that determine what component can write to bus, what registers are loaded from the bus and what arithmetical operation is performed. In normal operating mode, controller performs the periodically repeated sequence that lasts exactly 1024 clock periods.

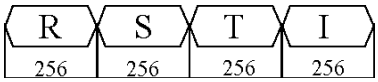

Figure 5. *Controller's subsequences*

The sequence of control unit is divided into four subsequences. Each subsequence lasts exactly 256 clock periods (Fig. 5). The first three controller's subsequences, called R, S and T, control the calculations made for each phase of the three-phase energy system. During R, S and T subsequences intensive calculations are performed only within subpart 3 (Fig.2) and sequence of operation for I^2 accumulating is given in Fig.6.

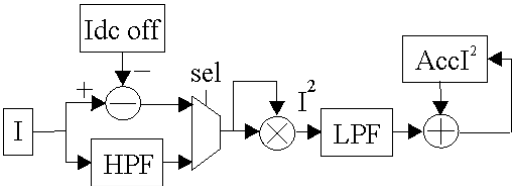

Figure 6. *Data processing chain for current-square accumulation*

At the beginning of each subsequence R, DC offset has to be removed from instantaneous current values. Two options are implemented. The first option is to subtract the offset which was determined during calibration procedure. The other is the use of first-order digital high pass filter which has unity gain and cut-off frequency of 5Hz. After, instantaneous sample of current I is squared in multiplication unit. Then, the value I^2 is passed through the single pole Low Pass Filter (LPF) with a cut-off frequency of 10Hz, and after that, it is accumulated into register $AccI^2$ (Fig. 6).

Low pass filter helps in reducing the calculation error that could exist due to the fact that time-interval of 1 second (that is, accumulating time for the value I^2) doesn't always happened to be integer number of power-line-signal half-periods.

In subsequence R, based on calculated values of P and Q, Whr (Watt-hour) pulses are generated for every Whr of the active and reactive energy. That pulses increment the content of the related registers in RAM block keeping the values of

positive and negative active energy, and positive and negative reactive energy.

The fourth subsequence of the controller, denoted I in Fig.5, manages the calculations that are periodically repeated every second. Based on accumulated squares of instantaneous current and voltage, and accumulated instantaneous active and reactive power during the last second, calculations are performed in order to generate voltage and current root mean square (RMS) and mean active and reactive power values. Block 4 in Fig. 2 implements all these calculations. Sequence of operations for Irms is given in Fig.7.

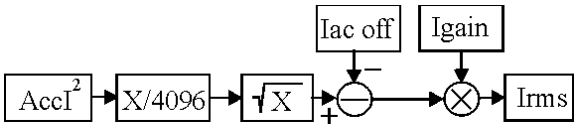

Figure 7. *Data processing chain for current RMS calculation*

Programmable energy-to-frequency converter within DSP part provides pulse response to consumed power over pin EOUT [5]. Therefore, chip allows interfacing with an external counter. By default, one pulse is generated on EOUT pin when active energy of 1 Whr (Watt-hour) is measured. This pulse-frequency to energy ratio can be changed by loading the appropriate value into special pulse-frequency to energy register within DSP part.

3. DSP INTERFACE

The measurement results calculated by DSP can be read from IPM in two different ways:

- Through Communication Serial Protocol block (CSP) which has I2C-like interface [6]. CSP is directly connected to DSP block (Fig.1).
- Over MCU ports or MCU serial communication pins. The connection between DSP and MCU is achieved through some of the MCU special function registers (SFR).

The method for accessing to the DSP block can be chosen by chip input pin SEL_MCU.

When SEL_MCU is in reset state, the DSP results are transferred to the following CSP. CSP also allows the user to calibrate components of the power meter (AD converter programmable gain, phase shift in decimation filters, various calibration parameters of DSP, etc.). Bidirectional pins SDA and SCL are used as communication interface. One additional pin CSN enables the CSP. Also, when SEL_MCU is in reset state, chip input pins MODE1 and MODE0 are used for selecting one of four chip operating modes [5]: normal operating mode (NOM), testing (TM), initialization (INI) and reset (RST). Additional RUNSTOP (input) and WAIT (output) pins are used only during DSP testing mode [4].

The other option for access to the DSP is when input pin SEL_MCU is set. The control of DSP is achieved through MCU block.

The read operation is started by loading the address into dedicated MCU register (address SFR), and after, by setting the dedicated bit of one of MCU registers. When this bit becomes zero, the read operation is completed. The content of addressed DSP register is stored automatically in three data SFRs.

When data is to be transferred to DSP (for calibrating purpose), data has to be stored first into three data SFRs. After, address has to be loaded into address SFR. The write operation is started by setting the dedicated bit of one MCU register. The write operation is completed when this bit becomes zero.

When SEL_MCU is set, the operating mode of DSP is governed by the content of one SFR which name is DSPCON. DSPCON has address E8h. The group of bits of DSPCON important for DSP control consists of: DSP_WAIT, DSP_RUN_STOP, DSP_MODE1, DSP_MODE0. Figure 8 shows positions of control bits within the register and Table 1 gives their explanation.

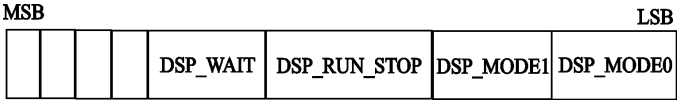

Figure 8. *The content of DSPCON Special Function Register*

Table 1.

Bit	Symbol	Function
DSPCON.3	DSP_WAIT	Used only during DSP testing procedure
DSPCON.2	DSP_RUN_STOP	
DSPCON.1	DSP_MODE1	DSP operating mode selection bits.
DSPCON.0	DSP_MODE0	

DSP_MODE1, DSP_MODE0 correspond to the input pins MODE1 and MODE0 and they determine the DSP operating mode. When both signals are set, DSP is in the reset state (RST). When DSP_MODE1=0 and DSP_MODE0=1, initialization mode is chosen (INI). Else, if both signals are in reset, DSP is in normal operating mode (NOM). DSP_WAIT and DSP_RUN_STOP are used only during DSP detail testing procedure [4] which is selected when DSP_MODE1=1 and DSP_MODE0=0.

The MCU communicates with DSP through the following SFR registers: DSP_BYTE1 (address A7h), DSP_BYTE2 (address A6h), DSP_BYTE3 (address A5h), DSP_ADDR (address A5h) and PSW (address D0h).

The PSW register contains command bits for access to the DSP block registers - bits REQ_DSP and R_NW_DSP.

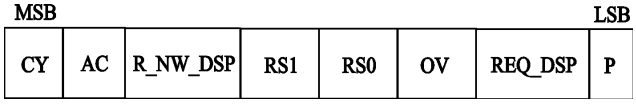

Figure 9. *PSW Special Function Register*

Table 2.

Bit	Symbol	Function
PSW.5	R_NW_DSP	Control bit for read/ write access to the DSP registers.
PSW.1	REQ_DSP	DSP request control bit

If the 5th bit in PSW (R_NW_DSP) is set, read operation is selected. The address has to be loaded into DSP_ADDR. The read operation is started by setting the REQ_DSP control bit. The operation lasts several clock periods depending on the current state of DSP's control unit. The maximal number of clock periods required for read operation is 10. When operation is completed, on-chip

hardware automatically clears control bit REQ_DSP. The 24-bit result (the content of addressed DSP register) is stored into SFRs: DSP_BYTE3 (the msb 8 bits), DSP_BYTE2 (the middle 8 bits), DSP_BYTE1 (the 8 lsb bits).

The assembler instruction sequence illustrating the reading operation is given in the Table 3. In the example, DSP register which address is 33h is read and result is stored into registers byte3, byte2 and byte1.

Table 3.

```
        setb R_NW_DSP
        mov DSP_ADDR,  #033h
        setb REQ_DSP
loop1: jb REQ_DSP, loop1
        mov byte3, DSP_BYTE3
        mov byte2, DSP_BYTE2
        mov byte1, DSP_BYTE1
```

When bit R_NW_DSP is zero, write operation is selected. The address has to be loaded into DSP_ADDR register and data bytes into DSP_BYTE3, DSP_BYTE2 and DSP_BYTE1. The write operation is started by setting the REQ_DSP. On-chip hardware clears the REQ_DSP when write operation is completed. The maximal number of clock periods required for data transfer is 10. The example for write operation is given in Table 4. The sequence of 8052 assembler instructions describes the loading of 24-bit data (hexadecimal value AA7005h) into DSP register which address is 33h.

Table 4.

```
        mov DSP_BYTE3, #0AAh
        mov DSP_BYTE2, #070h
        mov DSP_BYTE1, #005h
        mov DSP_ADDR,  #033h
        clr R_NW_DSP
        setb REQ_DSP
loop1: jb REQ_DSP, loop1
        setb R_NW_DSP
```

4. DSP VERIFICATION

The DSP block was described in VHDL and simulated in NCSim. After, VHDL model of MCU part was added and both models were simulated again.

Several tasks had to be accomplished during logical verification:

- Since one method for communication with the DSP is through CSP block, I2C-like protocol had to be described in testbench.
- MCU communicates with DSP. Therefore, assembler program had to be written and translated into hex file. Special program was written for translating the hex file into VHDL code. After, VHDL code was used in testbench for initialization of MCU program memory.
- Digital filter's outputs had to be generated in testbench and put on DSP's inputs.
- Verification outputs had to be written into textual files.

Since DSP calculates new values (current and voltage RMS values, apparent, active and reactive power, power factor, frequency, active and reactive energy) every second, simulation lasts several seconds of simulator's time. Because of limited computer resources, output data could not be observed in standard waveforms. Instead, the results are read every second, transformed into appropriate digital words and stored into textual output file.

Simulations were performed for three times using the same test bench: before synthesis process, after synthesis process and after clock tree generation. The obtained results were the same, proving the correctness of the design.

Integrated Power Meter IC is implemented in 0.35μm CMOS standard cell technology. After synthesis process in *Build Gates*, estimated DSP area expressed in logical NAND-gate units, was 55230 units.

Finally, *First Encounter* Tool was used for floorplanning, placement and routing, as well as clock and reset trees generation for complete circuit.

5. CONCLUSION

The DSP of an Integrated Power Meter IC is considered. Based on controller/datapath partition, fast DSP with low power consumption and small area has been developed. System was verified through the VHDL simulations and synthesized by Cadence tools..

REFERENCES

[1] M. Sokolović, B. Jovanović , M. Damnjanović, "Decimation Filter Design", Proc of *24th Int. Conf. on Microelectronics MIEL 2004,* pp. 601-604

[2] B. Jovanović, M. Damnjanović, "Digital system for power line frequency measurement", Proc. of *XLVI Conf. of ETRAN*, Čačak, 2004, pp 29-32

[3] M. Jevtić, B. Jovanović, S. Brankov, "The Control Unit For Energy Meter System-on-Chip", Proc. of *XLVI Conf. of ETRAN*, Čačak, 2004, pp 75-78

[4] B. Jovanović, M. Jevtić, S. Đošić, M. Sokolović, P.Petković, "BIST Logic Design for DSP of an Integrated Power Meter", Proc of *V Simp. Industrijska elektronika INDEL 2004*, Banjaluka, pp 120-125

[5] B. Jovanović, M. Damnjanovic, P.Petković, "Digital Signal Processing for an Integrated Power-Meter", Proc. of *49. Internationales Wissenschaftliches Kolloquium*, Technische Universirtat Ilmenau, 2004, pp 190-195

[6] M. Cvetković, M. Jevtić, "I2C-like communication for the Power Meter IC", Proc of *24th Int. Conf. on Microelectronics MIEL 2004,* pp.781-784.

Sadržaj – U ovom radu opisan je blok za procesiranje digitalnih signala (DSP) kod Integrisanog merača potrošnje električne energije. Arhitektura DSP-a zasnovana na *controller/datapath* podeli ima malu potrošnju energije i malo zauzeće površine čipa. DSP je verifikovan VHDL simulacijama i implementiran *Cadence* alatima.

DIGITALNO PROCESIRANJE SIGNALA U TROFAZNOM MERAČU POTROŠNJE ELEKTRIČNE ENERGIJE

Borisav Jovanović, Milunka Damnjanović