Zbornik radova 51. Konferencije za ETRAN, Herceg Novi – Igalo, 4-8. juna 2007.
Proc. 51st ETRAN Conference, Herceg Novi – Igalo, June 4-8, 2007

DESIGN FOR TESTABILITY OF DECIMATION FILTERS IN SOLID-STATE ENERGY METER

Miroslav Marinković, Borisav Jovanović, Predrag Petković, *Faculty of Electronic Engineering Niš*
Nagrađeni rad mladog autora

Abstract – *This paper considers the test logic architecture of digital decimation filters dedicated for a three-phase energy meter IC. Test logic architecture is based on Design For Testability (DFT) techniques. The testing method is achieved by periodical executing of special functions incorporated in the chip-testing mode. Filter testing completes with storing results in Status Register (SR). One easily can accesses the content of SR through Communication Serial Port (CSP). The filter architectures with embedded test logic are verified by simulations and synthesized using AMI Semiconductor CMOS 0.35μm technology library.*

1. INTRODUCTION

Nowadays, complete hardware and software systems are integrated into a single silicon circuit called System-on-Chip (SoC). Designers of such complex systems are unavoidably faced with many challenges. One of the most important demand is assigned to fast and efficient testability as well as the ability of defect diagnostics. The main influence of testing and diagnostics is reflected in cost-reliability trade. For prototype chip testing, the incorporated diagnostics is essential. It is not important just to determine if a system functionally fulfils all requirements, but also to discover where and why the problem occurs.

This paper considers test logic architecture for digital decimation filters. The proposed filters are part of a SoC dedicated for energy measurements in three-phase power system. The SoC is named – IMPEG3. It is the next generation of the prototyped single-phase power meter, IMPEG [1]. The testing concept is based on the application of widely used DFT strategies, the Built-In Self Test (BIST) [2]. BIST uses embedded hardware test generators and test response analyzers. It generates and applies test patterns on-chip at the speed of the circuit, thereby eliminating the need for an external tester. It is important to achieve high coverage of specific faults, minimal chip area and as low performance degradation as possible. The test logic is designed for diagnostics (DFD). Therefore, it contains circuits that diagnostics make easier.

IMPEG converts analogue voltage and current into digital form suitable for calculation of all parameters needed for energy metering. This includes voltage and current RMS values, frequency, active, reactive and apparent power, power factor, active and reactive energy. The IMPEG consists of analog and digital processing blocks. The main part of the digital block contains five digital filter blocks (two Sinc, two FIR filters and Hilbert transformer), DSP block for calculating mentioned parameters and CSP block for serial communication with external system.

The following section provides brief overview of proposed testing concept and describes relevant part of DSP control unit that controls the testing process. Because of differences in the Sinc and FIR filter architectures, the VLSI implementation of test logic is separately described for Sinc and FIR filters in the third and fourth section of the paper, respectively. The final section gives the results of simulation and synthesis process.

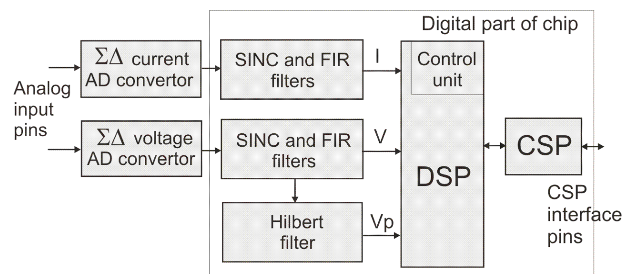

Fig. 1 *Block diagram of IMPEG3 Power meter System - on - Chip*

2. TESTING CONCEPT AND TESTING CONTROL LOGIC

In IMPEG prototype, DFT technique is implemented only on arithmetic modules, incorporated in DSP block. It will be spread on digital filters in IMPEG3. The following brief description of DFT concept incorporated in IMPEG will ease understanding of the testing concept proposed for IMPEG3. The details about testing and DSP test logic can be found in [1].

IMPEG's DSP operates at rate of 4096x1024Hz. The chip has four operating modes: reset, initialization, normal (or running mode) and special test mode. Operating modes are controlled with two input signals named MODE1 and MODE0.

The proposed testing approach assumes two main testing levels. One takes place during the normal operating mode, while the second one occurs throughout the special test mode.

During the normal operating mode, DSP provides on-line BIST routines that continuously, once after every second, check the arithmetical units. If error is detected in any of arithmetical blocks, the short pulse appears on the output ERROR pin. Besides, results of on-line testing can be obtained by reading the Status register that is accessed through CSP. For every tested arithmetical block one bit is reserved in the Status register. If during two consecutive testing procedures (in the two successive seconds of normal mode operation), the test logic concludes that an error exists in particular arithmetical unit, the appropriate bit of Status register will be flagged-on.

The test mode is dedicated to detailed testing of arithmetical units in DSP. Unlike on-line testing, the various values for input operands can be launched and stored into RAM memory of DSP. RAM registers where arithmetic operands are stored can be accessed through. Similarly, CSP provides the testing results as well.

DSP's control unit (CU) is implemented as Moore FSM (Finite State Machine). During the normal operating mode, CU periodically executes main state sequence that lasts 1024

clock periods and is divided into 4 sub-sequences called S, R, T and I. Each of these subsequences lasts 256 clock periods. Besides, CU contains 12-bit counter that increments value once during the main state sequence. Since the chip runs at 4096x1024 Hz, the counter overflows after every second of normal operation. Four *lsb* bits determine what arithmetical unit is going to be tested.

Testing technique for digital filters in IMPEG3 retains the similar approach. Fig. 2 shows the part of the main CU state sequence that is relevant for testing of IMPEG3.

The left-hand side of the diagram denotes DSP testing. States *WAIT1* and *WAIT2* control flow of operands and test results through CSP. These states do not affect decimation filter's testing. At the end of T sub-sequence, in state *Case1* (Fig.2), CU checks four LSBs of 12-bit counter. If their value is "1111", CU executes I sub-sequence (*IE*, Fig.2). If counter takes a value in range "0001" to "1001", CU starts the execution of arithmetic unit testing procedure (*IT1*, *IT2,…, IT9*).

If counter takes value in range "1010" to "1110", and current operating mode is test mode, CU starts the execution of digital filter testing procedure. The right-hand side of the diagram in Fig. 2 corresponds to these cases. Table 1 presents correlation between appropriate states (*IT10,…,IT14*) and particular filter .

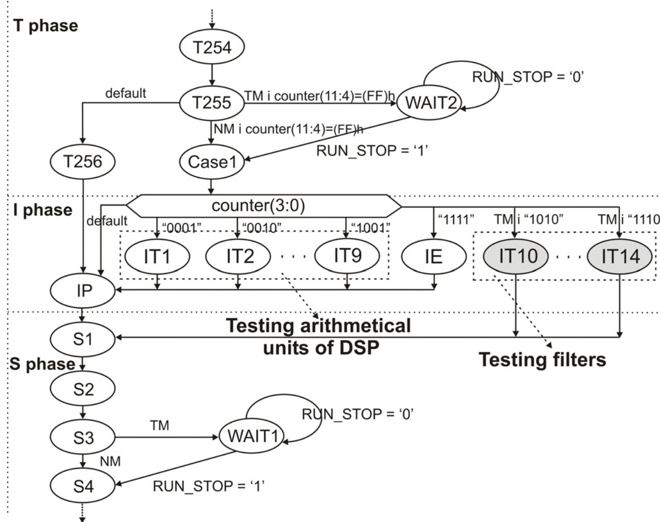

Fig. 2 *State details during I sub-sequence of DSP's control unit*

Table 1. *Counter values, states and corresponding filter*

Test operating mode		
Counter(3:0)	State	Tested filter
"1010"	IT10	Sinc_Voltage
"1011"	IT11	Sinc_Current
"1100"	IT12	FIR_Voltage
"1101"	IT13	FIR_Current
"1110"	IT14	Hilbert Transformer

In every of the specified states CU starts the testing procedure for one of the filters. At the end, if the result is incorrect, errors counter increments.

The testing procedure is in control of filter-test control logic (FCL), without any influence of DSP's CU. The testing procedure is repeated periodically in test mode with the period approximately of 1s. The Fig.3 presents the digital filters and DSP block with signals relevant for filter testing. Further, testing procedure will be described in detail on current channel Sinc filter example.

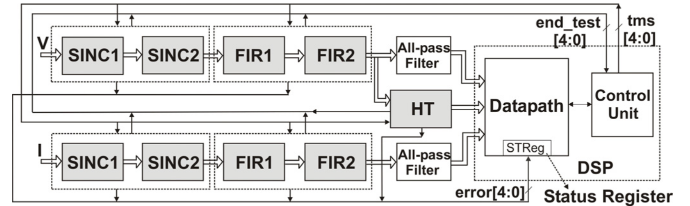

Fig. 3 *DSP/Digital filters communication with testing signals*

FCL sequence for Sinc filter testing is shown in Fig. 4. During *WAIT1*state, FCL activates signal *tms_sc*='1' (Test Mode Select signal for current channel Sinc filter) and the testing starts. DSP's CU stays unaffected until FCL sets signal *end_test_sv* = '1' that indicates end of testing. Then FCL enters the next state, *INC11*, activating the signal *en_inc_sc* = '1' that enables errors counter incrementing. s The counter increments, when the testing faults. If two consecutive testing procedures found error in Sinc filter, the error counter is incremented two times. Then, the error signal is stored at 11th bit position of Status register.

Similar state sequence corresponds for the other filters testing. Since there are five different filters (Table 1), Status register contains additional 5 bits (besides 9 bit positions already used for testing of DSP's arithmetical blocks). Testing results can be obtained by reading the status register through CSP.

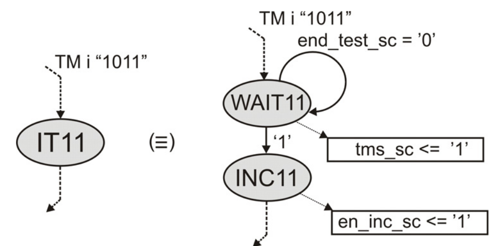

Fig. 4 *Control unit sequence for Sinc filter testing*

3. TEST LOGIC FOR SINC FILTERS

Sinc filters are the front-end part of the decimation filter processing chain. Fig. 5 gives the block diagram of current channel Sinc and incorporated BIST controller. The filter architecture is based on TDM (Time-Division Multiplexing) concept that saves chip area [4].

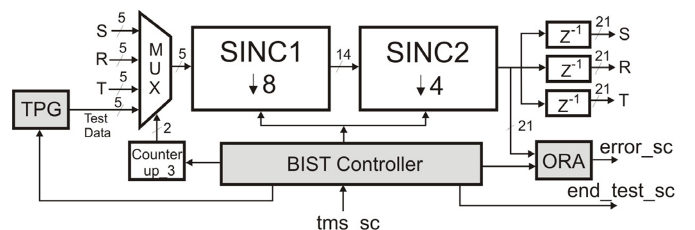

Fig. 5 *Block schematic of current channel Sinc filter*

The test logic consists of the test pattern generator (TPG), the BIST controller, and the output response analyzer (ORA). At the beginning, all registers incorporated in Sinc filters are in reset state and binary counter (*Counter_up_3*) is set into the state "11". Counter's output is fed as selector to the multiplexer. In state "11", the test sequence from TPG passes through the multiplexer and supplies digital filters. TPG block generates two different test patterns "11111" and "00000". The BIST controller contains a counter that defines the testing interval in terms of number of cycles. After a predetermined number of cycles, FCL stores the result of testing (comparison between signal obtained at Sinc output and expected result stored in ROM) into flip-flops within ORA block. Fig. 6 represents internal structure of ORA. Testing checks if all output bits of Sinc filter switch from '1' to '0' and from '0' to '1'. Therefore, the test detects stack-at faults. Explicitly the testing comprises three tests. With expected outputs presented in Table 2.If at least one test finds the error, ORA sets the signal *error_sc='1'* that forces incrementation of error counter (*Counter_up_2) of* DSP's control unit.

Table 2.

Test number	Correct value (21-bit word)
1.	(000000)h
2.	(1FFFFF)h
3.	(000000)h

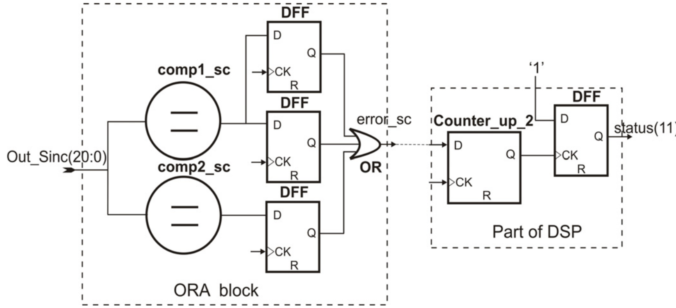

Fig. 6 *ORA block and the part of DSP block for storing the Sinc filter's test result*

4. TEST LOGIC FOR FIR FILTERS

FIR filters form the back-end part of the decimation filter chain. Its architecture is based on compact MAC (Multiply and Accumulate) approach and TDM concept [4].

FIR testing requires introducing additional states in the state diagram of controller that manage dataflow in FIR filters and Hilbert transformer (HT). The new states have minimal affect on filters architecture. The simplified state diagrams of FIR filter and Hilbert transformer controller are shown in Fig. 7 and Fig.8, respectively.

Additional states (shown in Fig.7 and Fig.8) are *INITIAL* and *RUN_BIST*. During *INITIAL* state the test words stored in TPG block are transferred into the RAM registers dedicated to keep input samples. During IDLE state, the control logic checks the operating mode by inspecting signal *tms_fc*. If *tms_fc* = '0', the system operates in normal mode. Then FIR and HT filters execute sequences *RUN_FIR1*, *RUN_FIR2* and *RUN_HT*. If *tms_fc* = '1' the system operates in test mode. Only then the testing function *RUN_BIST* is running providing output y_t according to (1).

$$y_t = h_{t1}(x_{t1}+x_{t2}) + h_{t2}(x_{t3}+x_{t4}) + h_{t3}(x_{t5}+x_{t6}), \qquad (1)$$

where x_{ti}, (i=1,…,6) represent the test samples, while h_{ti}, (i=1, 2, 3) are filter coefficients. Their values are listed in Table 3.

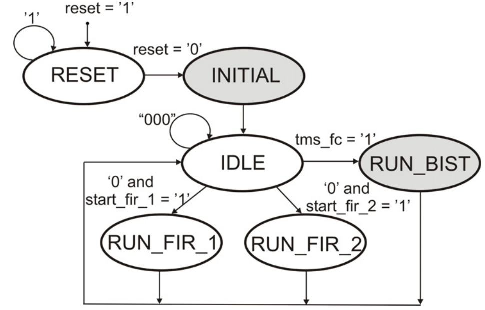

Fig. 7 *The simplified state diagram for FIR filter controller*

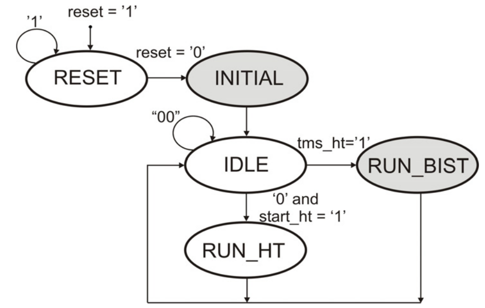

Fig. 8 *The simplified state diagram for Hilbert controller*

Table 3. Test samples and coefficient values for FIR filters

Test word	x_{t1}	x_{t2}	x_{t3}	x_{t4}	x_{t5}	x_{t6}	h_{t1}	h_{t2}	h_{t3}
Value (DEC)	-1	1	0	-1	0	1	-1	1	1

The testing procedure runs in three steps listed in Table 4. This allows intermediate result testing.

Table 4.

Arithmetical expression for testing	Expected value (34-bit word)
$h_{t1}(x_{t1}+x_{t2})$	(000000000)h
$h_{t1}(x_{t1}+x_{t2})+h_{t2}(x_{t3}+x_{t4})$	(3FFFFFFF)h
$h_{t1}(x_{t1}+x_{t2})+h_{t2}(x_{t3}+x_{t4})+h_{t3}(x_{t5}+x_{t6})$	(000000000)h

Fig.9 shows internal structure of FIR filters in current channel and incorporated test logic. The test pattern generator TPG is implemented as ROM that stores test samples x_{ti}, (i=1,…,6).

The testing procedure runs as follows.

In *INITIAL* state, controller (*FIR_C* controler in Fig 9) transfers the test samples into SRAM register cells putting one pair of samples into one SRAM cell. During testing, samples pass through *MUX_2* and enter registers *Reg_sym_1* and *Reg_sym_2*. Then they are summarized. According to (1), the sum is multiplied by appropriated test coefficients h_{ti}, (i=1, 2, 3) that are stored in ROM memory *ROM_Coefficient* together with the filter coefficients. The product is stored into accumulator register. After three cycles, y_t is completed.

Testing proceeds with comparing the obtained result with the expected value. Similar to Sinc filters, all bits in the output word are expected to switch from '1' to '0' and from '0' to '1'.

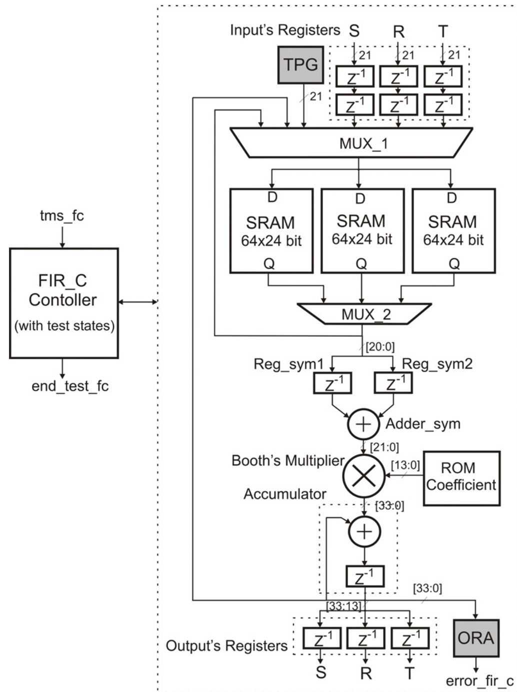

Fig. 9 *Internal structure of current FIR filters and incorporated test logic*

ORA block consists of the comparators and flip-flops for storing the test results. The architecture of ORA block is similar the structure shown in Fig.6. If an error is detected at least in one of the test steps (Table 4), DSP's control unit increments the appropriate errors counter

Hilbert transformer testing uses the same method. The only difference stands for opposite sign of test samples x_{t2}, x_{t4}, x_{t6}. Therefore, instead of *Adder_Sym* block (Fig.9) in FIR filter's architecture, Hilbert transformer incorporates block for subtraction.

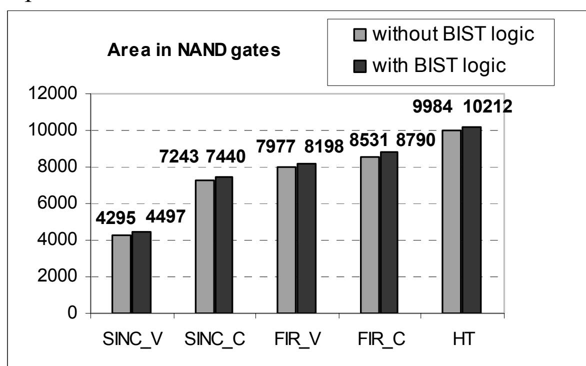

Fig.10 *The comparation of filter's area in terms of NAND gates*

5. SIMULATION AND SYNTHESIS RESULTS

Decimation filters and BIST logic are coded in VHDL and verified using simulator Active HDL [6]. The testbenches for normal and test-operating mode were stimulated with appropriate input pattern that was able to detect defects stuck-at-0 and stuck-at-1. Their effects were observed by reading the appropriate bits of Status register.

After, verification the code was synthesized using Cadence's *Build Gates* and Alcatel CMOS035 digital library cells.

Fig. 10 compares area of particular digital filters without and with incorporated BIST in terms of used logical gates. Obviously, the additional BIST logic has insignificant influence on area occupation. Namely, the area is increased for 2.3 to 4.7 % depending on type of filter. This approves that the main goal of this work is achieved.

6. CONCLUSION

Embedded DFT and DFD techniques enable testing and diagnostic capabilities of the digital filters. The realization of this concept requires relatively simple hardware that consists of comparators, counters, registers and an additional status register that is a part of the chip internal memory. The most of stuck-at-0 and stuck-at-1 defects inside the arithmetic units are covered by specially selected test patterns. The test results can be easily obtained by reading the Status register through serial communication port.

ACKNOWLEDGMENT

The work presented in this paper was supported by Serbian Ministry of Science and Environment Protection within the project cipher TR-6108B.

REFERENCES

[1] M. Zwolinski "*Digital system design with VHDL*", Pearson education Ltd, England, 2000.

[2] P.Petković, " Projekat IT.01.01.0076B - Razvoj uređaja i sistema za merenje i upravljanje potrošnjom električne energije u industriji – Završni izveštaj", Niš, januar 2005.

[3] B. Jovanović, M. Jevtić, S. Đošić, M. Sokolović, P.Petković, "Projektovanje BIST logike u DSP bloku integrisanog merača potrošnje električne energije", Zbornik radova V simpozijuma industrijska elektronika INDEL 2004, Banjaluka, pp. 120-125.

[4] D. Marković, "A Power/Area Optimal Approach to VLSI Signal Processing", PhD Thesis, University of California, Berkeley, Spring 2006.

[5] M. Marinković, B. Anđelković, P. Petković, "Kompaktna arhitektura digitalnih decimacionih filtara u integrisanom meraču potrošnje električne energije", Zbornik radova VI simpozijuma industrijska elektronika INDEL 2006, Banjaluka

Sadržaj – U radu je opisana arhitektura testne logike koja omogućava testiranje digitalnih decimacionih filtara namenjenih za primenu u čipu za merenje potrošnje električne energije u trofaznim sistemima. Ona je zasnovana na primeni tehnika projektovanja za testabilnost (Design For Testability, DFT). Testiranje se zasniva na periodičnom izvršenju ugrađenih testnih funkcija kada se čip nalazi u posebnom, testnom modu rada.

PROJEKTOVANJE ZA TESTABILNOST DECIMACIONIH FILTARA U INTEGRISANOM MERAČU POTROŠNJE ELEKTRIČNE ENERGIJE

Miroslav Marinković, Borisav Jovanović, Predrag Petković