The Decomposition of DSP's Control Logic Block

Borisav Jovanović, Milunka Damnjanović, Dejan Stevanović

Abstract – The paper considers the architecture and low power design aspects of the digital signal processing block embedded into a three-phase integrated power meter IC. Utilized power reduction techniques were focused on the optimization of control logic block. The operations that control unit performs are described together with power-optimization results.

Keywords - digital signal processing, power optimization.

I. Introduction

Nowadays, the most of circuits used for measurement of power line parameters embed digital signal processors (DSP). This paper proposes a DSP circuit which enables high performances at the level as those obtained with commercial DSP microprocessors, and at the same time, saves the occupied chip area and minimizes power consumption.

The proposed DSP circuit is incorporated into Integrated Power Meter (IPM) system-on-chip. DSP receives from AD converters [1] and digital filters [2] 16-bit digital samples of voltage, current and phase-shifted voltage signals at data-rate of 4096 samples per second, and calculates following power-line parameters:

- root mean square values for voltage and current,
- mean values for active power, reactive power, distortion and apparent power,
- active and reactive energy,
- power factor , and
- frequency.

The measurement range for current signal is from 10mA RMS to 100A RMS, and for voltage it is up to 300V RMS. The results are obtained for three power line phases.

The paper explains the operations performed by DSP, including the novel digital filtering methods, used for processing the instantaneous values of current- and voltage-sample signals. Besides, new circuit for distortion power measurement is presented. Since DSP's control unit is one of largest and most power consuming DSP's part, the paper presents the utilized techniques for power minimization, which are mainly focused on optimization of control logic block.

Borisav Jovanović and Milunka Damnjanović are with the Department of Electronics, Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia,E-mail:
{borisav.jovanovic, milunka.damnjanovic} @elfak.ni.ac.rs.

Dejan Stevanović is with The Innovation Center, School of Electrical Engineering, University of Belgrade, d.o.o. (ICEF), Bul. Kralja Aleksandra 73, 11120 Belgrade, Serbia, E-mail:
dejan.stevanovic@venus.elfak.ni.ac.rs.

II. DSP's operation

A. *Controller/datapath architecture*

The architecture of DSP [3, 4] utilizes controller/ datapath architecture and consists of several blocks:

- Block 1 – the part which consists of arithmetical units used for I^2, V^2, P, Q accumulating and energy calculation
- Block 2 - including arithmetical operators used for calculation of current and voltage RMS, power factor, active, reactive, distortion and apparent power
- Block 3 - control unit that controls all other parts of DSP.
- Block 4 - frequency measurement circuit
- Block 5 - RAM memory block storing the measurement results

DSP's control unit (Block 3) is implemented as finite state machine (FSM). During DSP's measurement operation, the control unit periodically executes main state sequence that lasts 1024 clock periods [5], repeated 4096 times during the time interval of one second. The sequence is divided into four sub-sequences called R, S, T and E that lasts 256 clock periods each. The first three sub-sequences R, S and T control the calculations made for each phase of the three-phase energy system. The fourth sub-sequence, denoted E, manages the calculations that are periodically repeated every second [5].

The control unit is composed of four smaller finite state machines: named F0, F1, F2 and F3. The reason for dividing the control unit is significant power consumption reduction which will be examined in following sections. Two sub-FSMs, F1 and F2, perform arithmetical operations within the Block1 during the phases R, S and T, while sub-FSM F3 - performs operations within Block 2 during E period. The F0 is intended for RAM memory initialization and F0 is active only at the beginning of chip operation, after the main reset state. The operations that F1 and F2 perform will be described in detail.

B. *The operation of F1*

The FSM F1 executes the state sequence during the phases R, S and T and consists of one hundred and two states.

At the beginning of the F1 operation sequence, the AC part of instantaneous samples of current m_I_{ac} (stored in RAM block) is squared in the multiplication unit within the

Block1. The squared value m_I$_{ac}^{2}$ is then passed through the digital Low Pass Filter (LPF), and after, it is accumulated into the accumulation register m_AccI$_{ac}^{2}$.

The LPF is implemented as Infinite Impulse Response (IIR) digital filter and helps in reducing the I$_{RMS}$ calculation error. The error could exist because the time interval of one second (that is, accumulating time of the value m_I$_{ac}^{2}$) is not always equal to the integer number of power-line-signal half-periods. LPF has cut-off frequency 10Hz and its transfer function is given by Eq.1.

$$H_{LPF}(z) = \frac{2^{-6}}{1 - z^{-1}(1 - 2^{-6})} \tag{1}$$

The filter transfer function can be transformed into the following equations performed by DSP:

$$m\_FI_{ac}^{2}x64_{NEW} = m\_FI_{ac}^{2}x64(1 - \frac{1}{2^{6}}) + m\_I_{ac}^{2} \tag{2}$$

$$(m\_I_{ac}^{2})_{DC} = (m\_FI_{ac}^{2}x64)/64 \tag{3}$$

All these operations are done by arithmetical circuits within the Block 1. The structure of Block 1 is given in Fig.1 and includes one multiplication unit and one circuit for addition and subtraction. Only the inputs (the AC part of current signal m_I$_{ac}$, values of LPF register m_FI$_{ac}^{2}$x64 and accumulation register m_AccI$_{ac}^{2}$) are stored in the RAM memory block. The transfer of data between the Block 5 (RAM memory) and Block 1 is achieved through 24-bit data bus. The intermediate results of operations are temporarily stored in the registers RegA and RegB of Block1 (Fig.1).

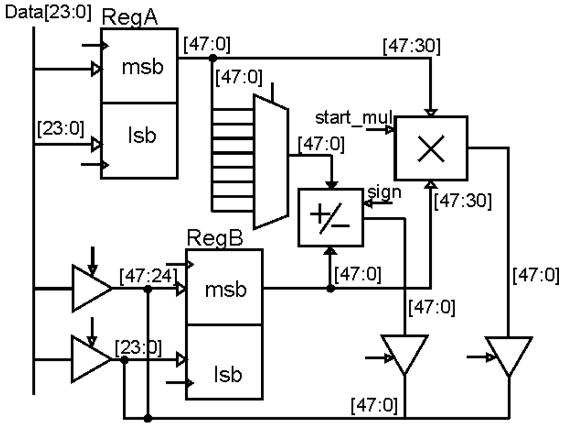

Fig.1 The structure of Block 1

The sequence of operations for the accumulation of squared current values is given by the Fig. 2. The sequence consists of simple data transfer, shifting, multiplication and addition operations which are performed at registers RegA and RegB.

The operations utilize contents of RAM memory registers:

- m_I$_{ac}$ – which contains the AC part of instantaneous current sample
- m_FI$_{ac}^{2}$x64_h and m_FI$_{ac}^{2}$x64_l are the 24-bit MSB and 24-bit LSB parts of 48-bit LPF register m_FI$_{ac}^{2}$x64, which contains the DC value of I$_{ac}^{2}$, multiplied by constant value equal to 64.
- m_AccI$_{ac}^{2}$ is 48-bit register for the accumulation of squared current samples.

$$\begin{array}{l}
\mathrm{m\_I_{ac}} \rightarrow \mathrm{RegA\_h, RegB\_h} \\
\mathrm{RegA\_h} \times \mathrm{RegB\_h} \rightarrow \mathrm{RegA} \\
\mathrm{m\_FI_{ac}^{2}x64\_h} \rightarrow \mathrm{RegB\_h} \\
\mathrm{m\_FI_{ac}^{2}x64\_l} \rightarrow \mathrm{RegB\_l} \\
\mathrm{RegA - (RegB >> 6)} \rightarrow \mathrm{RegA} \\
\mathrm{RegA + RegB} \rightarrow \mathrm{RegA} \\
\mathrm{RegA\_h} \rightarrow \mathrm{m\_FI_{ac}^{2}x64\_h, RegB\_h} \\
\mathrm{RegA\_l} \rightarrow \mathrm{m\_FI_{ac}^{2}x64\_l, RegB\_l} \\
\mathrm{m\_AccI_{ac}^{2}\_h} \rightarrow \mathrm{RegA\_h} \\
\mathrm{m\_AccI_{ac}^{2}\_l} \rightarrow \mathrm{RegA\_l} \\
\mathrm{RegA + (RegB >> 6)} \rightarrow \mathrm{RegA} \\
\mathrm{RegA\_h} \rightarrow \mathrm{m\_AccI_{ac}^{2}\_h} \\
\mathrm{RegA\_l} \rightarrow \mathrm{m\_AccI_{ac}^{2}\_l}
\end{array}$$

Fig.2 The sequence of accumulation of squared current values controlled by F1

The similar procedure is performed by Block 1 for processing the V$_{ac}^{2}$ (necessary for obtaining V$_{RMS}$) and instantaneous values of active and reactive power. The results are stored in the RAM registers: m_AccV$_{ac}^{2}$, m_AccP and m_AccQ. The difference is in the multiplication operands: voltage samples are multiplied to obtain V$_{RMS}$; voltage and current sample values for active power, and current-sample value is multiplied with phase-shifted voltage-sample for reactive power processing.

C. The operation of F2

The F2 is active during the phases R, S and T. It controls the energy pulses generation for measured active and reactive energy. It consists of one hundred and ninety three states. A pulse is generated when measured energy exceeds some predetermined energy level. The default energy level is one Whr (Watt-hour) for active and VAR (Volt-Ampere reactive) for reactive energy.

The DSP has four outputs producing the narrow pulses: Ea_pos – for consumed active, Ea_neg – generated active, Eq_pos –inductive reactive, and Eq_neg –capacitive reactive energy. The energy level is stored in m_Whr register, the part of RAM memory block, and can be modified. The operations are carried out by Block 1 using the adder/subtractor and registers RegA and RegB.

The sequence of operations is given in Fig.3. At the beginning of each sequence, performed exactly 4096 times during the time interval of one second, the active power value m_P, is added to the value of 48-bit register m_AccEa. The m_AccEa consists of two parts: the MSB part - m_AccEa_h and the LSB part - m_AccEa_l, both stored in RAM. After addition operation is done, the value of m_P and new value of m_AccEa are compared with zero. If value of m_P is positive and if new value of m_AccEa is greater than the energy level equivalent (given by m_Whr), a pulse on Ea_pos is generated and m_AccEa is subtracted by the m_Whr value. Else, if both m_P and m_AccEa are negative, a pulse on Ea_neg is generated, and value of m_Whr is added to m_AccEa.

The similar procedure stands for the reactive energy processing. Accompanied registers are m_AccEq_h and m_AccEq_h.

```
m_P → RegB_l
m_AccEa_h → RegA_h
m_AccEa_l → RegA_l
RegA + RegB → RegA
if (RegB > 0){
  m_Whr → RegB_l
  if ((RegA - (RegB << 12)) > 0){
     RegA - (RegB << 12) → RegA
     genarate pulse for positive Ea;
  }
} else{
   if (RegA < 0){
      m_Whr → RegB_l
      RegA + (RegB << 12) → RegA
      genarate pulse for negative Ea;
   }
}
RegA_h → m_AccEa_h
RegA_l → m_AccEa_l
```

Fig.3 The sequence of operations producing the energy pulses on Ea_neg and Ea_pos pins

Besides dealing with energy pulses, the F2 eliminates DC offsets from instantiations current and voltage signals that are derived from digital filters. This is necessary for the calculation of current and voltage RMS value. The DC offset will give a DC component after squaring operation. Since this DC component is extracted by LPF, this offsets can induce the error to RMS values. This problem is avoided by introducing the HPF in voltage and current signal processing chains. The HPF, applied to instantaneous current and voltage signals, is implemented as Infinite Impulse Response (IIR) digital filter with cut-off frequency 5Hz and transfer function as given by Eq.4:

$$H_{HPF}(z) = (1-2^{-10})\frac{(1-z^{-1})}{1-z^{-1}(1-2^{-9})} \tag{4}$$

The HPF transfer function can be transformed into the equations (5) and (6) performed by DSP.

$$m\_FIx1024_{NEW} = m\_FIx1024(1-\frac{1}{2^{9}}) + (2^{10}-1)(m\_I - m\_I\_p) \tag{5}$$

$$m\_I_{ac} = m\_FIx1024/1024 \tag{6}$$

The following registers values are used in the equations (5) and (6):

- m_I and m_I_p - two consecutive current samples
- m_FIx1024 is 48-bit HPF register, which contains the AC value of I, multiplied by constant value 1024. The register consists of two parts: the MSB part - m_FIx1024_h and LSB part - m_Fix1024_l.
- m_I$_{ac}$ is AC part of instantaneous sample of current signal. It represents the result of filtering operation and it is further used by FSM F1.

```
m_I_p → RegA_l
m_I → RegB_l
RegA_l - RegB_l → RegA_l
RegA → RegB
RegA - (RegB << 10) → RegA
//RegA = ((1024 - 1)(m_I - m_I_p))
m_FIx1024_h → RegB_h
m_FIx1024_l → RegB_l
RegA + RegB → RegA
RegA − (RegB >> 9) → RegA
//RegA = (1024 - 1)(m_I - m_I_p) +
//          m_FIx1024 (1 - 2^-9)
RegA_h → m_FIx1024_h
RegA_l → m_FIx1024_l
RegA → RegB
0 → RegA_l
RegA + (RegB >> 10) → RegA
RegA_l → m_I_ac
```

Fig.4 The sequence for high pass filtering of instantiations current sample signals, done by F2

The operation sequence for the offset elimination, performed by F2, is given in the Fig.4. The operations are carried out by Block 1.

The similar procedure is made for processing of m_Vac (necessary for obtaining V_{RMS}). The intermediate results are stored in 24-bit RAM registers: m_FVx1024_h and m_FVx1024_l.

D. *The operation of F3 FSM*

The fourth sub-sequence of the control unit, manages the calculations that are periodically repeated every second and consists of one three hundred and four states.

Based on accumulating sums $m\_AccI_{ac}^2$, $m\_AccV_{ac}^2$, m_AccP and, m_AccQ, arithmetical operations are performed by Block 2 to generate voltage and current root mean square values m_I_{RMS} and m_V_{RMS} and mean active and reactive power values m_P and m_Q. The sequence of operations is performed by FSM F3.

The interior structure of Block 2 is given in Fig.5. It consists of two registers named RegC and RegD and arithmetical units that implement square rooting, subtraction, multiplication and division.

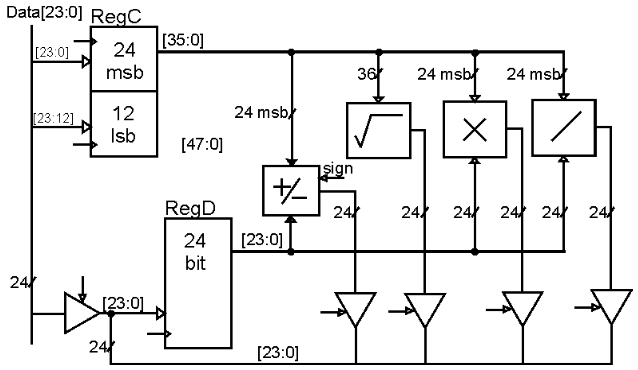

Fig.5 The structure of Block 2

The sequence, controlled by F3 that generates current root mean square m_I_{RMS}, is given in Fig.6.

To generate m_I_{RMS}, accumulated sum $m\_AccI_{ac}^2$ is stored into RegC and then, it is divided by 4096. Next, square rooting operation is performed over the average value of voltage square. Then, current offset m_I_{ACoff} is subtracted, multiplied with gain correction m_Igain and root mean square of current is obtained (Fig.6).

The similar processing steps are conducted for m_V_{RMS}. For mean active and reactive power calculation the square root calculation is avoided. Apparent power m_S is obtained by multiplying m_I_{RMS} and m_V_{RMS}, and power factor m_CosF – by dividing active m_P and apparent power m_S.

$$m\_AccI^2\_h \rightarrow RegC\_h$$
$$m\_AccI^2\_l \rightarrow RegC\_l$$
$$\sqrt{RegC} \rightarrow RegD$$
$$0 \rightarrow m\_AccI^2$$
$$m\_I_{ACoff} \rightarrow RegC\_h$$
$$RegC\_h - RegD \rightarrow RegD$$
$$m\_Igain \rightarrow RegC\_h$$
$$RegC\_h \times RegD \rightarrow RegD$$
$$RegD \rightarrow m\_I_{RMS}$$

Fig.6 The sequence that generates current root mean square m_IRMS

In addition to finding mean active (m_P), reactive (m_Q) and apparent power (m_S), the distortion power [6] (stored in the register m_D) calculation is provided.. F3 controls the operations producing the m_D. Arithmetical operators used to calculate the value of m_D, belong to blocks 1 and 2. The structure of Block 1 had to be slightly modified. The new input is introduced to RegB which makes the connection from the multiplication unit from Block2. The result of multiplication operation, done by arithmetical operator within Block2, has to be transferred to the RegB in Block1. The sequence is given in Fig.7.

At the beginning, the register RegA is reset to zero, and the content of register m_S is copied to both of the registers RegC and RegD. The squaring operation is performed and the result is moved to the RegA. Then, the active power m_P is moved to RegC and RegD, and the multiplication is performed. The result is subtracted from register RegA. The same operations are done with the value m_Q. After, the content of RegA is moved to the RegC, and square root operation is performed. Finally, the result is moved from RegD into the m_D, which is stored in the RAM memory.

$$0 \rightarrow RegA\_h, RegA\_l$$
$$m\_S \rightarrow RegC\_h, RegD$$
$$RegC\_h \times RegD \rightarrow RegB$$
$$RegA + RegB \rightarrow RegA$$
$$m\_P \rightarrow RegC\_h, RegD$$
$$RegC\_h \times RegD \rightarrow RegB$$
$$RegA - RegB \rightarrow RegA$$
$$m\_Q \rightarrow RegC\_h, RegD$$
$$RegC\_h \times RegD \rightarrow RegB$$
$$RegA - RegB \rightarrow RegA$$
$$RegA \rightarrow RegC$$
$$\sqrt{RegC} \rightarrow RegD$$
$$RegD \rightarrow m\_D$$

Fig.7 The sequence that generates distortion power m_D

III. THE IMPLEMENTATION RESULTS

The most of optimization process considered the DSP's control unit. The control unit incorporates over six hundred states and this large number of states required huge combinational logic of synthesized FSM. The implementation occupies large portion of DSP's area. Also, it represents one of the largest power consumers among DSP's blocks.

The following power minimization techniques were used: FSM decomposition [7, 8], clock gating and Grey code encoding [9]. The first technique divides large control unit into several smaller state machines, simplifying their combinatorial logic blocks. The division of control unit into smaller state machines has positive effect on power dissipation. Furthermore, the clock gating disables inactive parts of FSM by stopping its clock signal, and, reduces the switching activity within the combinatorial logic blocks. Besides, Gray binary encodings are assigned to the FSM's states.

The transition graph of original FSM was considered first, and after, divided into four sub-graphs (F0, F1, F2 and F3) that jointly produce the equivalent behaviour as the original FSM. The decomposition is performed by considering the datapath architecture. The states within one subset control the arithmetical operations performed by same part of DSP. As stated earlier, F1 and F2 perform the operations within the Block1, while F3 – mainly within the Block 2.

After the FSM decomposition is done, the clock gating is introduced in the FSM's implementation. New circuit is added into control logic block which identifies currently active sub FSM. The circuit also provides clock input signals to sub FSMs. The clock signal is present only at the input of active sub FSM, and the other three sub FSMs are blocked.

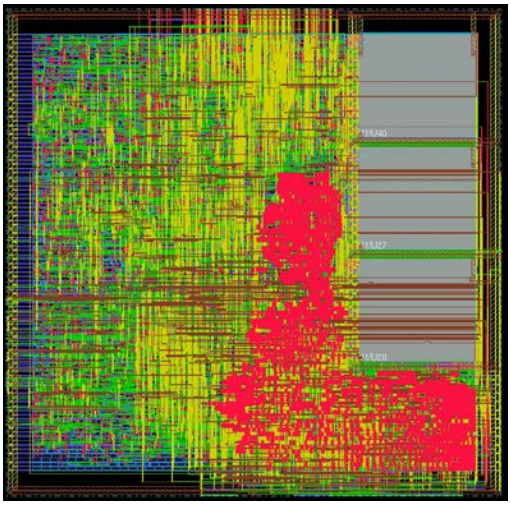

Fig.8 The layout of DSP

The DSP was implemented in technology AMI CMOS 350nm with power supply voltage of 3.3V.

When the design was verified by RTL simulation, the RTL descriptions were loaded into program for logical synthesis, Cadence's RTL Compiler that generated the netlist of digital library cells. The extracted netlist was loaded back to Verilog simulator and the simulation was performed using Cadence' NCsim tool.

SoC Encounter has performed floorplanning, placement and routing, as well as clock and reset trees generation for complete circuit (Fig.8). At the end of logical verification process, Verilog file was extracted from layout and brought back to NCsim simulator where final check of the total digital part of the IC was performed.

During the post-layout simulation, switching activity file was obtained and the power consumption results are obtained by the SoC Encounter taking account the parasitic capacitances from layout and switching activity file.

The estimation of DSP's power consumption gave the valuable information about the energy budget and identified all power hungry components. Three power analyses were performed: for the: (a) original design (before the power minimization techniques where applied), (b) DSP design which is optimized by gating and FSM decomposition, and finally, (c) design where all proposed techniques were applied: FSM decomposition, clock gating and Gray state encoding. The Table 1 gives the simulated power consumption values of different DSP cores, derived after layout generation. The power consumption of not-optimized design was 1.82mW .When all these techniques were applied, the total power became only 1.043mW resulting in the 42% switching power reduction, comparing to the non-optimized implementation.

TABLE I

THE RESULTS OF POWER OPTIMIZATION

	Not optimized	Decomposition & clock gating	Decomp., clock gating & Grey encoding
Area	1.84mm^2	1.831mm^2	1.823mm^2
Clock tree power	0.732mW	0.263mW	0.227mW
Control unit power	0.407mW	0.172mW	0.172mW
DSP's power	1.82mW	1.117mW	1.043mW

IV. CONCLUSION

The architecture and the low power design aspects of the digital signal processing block embedded into a three-phase integrated power meter IC, are considered. The operations that control unit performs are described together with power-optimization results.

The power reduction techniques were successfully implemented on the optimization of the control logic block.

The control unit of DSP block, implemented as finite state machine, was decomposed into four smaller state machines, clock gating was completely introduced and Gray finite state machine encoding used. The resulting effect was the significant reduction of the power consumption.

ACKNOWLEDGEMENT

This research was partially funded by The Ministry of Education and Science of Republic of Serbia under contract No. TR32004

.

REFERENCES

[1] Mirković, D., Petković, P. *"Multi channel Sigma-Delta A/D converter for integrated power meter"*, Proceedings of the Small Systems Simulation Symposium 2010, Niš, ISBN 987-86-6125-006-4, Feb., 2010, pp. 90-93.

[2] Marinković, M., Andjelković, B., Petković, P. "*Compact Architecture of Digital Decimation Filters in Solid-State Energy Meter*", Electronics, Vol. 10, No. 2, University of Banja Luka, ISSN 1450, December 2006, pp. 28-32.

[3] Jovanović, B., Damnjanović, M., Petković, P.*" Digital Signal Processing for an Integrated Power Meter"*, Proceedings of 49. Internationales Wissenschaftliches Kolloquium, Technische Universirtat Ilmenau, Ilmenau, ISBN 3-8322-2824-1, Vol.2, September, 2004, pp. 190-195.

[4] Jovanović, B., Zwolinski, M., Damnjanović, M. *"Low power digital design in Integrated Power Meter IC"*, Proceedings of the Small Systems Simulation Symposium 2010, Niš, ISBN 987-86-6125-006-4, Feb., 2010, pp. 49-55

[5] Jevtić, M., Jovanović, B., Brankov, S.*" Upravljačka jedinica sistema na čipu za registrovanje potrošnje električne energije"*, Zbornik radova XLVIII konferencije Etran 2004, Čačak, ISBN 86-80509-49-3, Vol.1, June, 2004, pp. 75-78

[6] Stevanović, D., Jovanović, B., Petković, P., Litovski, V. *"Korišćenje snage distorzije za identifikaciju izvora harmonijskog zagađenja na mreži"*, Tehnika - Elektrotehnika, 6/2011, Savez inženjera i tehničara Srbije, ISSN 0040-2176, 2011, accepted for publication.

[7] Chow, S.H., Yi-Cheng Ho, Y.C., Hwang, T. "*Low power realization of finite state machines - a decomposition approach*", ACM Transactions on Design Automation of Electronic Systems (TODAES) Volume 1, Issue 3 (July 1996) pp.: 315 – 340, ISSN:1084-4309

[8] Lee, W.K., Chi-Ying Tsui, C.Y. "*Finite state machine partitioning for low power*", Circuits and Systems, 1999. ISCAS'99, Proceedings of the 1999 IEEE International Symposium, Volume 1, June 1999, pp. 306 – 309

[9] Benini, L., De Micheli, G. "*State assignment for low power dissipation*", Solid-State Circuits, IEEE Journal of Volume 30, Issue 3, Mar 1995 pp.:258 – 268