Fuzzy Control of Coding Schemes for Reducing Energy Dissipation in Off-Chip Buses

Giuseppe Visalli
Serial Non-Volatile Memories Group, Numonyx, Catania Italy, 95121
(Received: 9 February 2008; Accepted: 9 June 2008)

In this paper, we proposed an high-speed and low-power off-chip data bus interface based on the best coding schemes in this hard operative condition. We analyzed the clustered bus invert method and the bus switch coding, a newly proposed approach based on bus lines logically re-ordered. We proposed an high speed and low-power bus interface based on the combined employment of these two approaches controlled by a 9-rules Takagi-Sugeno analog fuzzy controller. The controller analyzes the binary traffic statistical property changing on the fly the used coding scheme. The fuzzy controller has been designed taking care of total energy dissipation such to do not compromise the benefit of coding approaches. The controller is able also to re-configure the bus switch sub-section in an operative condition where original approach introduces strong power losses. We demonstrated the effectiveness of the approach designing at transistor level the analog fuzzy controller and the digital part of the bus interface. Simulation conducted with H-SPICE and NANOSIM confirmed the bus interface is the optimal trade-off for reducing dynamic energy in off-chip buses.

Keywords: Bus Transfer, Encoding, Fuzzy Control, Fuzzy Systems, Power Demand, Very Large Scale Integration (VLSI).

1. INTRODUCTION

High-speed and low-power data transfer is a major concern in the design of very complex system on chip (SoC). In the last decade, data communication has been faced using standard parallel buses such as PC-AT ISA and recently PCI and AGP data buses. Table I summarizes the principal electrical and operative specifications. Applications in portable devices (e.g., laptop) suggest the employment of some technological approach or system solution to satisfy the dramatically grow of the data rate. Voltage swing reduction\(^\text{14}\) represents a technological approach, such to reduce power and to increase data transfer. Coding instead represents another approach which is extensively tackled in the past. Prior works own into different categories: power reduction in address or data buses,\(^\text{2, 13, 19, 20, 24}\) on or off\(^\text{6}\), chip buses, high, or low data rate. Bus invert represented a first tentative to encode information for low-power in data buses.\(^\text{21}\) This simple circuit has been employed in different design contexts: parallel off-chip buses, low-data rate transmission and embedded processors.\(^\text{19}\)

Bus switch (BS) coding has been introduced faced the problem of reducing dynamic energy on external buses.\(^\text{15, 13, 26, 27}\) Bus switch reduces energy better than prior approach in particular operative conditions. Clustered bus invert (CBI) represents the best BS competitor since the activity ratio reduction and hardware complexity. Data coding, for reducing energy on buses has been validated using a set of limited benchmarks in the past. Instead, simulations certified the effectiveness of a coding scheme varying the bus lines main statistical momentum. Table II summarized the activity saving (SA = switching activity) using different coding strategies: the clustered bus invert the bus switch both in 3X or 4X cluster size, the adaptive partial bus invert (APBI)\(^\text{19}\) and finally the bus invert approach (BI). Results in Table II show how cluster-based bus encoding achieves maximum performance. Additionally, bus switch approach has maximum activity performance operating with lower logical 1 density, while a disastrous performance in higher densities suggest the utilization of an inverted version. This last approach differs from original BS coding the 1’s complement data. The activity reduction in Table II indicated three operative conditions, using BS, CBI, and “inverted” BS (\(\overline{\text{BS}}\)).

Although compressed streams reach the 1 density close to 50% where the CBI is the best encoding algorithm, the proposed approach is convenient when time-expensive online compression strategies compromise the data rate. Off-line compression algorithms cannot be used in real-time systems. The work\(^\text{3}\) illustrated off-line compression
strategies where power saving is measured in terms of bus accesses reduction. The authors did not provide any cod-
ing approach to reduce the toggle rate. Instead, the work\cite{17} illustrated the basic theory to derive simple procedure to modify the bus entropy, directly related to the 1 density used in this work. An approach similar to the proposed probability-based mapping can change the 1 density in the bus switch regions where the activity saving is twice com-
pared to CBI.

Fuzzy control represented a good strategy to exchange the interface operative condition dynamically in the set \{BS, CBI, B\}.\cite{22} Analog fuzzy controller permits higher operable bus frequency compared to a digital implemen-
tation. We employed a low-complexity 9-rules zero order Takagi-Sugeno controller. The choice of fuzzy decision mechanism is based on the problem’s nature. There are some important points to be recalled.

1. The bus pattern in a transmission interface is a non stationary random process at two distinct value (0, 1) in each line. The non stationary properties imply the statistics of any order change in the time. This means the 1 density changes in the time in the full range 0\%–100\%.
2. Each bus line has a different time-variant statistics, that possibly implies a wrong encoder selection based on the 1 density estimation in a much different line compared to the average density.
3. It is very difficult to compare any kind of bus encoding technique using a much reduced set of benchmarks (data, code, mixed) or to profile the encoding technique using constant-statistic bus traffic, varying the 1 density from 0\% to 100\%. We simulated in this last condition and the results are in Table II.
4. The 1 density can be considered a crisp set (using deterministic regulation such as the history-based

### Table II. Activity savings varying logic “1” density (30-line bus).

<table>
<thead>
<tr>
<th>Density (%)</th>
<th>CBI (%)</th>
<th>BSX (%)</th>
<th>BSX (%)</th>
<th>API (%)</th>
<th>BI (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.076</td>
<td>44.19</td>
<td>43.736</td>
<td>-0.0387</td>
<td>0.00</td>
</tr>
<tr>
<td>20</td>
<td>2.262</td>
<td>37.435</td>
<td>34.716</td>
<td>0.404</td>
<td>0.20</td>
</tr>
<tr>
<td>50</td>
<td>15.168</td>
<td>12.711</td>
<td>16.734</td>
<td>9.870</td>
<td>11.00</td>
</tr>
<tr>
<td>60</td>
<td>12.950</td>
<td>3.772</td>
<td>11.026</td>
<td>7.612</td>
<td>8.38</td>
</tr>
<tr>
<td>70</td>
<td>7.434</td>
<td>-2.883</td>
<td>-1.724</td>
<td>2.851</td>
<td>2.93</td>
</tr>
<tr>
<td>80</td>
<td>2.109</td>
<td>-48.614</td>
<td>-32.958</td>
<td>0.407</td>
<td>0.20</td>
</tr>
<tr>
<td>90</td>
<td>0.151</td>
<td>-161.195</td>
<td>-151.384</td>
<td>-0.0664</td>
<td>0.038</td>
</tr>
<tr>
<td>100</td>
<td>0.1654</td>
<td>-856.111</td>
<td>-737.053</td>
<td>-0.0827</td>
<td>0.018</td>
</tr>
</tbody>
</table>

### Table III. 9-rules Takagi-Sugeno fuzzy controller

<table>
<thead>
<tr>
<th>Rule no.</th>
<th>Fuzzy IN0</th>
<th>Fuzzy IN1</th>
<th>Coding scheme</th>
<th>Singleton</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LOW</td>
<td>LOW</td>
<td>BS</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>LOW</td>
<td>MEDIUM</td>
<td>BS</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>LOW</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>MEDIUM</td>
<td>LOW</td>
<td>CBI</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>MEDIUM</td>
<td>MEDIUM</td>
<td>CBI</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>MEDIUM</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>HIGH</td>
<td>LOW</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>HIGH</td>
<td>MEDIUM</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>HIGH</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
</tbody>
</table>

\end{quote}

\begin{flushright}
\textsuperscript{20}tor\end{flushright} or a fuzzy set that take account of uncertainty and vagueness of the problem as described in the previous three points.

The point 4 is crucial to justify the fuzzy approach. The history-based predictor measures the 1 density using binary counter and final comparator enables the cor-
rect encoder in the set \{BS, CBI, B\}. The predictor VLSI circuit compared the estimated 1 density with fixed thresholds based on experiment and some rough statistical estimation. The choice of these thresholds has no any com-
plete theoretical justification based on the unpredictable nature of the 1 density in a real bus traffic.

The theory on analog fuzzy controller introduced two approaches: the Magdami and the Takagi-Sugeno controller. These two approaches differ from the de-
fuzzification circuit. The Takagi-Sugeno controller has simple de-fuzzifiers implemented in this work as center of gravity (COGI) using a MOS resistor network. The con-
troller analyzed fuzzy sets receiving an estimation of logical “1” density in some representative input bus lines. The fuzzy variables own to the linguistic set: \{LOW, MEDIUM, HIGH\}. The fuzzy controller elaborates nine rules accord-
ing to Table III. The bus interface now admits three dif-
ferent coding schemes for reducing energy dissipation in off-chip buses. These coding circuitry have been selected dynamically using the linear fuzzy controller. The VLSI circuit implementation and post-layout evaluation indicated better energy gains, compared to prior art, in every bus statistical operative condition. The paper follows this orga-
nization: Section 2 is an overview of Bus Switch, inverted Bus-Switch, and Clustered Bus invert. Section 3 defines the Fuzzy controller functionality and VLSI circuit implementa-
tion. Section 4 is reserved for experimental results. Lastly Section 5 concluded the paper.

### 2. CLUSTERED CODING SCHEMES FOR REDUCING BUS ENERGY

Experimental results in Table II showed how cluster-
based coding approaches permit significant energy reduc-
tion at high data-rate. In this work, we considered the BS mechanism, a coding approach based on bus clustering, and CBI derived from Ref. \cite{21}.

---

**Table I. Standard off-chip buses main specifications.**

<table>
<thead>
<tr>
<th>Bus</th>
<th>Voltage swing</th>
<th>Frequency</th>
<th>Data rate</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISA PC-AT</td>
<td>5 V</td>
<td>10 MHz</td>
<td>2 BYTES/CLK</td>
<td>20 MB/S</td>
</tr>
<tr>
<td>VESA</td>
<td>3.3 V</td>
<td>33 MHz</td>
<td>4 BYTES/CLK</td>
<td>133 MB/S</td>
</tr>
<tr>
<td>PCI-104</td>
<td>3.3 V</td>
<td>33 MHz</td>
<td>4 BYTES/CLK</td>
<td>133 MB/S</td>
</tr>
<tr>
<td>AGP-1X</td>
<td>3.3 V/1.5 V</td>
<td>66 MHz</td>
<td>8 BYTES/CLK</td>
<td>266 MB/S</td>
</tr>
<tr>
<td>AGP-2X</td>
<td>3.3 V/1.5 V</td>
<td>133 MHz</td>
<td>8 BYTES/CLK</td>
<td>553 MB/S</td>
</tr>
<tr>
<td>AGP-4X</td>
<td>1.5 V</td>
<td>266 MHz</td>
<td>16 BYTES/CLK</td>
<td>1066 MB/S</td>
</tr>
<tr>
<td>AGP-8X</td>
<td>0.8 V</td>
<td>533 MHz</td>
<td>32 BYTES/CLK</td>
<td>2.1 GB/S</td>
</tr>
</tbody>
</table>

---

**Table III. 9-rules Takagi-Sugeno fuzzy controller.**

<table>
<thead>
<tr>
<th>Rule no.</th>
<th>Fuzzy IN0</th>
<th>Fuzzy IN1</th>
<th>Coding scheme</th>
<th>Singleton</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LOW</td>
<td>LOW</td>
<td>BS</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>LOW</td>
<td>MEDIUM</td>
<td>BS</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>LOW</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>MEDIUM</td>
<td>LOW</td>
<td>CBI</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>MEDIUM</td>
<td>MEDIUM</td>
<td>CBI</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>MEDIUM</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>HIGH</td>
<td>LOW</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>HIGH</td>
<td>MEDIUM</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>HIGH</td>
<td>HIGH</td>
<td>B\textsuperscript{\textregistered}</td>
<td>2</td>
</tr>
</tbody>
</table>
In the work\textsuperscript{12} the authors defined the energy saving compared to original bus power dissipation. The used coding approach reduced bus energy dissipation conveniently, when the parameter $E\%$, defined in the Eq. (1), is less than 100.0%.

$$E\% = \frac{0.5 \cdot C_{bus} \cdot V_d^2 - 0.5 \cdot \alpha \cdot C_{bus} \cdot N \cdot V_d^2 - E_{overhead}}{0.5 \cdot C_{bus} \cdot V_d^2} \cdot 100.0\%$$  

(1)

“$N$” represents the line overhead due coding (extra bus) in Eq. (1). Our simulations considered 30-line bus with additional four-wires extra bus (pattern and inv), so $N$ is 34/30. $C_{bus}$ is the bus capacitance and $V_d$ is the power supply. The parameter $\alpha$ represents the activity reduction related to the transition density defined in Ref. [11]. If $E\%$ is greater than 100.0% the encoder power dissipation (Eoverhead) overcome the energy saving in the bus. Equation (2) represents the minimum bus capacitance to make effective the used coding.

$$C_{bus} = \frac{E_{overhead}}{0.5 \cdot \alpha \cdot N \cdot V_d^2}$$  

(2)

2.1. The Bus Switch Mechanism

2.1.1. Theory

In principle, the BS technique can be logically (i.e., not corresponding to a physical realization) expressed as a four-step process:

1. A large bus is divided into several identical buses (clusters of $M$ lines each ($M$ is cluster depth).)
2. Each $M$-line bus is coded by swapping the input lines using a particular reordering pattern.
3. A tentative data encoding is obtained by applying to the swapped $M$ lines a fixed coding function.
4. The process is repeated $M!$ times from step 2 until the optimal reordering pattern is found, that minimizes the output switching activity in the encoded data of the whole bus.

In the following a formal definition of the process is given. Let $b(t)$ be input data word to the bus encoder and $B_{opt}(t)$ the encoded data word on the bus, at clock cycle $t$. The single bits of any $M$-bit data word $x(t)$ will be indicated as $x(t)(0), x(t)(1), \ldots, x(t)(M - 1)$.

DEFINITION 1. A reordering pattern $p(t)$ is an ordered set of $M$ indices $i_0, \ldots, i_{M-1}$, associated with clock cycle $t$. Given a data word $x(t)$, the swap operator $S$ with reordering pattern $p$ is a combinational logic function producing a swapped data word $y(t) = S(x(t), p(t))$, such that:

$$y(t)(0) = x(t)(i_0)$$
$$y(t)(1) = x(t)(i_1)$$
$$\ldots$$
$$y(t)(M-1) = x(t)(i_{M-1})$$

As an example, if $p(t) = \{1, 2, 3, 0\}$ and $x(t) = \langle 0100 \rangle$, then $S(x(t), p(t)) = \langle 1001 \rangle$. Note that each reordering pattern $p(t)$ has a unique inverse $p^{-1}(t)$, such that $x(t) = S(S(x(t), p(t)), p^{-1}(t))$.

DEFINITION 2. A coding function is a combinational logic function producing a data word $B(t)$ using the actual data $b(t)$, the previous encoded data $B_{opt}(t - 1)$, and any other word resulting from applying swapping to $b(t)$ and $B_{opt}(t - 1)$.

In the work\textsuperscript{12} authors evaluated different coding/decoding functions. ANSI C simulation showed the best activity reduction using $B(t)$ and $b(t)$ as follows:

$$B_{opt}(t) = S(b(t), p) \oplus S(B_{opt}(t - 1), p^{-1})$$  

(3)

$$b(t) = S(S(B_{opt}(t - 1), p^{-1}) \oplus B_{opt}(t), p^{-1})$$  

(4)

The symbol $\oplus$ in (3) and (4) represents the bit-wise XOR operator. $B_{opt}$ represents the encoder output bus, $b(t)$ the input bus, “$p$” and “$p^{-1}$” the current optimal direct and inverse reordering patterns respectively.

DEFINITION 3. The optimal reordering pattern $p_{opt}(t)$ is the pattern that minimizes the switching activity $H[B_{opt}(t - 1) \oplus B(t)]$, where $H$ is the Hamming distance from previous transmission $B_{opt}(t - 1)$ and the coding function result, varying the reordering pattern.

2.1.2. VLSI Architecture Design and Implementation

The BS encoder maximum parallelism architecture\textsuperscript{11} is organized in $M!$ (factorial) ways. Each way provided a tentative encoded word using a single reordering pattern “$p$” applied to the input bus $b(t)$. In the work\textsuperscript{12} authors considered $M = 4$ and $M = 5$ and a procedure of pattern set reduction to decrease hardware complexity and therefore increasing the minimum bus capacitance (2). Recent studies confirmed $M = 3$ a better hardware complexity and activity reduction tradeoff that improves the minimum bus capacitance. The single way provided also the tentative encoded word transition count (score) applying the Hamming operator ($H$) to the bit-wise XOR between the encoder word and the registered output bus (using clocked registers). The encoder selects the optimal code-word searching for the minimum score using the module “best” (see Fig. 1). The same module provides the signal used to drive the output multiplexer that receives the six (3!) tentative encoded words. This signal represents also the way number which competes the minimum score (and therefore the ordinal position of the optimal reordering pattern in the set) and it is subsequently encoded using a narrow bus and transmitted to permit the data recovery. This extra bus is composed by three lines if $M = 3$ (the minimum integer of log2($M!$)) and we can use the bus invert method to reduce power providing an additional INV line. Lastly, the encoder registers internally the output.
bus \( B(t - 1) \), used to evaluate the coding function (3). In the work\(^{12} \) authors evaluated a possible alternative to the maximum parallelism architecture to decrease the encoder area and power dissipation. The work\(^{14} \) finally excluded this approach that represents a strong limit to the maximum operative bus frequency. The static timing analysis (STA) applied to the single way concluded the critical timing path moves across the \( H \) module that performs the Hamming operator to calculate the transition count. This circuit represents the most delay consuming unit in the critical path.

### 2.2. The Inverted Bus Switch Coding (BS)

The BS coding has worse performance working at higher “1” density as showed by our simulations conducted with ANSI C encoder model. As example, at time \( t = 0 \) the internal register is all zero:

\[
B(t - 1) = "0000 \ldots 000"
\]

If we suppose to operate at “1” density close to 100.0\%: \( b(t) = "1111 \ldots 111" \). The best encoded word is therefore:

\[
B(t) = "1111 \ldots 111", \quad SA = N
\]

In the following step, \( B(t - 1) = "1111 \ldots 111" \) and \( b(t) = "1111 \ldots 111" \). In this operative condition, the best encoded word is therefore:

\[
B(t) = "0000 \ldots 000", \quad SA = N
\]

The BS is still useful in this operative condition, coding the data with the 1’s complement, moving the performance toward the best working zone.

### 2.3. The Clustered Bus Invert Mechanism

#### 2.3.1. Theory

The CBI approach organizes the large bus into \( N \)-line clusters. The encoder implements the BI approach according to the work\(^{23} \) in each cluster:

\[
B(t) = \begin{cases} 
    b(t) & SA < N/2 \\
    \tilde{b}(t) & SA \geq N/2
\end{cases}
\]

#### 2.3.2. VLSI Architecture: Design and Implementation

The CBI is a set of BI encoders. In this work, the 30-line bus is organized in three clusters of \( N = 10 \)-line each. This choice admits a total of three extra lines organized in a narrow bus (INV[2:0]) according to the BS extra bus of the same size.

Table IV summarized the most important technology data, translating the VLSI architectural description (Verilog) in a 90 nm CMOS. We considered the Bus Invert, the Bus Switch 3X (6 patterns) and reduced 4X (16 patterns see Ref.\([12]\)) and the adaptive partial bus invert. The BS mechanism is area and power hungry, but permits energy saving more than 40\%. Instead CBI saves a lot of power and area, but the performance are relevant only in a small operative condition, at density in the range 45\% to 55\%.

The proposed bus interface includes a BS3X and a CBI encoder as depicted in the Figure 1. The analog fuzzy controller included in the transmitter estimated the 1 density in some representative bus lines to select the interface operative mode in the set \{BS, CBI, BS\}. The analog fuzzy controller selects a wrong encoder mode when the used lines have time-variant statistics much different from the average. The wrong estimation can significantly decrease the activity reduction under the 0\% using BS and BS only. We included the CBI encoding that introduces superior activity reduction in the range 40\%–60\% (compared to BS see Table II) and also to avoid strong negative activity savings in the case of fuzzy wrong decision. The wrong decision event can be mitigated increasing the monitored bus lines. The analog fuzzy controller controls the coding mode providing the additional signals “BS” (bus switch mode) and “encm” (encoder mode). The former electrical

### Table IV. Coding schemes area and energy waste comparison (30-line bus and 90 nm CMOS, \( f_s = 200 \text{ MHz} \))

<table>
<thead>
<tr>
<th>Codec</th>
<th>Area (( \mu m^2 ))</th>
<th>Power (( \mu W ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>BI</td>
<td>1779.2 ( \mu m^2 )</td>
<td>316.03 ( \mu W )</td>
</tr>
<tr>
<td>CBI</td>
<td>1606.8 ( \mu m^2 )</td>
<td>283.13 ( \mu W )</td>
</tr>
<tr>
<td>AP8I</td>
<td>15085.4 ( \mu m^2 )</td>
<td>1.328 mW</td>
</tr>
<tr>
<td>BS3X</td>
<td>13211.8 ( \mu m^2 )</td>
<td>2.70 mW</td>
</tr>
<tr>
<td>BS4X*</td>
<td>45483.2 ( \mu m^2 )</td>
<td>11.09 mW</td>
</tr>
</tbody>
</table>

*28-line bus, 36 ways.
Wire selects the BS mode coding, original data or 1’s complement. Instead the "encm" wire selects the BS (or B̅S) or the CBI encoder mode.

3. A 9-RULES ZERO ORDER TAKAGI-SUGENO FUZZY CONTROLLER

There are several regulation methods suitable for this kind of problem. However, fuzzy control represents a good choice where uncertainty and vagueness characterize the bus line statistical distribution. Fuzzy processor has been tackled in the past, both for analog and digital design contexts. In fact, each bus line follows a particular statistical distribution in some case with different linguistic values (that own to the used term set: LOW, MEDIUM, HIGH). Additionally, fuzzy system are relatively small and fast, able to support bus encoding circuits at higher frequencies.

A fuzzy system analyzes fuzzy sets that own to the term set above. The system admits a total of 3^L fuzzy inferences, where L represent the monitored bus lines. For this reason, we operate with L = 2 to control the hardware complexity and in fact to reduce as most as possible the power consumption. Table III reported the used fuzzy rules. We derived these rules in a conservative way: the controller selects the BS when at least one monitored bus line is HIGH. This choice permits the fuzzification of two crisp set representing the logical one density in two representative bus lines in the set of 30 available.

3.1. VLSI Circuit Implementation

The fuzzification process has been accomplished by two 4-bit digital to analog converter (DAC) in the range 0–500 mV. The crisp sets represent the logical 1 density measured by binary counters in a 16 clock cycles observation window. Power supply in 90 nm HVT (low leakage) CMOS is 1.08 Volt. The controller evaluates the firing degrees using the signals from DACs. We made reference to the work, using the Differential regulated-cascode triode trans-conductors for the complementary membership functions (CFMF) illustrated in Figure 2. We used the complementary membership function since S-Norm operators are easier to implement than T-Norm with analog current-mode hardware. We made reference to the works (Ref. [8]) and (Ref. [9]) to derive the low-power CMOS architecture for the fuzzy inference and the de-fuzzifier. The current mode rule evaluation is a new elaboration of the voltage mode introduced in the work. In that work, the authors provided a resistor network based on voltage controlled MOSFET in weak inversion. The paper explained also how it is possible to derive a similar current mode evaluation based on the same approach. The Figure 3 illustrated the proposed current mode fuzzy inference based on parallel resistor network. The current Ij is splitted into two unequal components Ij ⋅ Aj and Ij ⋅ (1 − Aj), where Aj are the singletons reported in Table III. The total current Ij is therefore:

\[ I_j = \sum_{j=1}^{9} A_j \cdot I_j \]  \hspace{1cm} (6)

The equivalent conductance Gj of the two loads g1, and g2, is:

\[ G_j = \frac{1}{g_1} + \frac{1}{g_2} \]  \hspace{1cm} (7)

The constant "gm" represents the conductance of the differential regulated-cascode triode transconductors. The single current Ii flowing in Gi is given by:

\[ I_i = \frac{G_i}{\sum_{j=1}^{9} G_j} \]  \hspace{1cm} (8)

Finally, the formula of the total current Iy simplifies the de-fuzzification circuit as center of gravity method. This
The fuzzy control mechanism: the 9-rules zero-order Takagi-Sugeno analog controller architecture.

Fig. 4

property derived by the parallel arrangement of the nine rules and related conductance \( G_i \):

\[
I_i = \sum_{j=1}^{9} A_j \cdot G_j
\]

The mixed mode \((BS, CBI, \overline{BS})\) encoder has a VLSI circuit implementation as follows. The encoder employed the BS3X VLSI circuit composed by six ways. As mentioned, the fuzzy controller provided additional wires representing the encoder state: “encm” (encoder mode) and “bsi” (bus switch mode) according to the table showed in Figure 1. The BS sub-section received the input stream depending on “bsi:” original traffic or 1’s complement using inverters. The “encm” wire is used to drive the encoder output and the extra bus: BS or CBI. In particular, this wire drives the 3-line extra-bus: PAT[2:0] (BS) or INV[2:0] (CBI). In this way, we used the 3-line extra bus in both encoding modes. This last bus has been encoded using the bus invert method. The two latches (see Fig. 4) make the signals “encm” and “bsi” stable during the observation window. In fact, we supply a finite state machine (FSM) that counts 16 \((2^4)\) clock cycles according to the DAC precision and the length of the observation window. The encoder sends the information “encm” and “bsi” using a single line (slane) and a dual-edge flip-flop as depicted in Figure 4.

Figure 5 illustrates the single way circuit, when BS follows the maximum parallelism architecture. The “H” block represents the worst delay-consuming unit; it performs the Hamming distance calculation over 30-line bus. The module \( H \) is organized in two level of parallelism. The first level provides three fast adders (ADD) over 10-bit lines; the final carry save adder (CSA) calculates the total score. The sixth way (way 5) has a different architecture and a dual operation mode as illustrated in the Figure 6. The three adders performs the Hamming distance of the entire bus when the mode is BS \((enc = 1, bsi = \text{don’t care})\), otherwise the three ADD circuit performs the Hamming distance over the three 10-bit clusters \((enc = 0)\). This last operation mode participated to realize conveniently the CBI circuit.

The decoder architecture is rather simple and illustrated in the Figure 7. This circuit included the BS and CBI...
decoders, mostly a collection of multiplexers (BS, 22.3% of total cells) and inverters (CBI, 14.1% of total cells). The dual edge “sine” has been sampled both in the rising and fall clock transitions. We provided a clock gate controlled by a FSM identical to those included in the encoder. More precisely, the two FSMS have a common RESET, that ensure the exact synchronization.

4. EXPERIMENTAL RESULTS

In this section we illustrated the most important results, simulating the encoder/decoder system, translating the HDL net list at transistor level using Cadence Design Framework II. Additionally, the mixed-signal encoder/fuzzy controller has been simulated using Mentor AMS mixed-signal simulator. A Monte Carlo simulator (MC-SIM) generated the test bench input vectors used to simulate the encoders at constant 1 density. This tool performs stochastic simulations employing an user-defined statistical model. In particular, we defined the statistical model providing the “1” static probability variable from 0% to 100%. We initially analyzed the proposed approach at constant bus lines statistics, meaning the real word is a non stationary time variant random process. As mentioned, we considered \( L = 2 \) to reduce the fuzzy controller area and power dissipation. We demonstrated in this section \( L = 2 \) represents an ideal tradeoff between the activity savings using the fuzzy regulation method and the encoder hardware complexity. For this purposes, we optimized the constant \( L \) and the positions of the bus lines to be analyzed. Let be \( b_t \) the random process at two distinct values (0, 1) that represents the encoder input bus at the discrete time step \( n \). The process \( b_t \) introduced spatial and temporal correlation. The spatial correlation \( r(i,j) \) of two distinct bus lines \( b_{i,s} \) and \( b_{j,s} \) used the expectation operator “E”, that is the statistical average operator:

\[
r_{i,j} = E[b_{i,s} \cdot b_{j,s}] = P_b[b_{i,s} = 1, b_{j,s} = 1]
\]

The correlation represents the second order statistic and it is related to the 1 density as illustrated in the previous equation. Moreover, the mean value represents a first order statistics related to the 1 density. The first order statistic

\[
\mu = P_b[b_{i,s} = 1] = \frac{1}{L} \sum_{i=0}^{L-1} m_i
\]

neglects the effect of spatial correlation among the lines. We proposed a six-steps algorithm to select the best bus lines to be monitored:

1. Let \( M \) the mean value vector of the encoder input bus “\( b \)”, The \( j \)-th component is \( m_j \).
2. The \( N \)-line bus is divided in \( L \) identical clusters.
3. For each cluster, the algorithm calculates the average mean value \( \mu \). As example for cluster 0: \( \mu(0) = (L/N) \cdot \sum_{i=0}^{L-1} m_i \).
4. The optimal bus line to be monitored in the cluster has the mean value with minimal Euclidean distance from \( \mu \).
5. We defined “dispersion” the standard deviation of mean values \( m_i \) from \( \mu \): \( \rho_c = \frac{1}{N-L} \sum |m_i - \mu(0)|^2 \)
6. If the dispersion is unacceptable, the algorithm increments \( L \) and goes to point 2, otherwise exits.

The proposed algorithm used first order statistics that are easy to compute. Instead, the algorithm complexity can be improved considering the second order statistics related

### Table V. Activity performance of the analyzed coding schemes.

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>CBI</th>
<th>BS</th>
<th>Fuzzy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latex</td>
<td>13.66</td>
<td>21.60</td>
<td>-2.0</td>
</tr>
<tr>
<td>Spice</td>
<td>15.68</td>
<td>18.27</td>
<td>5.28</td>
</tr>
<tr>
<td>Gcc</td>
<td>14.05</td>
<td>16.27</td>
<td>-6.5</td>
</tr>
<tr>
<td>JPEG</td>
<td>14.49</td>
<td>9.72</td>
<td>9.68</td>
</tr>
<tr>
<td>MP3</td>
<td>14.64</td>
<td>11.46</td>
<td>15.16</td>
</tr>
<tr>
<td>AVI1</td>
<td>01.43</td>
<td>25.06</td>
<td>-71.5</td>
</tr>
<tr>
<td>AVI2</td>
<td>01.54</td>
<td>72.00</td>
<td>25.42</td>
</tr>
<tr>
<td>Average</td>
<td>10.78</td>
<td>04.34</td>
<td>-3.78</td>
</tr>
</tbody>
</table>

### Table VI. Bus interface energy dissipation.

<table>
<thead>
<tr>
<th></th>
<th>H-SPICE</th>
<th>NANO-SIM</th>
<th>Power comp.</th>
</tr>
</thead>
</table>

Visalli  
Fuzzy Control of Coding Schemes for Reducing Energy Dissipation in Off-Chip Buses

J. Low Power Electronics 4, 1–9, 2008
Fuzzy Control of Coding Schemes for Reducing Energy Dissipation in Off-Chip Buses

Table VII. Bus coding electrical specifications (\(V_{dd} = 3.3\) V, 90 nm CMOS).

<table>
<thead>
<tr>
<th>System</th>
<th>(S_{BHI})</th>
<th>(S_{BNA})</th>
<th>(N)</th>
<th>Energy</th>
<th>(C_{MIN,LOW})</th>
<th>(C_{MIN,HIGH})</th>
<th>(\Delta)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BI</td>
<td>0.0</td>
<td>0.11</td>
<td>31/30</td>
<td>1.58 pJ</td>
<td>2.55 pF</td>
<td>309.75 pF</td>
<td>308.19 pF</td>
</tr>
<tr>
<td>CPI</td>
<td>0.00076</td>
<td>0.15</td>
<td>33/30</td>
<td>1.41 pJ</td>
<td>1.56 pF</td>
<td>309.75 pF</td>
<td>308.19 pF</td>
</tr>
<tr>
<td>APBI</td>
<td>-0.000187</td>
<td>0.098</td>
<td>31/30</td>
<td>6.9 pJ</td>
<td>12.42 pF</td>
<td>309.75 pF</td>
<td>308.19 pF</td>
</tr>
<tr>
<td>BS3X</td>
<td>-8.56</td>
<td>0.441</td>
<td>34/30</td>
<td>13.5 pJ</td>
<td>4.96 pF</td>
<td>309.75 pF</td>
<td>308.19 pF</td>
</tr>
<tr>
<td>Fuzzy</td>
<td>0.18</td>
<td>0.45</td>
<td>34/30</td>
<td>18.28 pJ</td>
<td>6.58 pF</td>
<td>309.75 pF</td>
<td>308.19 pF</td>
</tr>
</tbody>
</table>

architectural description translated in 90 nm CMOS has 2688 \(\mu m^2\) area and a energy dissipation of just 2 pJ/msec.

5. CONCLUSION

The work analyzed the most known bus-coding scheme for low power, varying the traffic statistical property. This analysis showed how the cluster-based approaches permit better energy saving. We proposed a bus interface as a collection of the most important coding approach at high data rate: the Bus Switch mechanism and the clustered Bus Invert. The proposed approach has practical application in real-time systems and scenario where time-expensive or hardware-expensive online compression strategies compromise the data rate or the energy savings. In this work we discovered the BS' weak point, proposing a simple modification based on a 1's complement coding and decoding system. The final circuit is composed by three different coding schemes controlled by a 9-rules Takagi-Sugeno zero order analog fuzzy controller. This work analyzed the interface energy performance using a 9-rules fuzzy controller, but the approach can be extended easily using any kind of fuzzy system (analog or digital) with whichever used fuzzy inferences. The results of our simulation indicated the encoder can estimate the 1 density in \(L = 2\) bus lines only with satisfactory results in terms of statistical dispersion and activity performance. Statistical simulations indicated the bus interface avoids the performance losses both for \(BI\) and \(BS3X\), in particular the former out of the range \(45\% - 55\%\) and the latter more than \(60\%\). We estimated the proposed interface power dissipation in 90 nm CMOS technology, but technology scaling improves the proposed approach field of application, since the off-chip buses electrical specifications do not significantly change as the use of modern CMOS processes decreases area and power consumption. Power estimations and post layout evaluations concluded the proposed bus interface as best choice compared to the utilization of the single coding scheme.

References


Giuseppe Visalli

Giuseppe Visalli received the M.S. Degree in Telecommunication Engineering from University of Pisa, Italy in 1996. From 2001 to 2007 he has been a Research engineer in the Advanced System Technology division at ST Microelectronics. Currently, he is working in a Serial Flash Memory Group in Namonyx, a newly ST Microelectronics and Intel joint venture in the field of non-volatile memories. His current research interests lie in the area of low-power architectures and methodologies, wireless sensor networks, computer architecture and telecommunication. Dr. Visalli is author of seven US and two European granted patents and six US patent applications.