ENERGY-AWARE DESIGN FLOW FOR EMBEDDED SYSTEMS

1. Today there is an increasing demand for miniaturized smart sensors embedded in sensorial materials.
2. With increasing miniaturization and sensor-actuator density, decentralized network and data-processing architectures are preferred, but energy supply is still centralized. Advanced smart low-power design methodologies and applications are required.
3. A new design methodology focuses on 1. smart energy management at runtime and 2. application-specific System-On-Chip (SoC) design at design time contributing to low-power systems on both algorithmic and technological level.

Data processing systems are modelled using signal flow diagrams (see Figure 1.) [2].

This initial specification is used to derive 1. a multi-process programming model, and 2. a hardware model for a System-On-Chip design on Register-Transfer level.

The signal flow diagram is first transformed into a S/T Petri Net representation (Figure 2.). The Petri Net is used to derive the communication architecture, and 2. to determine an initial configuration for the communication network.

The Petri Net is mapped to sequential processes performing functional operations and channels providing the inter-process communication.

Finally, a RTL SoC design is synthesized [1] and the circuit activity is analyzed regarding different algorithms and complexity (Figure 4.).

ENERGY MANAGEMENT AT RUNTIME

1. Smart energy management is performed spatially at runtime by a behaviour-based or state-action-driven selection from a set of different (implemented) algorithms classified by their demand of computation power, and temporally by varying data-processing rates (based on previous activity analysis).

Definition 1. Constraints net relations satisfying quality of service and minimizing power consumption to be fulfilled at runtime. Some values are derived from circuit activity analysis.

VARS = {Runtime, Rate, Level, Energy, Power, Error}
Rate = {1,5,10,50,100} ...
Runtime = {LOW=1,MED=2,HIGH=3}, Error = {0,5,10,100}, Level = {LOW=1,HIGH=2,3,100}
Power = (Power*Runtime)/2
Energy = {Power*Runtime}/2

Figure 3. System simulation (from Figure 1.) with different runtime behaviours using a decision tree which can be retrieved by machine learning methods. Parameters: Data-processing rates=(1,5,10,20,100), Algorithmic level=1,5,10.2

Using advanced methods from the artificial intelligence area enables dynamic adaption of smart sensors and actuators at runtime.

Definition 1 shows a constraints net approach performing energy management at runtime.

Figure 3. shows simulation results of a system using a decision tree and machine learning approaches to optimize the runtime behaviour.

Figure 4. SoC cell activity correlates strongly with computation and signal/data flow. The first five results are computed only with the P controller; after obtaining the fifth result values are derived from circuit activity analysis on Register-Transfer level.

Cell Activity

Digit1

<table>
<thead>
<tr>
<th>Clock Cycles</th>
<th>Processing of a data set with simplified algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>&lt;0.0003</td>
</tr>
<tr>
<td>5</td>
<td>&lt;0.125</td>
</tr>
<tr>
<td>10</td>
<td>&lt;0.25</td>
</tr>
<tr>
<td>15</td>
<td>&gt;0.5</td>
</tr>
<tr>
<td>20</td>
<td>&gt;1.0</td>
</tr>
</tbody>
</table>

Quality

Time [Arb. Units]

3.

| Rate=1, Level=1, Error=10 |
| Rate=5, Level=1, Error=10 |
| Rate=10, Level=1, Error=10 |
| Rate=100, Level=1, Error=10 |

REFERENCES