AutoBench™ Version 1.1

Benchmark Name: Infinite Impulse Response (IIR) Filter

Benchmark Description

This Embedded Microprocessor Benchmark Consortium (EEMBC) benchmark algorithm simulates an embedded automotive/industrial application where the CPU performs an Infinite Impulse Response (IIR) filtering sample on 16-bit or 32-bit fixed-point values. It implements a Direct-Form II N-cascaded, second-order IIR filter. IIR filters can often be more efficient than FIR filters, in terms of attaining better magnitude response with a given filter order. This is because IIR filters incorporate feedback and are capable of realizing both poles and zeros of a system, whereas FIR filters are not capable of realizing the zeros. The difference equation for a Direct Form II N-Cascaded Direct second-order IIR filter is:

\[
\begin{align*}
    u(n) &= x(n) + a(1)x(n-1) + a(2)x(n-2), \\
    y(n) &= b(0)u(n) + b(1)u(n-1) + b(2)u(n-2);
\end{align*}
\]

where:

- \( x(n) \) = input signal of the biquad at time \( n \)
- \( u(n) \) = state variable of the biquad at time \( n \)
- \( y(n) \) = output signal of the biquad at time \( n \)
- \( a(n) \), \( b(n) \) = coefficients of the biquad

High- and low-pass IIR filters process the input signal data. Binary comparators also digitize the outputs of the filters. This IIR filter benchmark explores a CPU’s ability to perform multiply-accumulates and rounding. It employs typical DSP functions that would replace an analog signal chain comprised of op-amps and comparators.

Optimization Rules

<table>
<thead>
<tr>
<th>Category</th>
<th>Allowed</th>
<th>Disallowed</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANSI C</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Intrinsics/Language</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Extensions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom Libraries</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Assembly Language</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>HW Accelerators</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

Algorithm Flowchart (page 2)
Algorithm Flowchart

START

Initialize and Send Data

Get Input Signal

Compute Low Pass Filter

Compute IR High Pass Filter

Detect IP Signals

Detect LP Signals

INC

Clamp, Report Results

Stop

No

Yes