Automatic Acceleration of Computationally Intensive Applications onto an FPGA

Wong Po Ping
13038118D
Supervisor: Dr. Benjamin Carrion Schafer
• Introduction

• Interface on PC and FPGA

• Evaluation

Contents
Introduction
1. Extract the computationally intensive function
2. Convert the C function into VHDL
3. Configure the FPGA
4. Execute the program with FPGA
5. Conduct evaluation

Introduction
Hardware

- PC
  - CPU @ 3.0 GHz

- FPGA
  - Xilinx ML402 Virtex-4 SX Evaluation Platform
  - XC4VSX35
- Software Profiler
- gprof

Each sample counts as 0.01 seconds.

<table>
<thead>
<tr>
<th>% time</th>
<th>cumulative</th>
<th>self</th>
<th>calls</th>
<th>self</th>
<th>total</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>66.67</td>
<td>0.02</td>
<td>0.02</td>
<td>261632</td>
<td>76.44</td>
<td>76.44</td>
<td>soebel</td>
</tr>
<tr>
<td>33.33</td>
<td>0.03</td>
<td>0.01</td>
<td></td>
<td></td>
<td></td>
<td>main</td>
</tr>
<tr>
<td>0.00</td>
<td>0.03</td>
<td>0.00</td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>LoadBitmapFile</td>
</tr>
</tbody>
</table>

Function Extraction
• Modify C into HW-C

• High Level synthesis tool
  • CyberWorkBench

Conversion (C->VHDL)
Configuration of FPGA

- Pins Assignment
  - Clock (AE14), reset (D6), Rx (W2), Tx (W1)
  - Led indicators (G5, G6, A11, A12)

- Clock frequency = 100 MHz
• 115200 Baud rate
• 8 Date bits
• 1 Stop bit
• No Parity bit
ready flag of Tx => reset
enable flag of Tx => reset

idle

waiting for the enable signal from Rx and the received bit => input row

transmitting

When Tx set a flag processing result => transmission

processing

the flag of processing would be set and this signal triggered the function

receiving

Interface on FPGA
C API

- `serial_open` – Configure the serial communication
- `serial_send` - Send 1 byte
- `serial_receive` - Receive 1 byte
- `serial_close` – Close the COM port
• 3 8-bit inputs; 1 8-bit output
• Data bits = (512 x 512) x 8 x 4 = 8388608 bits
• Each frame: 8 data bits, 1 start bit, 1 stop bit
• Bandwidth
  = 80% of baud rate = 115200 x 0.8 = 92160 bps
• Transmission time = 8388608 / 92160 = 91s

Theoretical estimation
• SW execution time = 0.0469s
• SW and HW execution time = 197.5s
• Speedup = 0.000237

=> Slower than the SW only program

Evaluation
Critical Problem

• Some parts of the function remain sequential

• UART is too slow
Alternatives

- USB
- PCI
- PCI-X

Alternatives
• UART – 0.011 MB/s
• USB 2.0 – 60 MB/s
• PCI 2.0 – 133 MB/s
• PCI-X 2.0 – 4300 MB/s

Theoretical Bandwidth

Source form: http://www.pixelbeat.org/speeds.html