Static Checking of Interrupt-driven Software

Jens Palsberg
Purdue University
CERIAS and Department of Computer Science
www.cs.purdue.edu/people/palsberg

Joint work with Dennis Brylow and Niels Damgaard.
Supported by an NSF CAREER award.
Fan control signal

Internal Timer

Microcontroller

Power Pulse

Network
Interrupt-driven Control

Interrupt Sources

Interrupt Vectors

Handler 1

Handler 2

Handler 3

Handler 4
Example Program in Z86 Assembly Language

; Constant Pool (Symbol Table).
; Bit Flags for IMR and IRQ.
IRQ0 .EQU  #00000001b
; Bit Flags for external devices
; on Port 0 and Port 3.
DEV2 .EQU  #00010000b

; Interrupt Vectors.
   .ORG  %00h
   .WORD  #HANDLER  ; Device 0

; Main Program Code.
   .ORG  0Ch
   INIT:  ; Initialization section.
   OC   LD  SPL, #0F0h ; Initialize Stack Pointer.
   OF   LD  RP,  #10h ; Work in register bank 1.
   12   LD  P2M, #00h ; Set Port 2 lines to
                   ; all outputs.
   15   LD  IRQ, #00h ; Clear IRQ.
   18   LD  IMR, #IRQ0
   1B   EI       ; Enable Interrupt 0.
Example Program in Z86 Assembly Language

START: ; Start of main program loop.
     1C   DJNZ r2, START ; If our counter expires,
     1E   LD r1, P3 ; send this sensor’s reading
     20   CALL SEND ; to the output device.
     23   JP START

SEND: ; Send Data to Device 2.
     26   PUSH IMR ; Remember what IMR was.
     28   DI ; Musn’t be interrupted
          ; during pulse.
     29   LD P0, #DEV2 ; Select control line
          ; for Device 2.
     2C   DJNZ r3, DELAY ; Short delay.
     2E   CLR P0
     30   POP IMR ; Reactivate interrupts.
     32   RET

HANDLER: ; Interrupt for Device 0.
     33   LD r2, #00h ; Reset counter in main loop.
     35   CALL SEND
     38   IRET ; Interrupt Handler is done.
.END
Program

Model Extraction

Model Checking

Model

Properties
Stack-Size Analysis

<table>
<thead>
<tr>
<th>Program</th>
<th>Lower</th>
<th>Upper</th>
<th>Time</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTurk</td>
<td>17</td>
<td>18</td>
<td>4.11 s</td>
<td>31.6 MB</td>
</tr>
<tr>
<td>GTurk</td>
<td>16</td>
<td>17</td>
<td>4.31 s</td>
<td>32.2 MB</td>
</tr>
<tr>
<td>ZTurk</td>
<td>16</td>
<td>17</td>
<td>4.22 s</td>
<td>32.1 MB</td>
</tr>
<tr>
<td>DRop</td>
<td>12</td>
<td>14</td>
<td>4.14 s</td>
<td>31.1 MB</td>
</tr>
<tr>
<td>Rop</td>
<td>12</td>
<td>14</td>
<td>4.18 s</td>
<td>31.8 MB</td>
</tr>
<tr>
<td>Fan</td>
<td>11</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Serial</td>
<td>10</td>
<td>10</td>
<td>3.87 s</td>
<td>31.0 MB</td>
</tr>
<tr>
<td>Example</td>
<td>37</td>
<td>37</td>
<td>3.21 s</td>
<td>34.9 MB</td>
</tr>
</tbody>
</table>

The lower bounds were found with a software simulator for Z86 assembly language that we wrote.
## Interrupt Latency Analysis of the Highest Priority IRQ

<table>
<thead>
<tr>
<th>Program</th>
<th>Green</th>
<th>Yellow</th>
<th>Red</th>
<th>Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTurk</td>
<td>51%</td>
<td>49%</td>
<td>0%</td>
<td>260</td>
</tr>
<tr>
<td>GTurk</td>
<td>50%</td>
<td>50%</td>
<td>0%</td>
<td>272</td>
</tr>
<tr>
<td>ZTurk</td>
<td>50%</td>
<td>50%</td>
<td>0%</td>
<td>276</td>
</tr>
<tr>
<td>DRop</td>
<td>19%</td>
<td>81%</td>
<td>0%</td>
<td>312</td>
</tr>
<tr>
<td>Rop</td>
<td>19%</td>
<td>81%</td>
<td>0%</td>
<td>312</td>
</tr>
<tr>
<td>Fan</td>
<td>67%</td>
<td>33%</td>
<td>0%</td>
<td>310</td>
</tr>
<tr>
<td>Serial</td>
<td>79%</td>
<td>21%</td>
<td>0%</td>
<td>326</td>
</tr>
<tr>
<td>Example</td>
<td>46%</td>
<td>54%</td>
<td>0%</td>
<td>242</td>
</tr>
</tbody>
</table>

Latencies are given in machine cycles.

One machine cycle is executed in 1 microsecond.
Secure Software Systems Group

Department of Computer Science

http://www.cs.purdue.edu/s3/

Faculty: Antony Hosking, Jens Palsberg, Jan Vitek.

16 Ph.D. students, 2 M.S. students, 3 undergraduate students.

Sample projects: Java security, bytecode compression, interoperability of software systems, real-time system verification, software watermarking, high-performance persistent object storage.

Funding: NSF, DARPA, CERIAS, Sun Microsystems, Lockheed Martin, IBM, Motorola, and Intel.