Automatic Device Driver Synthesis with Termite

Leonid Ryzhyk, Peter Chubb, Ihor Kuz, Etienne Le Sueur, Gernot Heiser
UNSW, NICTA, Open Kernel Labs (Australia),

John Keys, Intel
Conventional driver development

OS interface spec

device spec
Conventional driver development

OS interface spec

device spec
Conventional driver development

OS interface spec

requests

driver.c

device commands

device spec
Driver synthesis: high-level view

Formal OS interface spec

requests

driver.c

device commands

Formal device spec
Driver synthesis: high-level view

- Formal OS interface spec
  
  - requests
  
  - driver.c
    
    - device commands
      
      - Formal device spec
Driver synthesis: high-level view

Formal OS interface spec

requests

driver.c

device commands

Formal device spec

Advantages

• Separation of concerns
  – Know one thing well

• Reuse
  – Specify once, synthesise many
Driver synthesis by example

Formal OS interface spec

Formal device spec

GPIO controller

ctrl data

GPIO line
Driver synthesis by example

Formal
OS interface spec

Formal
device spec

GPIO controller registers

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

GPIO controller

GPIO line
**Driver synthesis by example view**

**Formal OS interface spec**

**GPIO controller registers**

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

**Formal device spec**

- \(\text{!ctrlWrite}(1)\)
- \(\text{!ctrlWrite}(0)\)
- \(\text{!dataWrite/changeLevel}\)
Driver synthesis by example view

Formal OS interface spec

GPIO controller registers

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

Formal device spec

- `!ctrlWrite(1)`
- `!ctrlWrite(0)`
- `!dataWrite/ changeLevel`
Driver synthesis by example view

**Formal OS interface spec**

- `!ctrlWrite(1)`: off
- `!ctrlWrite(0)`: on
- `!dataWrite/changeLevel`

**GPIO controller registers**

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

**Formal device spec**

- Go from off to on: `!ctrlWrite(1)`
- Go from on to off: `!ctrlWrite(0)`
- `!dataWrite/changeLevel`
Driver synthesis by example view

Formal OS interface spec

GPIO controller registers

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

Formal device spec

off
!ctrlWrite(1)
!ctrlWrite(0)
on!
dataWrite/changeLevel

GPIO line
Driver synthesis by example

Formal OS interface spec

- ?set
- changeLevel
- !setComplete

Formal device spec

- off
- on
- !ctrlWrite(1)
- !ctrlWrite(0)
- !dataWrite/changeLevel

GPIO controller registers

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

GPIO controller

GPIO line
Driver synthesis by example

**Formal OS interface spec**

- ?set
- changeLevel
- !setComplete

**GPIO controller registers**

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

**Formal device spec**

- off
- on
- !ctrlWrite(1)
- !ctrlWrite(0)
- !dataWrite/changeLevel

**GPIO controller**

- ctrl/data

**GPIO line**
Driver synthesis by example

**Formal OS interface spec**

```
?set  changeLevel  !setComplete
```

**GPIO controller registers**

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

**Formal device spec**

```
!ctrlWrite(1)  !dataWrite/ changeLevel
!ctrlWrite(0)  off  on
```
Driver synthesis by example

Formal OS interface spec

?set -> changeLevel
!setComplete

GPIO controller registers

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

Formal device spec

GPIO controller

GPIO line

!ctrlWrite(1) -> !ctrlWrite(0)

!ctrlWrite/changeLevel

!dataWrite/changeLevel

0=off
1=on
Driver synthesis by example

**Formal OS interface spec**

- ?set
- changeLevel
- !setComplete

**GPIO controller registers**

<table>
<thead>
<tr>
<th>ctrl</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=off</td>
<td>0=low</td>
</tr>
<tr>
<td>1=on</td>
<td>1=high</td>
</tr>
</tbody>
</table>

**Formal device spec**

- !ctrlWrite(1)
- !ctrlWrite(0)
- !dataWrite/
- changeLevel

**GPIO controller**

- ctrl
- data

**GPIO line**
Driver synthesis by example

Formal OS interface spec

Formal device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec || Device spec
Driver synthesis by example

Formal OS interface spec

![chart](chart1.png)

Formal device spec

![chart](chart2.png)

OS spec || Device spec

![chart](chart3.png)
Driver synthesis by example

**Formal OS interface spec**

- `?set`
- `changeLevel`
- `!setComplete`

**Formal device spec**

- `off`
- `on`
- `!ctrlWrite(1)`
- `!ctrlWrite(0)`
- `!dataWrite/changeLevel`

**OS spec || Device spec**

- `?set`
- `!ctrlWrite(1)`
Driver synthesis by example

Formal OS interface spec

?set \rightarrow \text{changeLevel} \rightarrow !\text{setComplete}

Formal device spec

on \quad \text{off}

!\text{ctrlWrite}(1) \rightarrow !\text{ctrlWrite}(0) \rightarrow \text{!dataWrite/ changeLevel}

OS spec \ || \ Device spec

!\text{ctrlWrite}(1) \rightarrow ?\text{set} \rightarrow !\text{ctrlWrite}(1)

Driver synthesis by example

Formal OS interface spec

- \(?\text{set}\)
- \(\text{changeLevel}\)
- \(!\text{setComplete}\)

Formal device spec

- \(\text{off}\)
- \(!\text{ctrlWrite}(1)\)
- \(!\text{ctrlWrite}(0)\)
- \(\text{on}\)
- \(!\text{dataWrite}/\text{changeLevel}\)

OS spec \(\parallel\) Device spec
Driver synthesis by example

Formal OS interface spec

- `?set`
- `changeLevel`
- `!setComplete`

Formal device spec

- `off`
- `on`
- `!ctrlWrite(1)`
- `!ctrlWrite(0)`
- `!dataWrite/changeLevel`

OS spec || Device spec

- `?set`
- `!ctrlWrite(1)`
- `!ctrlWrite(1)`
- `!ctrlWrite(0)`
Driver synthesis by example

Formal OS interface spec
- ?set changeLevel
- !setComplete

Formal device spec
- off
- !ctrlWrite(1)
- !ctrlWrite(0)
- on
- !ctrlWrite(1)

OS spec || Device spec
- ?set
- !ctrlWrite(1)
- !ctrlWrite(1)
- !ctrlWrite(0)
- !dataWrite/changeLevel
- !dataWrite/changeLevel
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec || Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec \| Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec ⊕ Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec \parallel Device spec
Driver synthesis by example

Formal OS interface spec
- ?set
- changeLevel
- !setComplete

Formal device spec
- off
- !ctrlWrite(1)
- !ctrlWrite(0)
- on
- !dataWrite/changeLevel

OS spec || Device spec
- ?set
- !ctrlWrite(1)
- !dataWrite/changeLevel
- !setComplete
**Driver synthesis by example**

**Formal OS interface spec**

- `?set` to `changeLevel`
- `!setComplete`

**Formal device spec**

- `off` to `!ctrlWrite(1)`
- `on` to `!ctrlWrite(0)`
- `!dataWrite/changeLevel`

**OS spec || Device spec**

- `?set` to `!ctrlWrite(1)`
- `!dataWrite/changeLevel`
- `!setComplete`
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec \parallel Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec \parallel Device spec
Driver synthesis by example

Formal OS interface spec

OS spec \parallel Device spec

Formal device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec  Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec || Device spec
Driver synthesis by example

Formal OS interface spec

Formal device spec

OS spec $\parallel$ Device spec
Modelling real device interfaces

SD host controller device

REGISTERS
ARG
CMD
STAT
RESP
ISR
DIV
RST
DISR
BDST
BDRX
BDTX

CMD_MASTER
CLK DIVIDER
DATA_MASTER
Modelling real device interfaces

SD host controller device

REGISTERS
ARG
CMD
STAT
RESP
ISR

CDM_MASTER

CLK DIVIDER

DATA_MASTER

Multiple functional units
Modelling real device interfaces

SD host controller device

REGISTERS
- ARG
- CMD
- STAT
- RESP
- ISR
- DIV
- RST
- DISR
- BDST
- BDRX
- BDTX

Multiple functional units

REGISTERS
|||
CLOCK_DIVIDER
|||
(COMMAND_MASTER | [class.off] | DATA_MASTER)
Modelling real device interfaces

SD host controller device

REGISTERS

ARG
CMD
STAT
RESP
ISR

DIV
RST

DISR
BDST
BDRX
BDTX

CMD_MASTER
CLK_DIVIDER
DATA_MASTER

Multiple functional units

REGISTERS

|||
CLOCK_DIVIDER
|||
(COMMAND_MASTER |[class.off]| DATA_MASTER)
Modelling real device interfaces

```c
comand_reg m_command_reg;
...
write_reset_reg
/m_command_reg = 0
```

Multiple functional units

```
REGISTERS
|||
CLOCK_DIVIDER
|||
(COMMAND_MASTER │[class.off]│ DATA_MASTER)
```
The synthesis algorithm

• The state explosion problem
  - **Problem:** The product state space can be huge
  - **Solution:** Explore the product state space incrementally

• Dealing with data
  - **Problem:** Enumerating all variable assignments is infeasible
  - **Solution:** Manipulate data symbolically
Results

• Successfully synthesised drivers for real devices:
  - Asix AX88772 USB-to-Ethernet adapter
    • Linux
  - Ricoh R5C822 SD host controller
    • Linux
    • FreeBSD
<table>
<thead>
<tr>
<th></th>
<th>USB-to-Ethernet</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS interface spec</td>
<td>309 loc</td>
<td>641 loc</td>
</tr>
<tr>
<td>Device spec</td>
<td>463 loc</td>
<td>653 loc</td>
</tr>
<tr>
<td>Synthesised driver</td>
<td>2620 loc</td>
<td>4667 loc</td>
</tr>
<tr>
<td>Linux driver</td>
<td>1200 loc</td>
<td>1174 loc</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th></th>
<th>USB-to-Ethernet</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS interface spec</td>
<td>309 loc</td>
<td>641 loc</td>
</tr>
<tr>
<td>Device spec</td>
<td><strong>463 loc</strong></td>
<td><strong>653 loc</strong></td>
</tr>
<tr>
<td>Synthesised driver</td>
<td>2620 loc</td>
<td>4667 loc</td>
</tr>
<tr>
<td>Linux driver</td>
<td><strong>1200 loc</strong></td>
<td><strong>1174 loc</strong></td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th></th>
<th>USB-to-Ethernet</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS interface spec</td>
<td>309 loc</td>
<td>641 loc</td>
</tr>
<tr>
<td>Device spec</td>
<td>463 loc</td>
<td>653 loc</td>
</tr>
<tr>
<td>Synthesised driver</td>
<td>2620 loc</td>
<td>4667 loc</td>
</tr>
<tr>
<td>Linux driver</td>
<td>1200 loc</td>
<td>1174 loc</td>
</tr>
</tbody>
</table>
The Termite Debugger
Performance of the AX88772 USB-to-Ethernet adapter driver

CPU Utilisation (%)

Packet size (bytes)

Throughput (Mbit/s)

Packet size (bytes)
Limitations

Limitations of Termite

- Cannot specify constraints on data in memory
  - alignment, fragmentation, etc
- Complex relations among variables are not supported
- Restrictions on the structure of specifications
- Termite drivers require runtime support
Future work

- Formal OS interface spec
- driver.c
- Formal device spec
Future work

- Formal OS interface spec
- driver.c
- Formal device spec
- HDL
Future work

HDL → driver.c → Formal device spec → Formal OS interface spec
Conclusions

- Driver synthesis is possible
  - Device experts provide device specs
  - OS experts provide OS specs
  - Termite does the rest

- Still work-in-progress
  - Addressing current limitations
  - Driver synthesis from HDL
Conclusions

• Driver synthesis is possible
  - Device experts provide device specs
  - OS experts provide OS specs
  - **Termite does the rest**

• Still work-in-progress
  - Addressing current limitations
  - Driver synthesis from HDL