STREAM MONITORING

Output Streams

VERDICT
VERDICT

FPGA RUNTIME MONITOR

VHDL
Spec

DLR
STREAM MONITORING

FPGA RUNTIME MONITOR

- Easy to Integrate
- Low Resource Consumption
- Formal Guarantees

Output Streams

VERDICT

VERDICT

Spec

VHDL
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
Formal Guarantees
Specification Logics
SPECIFICATION LANGUAGE

Formal Guarantees
Specification Logics

Expressiveness
Programming Languages
The GPS module operates with at least 5Hz.
input lat, lon, velo: Float64

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)

trigger gps_freq < 5 “GPS frequency less than 5 Hz.”
input lat, lon, velo: Float64

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)

trigger gps_freq < 5 “GPS frequency less than 5 Hz.”

Measured and computed velocities coincide.
input lat, lon, velo: Float64

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)
trigger gps_freq < 5 “GPS frequency less than 5 Hz.”

output gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)
output gps_velo := ∇(gps_dist)
trigger abs(gps_velo - velo) > 0.1 “Conflicting measurements for velocity.”
Deceleration is preceded by a slow-down command.
input lat, lon, velo: Float64
input slow_down_cmd: Bool

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)
trigger gps_freq < 5 “GPS frequency less than 5 Hz.”

output gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)
output gps_velo := ∇(gps_dist)
trigger abs(gps_velo - velo) > 0.1 “Conflicting measurements for velocity.”

output fast := velo > 700
output slow_down := fast.offset(by: -1).defaults(to: false) ∧ ¬fast
trigger @1Hz ¬slow_down_cmd.aggregate(over: 5s, using: ∃)
∧ slow_down.hold().defaults(to: false) “Spurious Slow-Down.”
CHALLENGES

Periodic versus Event-Based

Utilize Parallel Execution

Reduce Circuit Cost
input lat, lon, velo: Float64
input slow_down_cmd: Bool

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)

trigger gps_freq < 5 “GPS frequency less than 5 Hz.”

output gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)
output gps_velo := ∇(gps_dist)

trigger abs(gps_velo - velo) > 0.1 “Conflicting measurements for velocity.”

...
**PERIODIC v. EVENT-BASED**

input lat, lon, velo: Float64
input slow_down_cmd: Bool

output gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)

trigger gps_freq < 5 “GPS frequency less than 5 Hz.”

output gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)
output gps_velo := ∇(gps_dist)

trigger abs(gps_velo - velo) > 0.1 “Conflicting measurements for velocity.”

...
**input** lat, lon, velo: Float64  
**input** slow_down_cmd: Bool

**output** gps_freq @1Hz := lat.aggregate(over_exactly: 1s, using: count)  
**trigger** gps_freq < 5 “GPS frequency less than 5 Hz.”

**output** gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)  
**output** gps_velo := ∇(gps_dist)  
**trigger** abs(gps_velo - velo) > 0.1 “Conflicting measurements for velocity.”
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
High-Level Controller

Periodic
- Sys Clk
- Scheduler

Event-Based
- Ext I/F
- Event Prep

HLQ I/F

event data affected streams timestamp
**High-Level Controller**

**Periodic**
- Sys Clk
- Scheduler

**Event-Based**
- Ext I/F
- Event Prep

**HLQ I/F**

- Event data
- Affected streams
- Timestamp

\[
in_1 \rightarrow 3.4 \\
in_2 \rightarrow \top \\
in_4 \rightarrow 9 \\
\text{time} \rightarrow 4:15\text{pm}
\]
**High-Level Controller**

**Periodic**
- Sys Clk
- Scheduler

**Event-Based**
- Ext I/F
- Event Prep

**HLQ I/F**

- Event data
- Affected streams
- Timestamp

- $\text{in}_1 \rightarrow 3.4$
- $\text{in}_2 \rightarrow \top$
- $\text{in}_4 \rightarrow 9$
- Time $\rightarrow 4:15$pm

$3.4, \top, #, 9;$

$\text{in}_1, \text{in}_2, \text{in}_4, \text{out}_2, \ldots$

$4:15$pm
**High-Level Controller**

**Periodic**
- Sys Clk
- Scheduler

**Event-Based**
- Ext I/F
- Event Prep

**HLQ I/F**

**Event data**
- affected streams
- timestamp

- in$_1$ → 3.4
- in$_2$ → ⊤
- in$_4$ → 9
- time → 4:15pm

3.4, ⊤, #, 9;

3.4, ⊤, #, 9;

110101...
**High-Level Controller**

**Periodic**
- Sys Clk
- Scheduler

**Event-Based**
- Ext I/F
- Event Prep

**HLQ I/F**

- event data
- affected streams
- timestamp
**High-Level Controller**

- **Periodic**
  - Sys Clk
  - Scheduler

- **Event-Based**
  - Ext I/F
  - Event Prep

**Outputs:**
- Event data
- Affected streams
- Timestamp
High-Level Controller

4:21pm

\( \text{out}_1, \text{out}_3, \text{out}_4, \ldots \)

Periodic

- Sys Clk
- Scheduler

Event-Based

- Ext I/F
- Event Prep

HLQ I/F

event data, affected streams, timestamp
**High-Level Controller**

![Diagram of High-Level Controller]

- **Periodic**
  - Sys Clk
  - Scheduler

- **Event-Based**
  - Ext I/F
  - Event Prep

---

4:21pm

out₁, out₃, out₄, ...

0...0;
00001011...
4:21pm

---

**Event data**  **affected streams**  **timestamp**
Event data
affected streams
timestamp
High-Level Controller

**PERIODIC**
- Sys Clk
- Scheduler

**EVENT-BASED**
- Ext I/F
- Event Prep

**HLQ I/F**

*event data* | *affected streams* | *timestamp*
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
Low-Level Controller

Queue

empty \rightarrow \text{Idle}

\neg \text{empty} \rightarrow \text{Pop}

\neg \text{empty} \rightarrow \text{Eval}

\text{Eval} \rightarrow \neg \text{empty}

\text{Eval} \rightarrow \text{done} \land \neg \text{done}

\text{Eval} \rightarrow \neg \text{empty}

\neg \text{empty} \rightarrow \text{done} \land \neg \text{done}

\text{Eval} \rightarrow \top

\text{Eval} \rightarrow \text{enable}

\text{enable} \rightarrow \text{Prep}

\text{Prep} \rightarrow \text{Layer}_n

\text{Layer}_0 \rightarrow \text{Layer}_n

\text{Layer}_n \rightarrow \text{enable}

\text{enable} \rightarrow \text{Idle}

\text{Streams} \rightarrow \text{Windows}

LLC

Eval + Memory
**SPECIFICATION**

**input** lat, lon, velo: Float64

**input** slow_down_cmd: Bool

**output** gps_freq @1Hz :=

lat.aggregate(over_exactly: 1s, using: count)

**trigger** gps_freq < 5

“GPS frequency less than 5 Hz.”

**output** gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)

**output** gps_velo := ∇(gps_dist)

**trigger** abs(gps_velo - velo) > 0.1

“Conflicting measurements for velocity.”

**output** fast := velo > 700

**output** slow_down :=

fast.offset(by: -1).defaults(to: false) ∧ ¬fast

**trigger** @1Hz ¬slow_down_cmd.aggregate(over: 5s, using: ∃)

∧ slow_down.hold().defaults(to: false) “Spurious Slow-Down.”

**DEPENDENCY GRAPH**

```
lat ---> gps_freq

lon ---> gps_dist

velo ---> gps_velo

cmd

fast

slow_down

trig1

trig2

trig3
```
input lat, lon, velo: Float64
input slow_down_cmd: Bool
output gps_freq @1Hz :=
  lat.aggregate(over_exactly: 1s, using: count)
trigger gps_freq < 5
  “GPS frequency less than 5 Hz.”
output gps_dist := sqrt(δ(lon)^2 + δ(lat)^2)
output gps_velo := ∇(gps_dist)
trigger abs(gps_velo - velo) > 0.1
  “Conflicting measurements for velocity.”
output fast := velo > 700
output slow_down :=
  fast.offset(by: -1).defaults(to: false) ∧ ¬fast
trigger @1Hz ¬slow_down_cmd.aggregate(over: 5s, using: ∃)
  ∧ slow_down.hold().defaults(to: false) “Spurious Slow-Down.”
Low-Level Controller

EvalController

Idle
Prep
Layer_n
Layer_0

Dependency Graph

lat
lon
velo
cmd
gps_freq
gps_dist
gps_velo
fast
slow_dn
trig_1
trig_2
trig_3
**Low-Level Controller**

- **Idle** → **Prep**
- **Layerₙ** → **Layer₀**

**Dependency Graph**

- **Layer₀**
  - **lat** → **lon** → **velo** → **cmd**
  - **gps_freq** → **gps_dist** → **fast**

- **Layer₁**
  - **trig₁** → **gps_velo** → **slow_dn**

- **Layer₂**
  - **trig₂** → **trig₃**
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
SLIDING WINDOWS

\[
\text{output } h @ p^{-1}\text{Hz} := \text{s.aggregate(over: } \eta\text{sec, using: } \gamma)\]

- Li et al.: “No Pane, No Gain: Efficient Evaluation of Sliding-window Aggregates over Data Streams”, SIGMOD Rec. 2005
- Meertens: “Algorithmics: Towards programming as a mathematical activity”, 1986
Honorable Mention: Sliding Windows

output $h \oplus p^{-1}\text{Hz} := s.\text{aggregate(over: } \eta \text{sec, using: } \gamma)$

After $p$ seconds:
$r := \text{fin}(v_1 \circ \gamma \ldots \circ \gamma v_8)$

new value $\alpha$ from stream $s$

result $r$

<table>
<thead>
<tr>
<th>$v_1$</th>
<th>$v_2$</th>
<th>$v_3$</th>
<th>$v_4$</th>
<th>$v_5$</th>
<th>$v_6$</th>
<th>$v_7$</th>
<th>$v_8$</th>
</tr>
</thead>
</table>

- Li et al.: “No Pane, No Gain: Efficient Evaluation of Sliding-window Aggregates over Data Streams”, SIGMOD Rec. 2005
- Meertens: “Algorithmics: Towards programming as a mathematical activity”, 1986
1. Never injure humans.
2. Obey orders.
3. Protect yourself.
Never injure humans.

Obey orders.

Protect yourself.

From RTLola to VHDL

1. System
2. Monitor
3. Health

SPEC

1. Never injure humans.
2. Obey orders.
3. Protect yourself.
## Evaluation

<table>
<thead>
<tr>
<th></th>
<th>FF</th>
<th>LUT</th>
<th>MUX</th>
<th>CA</th>
<th>MULT</th>
<th>Pwr [W]</th>
<th>Time [μs]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Drone</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mon</td>
<td>3036</td>
<td>3685</td>
<td>26</td>
<td>656</td>
<td>10</td>
<td>1.620</td>
<td>4.28</td>
</tr>
<tr>
<td>HLC</td>
<td>901</td>
<td>156</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>1.620</td>
<td>4.28</td>
</tr>
<tr>
<td>Q</td>
<td>543</td>
<td>442</td>
<td>0</td>
<td>43</td>
<td>0</td>
<td>1.620</td>
<td>4.28</td>
</tr>
<tr>
<td>LLC</td>
<td>1281</td>
<td>2820</td>
<td>0</td>
<td>576</td>
<td>10</td>
<td>1.620</td>
<td>4.28</td>
</tr>
<tr>
<td><strong>Network</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mon</td>
<td>1905</td>
<td>1533</td>
<td>23</td>
<td>226</td>
<td>23</td>
<td>1.570</td>
<td>3.20</td>
</tr>
<tr>
<td>HLC</td>
<td>550</td>
<td>161</td>
<td>0</td>
<td>37</td>
<td>0</td>
<td>1.570</td>
<td>3.20</td>
</tr>
<tr>
<td>Q</td>
<td>330</td>
<td>342</td>
<td>0</td>
<td>28</td>
<td>0</td>
<td>1.570</td>
<td>3.20</td>
</tr>
<tr>
<td>LLC</td>
<td>895</td>
<td>927</td>
<td>0</td>
<td>161</td>
<td>0</td>
<td>1.570</td>
<td>3.20</td>
</tr>
<tr>
<td><strong>Cmd Resp</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mon</td>
<td>6379</td>
<td>13794</td>
<td>0</td>
<td>849</td>
<td>0</td>
<td>1.582</td>
<td>3.77</td>
</tr>
<tr>
<td>HLC</td>
<td>936</td>
<td>232</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>1.582</td>
<td>3.77</td>
</tr>
<tr>
<td>Q</td>
<td>540</td>
<td>326</td>
<td>0</td>
<td>28</td>
<td>0</td>
<td>1.582</td>
<td>3.77</td>
</tr>
<tr>
<td>LLC</td>
<td>4903</td>
<td>13236</td>
<td>0</td>
<td>971</td>
<td>0</td>
<td>1.582</td>
<td>3.77</td>
</tr>
<tr>
<td><strong>Cmd Resp</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mon</td>
<td>6909</td>
<td>14768</td>
<td>0</td>
<td>851</td>
<td>0</td>
<td>1.581</td>
<td>43.83</td>
</tr>
<tr>
<td>HLC</td>
<td>936</td>
<td>232</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>1.581</td>
<td>43.83</td>
</tr>
<tr>
<td>Q</td>
<td>534</td>
<td>326</td>
<td>0</td>
<td>28</td>
<td>0</td>
<td>1.581</td>
<td>43.83</td>
</tr>
<tr>
<td>LLC</td>
<td>5433</td>
<td>14210</td>
<td>0</td>
<td>973</td>
<td>0</td>
<td>1.581</td>
<td>43.83</td>
</tr>
</tbody>
</table>
CONCLUSION

HLC

Queue

LLC

SYSTEM

MONITOR

HEALTH

Periodic versus Event-Based

Utilize Parallel Execution

Reduce Circuit Cost
OUTLOOK

LEARN MORE:
stream-lab.eu