Timed and Resource-oriented Statecharts for Embedded Software

Jin Hyun Kim
jhkim@formal.korea.ac.kr
Formal Methods Lab. Korea University
Contents

• Introduction

• TRoS (Timed and Resource-oriented Statecharts)

• Case Study

• Conclusions
Introduction

• Model-based analysis
  – Specification (Modeling) with modeling (formal) languages
  – Verification with mathematical methods
Model-based Analysis

- External Input
- Functional Behavior
- Timed Behavior
- Implementation

Context-Diagram
- Functional Reqs.
- Time & Res. Reqs.

External Output
Top-Down
Analysis

• The correctness of real-time embedded software
  – Functionality
  – Timed behavior (Concurrency related)
Correctness

• Functionality
Correctness

- Timed behavior

$I_t \rightarrow Schedulability \rightarrow O_{t+n}$
Correctness

- Timed behaviors constrained by

Time + Resource

- Execution Time

- Sharing Relation
  - Priorities
Our Approach

External Input

Context-Diagram

External Output

Top-Down

Functional Behavior

Statecharts

Functional Reqs.

Timed Actions

TRoS

Timed Behavior

Implementation
TIMED AND RESOURCE-ORIENTED STATECHARTS
TRoS
(Timed and Resource-Oriented Statecharts) =

Statecharts + Timed Actions
Railroad Crossing in Statecharts

![Statechart Diagram]

- No Approaching Train
- Train Approaching
  /CloseGate
- Train Moved Out
  /OpenGate
- Train Passing By
Timed Actions Expressions

\[ RS ::= (r, p), RS \mid (r, p) \]

\[ TA ::= \{RS\} \]
Annotated with Timed Actions

TrainApproaching
\{((\text{cpu},1), (\text{gate},1))^{20}\}
/CloseGate

TrainMovedOut
\{((\text{cpu},1), (\text{gate},1))^{25}\}
/OpenGate
Syntactic Sugar

S1

\[ E[C] \ TA/A \]

S2

S1

\[ E[C] \]

TA

\[ /A \]

S2
Syntactic Sugar

Diagram:

S1

E[C] TA^n/A

S2

S1

E[C]

TA^n

/A

S2
Railroad Crossing in TRoS

- TrainApproaching
- No Approaching Train
  - /OpenGate
  - GateOpenning \{(cpu,1),(gate,1)\}^{25}
- GateClosing \{(cpu,1),(gate,1)\}^{20}
  - /CloseGate
- TrainPassingBy
- TrainMovedOut
Railroad Crossing in TRoS
Railroad Crossing in TRoS

- **GateClosing** \{cpu,1\}, \{gate,1\}\(^{20}\)
- **GateOpenning** \{cpu,1\}, \{gate,1\}\(^{25}\)
- **TrainApproaching**
- **No Approaching Train**
- **TrainPassingBy**
- **TrainMovedOut**
- **/CloseGate**
- **/OpenGate**
Timed Behavior of TRoS

Micro-steps

Timed-step

Macro-step

\( t \) \hspace{1cm} \text{Timed-step} \hspace{1cm} t+1
Timed Behavior of TRoS

\[ \text{while true do} \]

Read external input events into \( I \);

\[ \text{while (there are any events to be handled) do} \]

\textit{Executes instantaneous steps;}

\textit{Execution Timed step;}

22
Case Study: DISTANCE CONTROL MODULE
PMA: Green

PMA: Yellow

PMA: Red

Occupied
Scheduling Functions

tm(DCM_CLK, 2)

```plaintext
SP(DCMTL_SLEEP)
/st!(BLN_OPEN_CLOSE)

HANDLE ATS_CNTL,Th
sp(BLC_OPEN_CLOSE)
/st!(SET_TEMP_SPEED)

HANDLE ATS_UNUSED
sp(UNSTABLE_SPEED)

TRN_SCHEDULING

MAKE_PROCESS_Q>

READ_TRN_SCHED
/TTNSCHED=TRN_STAT(TTRNID).TRN_SCHED; TTRNID-;

VERIFYING

S8
/ST!(VERIFY_TRAINE_POSITION)

VERIFYING_TRAINE

S9
/ST!(VERIFY_TRAINE_POSITION)

MONITORING_TRN_DIRECTION

S1
/ST!(MONITOR.TRAINE_DIRECTION)

UPDATE_BLC_STAT
sp(MANAGEMENT_BLC_STAT)
/st!(DISPLAY_DCM_STAT)

sp(TRAIN_DISTANCE_CONTROL)[TTRNID=0]
/st!(DISPLAY_DCM_STAT)

REPORT_TRN_PHA
sp(TRAIN_DISTANCE_CONTROL)[TTRNID>0]
```

S11
SP(DISPLAY_DCM_STAT) /DCM_CLK;

S12
```
Set Temporary Speed
Verify Train Position

Verify Train Position Activity

S2
[TRN_STAT_INITIALIZED]

[not TRN_STAT_INITIALIZED]

[TRN_CONN(TTRNID).LOST]

[TRN_CONN(TTRNID).LOST]
[TRN_STAT(TTRNID).TRN_STAT_UPDATE]

REFRESH_OLD_BLC_STAT>

[not TRN_STAT(TTRNID).TRN_STAT_UPDATE]

INIT_TRN_STATS>

UPDATE_TRN_LOST_IN_BLC_STAT>

UPDATE_TRN_POS_IN_BLC_STAT>

/TRN_STAT_INITIALIZED=true

/TRN_STAT(TTRNID).TRN_STAT_UPDATE=false;
for $I$ in STARTING_BLCID to ENDING_BLCID loop

    if ATS_CMD_TMPSPD($I)!=$NA then

        BLC_STAT($I).ATS_CMD_TMPSPD=ATS_CMD_TMPSPD($I);
        BLC_STAT($I).ATS_CMD_TMPSPDON=true;

    else

        BLC_STAT($I).ATS_CMD_TMPSPD=NA;
        BLC_STAT($I).ATS_CMD_TMPSPDON=false;

    end if;

end loop;
Simulation
Simulation Results
Model Checking Results
## Timing Properties

<table>
<thead>
<tr>
<th>Task</th>
<th>WCE (time unit)</th>
<th>Priority</th>
<th>HW Resource</th>
<th>Shared Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLC_OPEN_CLOSE</td>
<td>2</td>
<td>1</td>
<td></td>
<td>ATS_CMD_BLC_CLOSE, BLC_STAT</td>
</tr>
<tr>
<td>SET_TEMP_SPEED</td>
<td>2</td>
<td>2</td>
<td></td>
<td>ATS_CMD_BLC_TMPSPD, BLC_STAT</td>
</tr>
<tr>
<td>MANAGEMENT_BLC_STAT</td>
<td>2</td>
<td>6</td>
<td></td>
<td>BLC_STAT</td>
</tr>
<tr>
<td>MONITOR_TRAIN_DIRECTION</td>
<td>2</td>
<td>5</td>
<td></td>
<td>BLC_STAT, TRN_STAT, TTRNID, TRN_CONN</td>
</tr>
<tr>
<td>VERIFY_TRAIN_POSITION</td>
<td>11</td>
<td>4</td>
<td></td>
<td>BLC_STAT, TRN_STAT, TTRNID, TTRSCHED, DCM_TRN_CONN_OK, TRN_CONN</td>
</tr>
<tr>
<td>TRAIN_DISTANCE_CONTROL</td>
<td>6</td>
<td>7</td>
<td></td>
<td>BLC_STAT, TRN_STAT, TTRNID, TTRSCHED, DCM_TRN_CONN_OK, TRN_CONN</td>
</tr>
<tr>
<td>DISPLAY_DCM_STAT</td>
<td>4</td>
<td>8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
An Example of Scheduling DCM’s Functions

- MANAGEMENT_BLC_STAT
- BLC_OPEN_CLOSE
- SET_TEMP_SPEED
- VERIFY_TRAIN_POSITION
- MONITOR_TRAIN_DIRECTION
- TRAIN_DISTANCE_CONTROL
- DISPLAY_DCM_STAT
Verify Train Position in TRoS
Timed Behaviors in ACSR

\[
\begin{align*}
\text{MANAGEMENT\_BLC\_STAT} &= \{\} : \text{MANAGEMENT\_BLC\_STAT} \\
&\quad + (\text{runMANAGEMENT\_BLC\_STAT}, 4) \\
&\quad . \text{FIRST\_PHASE\_Ready}; \\
\text{FIRST\_PHASE\_Ready} &= \{\} : \text{FIRST\_PHASE\_Ready} \\
&\quad + \{(\text{cpu}, 4), (\text{BLC\_STAT}, 4)\} \\
&\quad : \text{FIRST\_PHASE\_}; \\
\text{FIRST\_PHASE\_} &= \text{SECOND\_PHASE\_Ready}; \\
\text{SECOND\_PHASE\_Ready} &= \{\} : \text{SECOND\_PHASE\_Ready} \\
&\quad + \{(\text{cpu}, 4), (\text{BLC\_STAT}, 4)\} \\
&\quad : \text{SECOND\_PHASE\_}; \\
\text{SECOND\_PHASE\_} &= \text{THIRD\_PHASE\_Ready}; \\
\text{THIRD\_PHASE\_Ready} &= \{\} : \text{THIRD\_PHASE\_Ready} \\
&\quad + \{(\text{cpu}, 4), (\text{BLC\_STAT}, 4)\} \\
&\quad : \text{THIRD\_PHASE\_}; \\
\text{THIRD\_PHASE\_} &= \text{FOURTH\_PHASE\_Ready}; \\
\text{FOURTH\_PHASE\_Ready} &= \{\} : \text{FOURTH\_PHASE\_Ready} \\
&\quad + \{(\text{cpu}, 4), (\text{BLC\_STAT}, 4)\} \\
&\quad : \text{FOURTH\_PHASE\_}; \\
\text{FOURTH\_PHASE\_} &= \text{mbsReturn}; \\
\text{mbsReturn} &= \{(\text{cpu}, 4)\} \\
&\quad : (\text{returnMANAGEMENT\_BLC\_STAT}, 4) \\
&\quad . \text{MANAGEMENT\_BLC\_STAT};
\end{align*}
\]
Schedulability Analysis

```
E:\Chi_Paper Work\ROD\ROD ACSR Models\DCM>uxp dcm_ac_v1.acsr

::> DCM == IDLE?
   ufi failed--following pair could not be matched:
     <DCMInit :: DCMFunction>
     dcm_system_on,runBLC_OPEN_CLOSE,returnBLC_OPEN_CLOSE,runSET_TEMP_SPEED,returnSET_TEMP_SPEED,runDISPLAY_DCM_STAT,returnDISPLAY_DCM_STAT,runMONITOR_TRAIN_DIRECTION,returnMONITOR_TRAIN_DIRECTION,runVERIFY_TRAIN_POSITION,returnVERIFY_TRAIN_POSITION,runTRAIN_DISTANCE_CONTROL,returnTRAIN_DISTANCE_CONTROL,runSOFTWARE_MANAGEMENT_BLC_STAT,returnSOFTWARE_MANAGEMENT_BLC_STAT>
     cpu,ATS_CMD_BLC_CLOSE,ATS_CMD_TMPSPD,BLC_STAT,TRN_CONN,TTRNID,TRN_STAT,DCM_RN_CONN_OK,TTRNCHED,DCM_TRN_CMD}, <:IDLE>
   --following pair was matched:
     <DCM,IDLE>
   true <by prioritized strong equivalence>
```
Conclusions

• Timing correctness of real-time embedded software is as important as functional correctness,

• The analyzing those behaviors needs multiple models,
Conclusions

• TRoS extends the functional behaviors of Statecharts with **timed actions**, 

**Timed actions** imply execution time and shared resource related information,
Conclusions

• Functional behaviors in Statecharts are extended into timed behaviors in TRoS by only annotating timed actions
Conclusions

• Applied to the development of a part of railway interlocking control systems,
Conclusions

• Easily getting the timed behavior model to verify timed behaviors constrained by time and resources

• using multiple verifications for Statecharts and TRoS,
Conclusions

• Future works
  – Specific platform-dependent analysis of application timed behavior,
    • RTOS
    • Platform
    • Etc,…