MODELING OF COMPLEX AUTOMATION SYSTEMS USING COLORED STATE CHARTS

Olga Fengler
Wolfgang Fengler
Bernd Däne
Vesselka Duridanova
Ilmenau Technical University
Department of Computer Architectures
E-mail: olga.fengler@tu-ilmenau.de
Web: http://tin.tu-ilmenau.de/ra
Outline

1. Motivation

2. State Charts

3. Colored State Charts

4. Conversion into High Level Petri Nets

5. Example model

6. Conclusion und Perspectives
Motivation

State Charts:
- many similar objects
- many elements
- very large and disturbing diagrams

Extending State Charts:
- Reducing complexity
- Partly introducing the well established concept of High Level Petri Nets
- Compact representation of similar parallel processes or objects
State Charts

- Graphical language for processes and sequences
- Usable even in early stages of system development
- Formally defined, textual description available
Colored State Charts

„Folding“

Matching states of similar content and structure

- some subclasses
- a class and its subclasses
- some objects of just one class
- some objects of several subclasses
Colored State Charts

- Example represents three objects from one parental class

- There are complex states, events, transitions

- Activities with time interval properties are available

\[ \text{axisMotion} \]

\[ \text{H} \]

\[ \text{forwards} \]

\[ \text{entry/PositionIncrease} \]

\[ \text{RA} \]

\[ \text{RA} \]

\[ \text{RA} \]

\[ \text{RA} \]

\[ \text{axisX axisY1 axisY2} \]
Colored State Charts

State

- Folded from several local states
- Complex capacity
- Current marking

active

name

intern activity

entry / validateRect

do / messageDispatch

WM_PAINT / OnPaint()

red || green || blue

red

capacity

marking

reference of decomposition

n Folded from several local states
n Complex capacity
n Current marking
Colored State Charts

Transitions

\[ \text{event\{red \&\& green\}\[requirement\{red\}\]/action} \]

state1 \rightarrow state2

function of objects (requirements)

transition spezification
Colored State Charts

- Colored states
- Complex transitions
Conversion into High Level Petri Nets

Transformation of the structure:

- decomposition into single modules
- transformation using existing algorithms
- "coloring" the net
- transformation of color dependencies
- transformation of state capacities

Key aspects of the transformation are:

- transformation of the place capacity,
- definition of the firing modes for transitions,
- transformation of hierarchical structures,
- transformation of time events and timed transitions
Conversion into High Level Petri Nets

<table>
<thead>
<tr>
<th>Name</th>
<th>R</th>
<th>G</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Marking</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Capacity</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Total capacity</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Idle capacity function

Red
Conversion into High Level Petri Nets
Example Model

axisMotion

axis X, axisY1, axis Y2

axisMotion

forward entry/PositionIncrease

backwards entry/PositionDecrease

RA RA RA RA

stop

sinusGeneration

entry/ SIN = sin(Pos)

cosinusGeneration

entry/ COS = cos(Pos)

creating disturbance

clock

clock

clock

Estimation of the period number

{Quadrant0, Quadrant1, Quadrant2, Quadrant3 }

system quadrants

QA1[W1]

actual quadrant

clock

Update the norm values

clock

clock

clock

Calculation of exact position

entry/ Pos = A1

calculation of Y-position and Y-divergence

entry/ Y = A2, dY = A3

axisY1 && axisY2

Output of the position

Output of actual position

system fault

Ilmenau Technical University • Department of Computer Architectures
Tool Development and Adaption

Conversion utility

Messsystem

TU Ilmenau, Fachgebiet Automatisierungsanlagen und Prozeßtechnik
Conclusion und Perspectives

- Compact and clear representation
- Formally defined description
- Available methods for analysis and transformation based on the Petri Net notation
- Available editors for colored diagrams
- Easy-to-use conversion utility