Radboud Universiteit Nijmegen

TU/e Technische Universiteit Eindhoven University of Technology

Embedded Systems INSTITUTE

océ Printing for Professionals

Formal Modeling and Scheduling of Data Paths of Digital Document Printers

Georgeta Igna
Frits Vaandrager
Venkatesh Kannan
Marc Voorhoeve
Yang Yang
Twan Basten
Marc Geilen
Sebastian de Smet
Lou Somers

Outline

- Introduction
  - The Octopus Project
  - Océ System Architecture
- Modelling and Analysis Approaches
  - Timed Automata
- Comparison
- Conclusions and Future Work

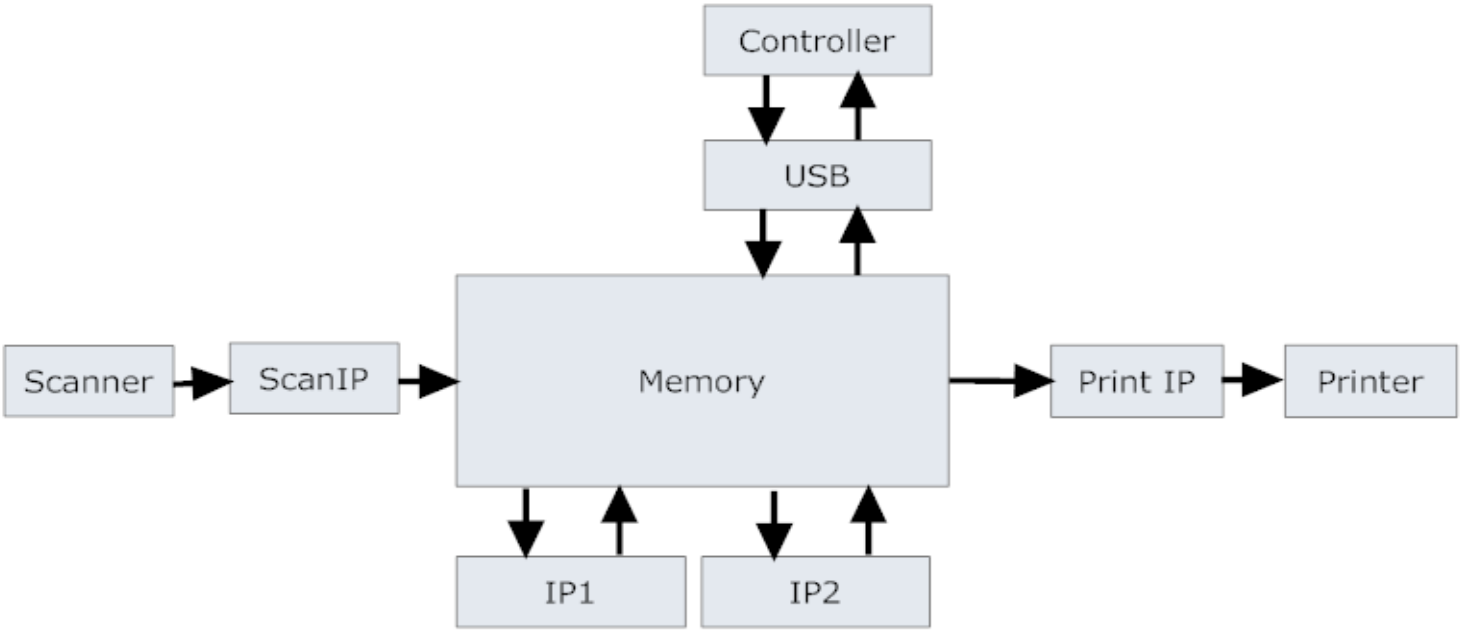
Outline

- **Introduction**
  - **The Octopus Project**
  - **Océ System Architecture**
- Modelling and Analysis Approaches
  - Timed Automata
- Comparison
- Conclusions and Future Work

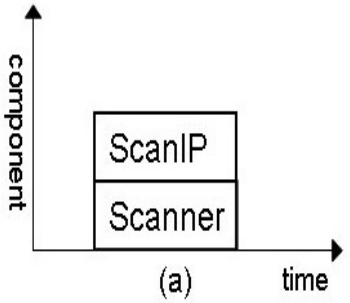
The Octopus Project

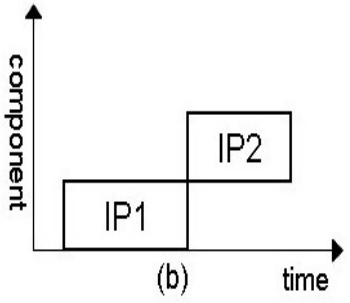
- **Aim**
  - new methods for designing adaptive datapaths of printers/copiers
- **Academic Partners**
  - *Radboud University of Nijmegen*
    - **Timed Automata**
  - *Technical University of Eindhoven*
    - **Colored Petri Nets**
  - *Technical University of Eindhoven*
    - **Synchronous Dataflow Graphs**

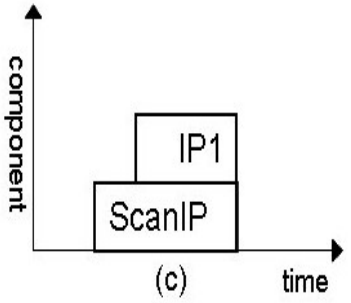
Océ System Architecture(1)

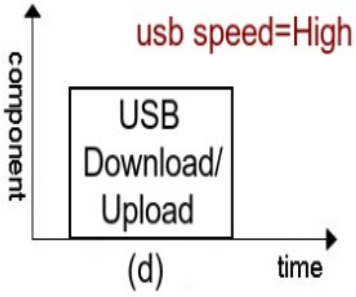


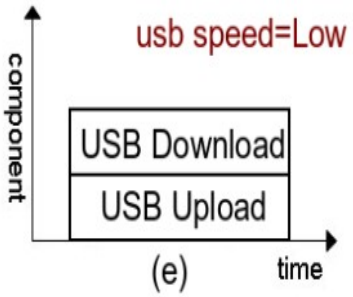
Constraints:



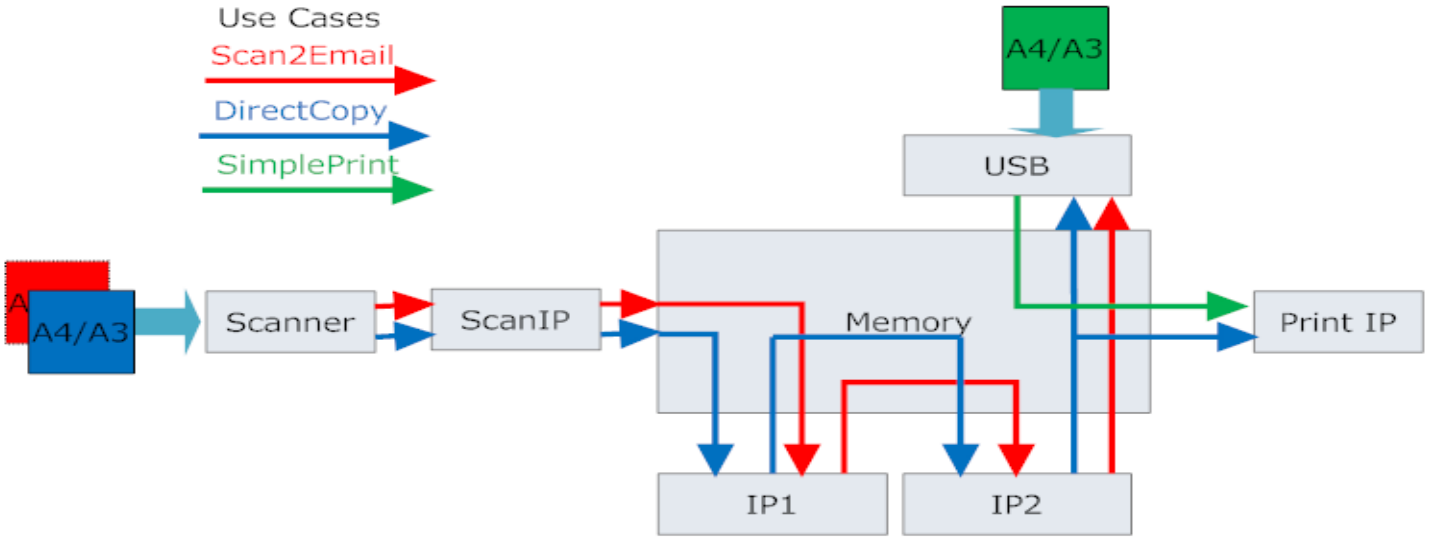


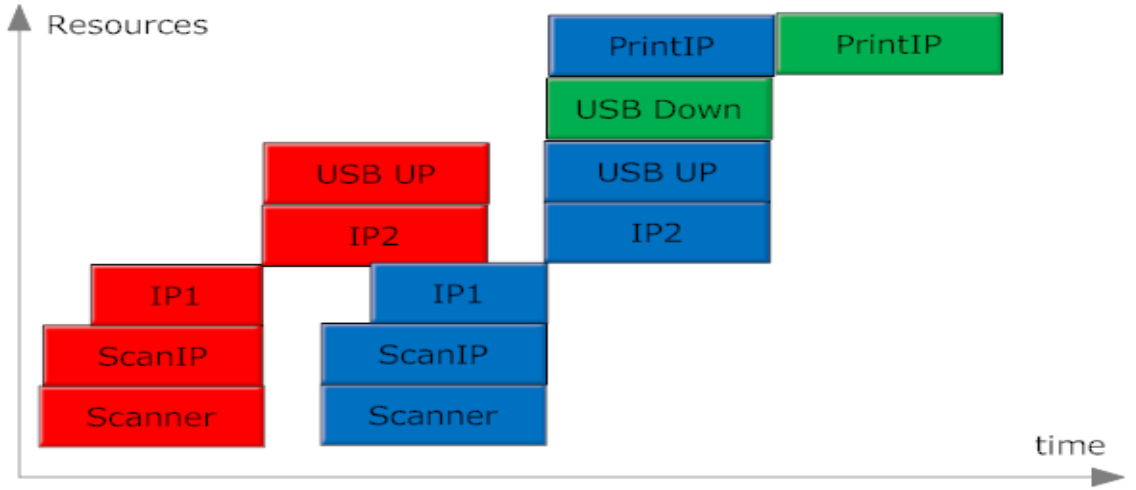






Océ System Architecture(2)



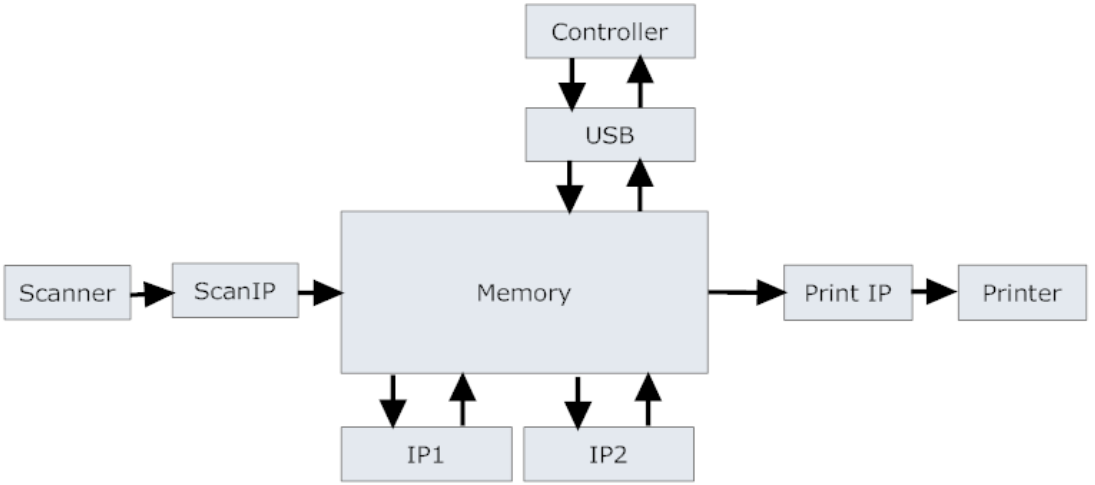


Outline

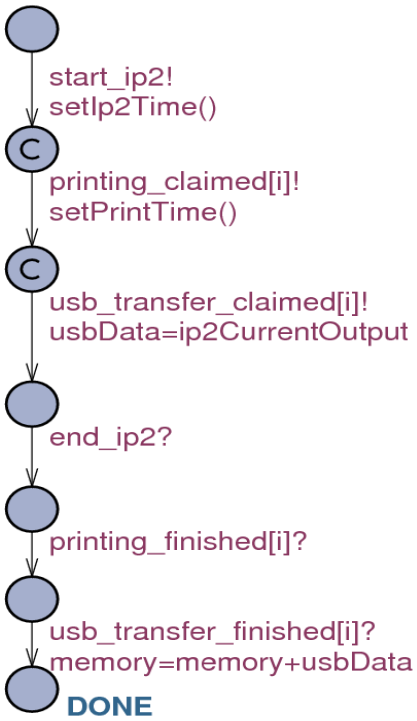
- Introduction
  - The Octopus Project
  - Océ System Architecture
  - Use Cases
- **Modelling and Analysis Approaches**
  - **Timed Automata**
- Comparison
- Conclusions and Future Work

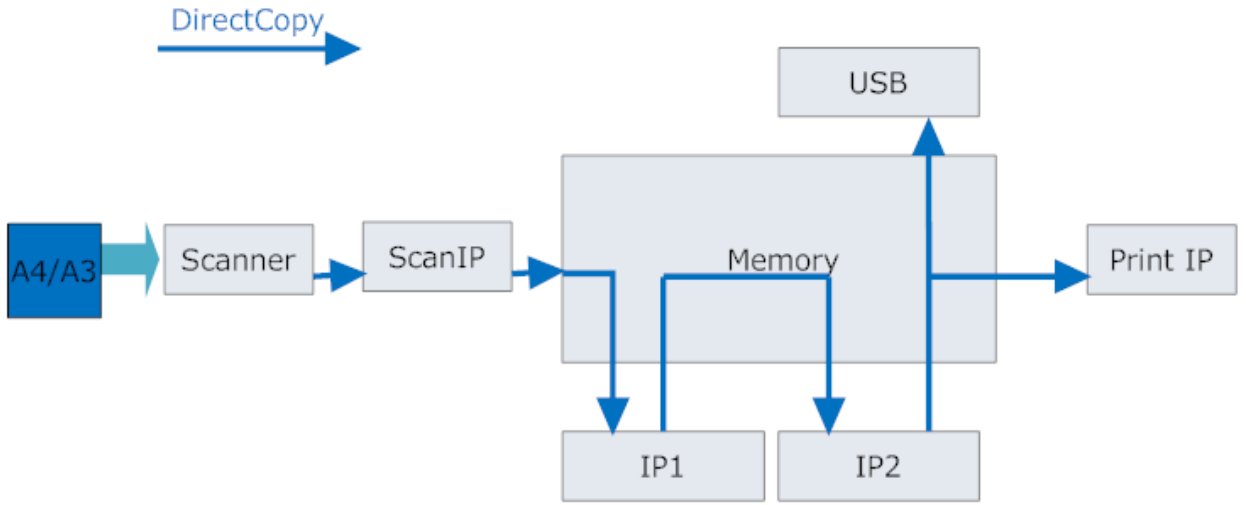
Timed Automata Model – General View

- Model
  - Automata:
    - Components: Scanner, ScanIP, IP1, IP2, and PrintIP
    - Use Case
    - USB model
  - Shared variables:
    - Memory
- Analysis
  - Uppaal model checker

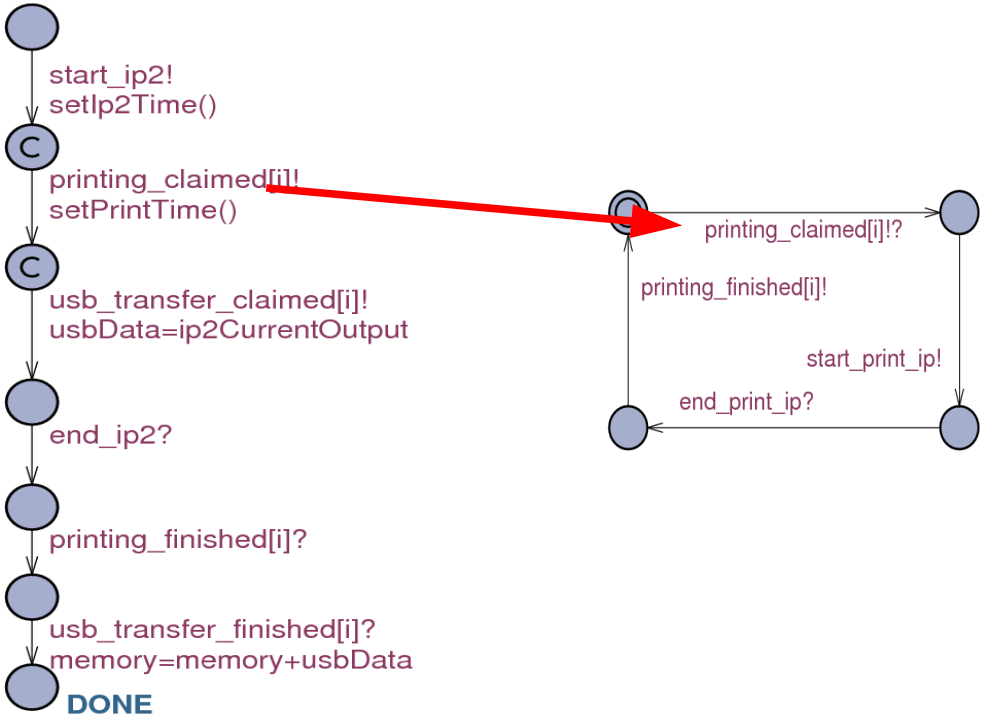


TA Model – Modelling Job's Parallel Activities(1)

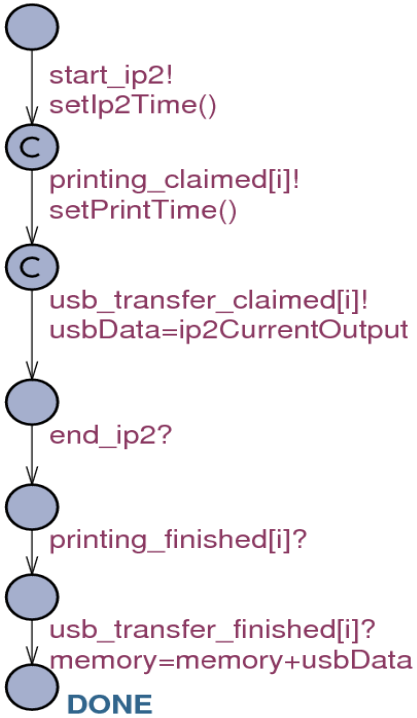


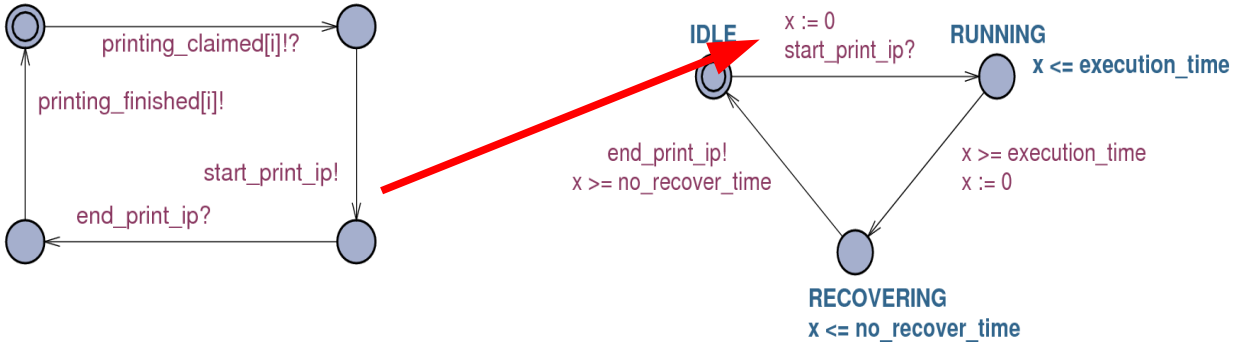


TA Model – Modelling Job's Parallel Activities(2)

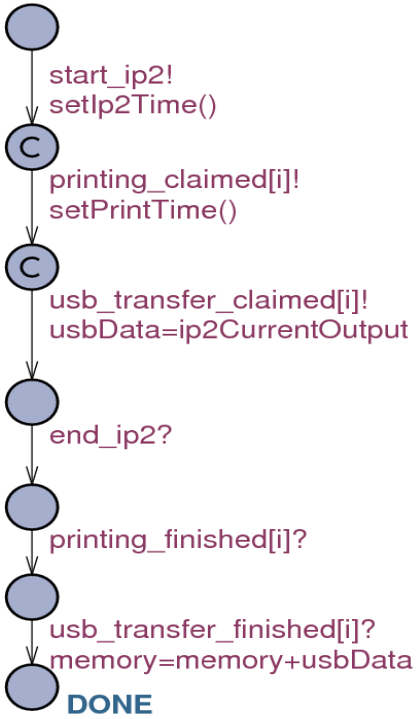


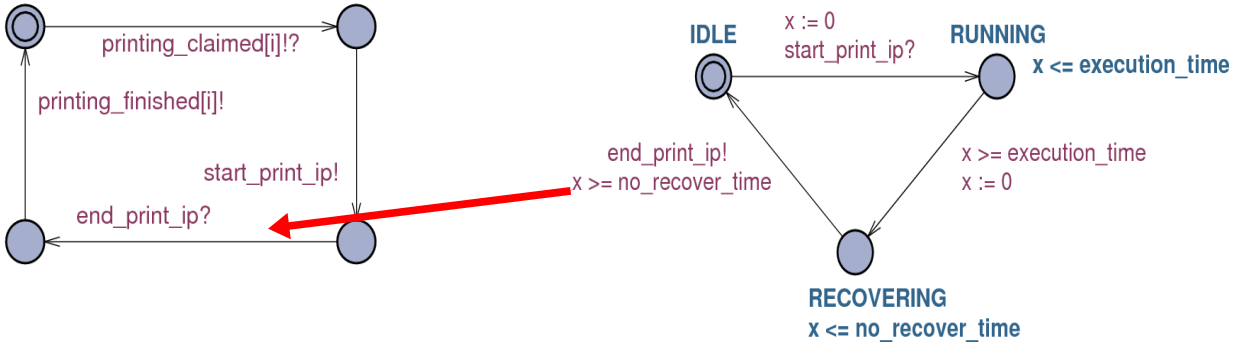
TA Model – Modelling Job's Parallel Activities(3)



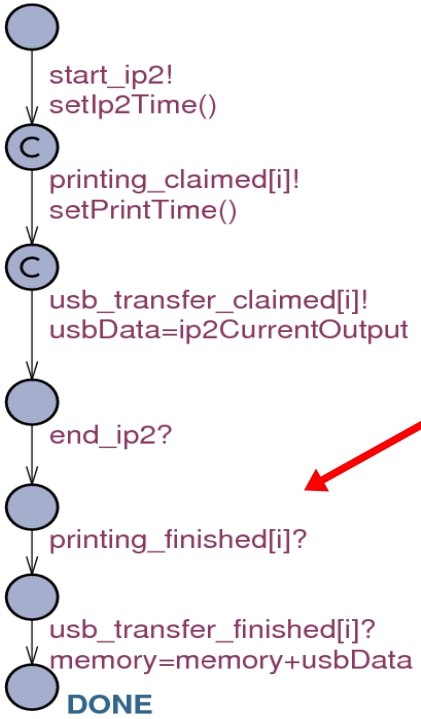


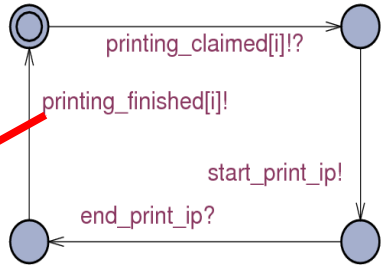
TA Model – Modelling Job's Parallel Activities(4)

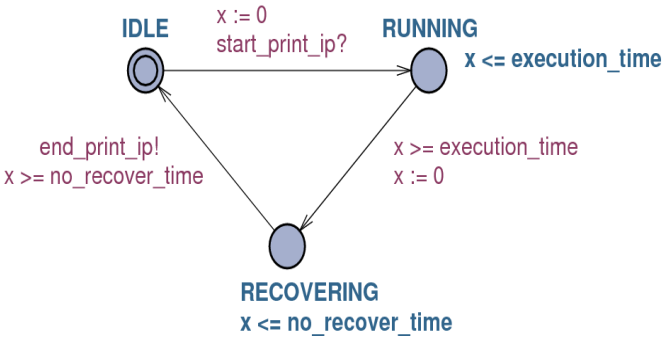




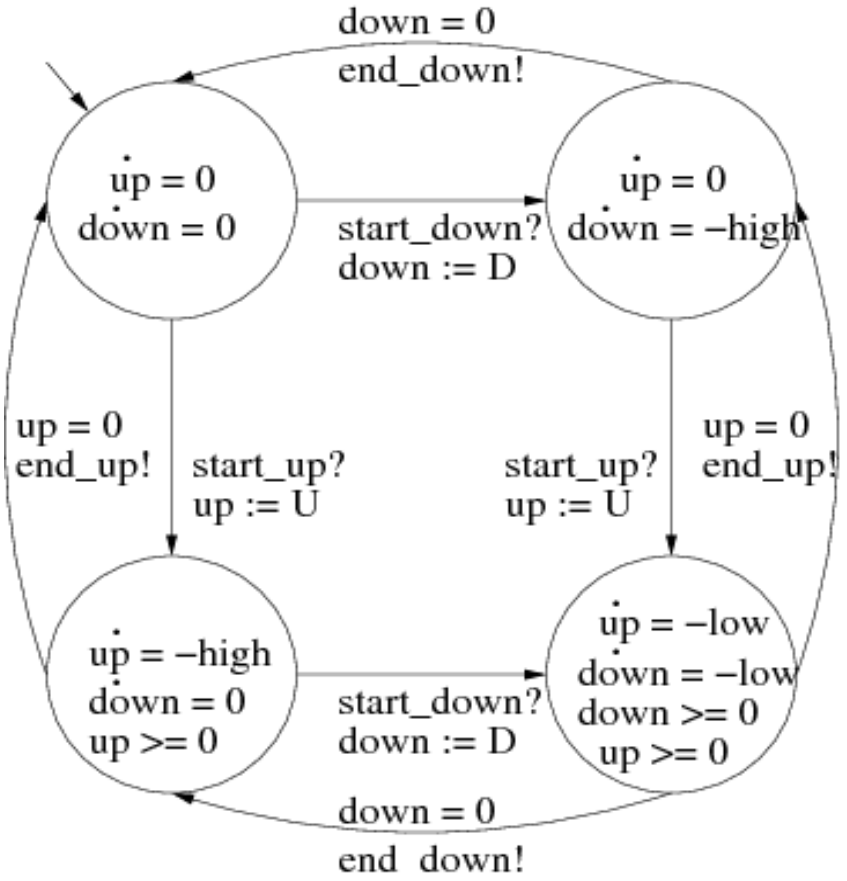
TA Model – Modelling Job's Parallel Activities(5)



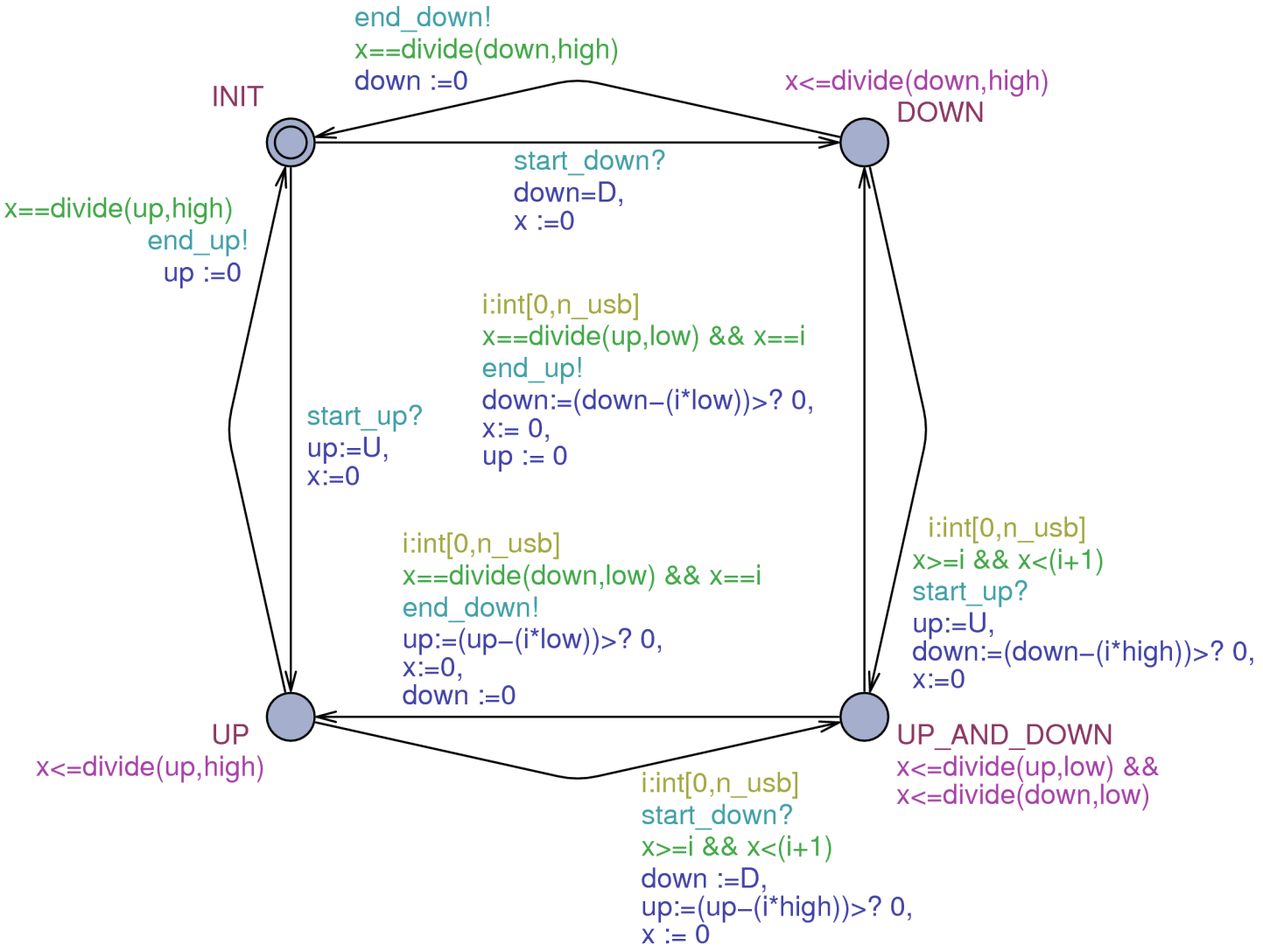




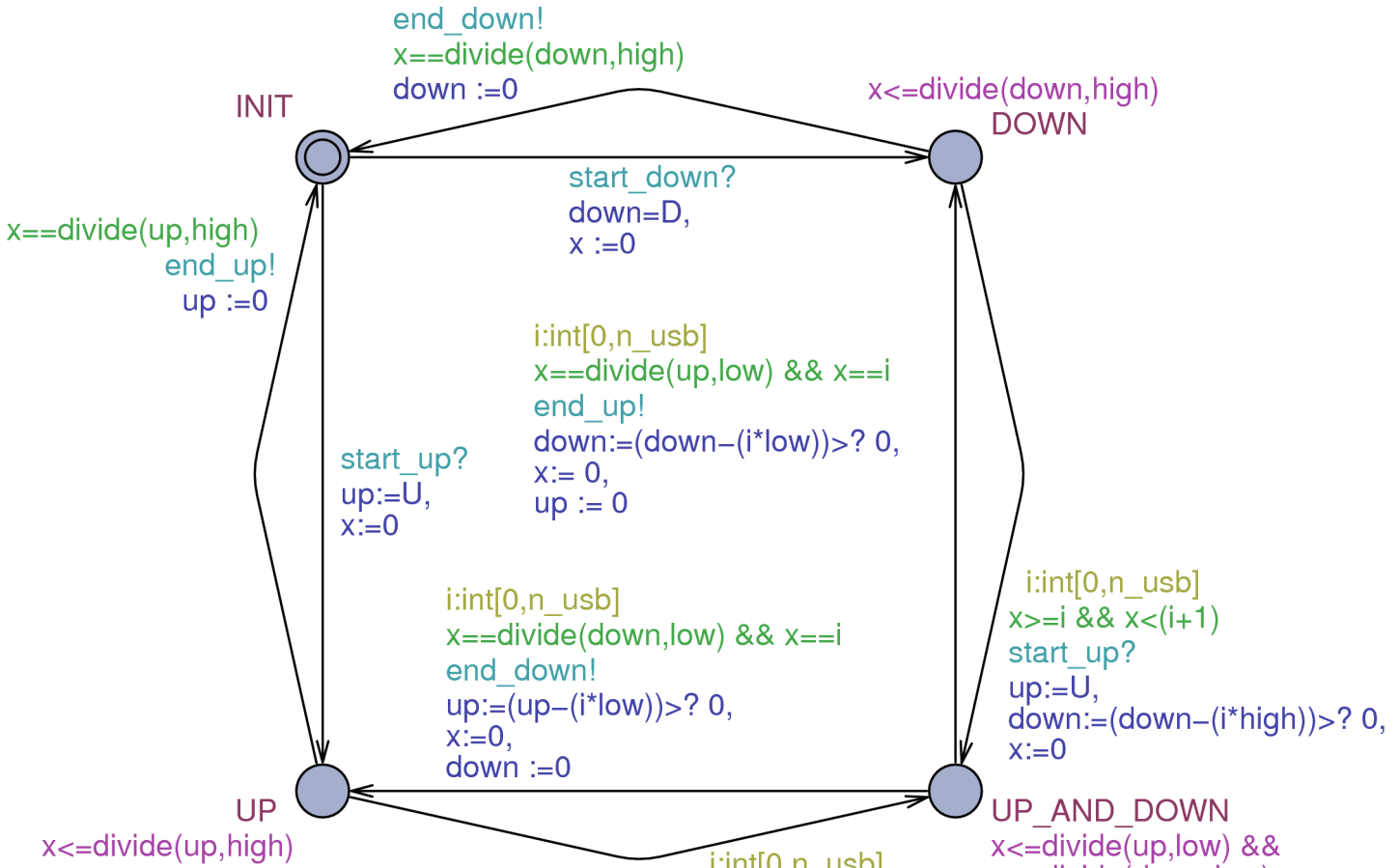
TA model – USB Linear Hybrid Automaton(1)



TA model - USB Automaton(1)



TA model - USB Automaton(2)



approximate the value of a clock to the closest integer

Outline

- Introduction
  - The Octopus Project
  - Océ System Architecture
  - Use Cases
- Modelling and Analysis Approaches
  - Timed Automata
  - Colored Petri Nets
  - Synchronous Dataflow Graphs
- **Comparison**
- Conclusions and Future Work

Comparison

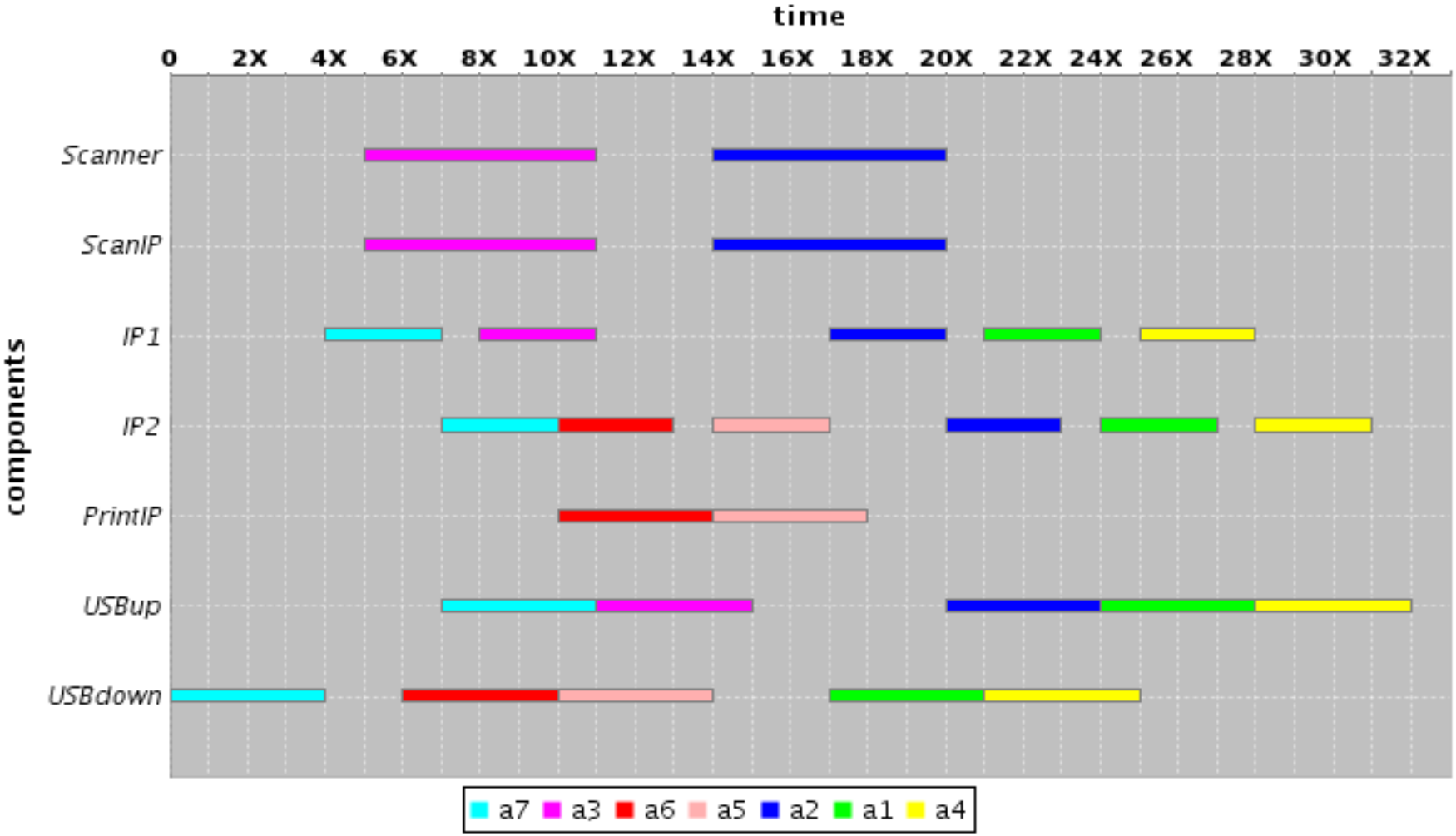
JobID	Use Case	Arrival time	Memory required
a1	ProcessFromStore	2X	24Y
a2	Scan2Email	1X	48Y
a3	Scan2Store	1X	36Y
a4	ProcessFromStore	3X	24Y
a5	PrintWithProcessing	1X	12Y
a6	PrintWithProcessing	0	12Y
a7	ProcessFromStore	0	24Y

X = 1s, 0.1s, 0.01s, ...
Y = 1MB, 32MB, 64MB, ...

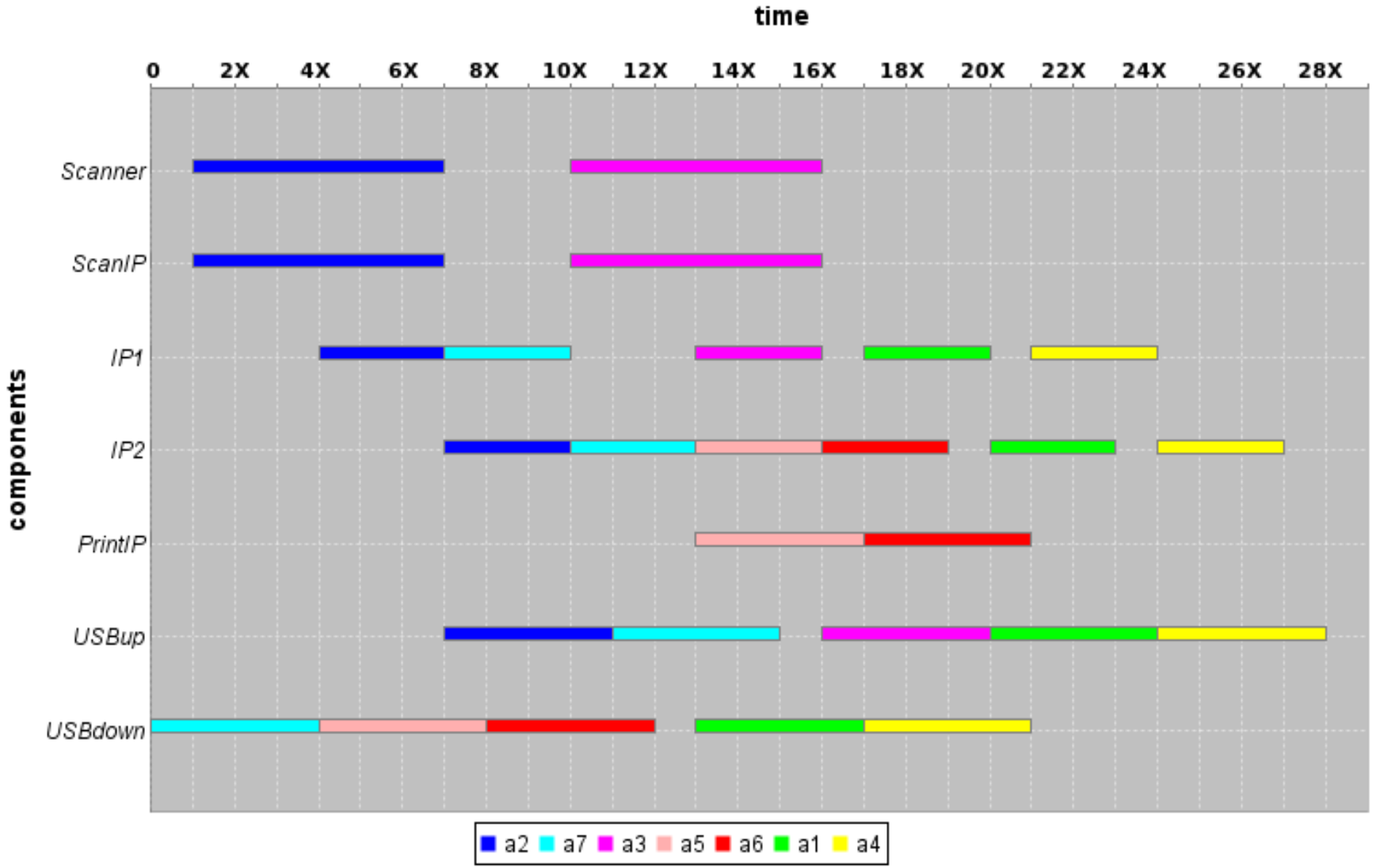
Completion time	**Static USB Model**	**Dynamic USB Model**
TA	27X	25X
CPN	28X	25.5X
SDF	32X	-

Static USB Model: USB speed=Low
Dynamic USB Model: USB speed=High/Low

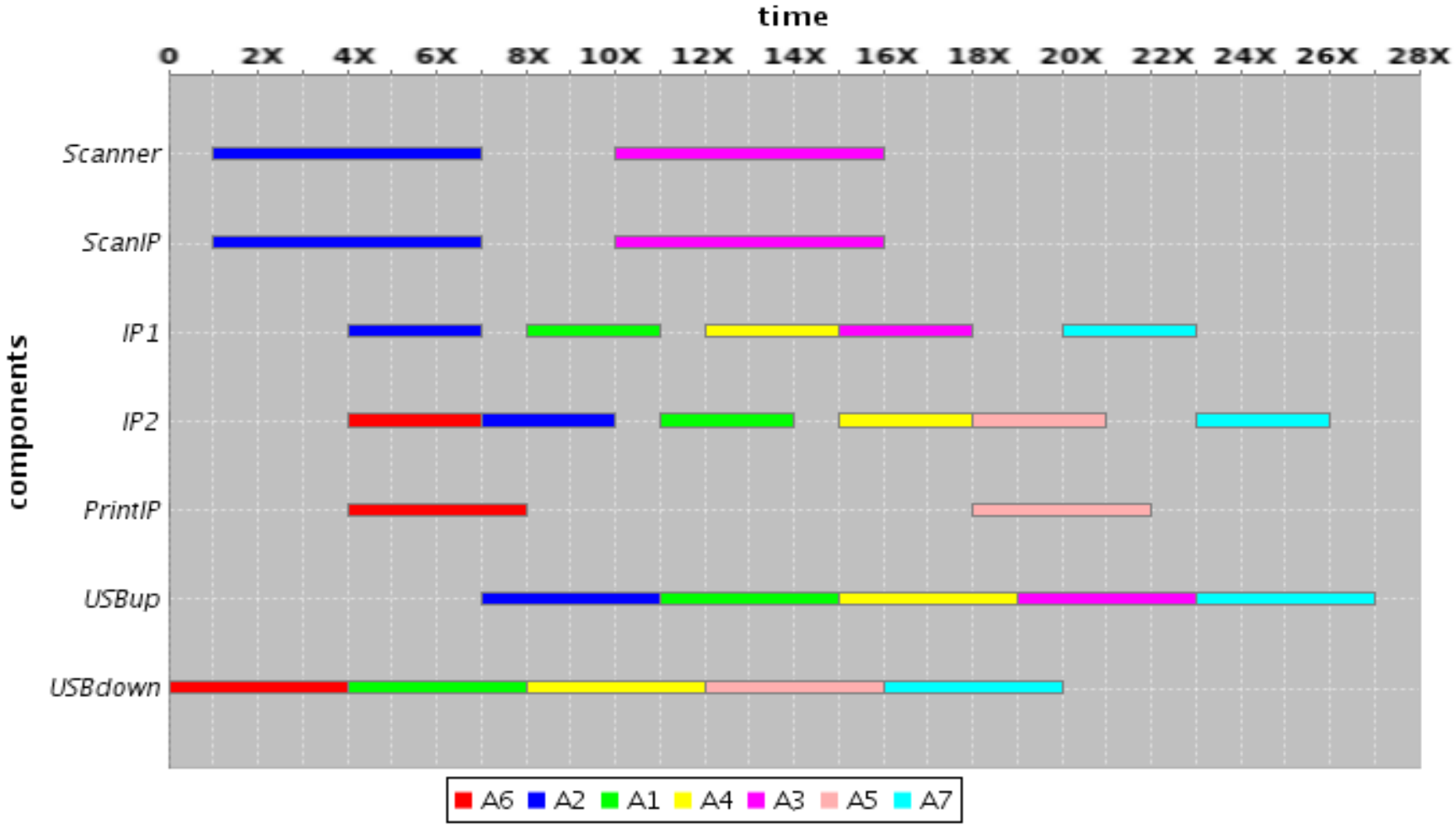
Static USB Model - SDF Results



Static USB Model - CPN Results



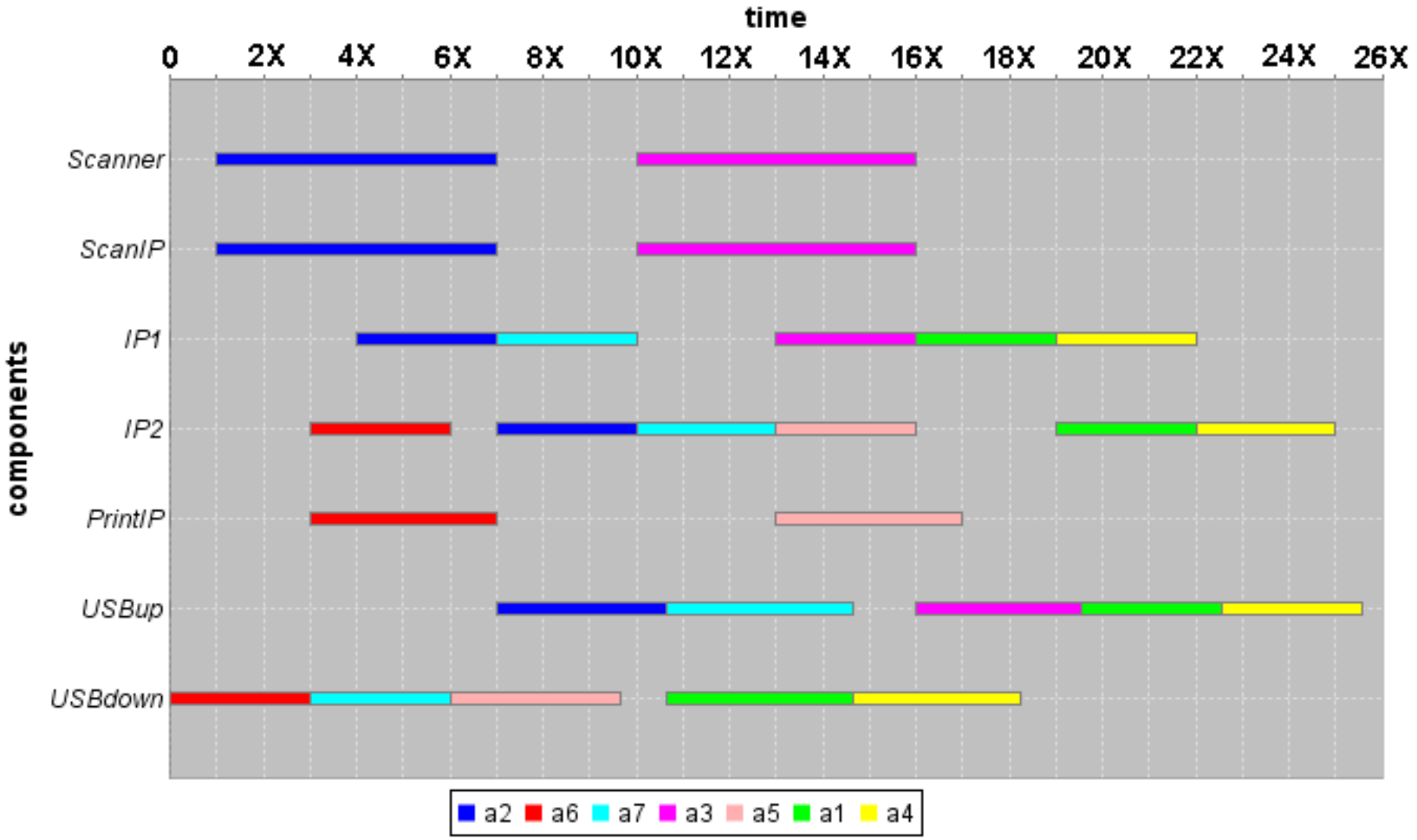
Static USB Model - TA Results



Conclusions and Future Work

- Model construction for the Océ system using three different models: TA, CPN, and SDF
- Comparison between models
  - TA gives the optimal schedules but close to state space explosion
  - CPN is the most expressive model
  - SDF can analyse a bigger number of input jobs
- *Current Status*
  - add another level of refinement: multipage jobs, real data for execution times of the components
  - comparison between results obtained with models and obtained using real printers
- *Next steps:*
  - model the memory bus
  - memory fragmentation

Dynamic USB Model – CPN Results



Dynamic USB Model – TA Results