

# Theory !

---

A Giotto program is a **single** set of tasks,  
so far

Is this enough?

# Theory = Practice ?

---

- **Yes**, we have sufficient computation and communication resources!  
=> A single set of tasks is enough

# Theory = Practice ?

---

- **Yes**, we have sufficient computation and communication resources!  
=> A single set of tasks is enough

Fin

# Theory = Practice ?

---

- **Yes**, we have sufficient computation and communication resources!  
=> A single set of tasks is enough

Fin

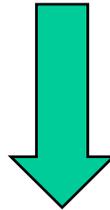
- **No**, we have to share resources!  
=> with modes

# Practice !

---

A mode is a **set** of tasks.

A Giotto program is a **set** of modes.

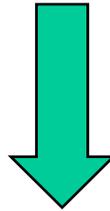


A Giotto program is a partitioning  
of the set of all **required** tasks.

# FAQ: Time vs. Events

---

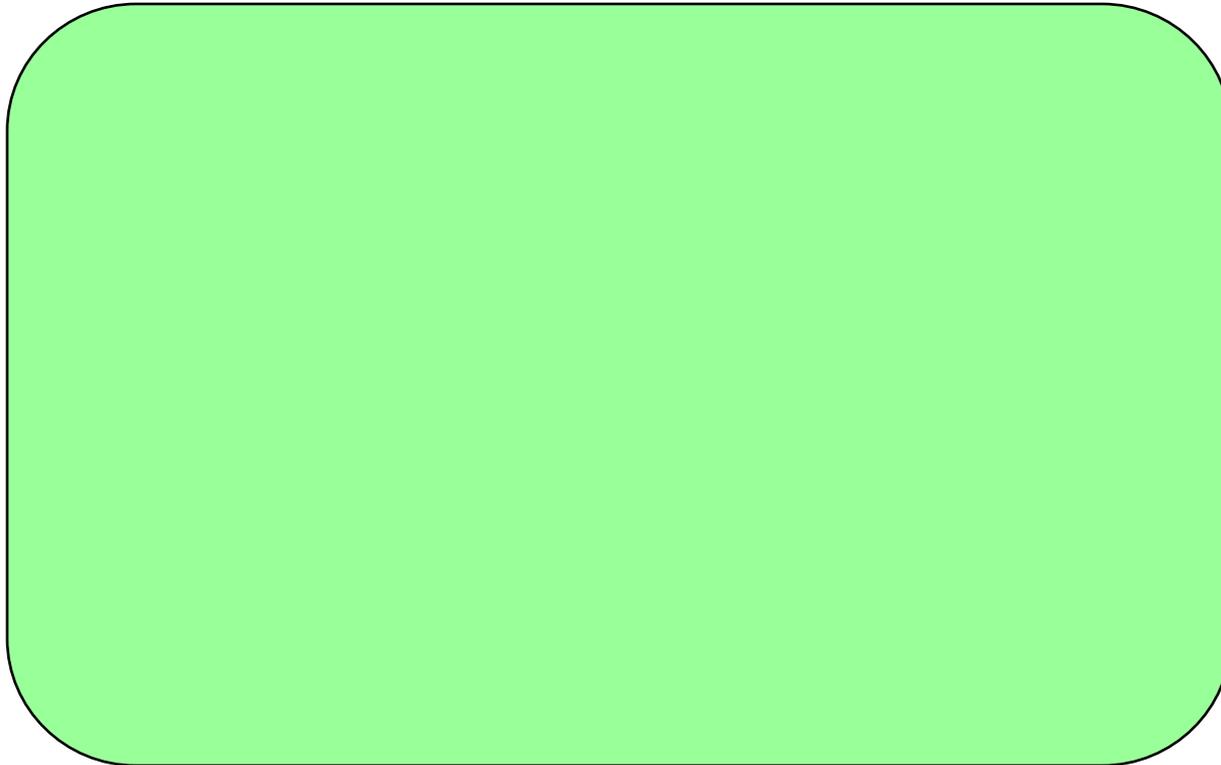
Q: How about events?



A: Use modes!

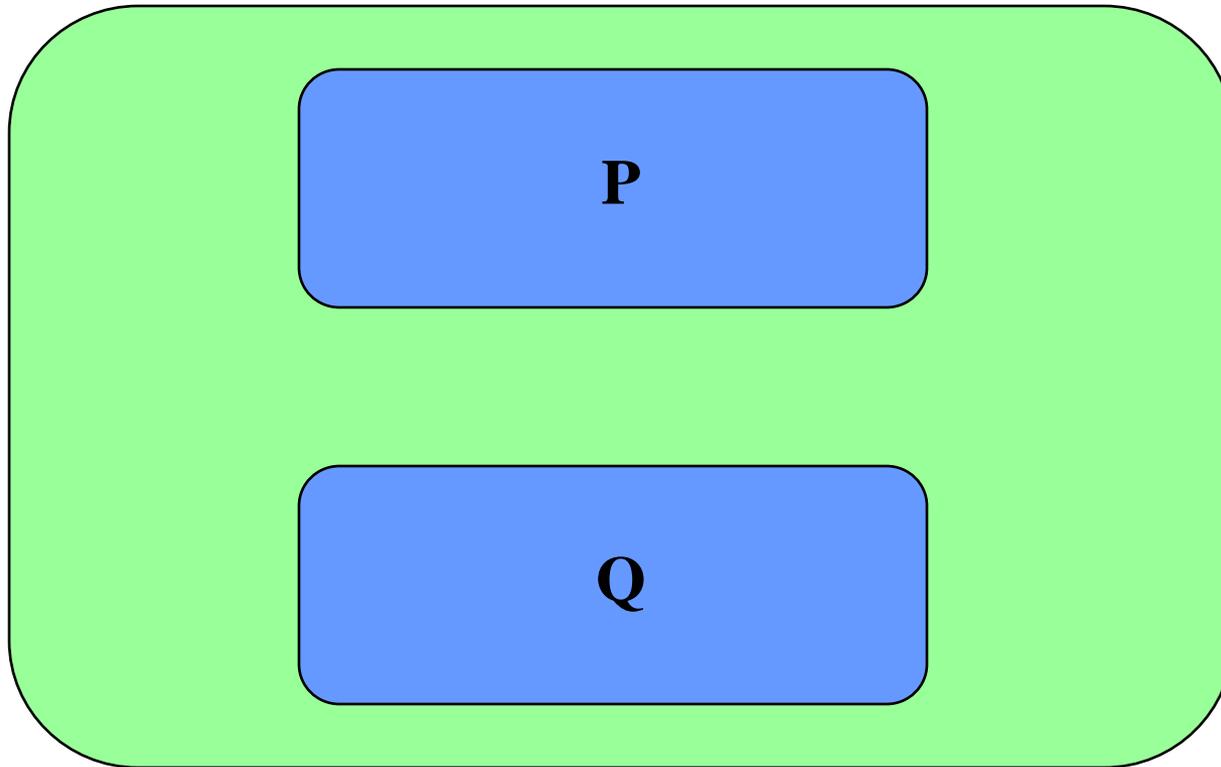
# A Mode

---



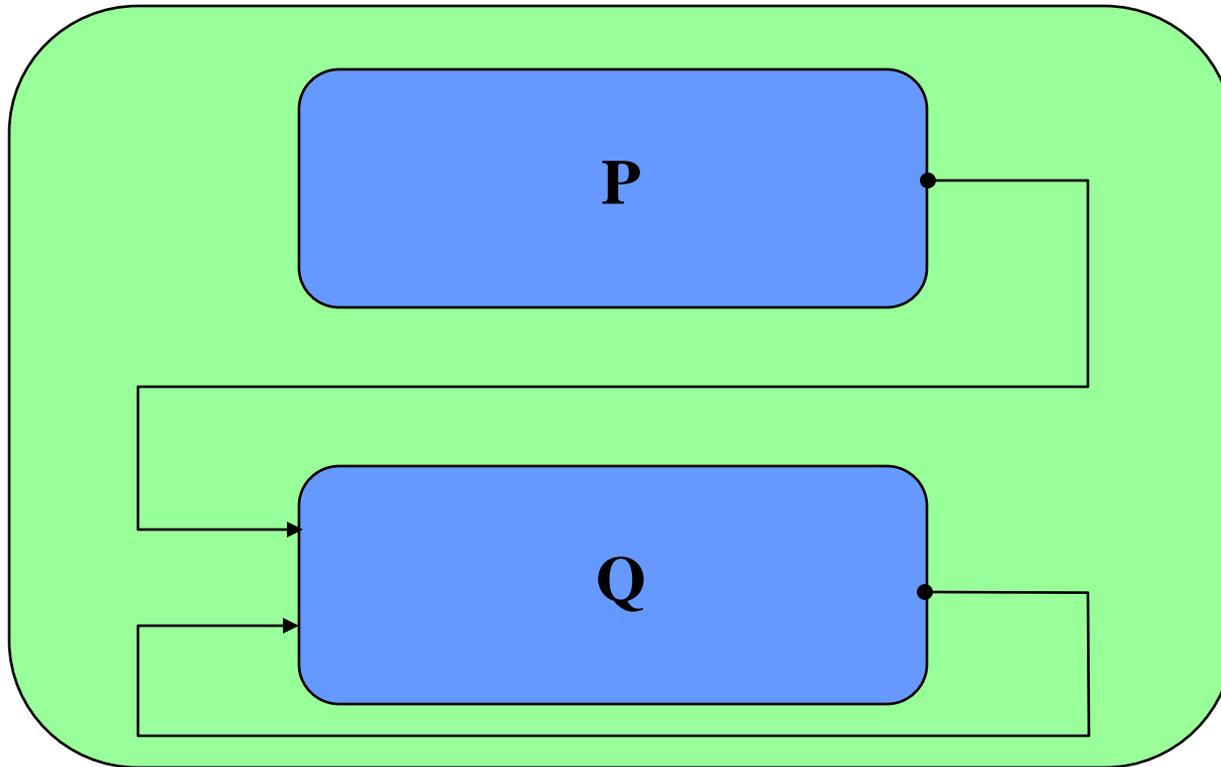
# Abstract Syntax of a Mode

---



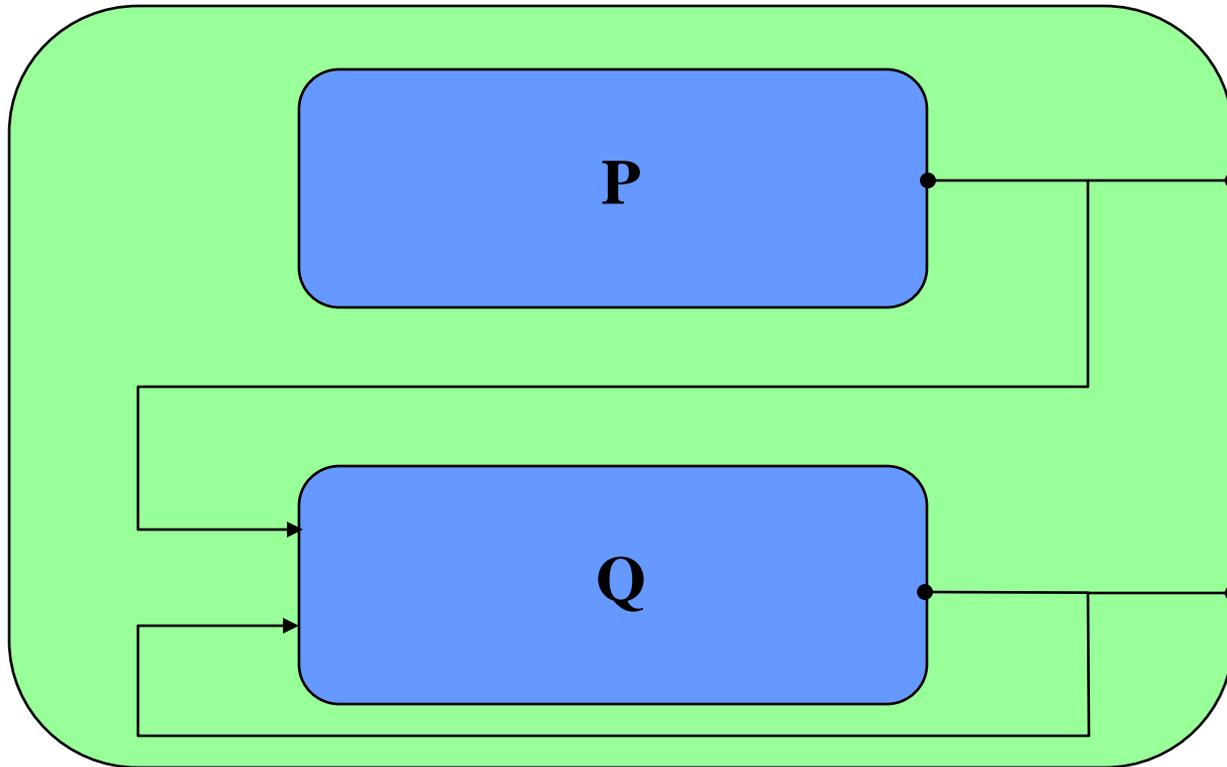
# Abstract Syntax of a Mode

---



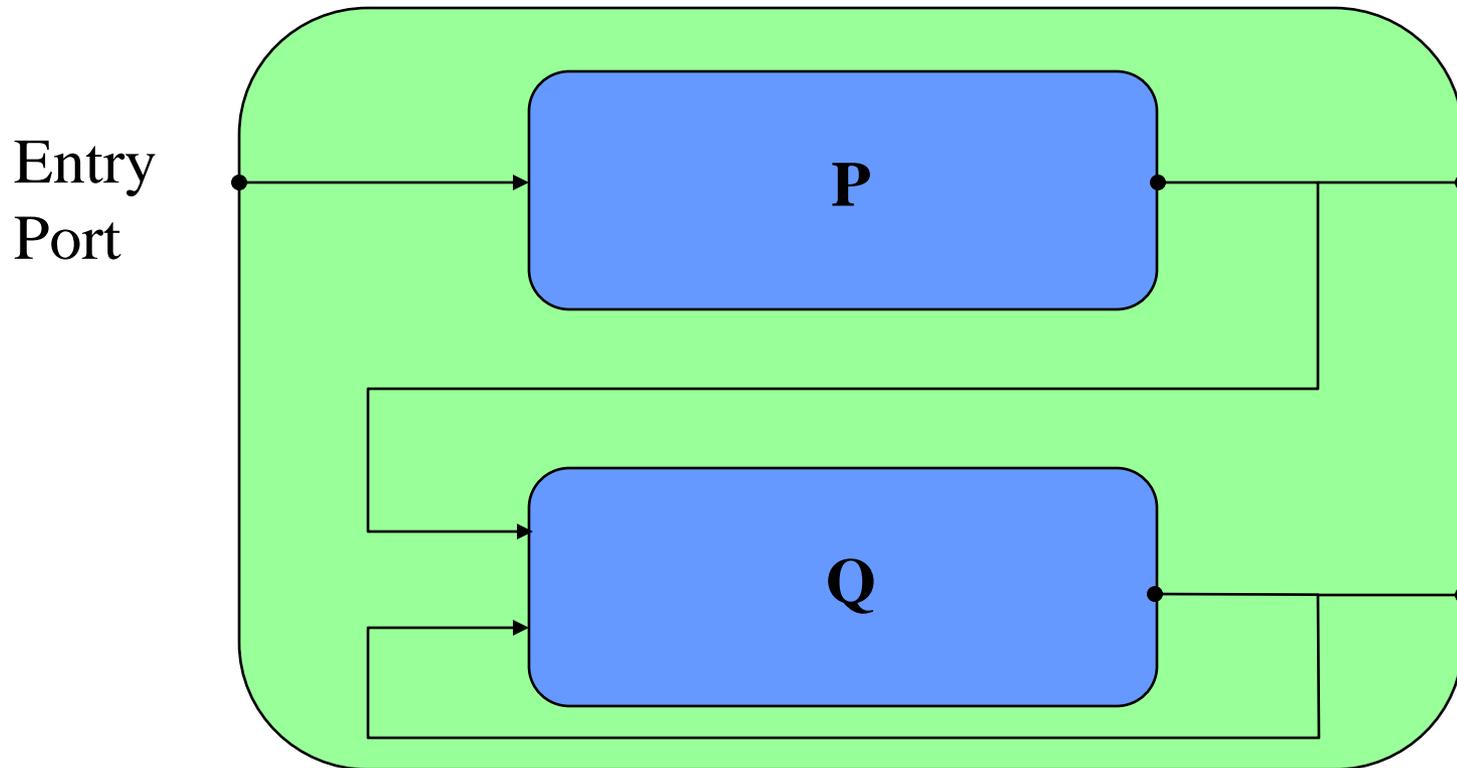
# Abstract Syntax of a Mode

---



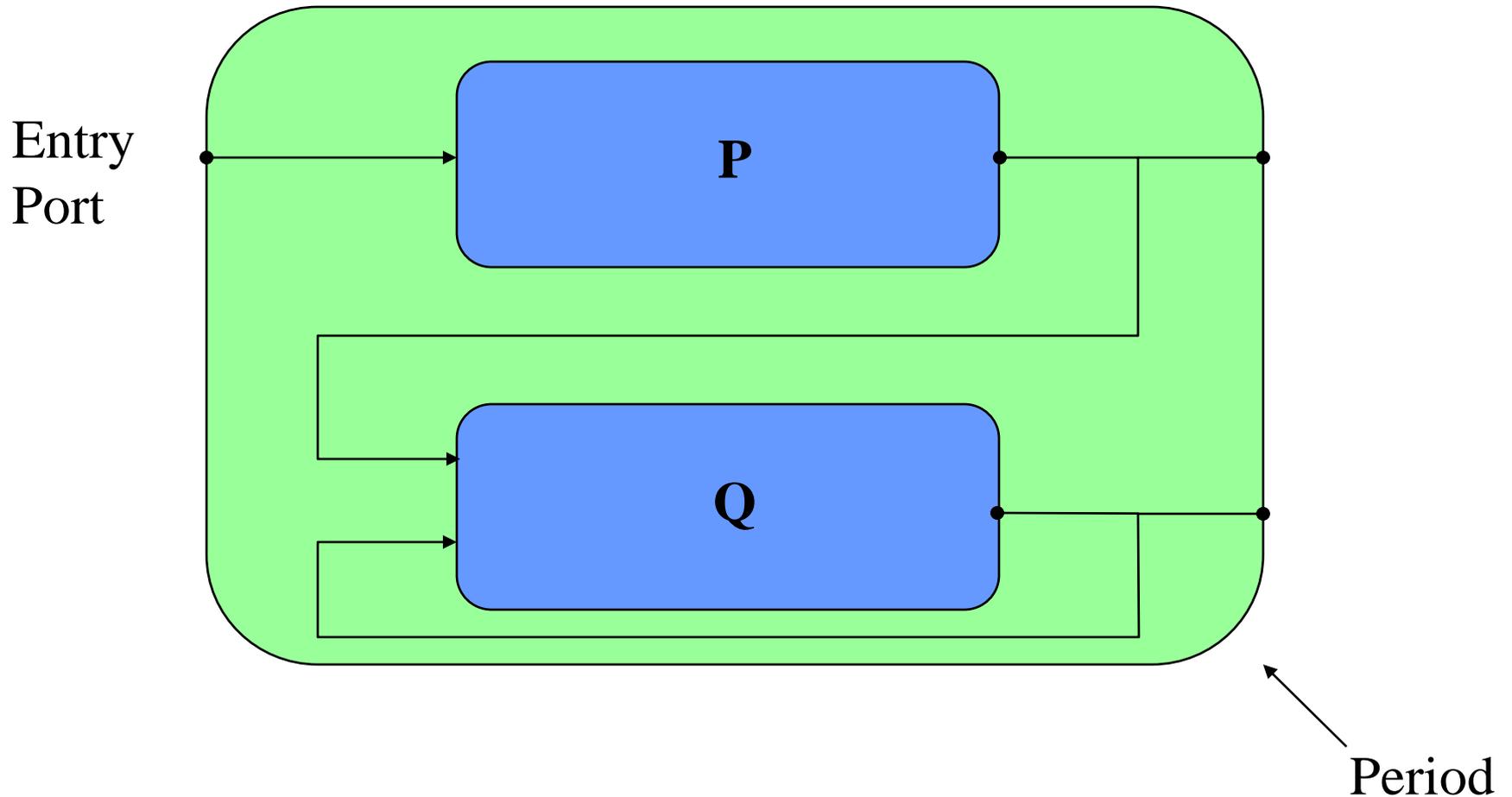
# Abstract Syntax of a Mode

---



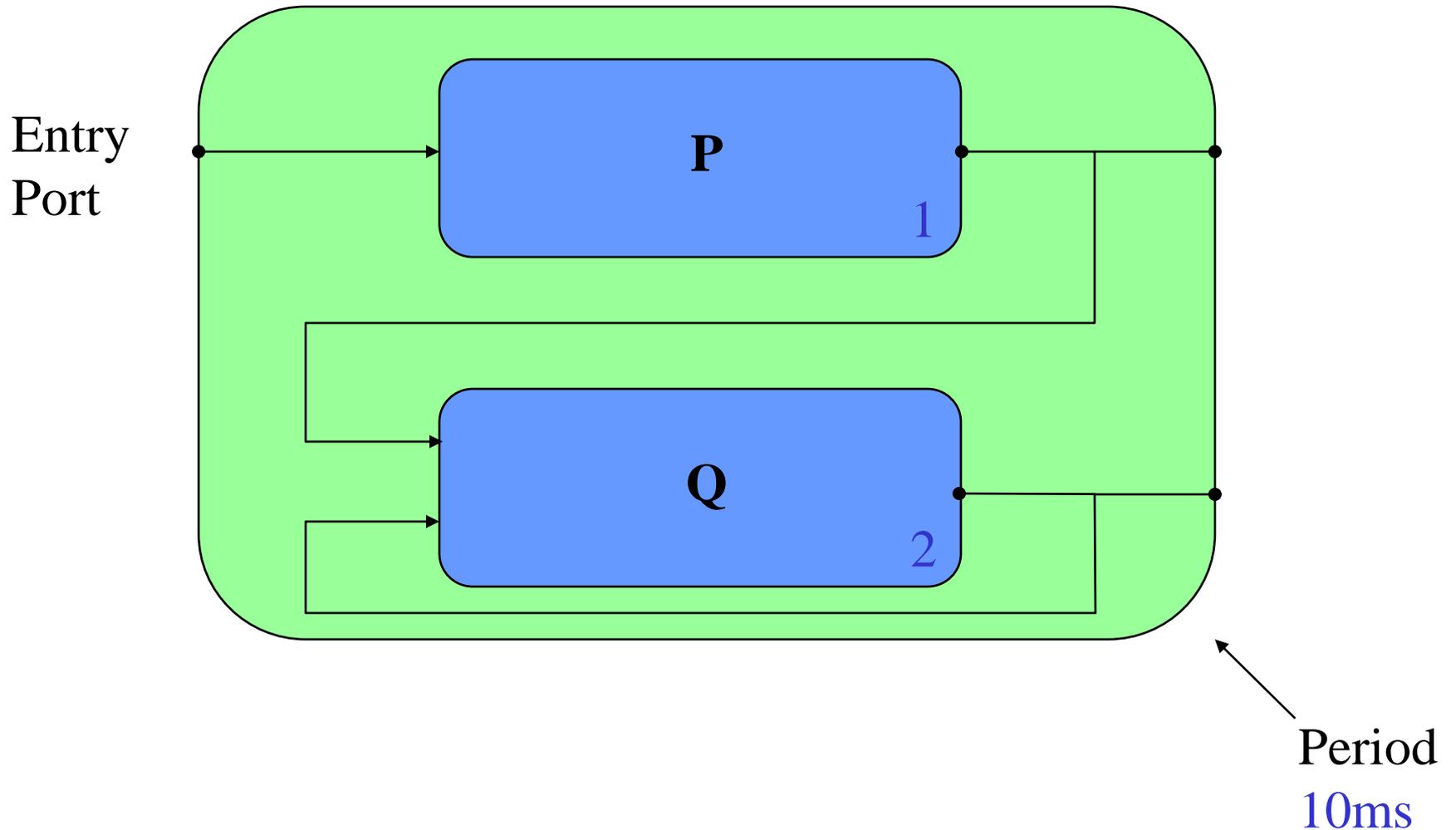
# Abstract Syntax of a Mode

---



# Semantics of a Mode

---



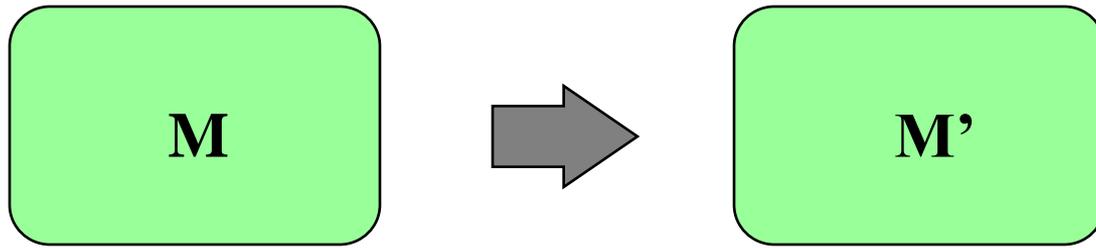
# Concrete Syntax

---

```
mode m ( ) period 10 ms
{
    taskfreq 1 do int x = P ( ) ;
    taskfreq 2 do int y = Q ( x, y ) ;
}
```

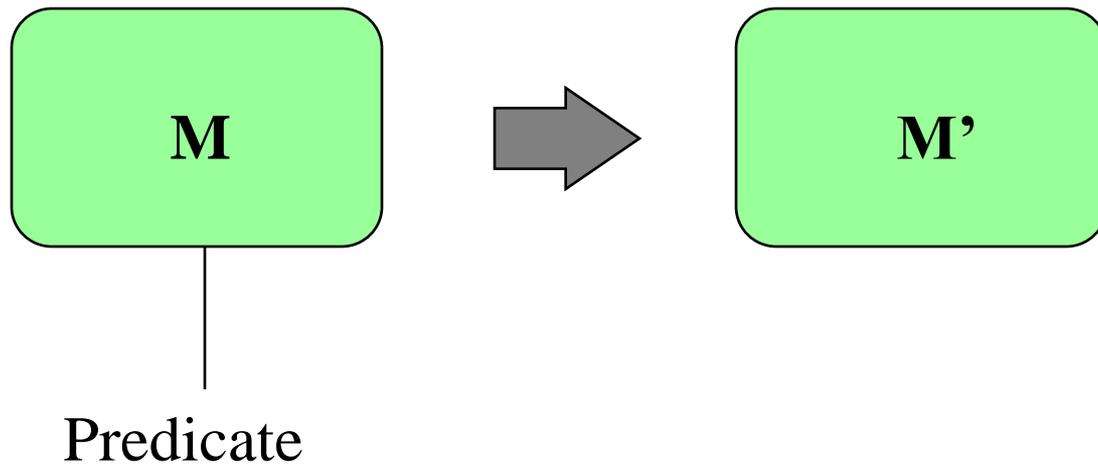
# A Mode Switch

---



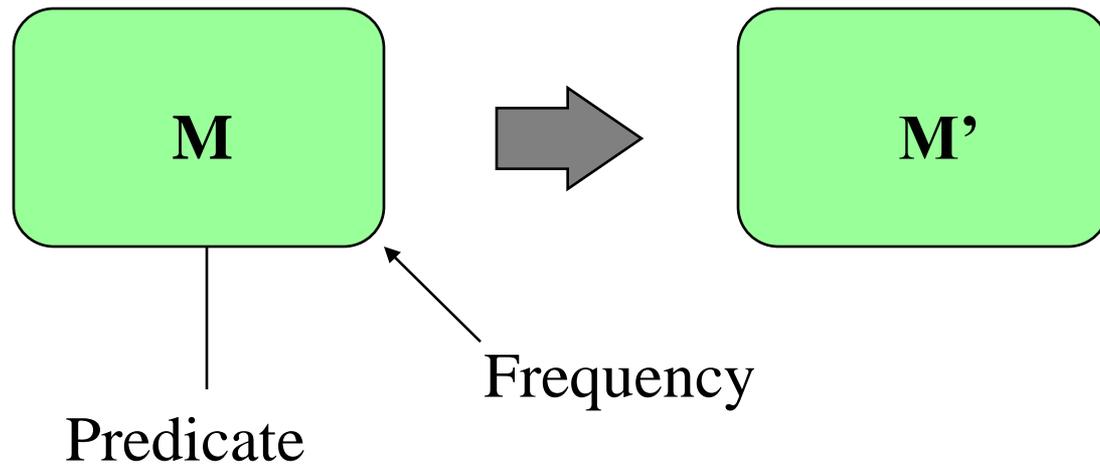
# Abstract Syntax of a Mode Switch

---



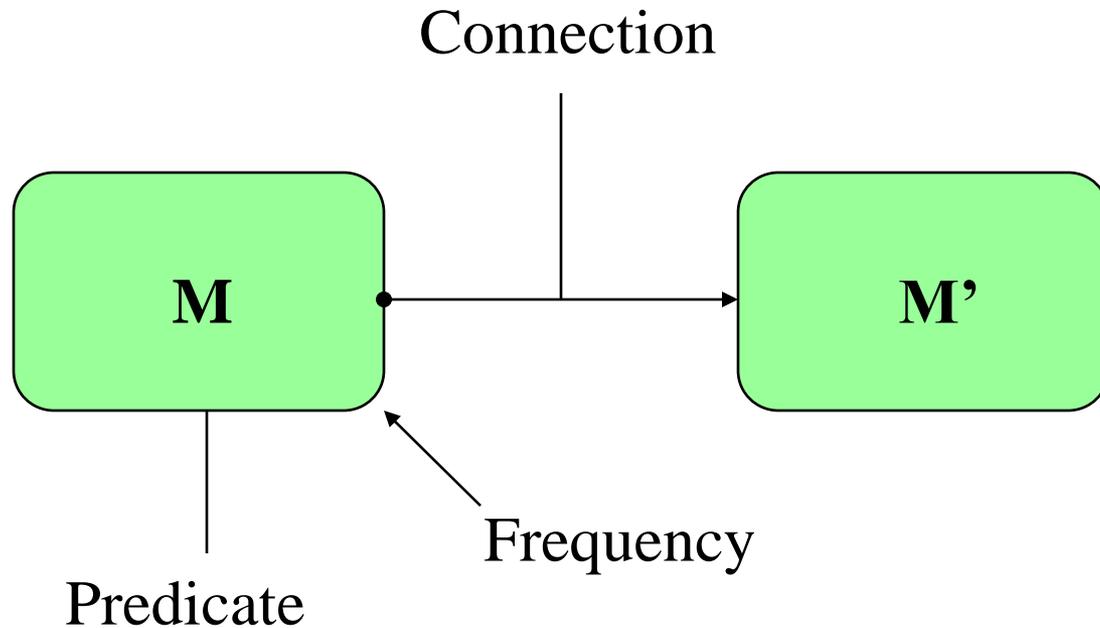
# Abstract Syntax of a Mode Switch

---



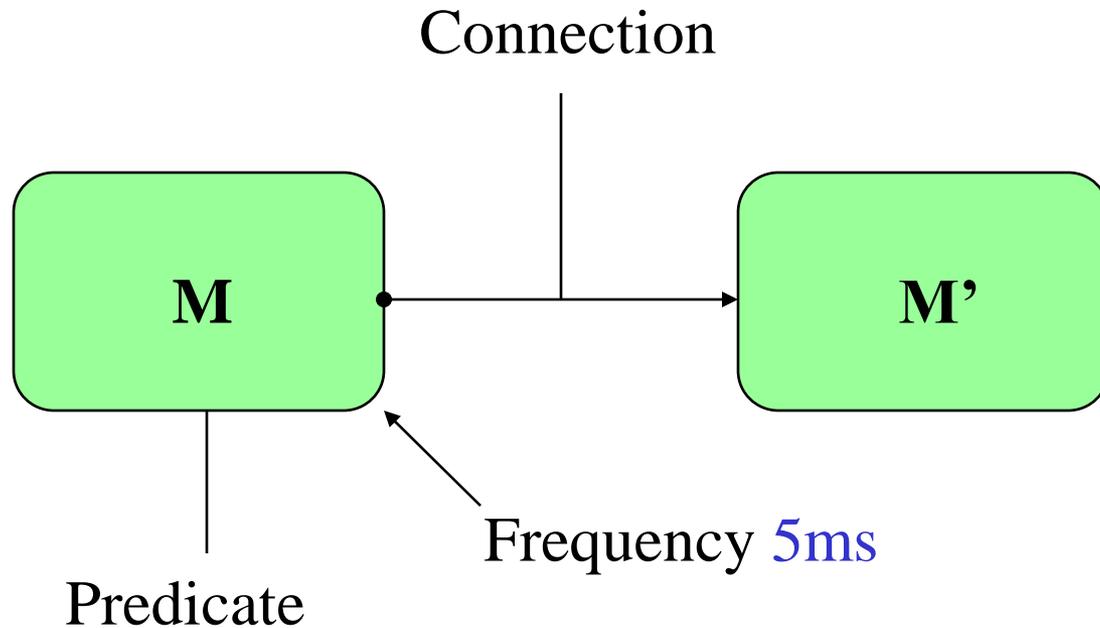
# Abstract Syntax of a Mode Switch

---

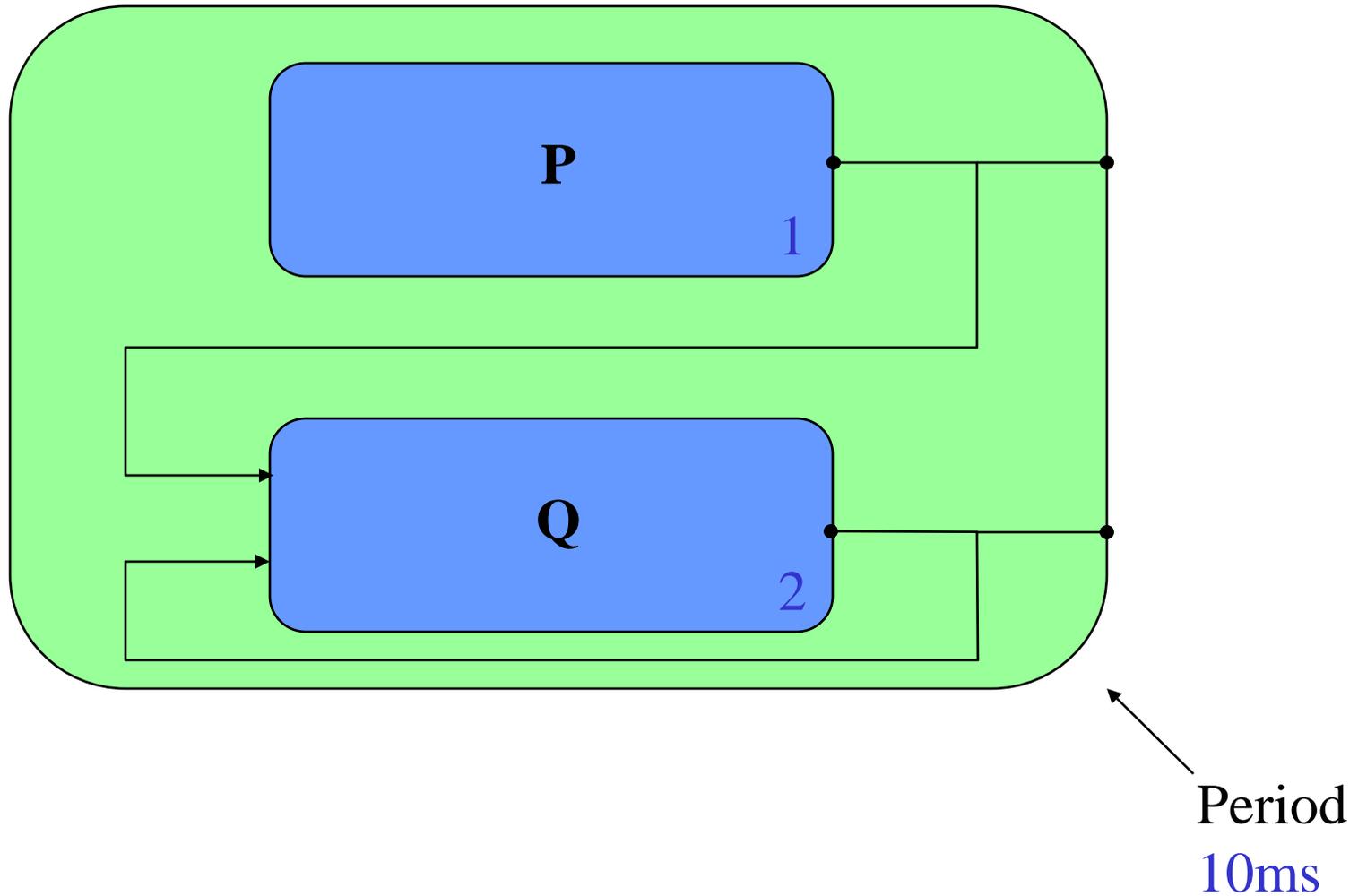


# Abstract Syntax of a Mode Switch

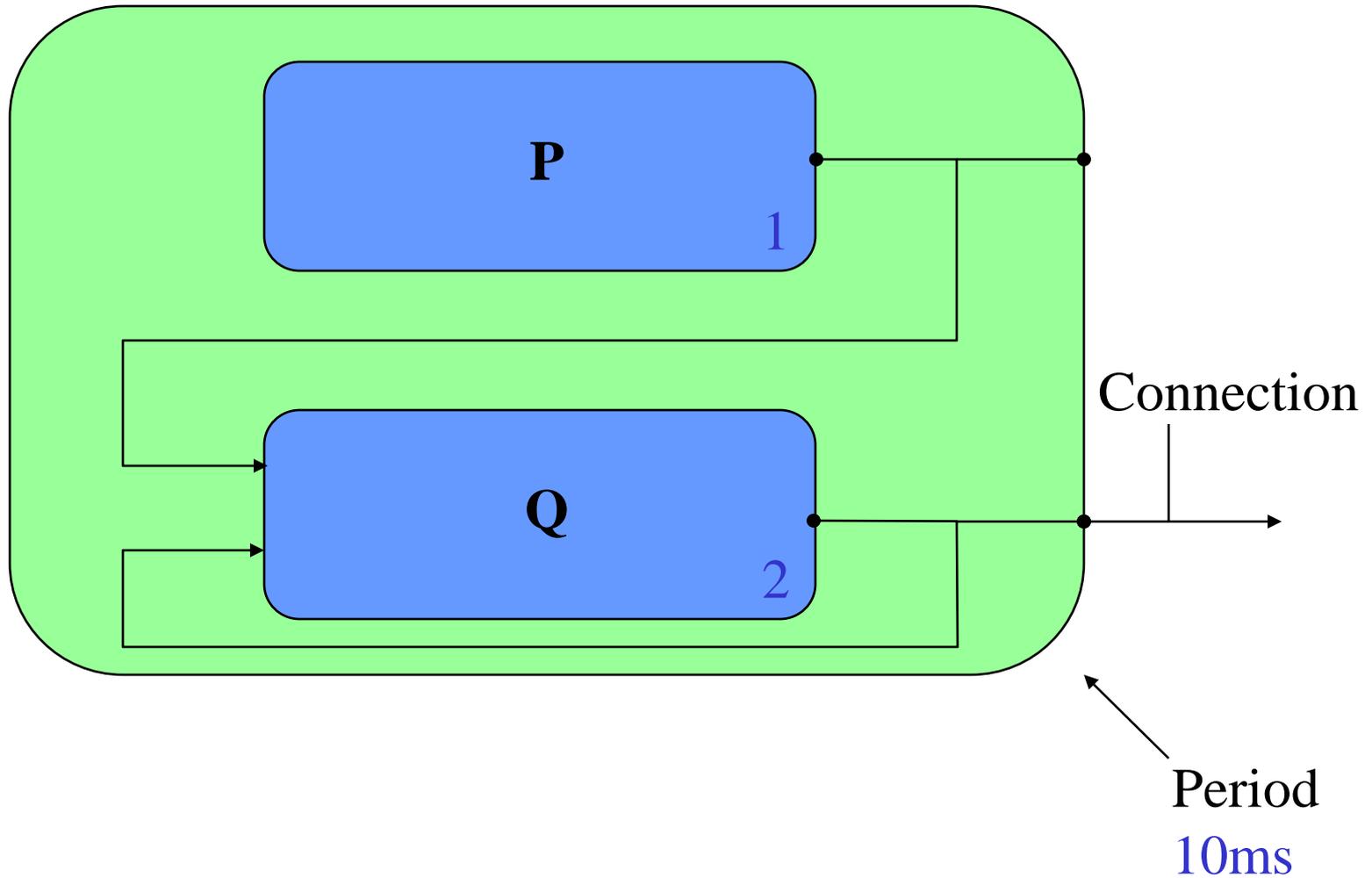
---



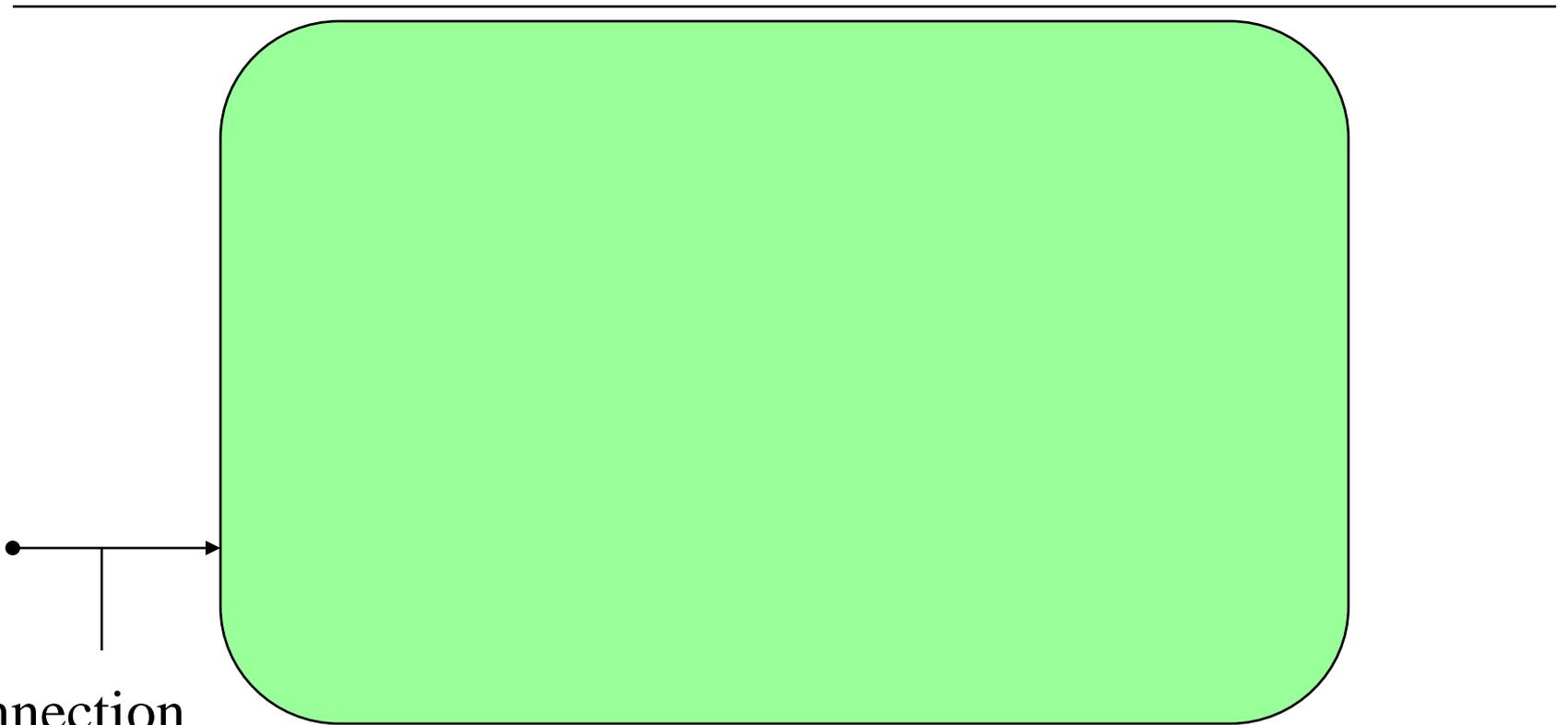
# Mode M



# Mode M

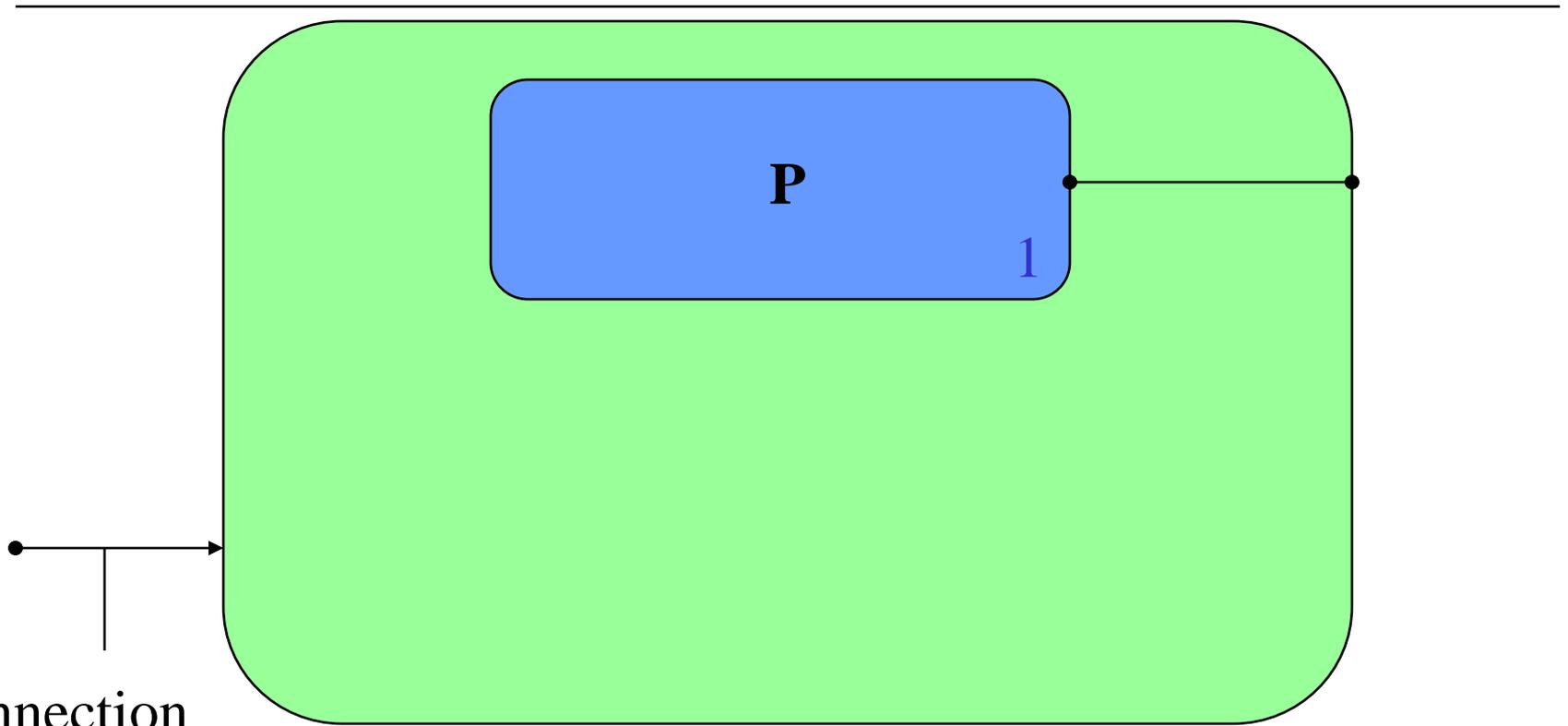


# Mode M'

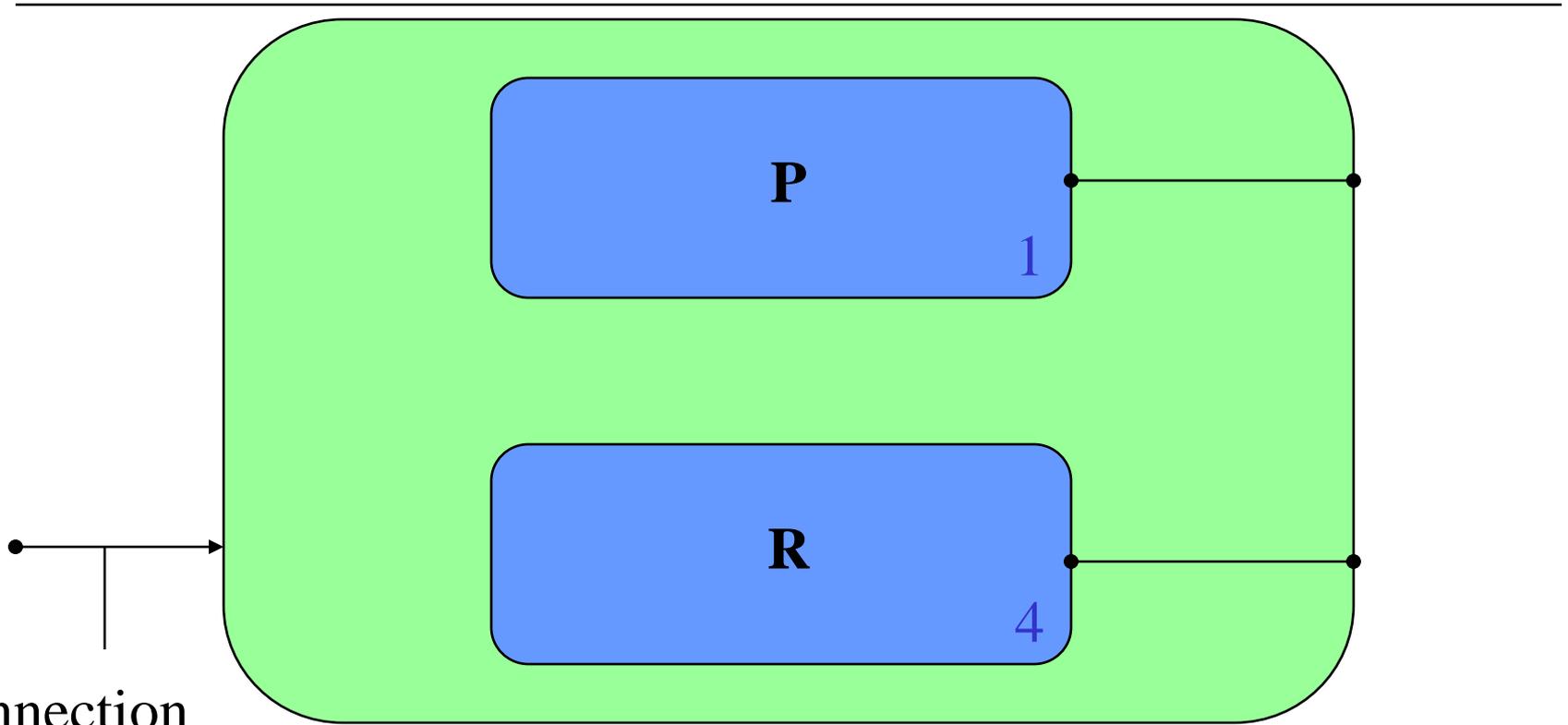


Connection

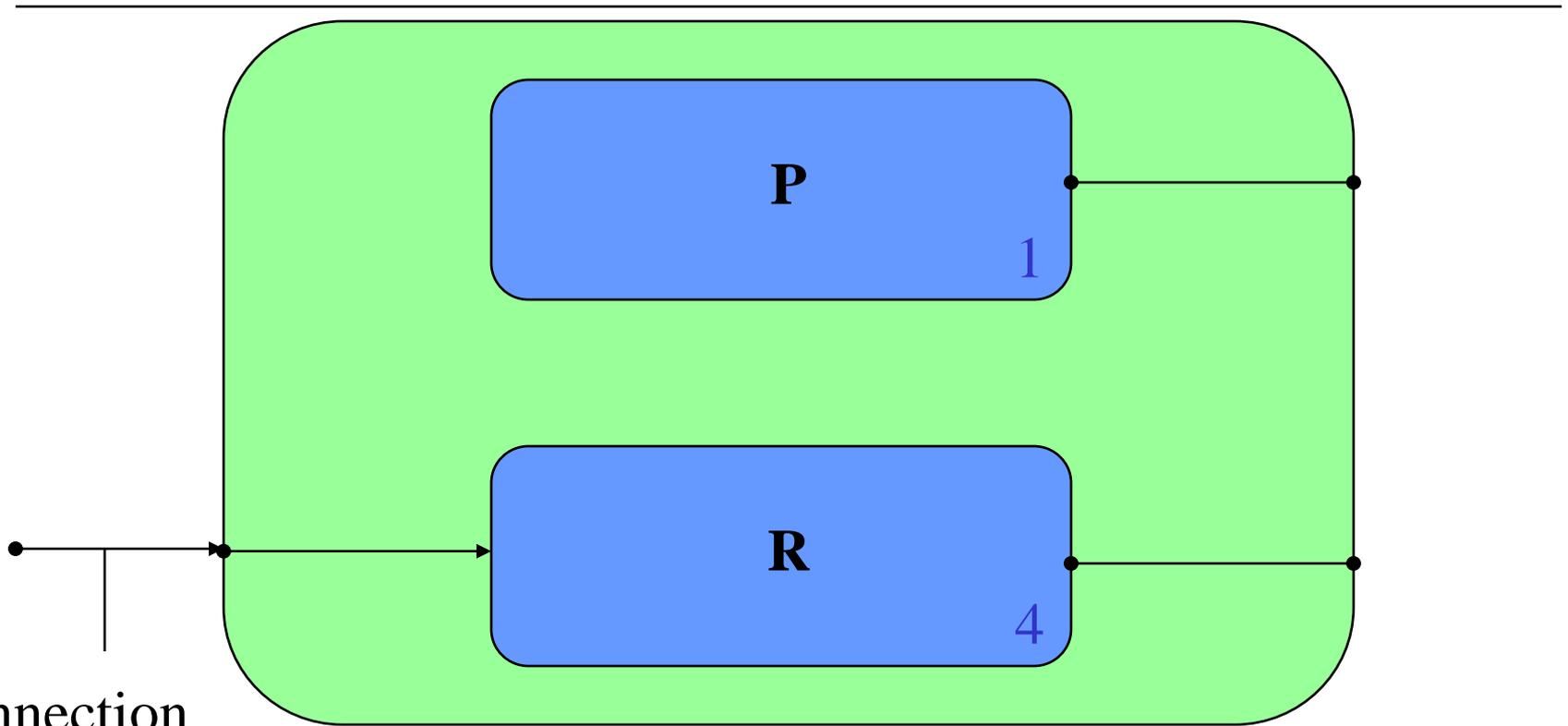
# Mode M'



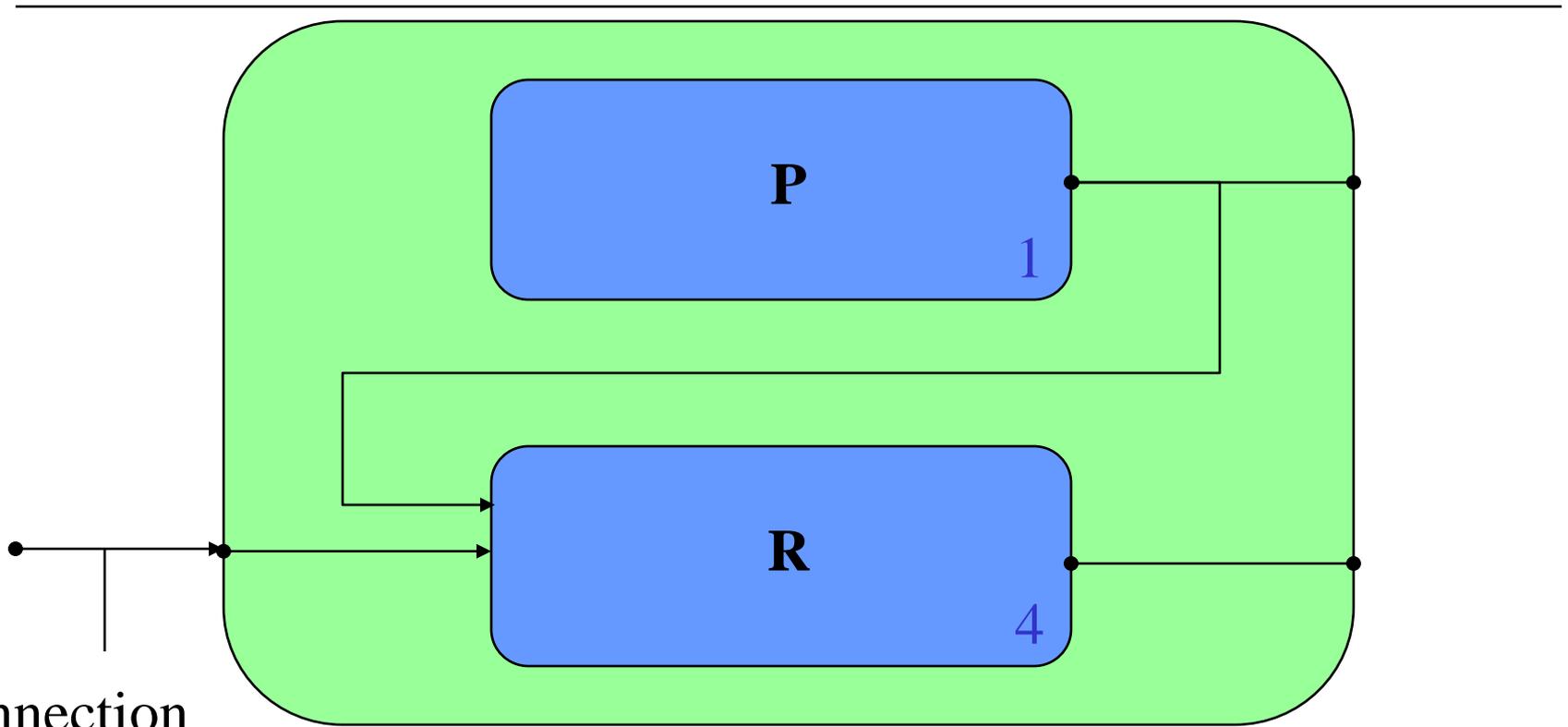
# Mode M'



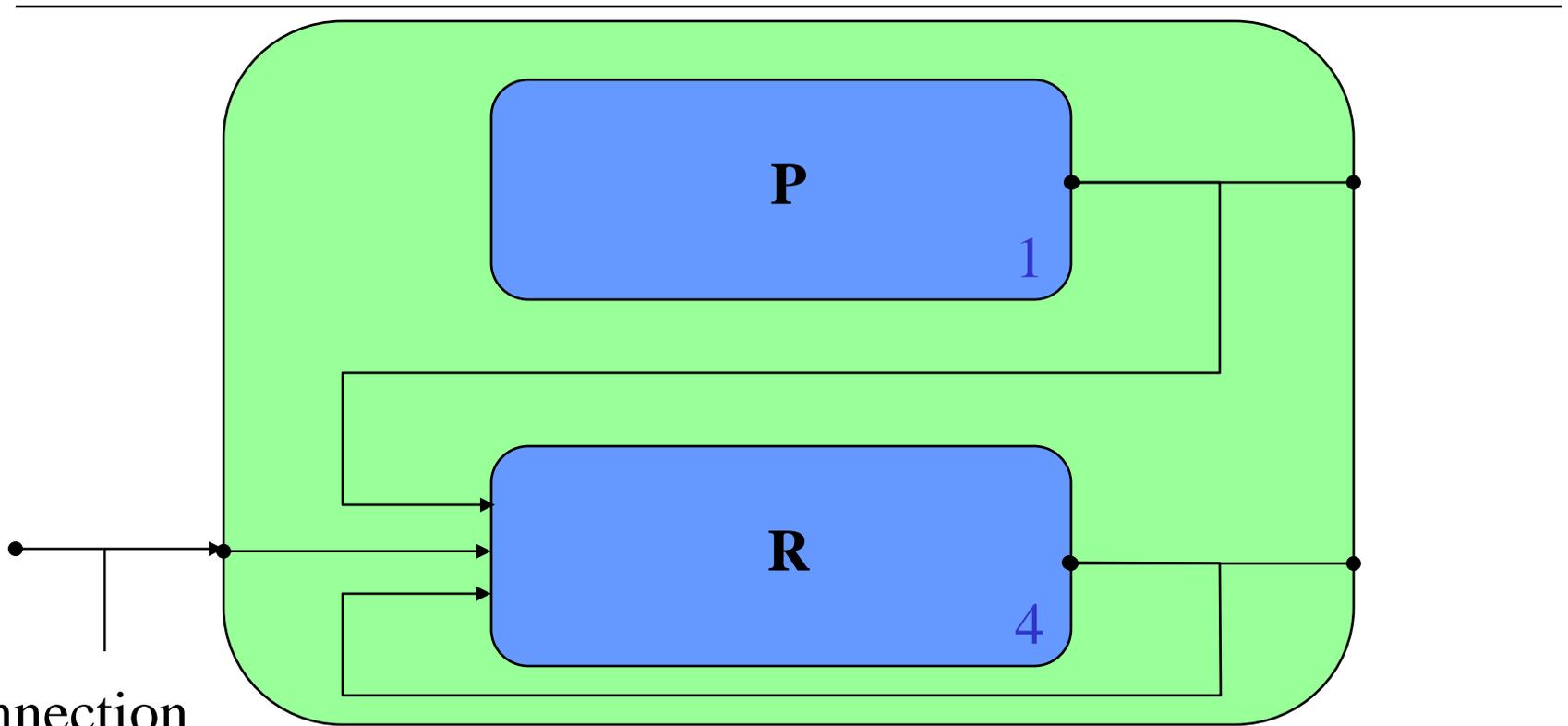
# Mode M'



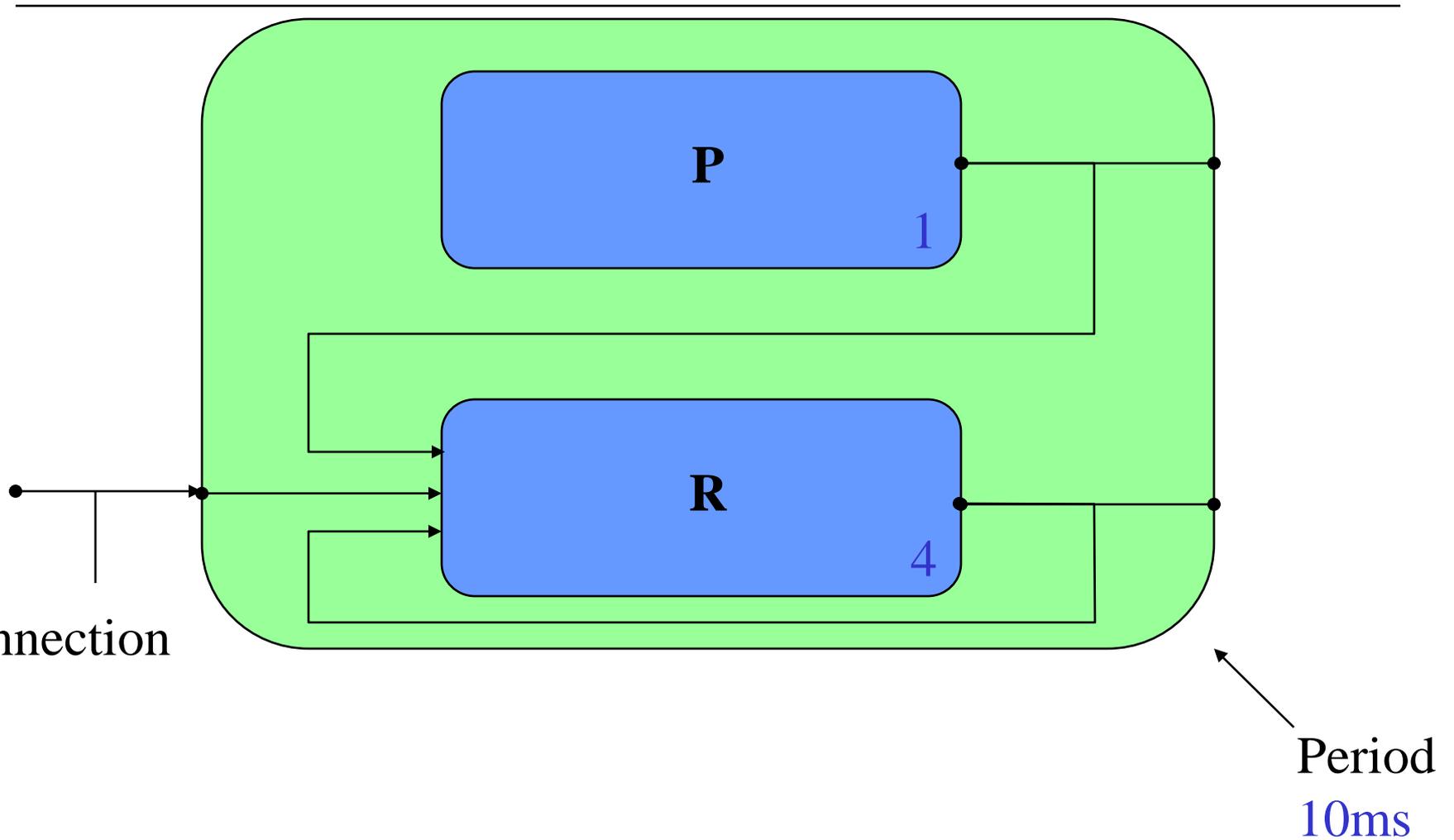
# Mode M'



# Mode M'



# Mode M'



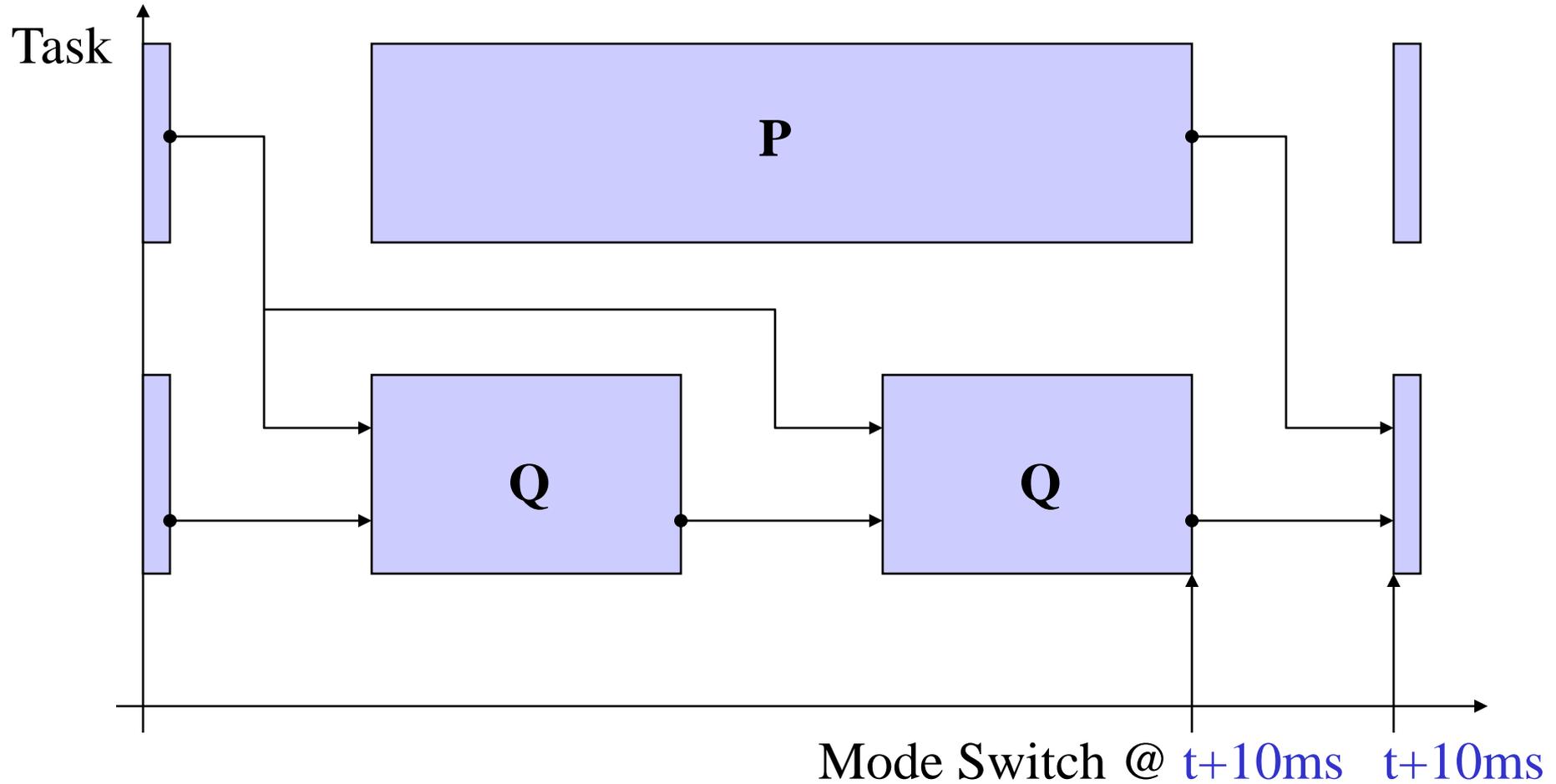
# Concrete Syntax

---

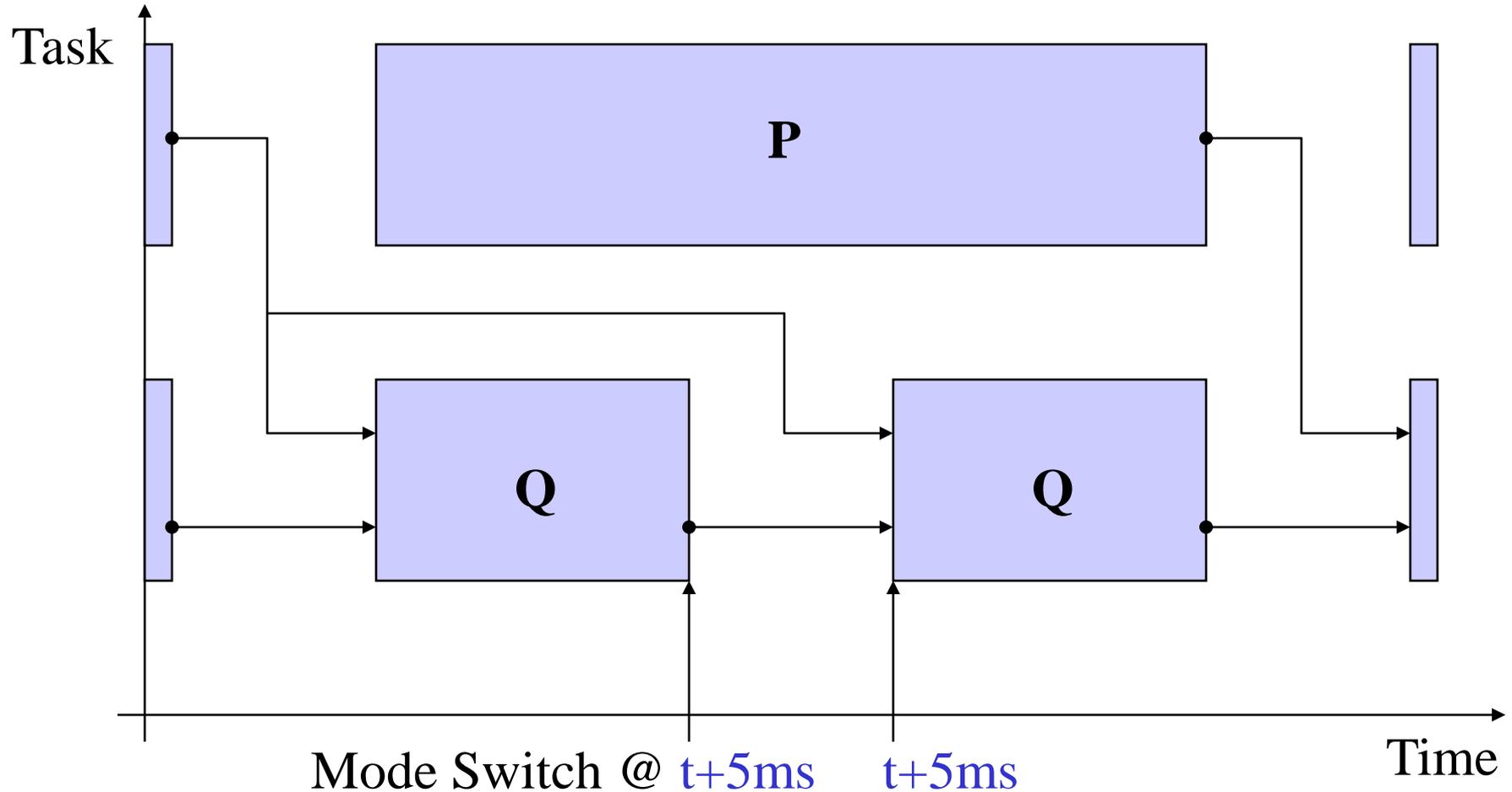
```
start m ( ) {  
  mode m ( ) period 10 ms entryfreq 1 {  
    taskfreq 1 do int x = P ( ) ;  
    taskfreq 2 do int y = Q ( x, y ) ;  
  
    exitfreq 2 if y = 5 then m' ( y ) ;  
  }  
  mode m' ( int z ) period 10 ms entryfreq 2 {  
    taskfreq 1 do int x = P ( ) ;  
    taskfreq 4 do int u = R ( x, z ) ;  
  }  
}
```



# Semantics of a Mode Switch

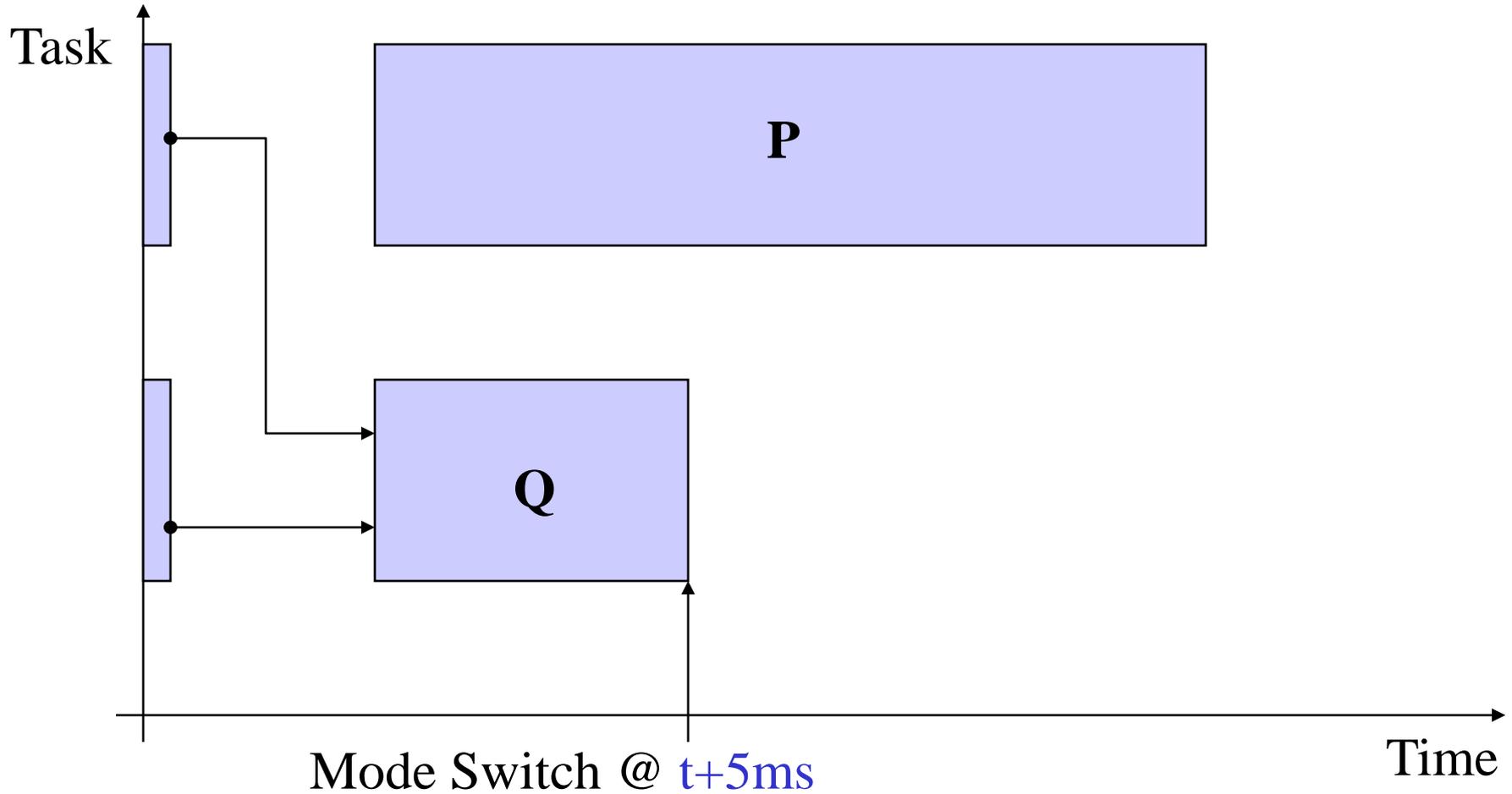


# Semantics of a Mode Switch



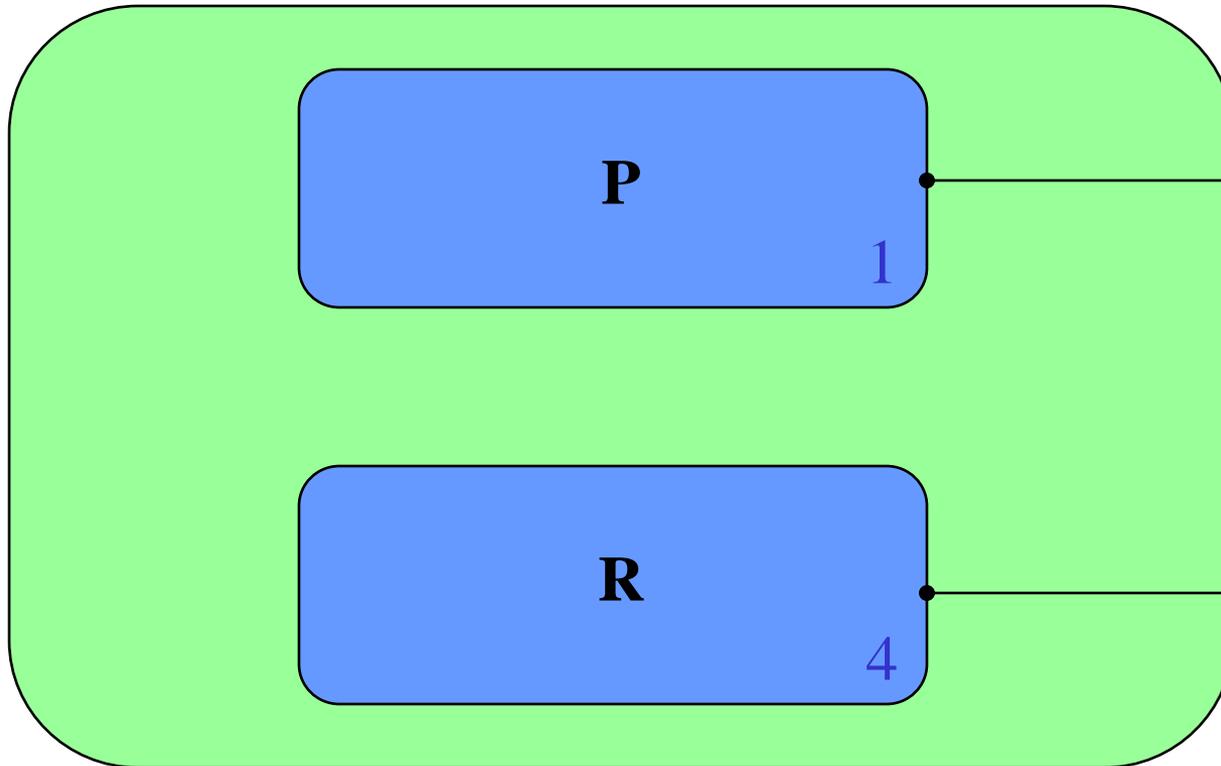
# Semantics of a Mode Switch

---



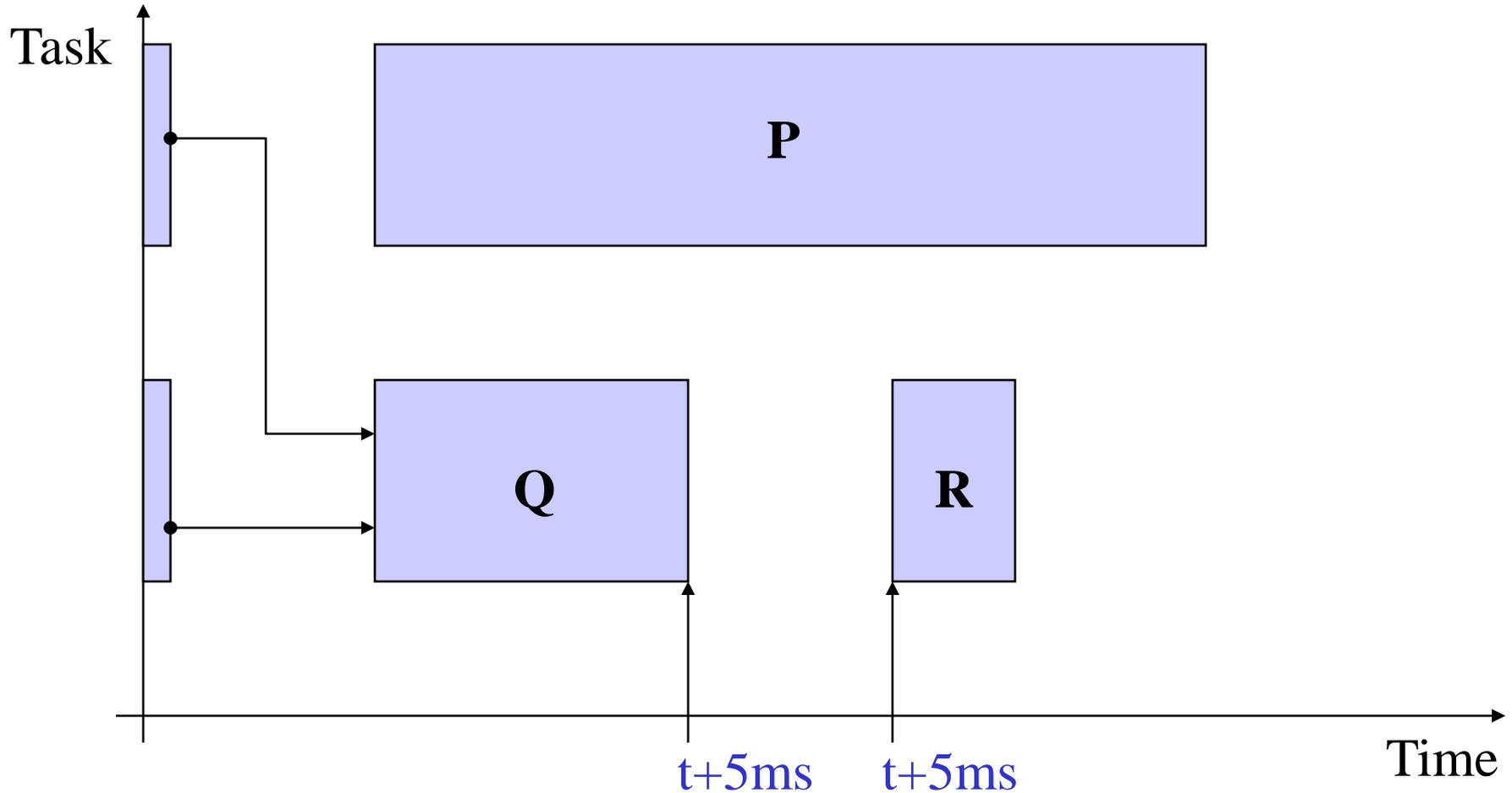
# Mode M'

---



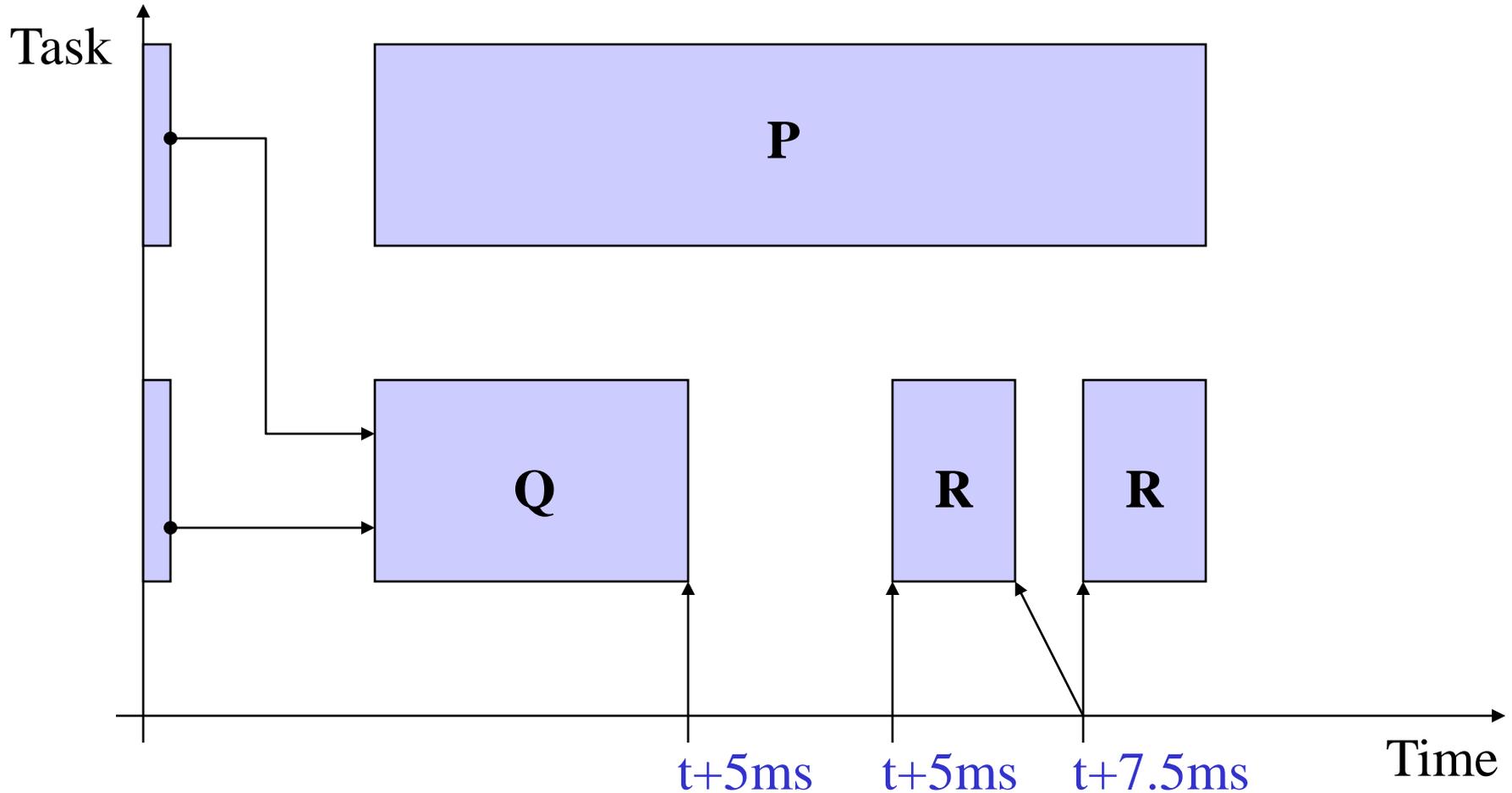
# Semantics of a Mode Switch

---

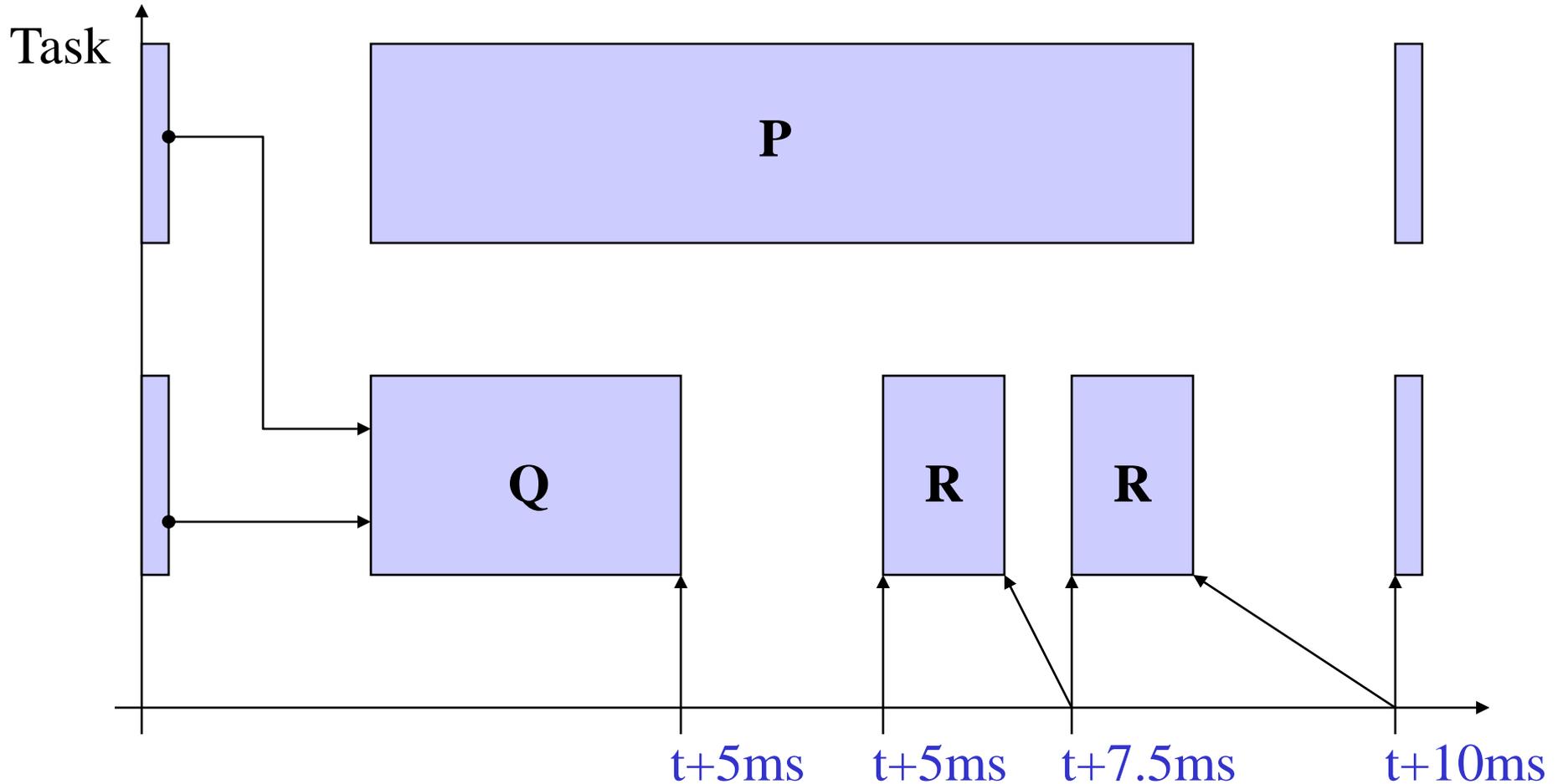


# Semantics of a Mode Switch

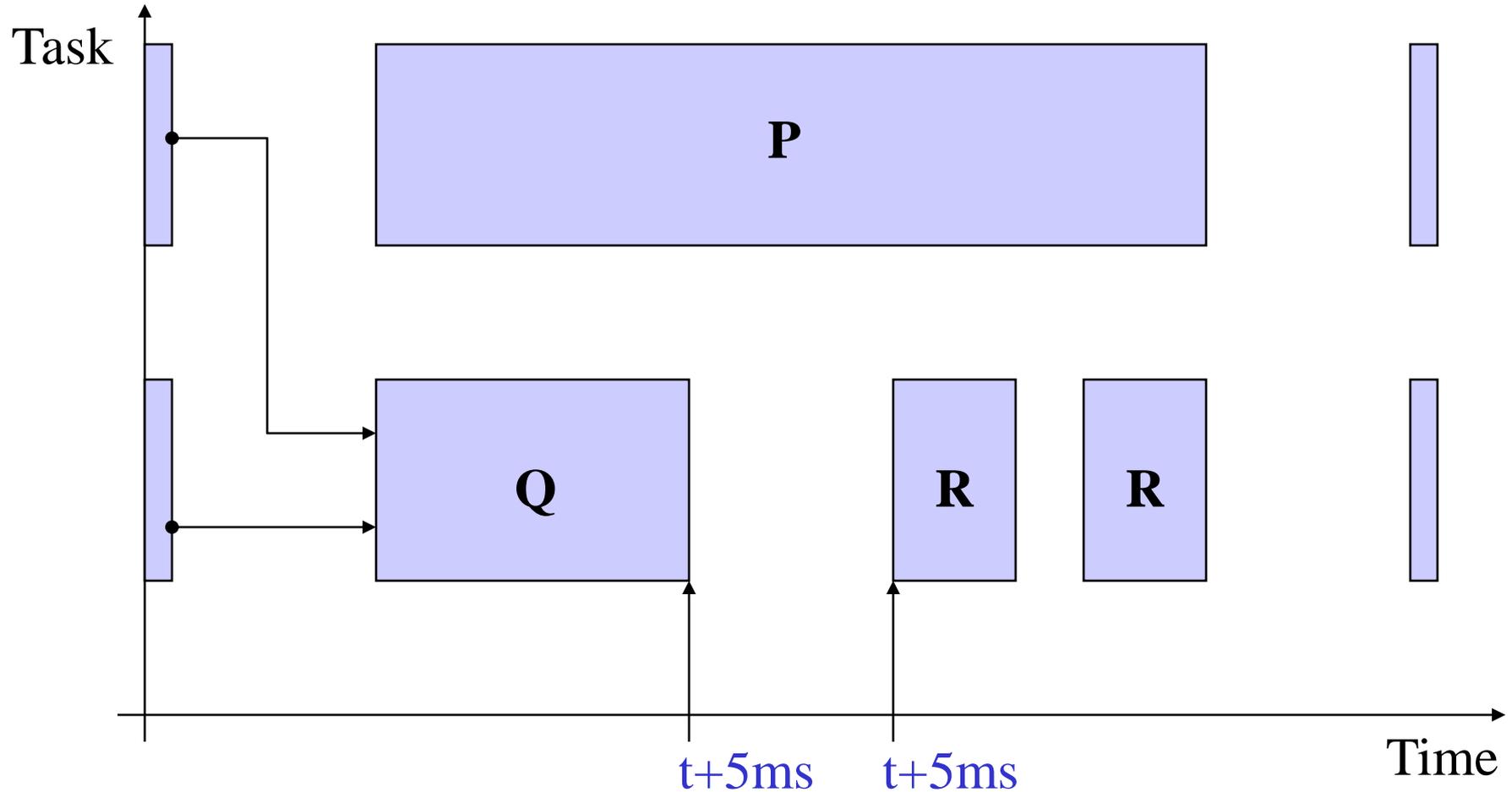
---



# Semantics of a Mode Switch

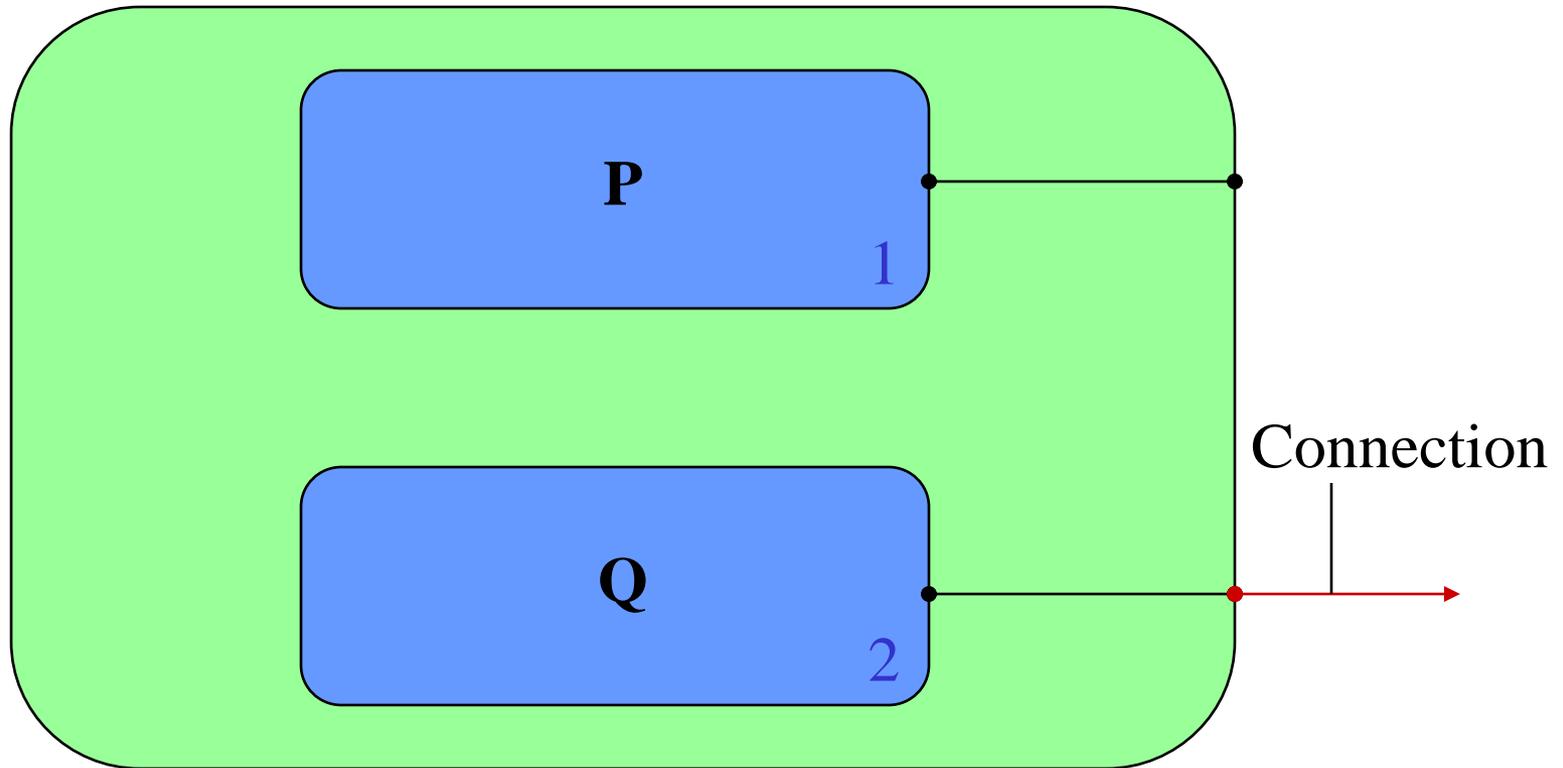


# Semantics of a Mode Switch

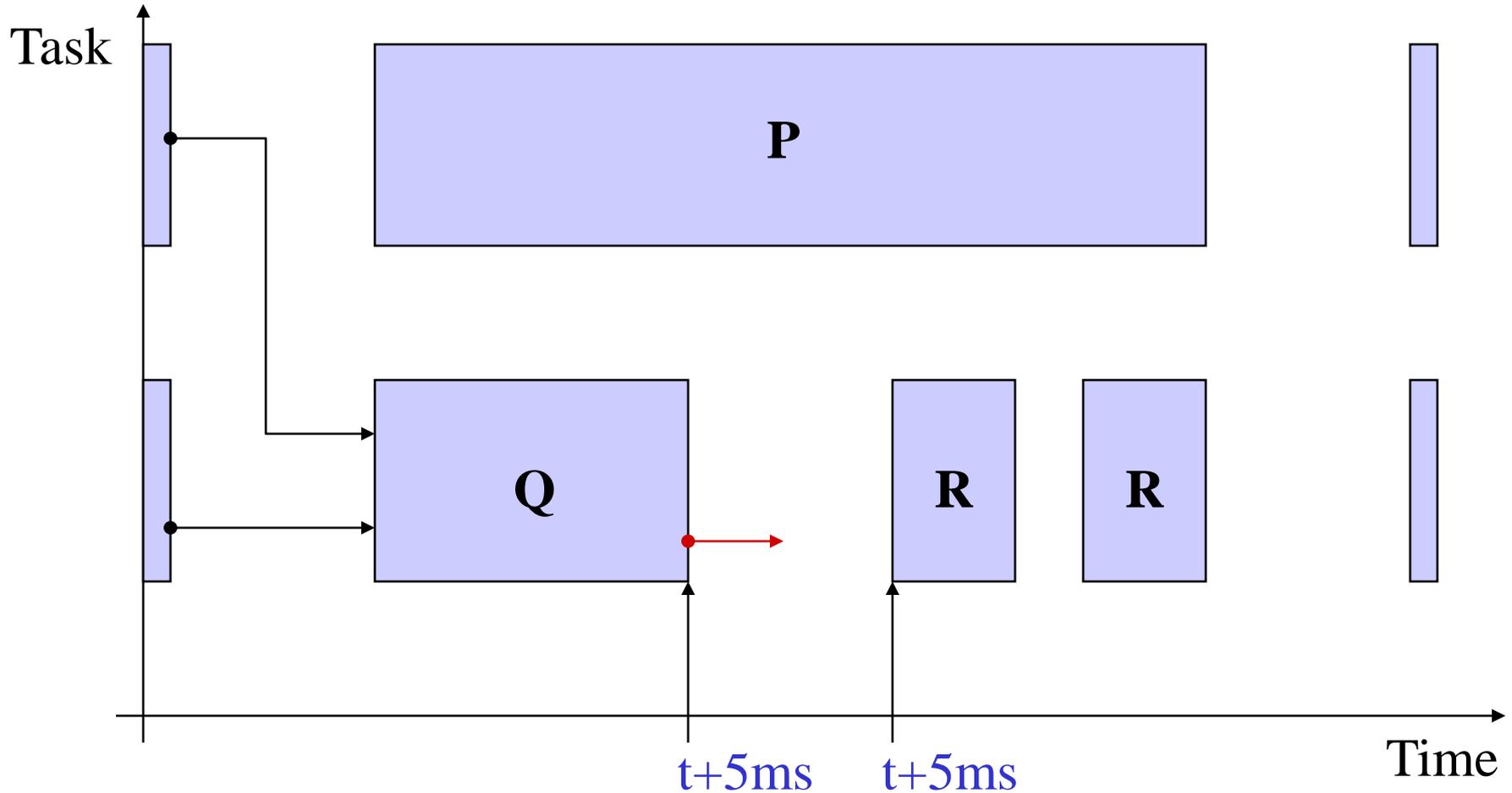


# Mode M

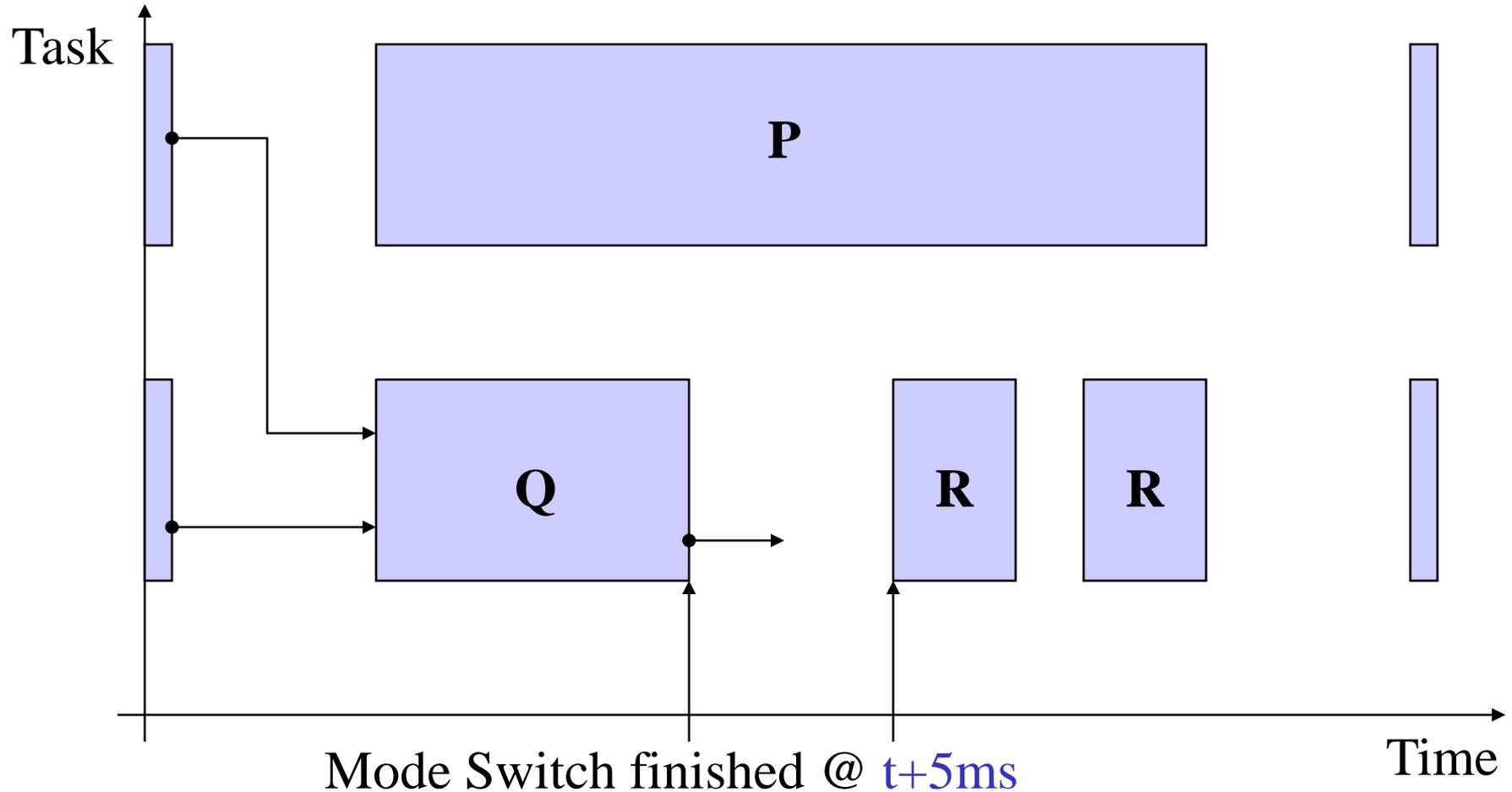
---



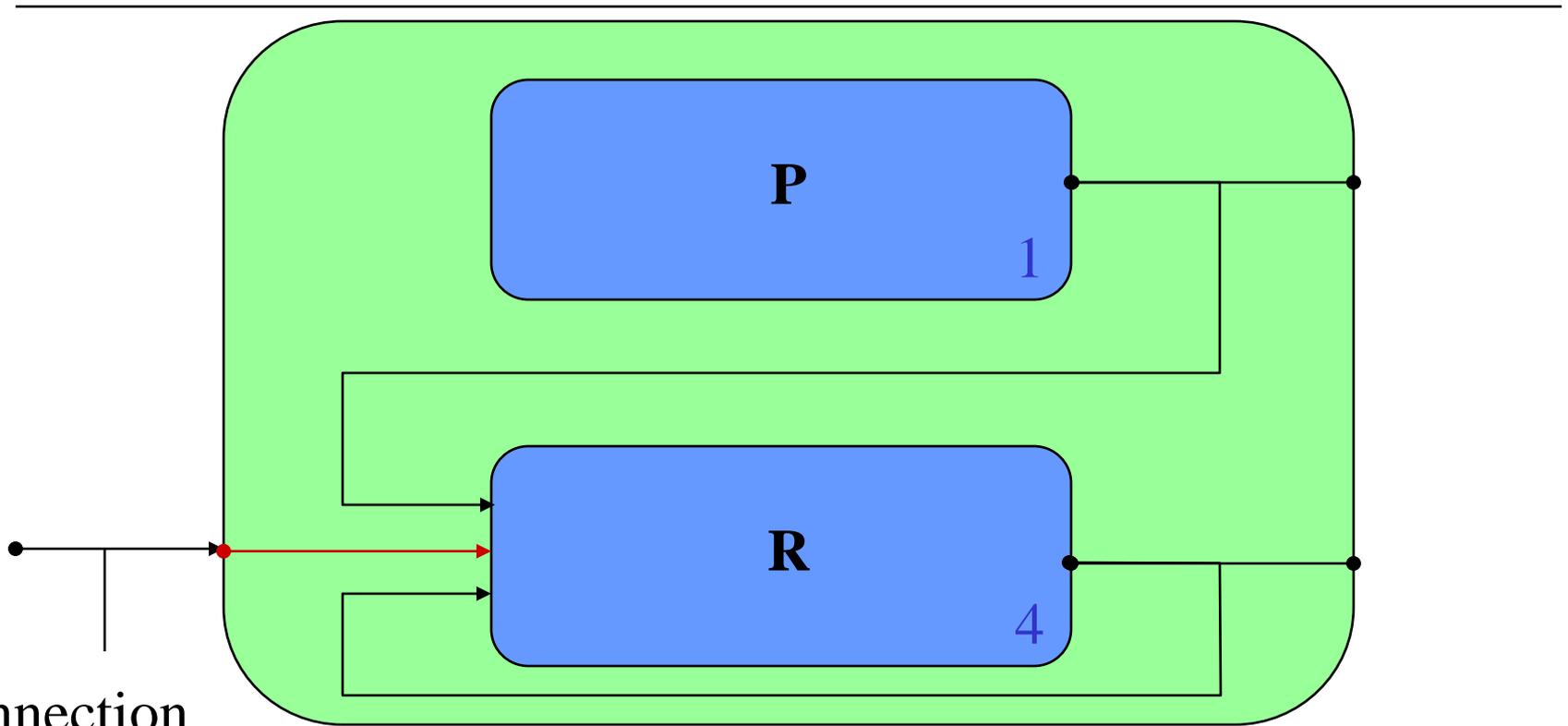
# Semantics of a Mode Switch



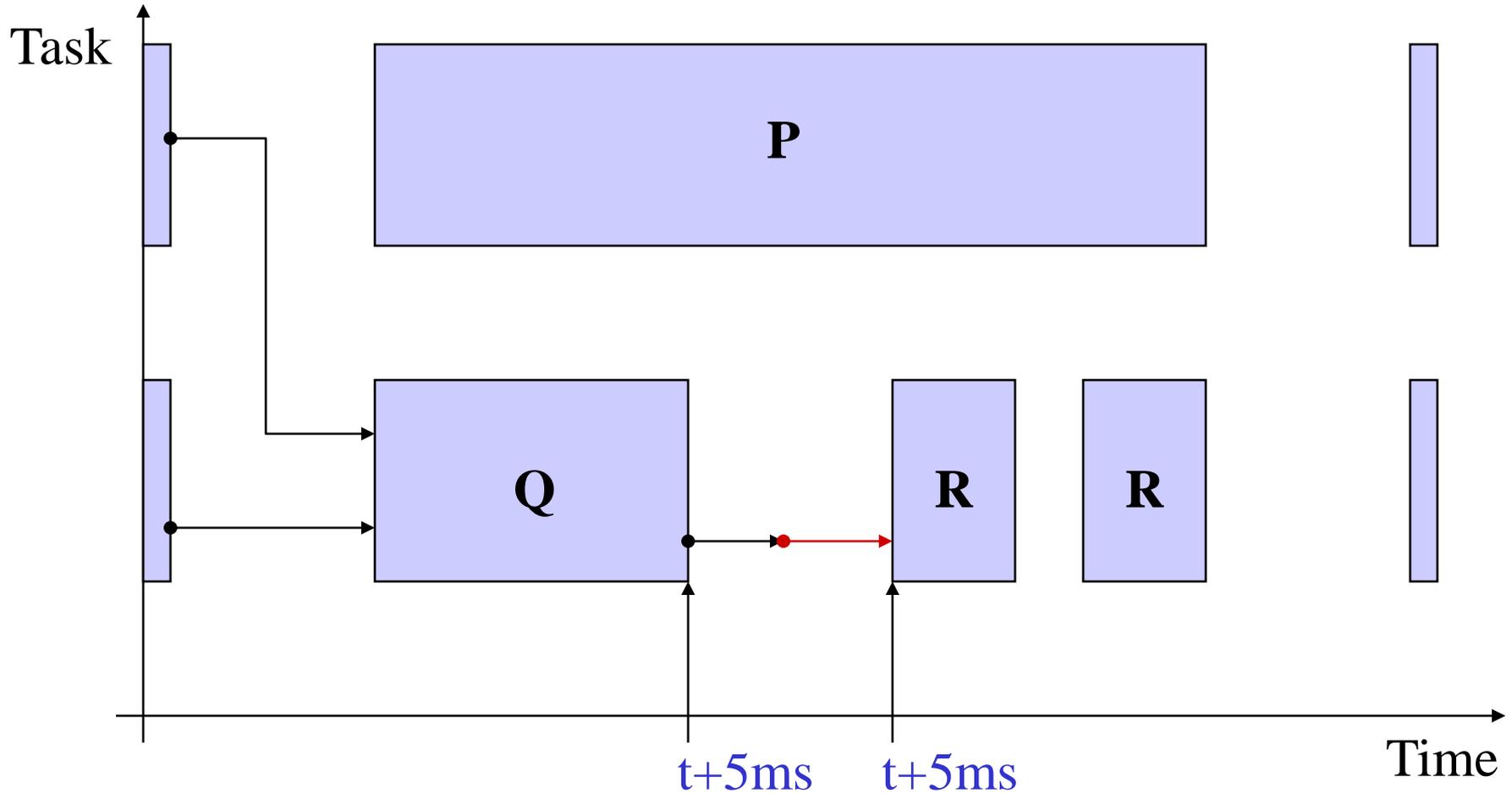
# Semantics of a Mode Switch



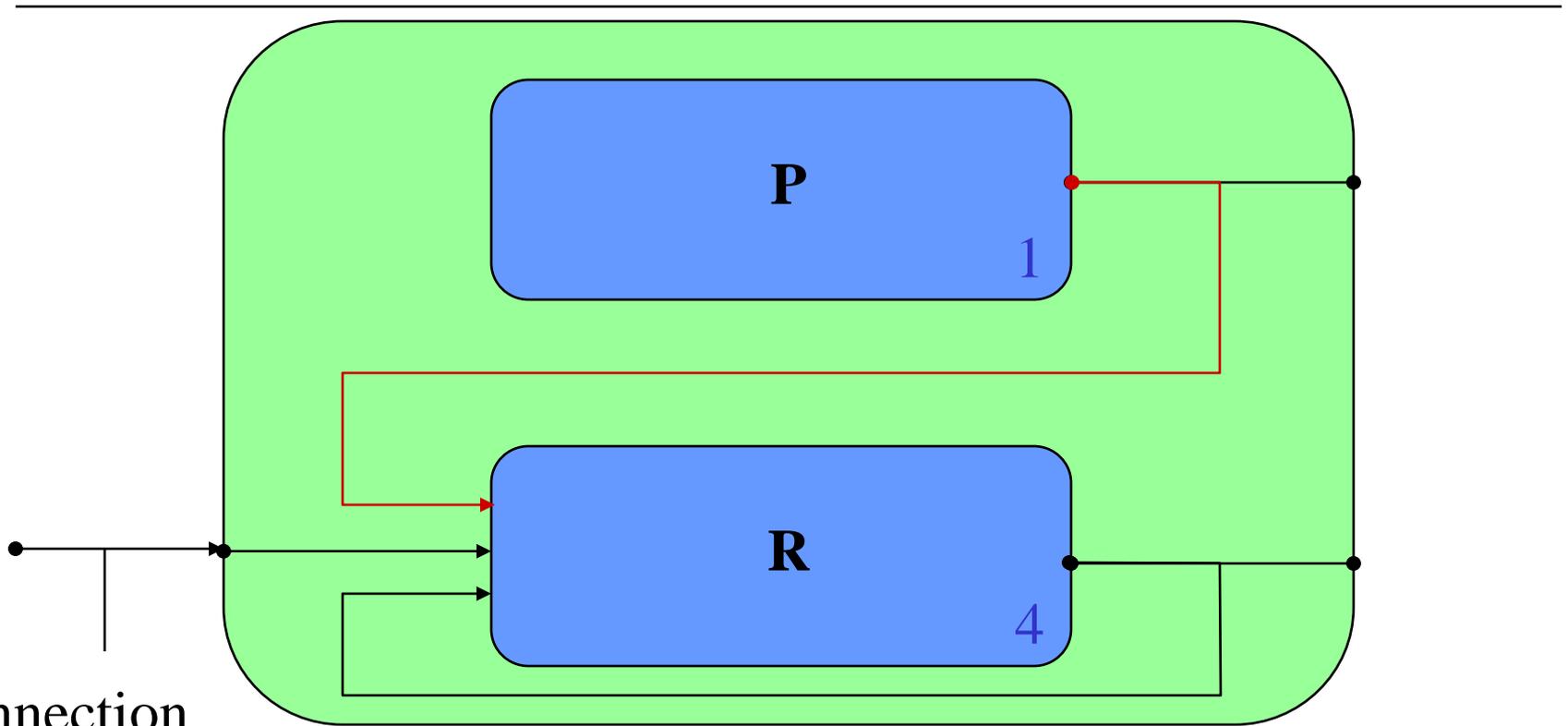
# Mode M'



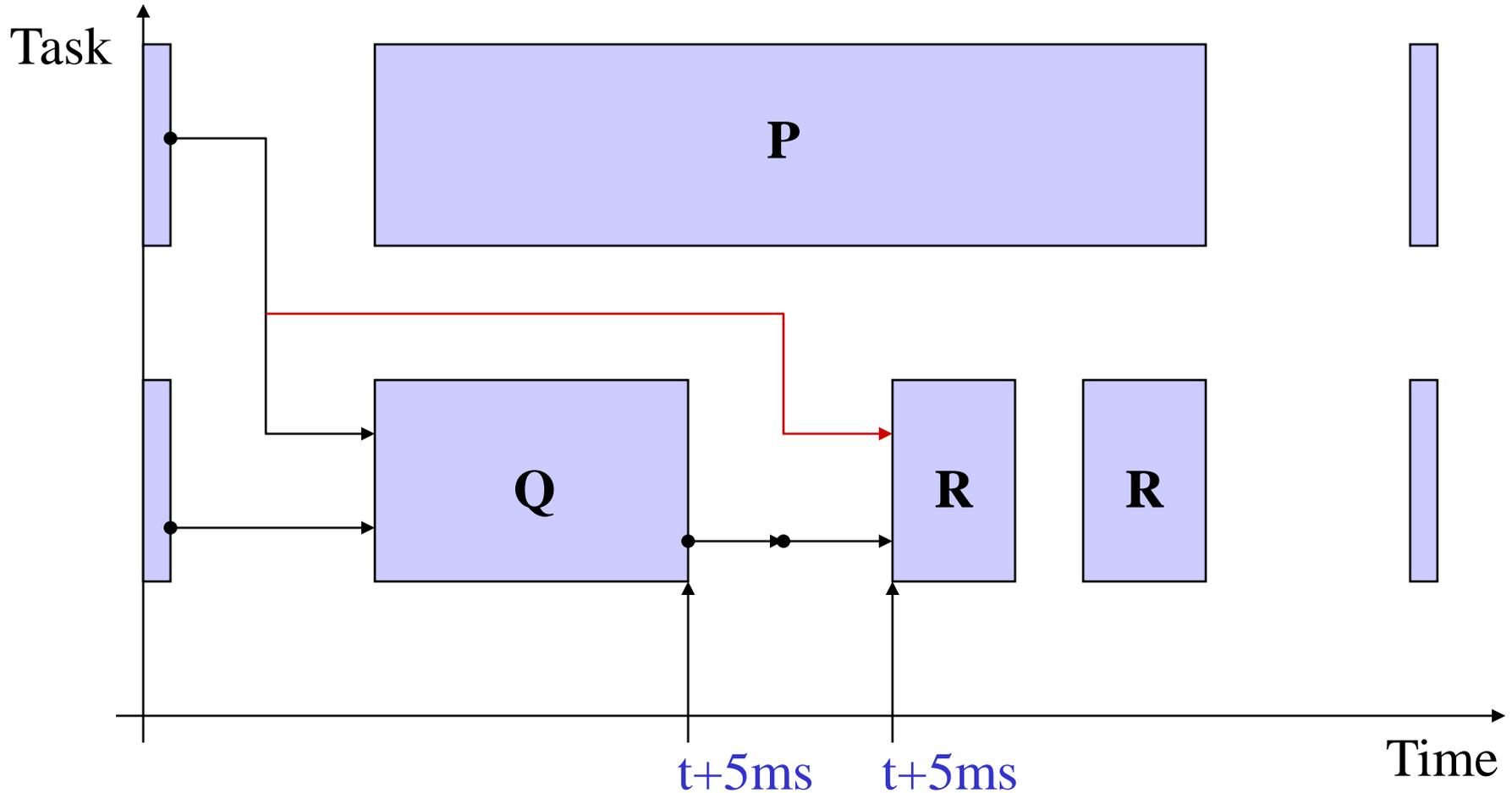
# Semantics of a Mode Switch



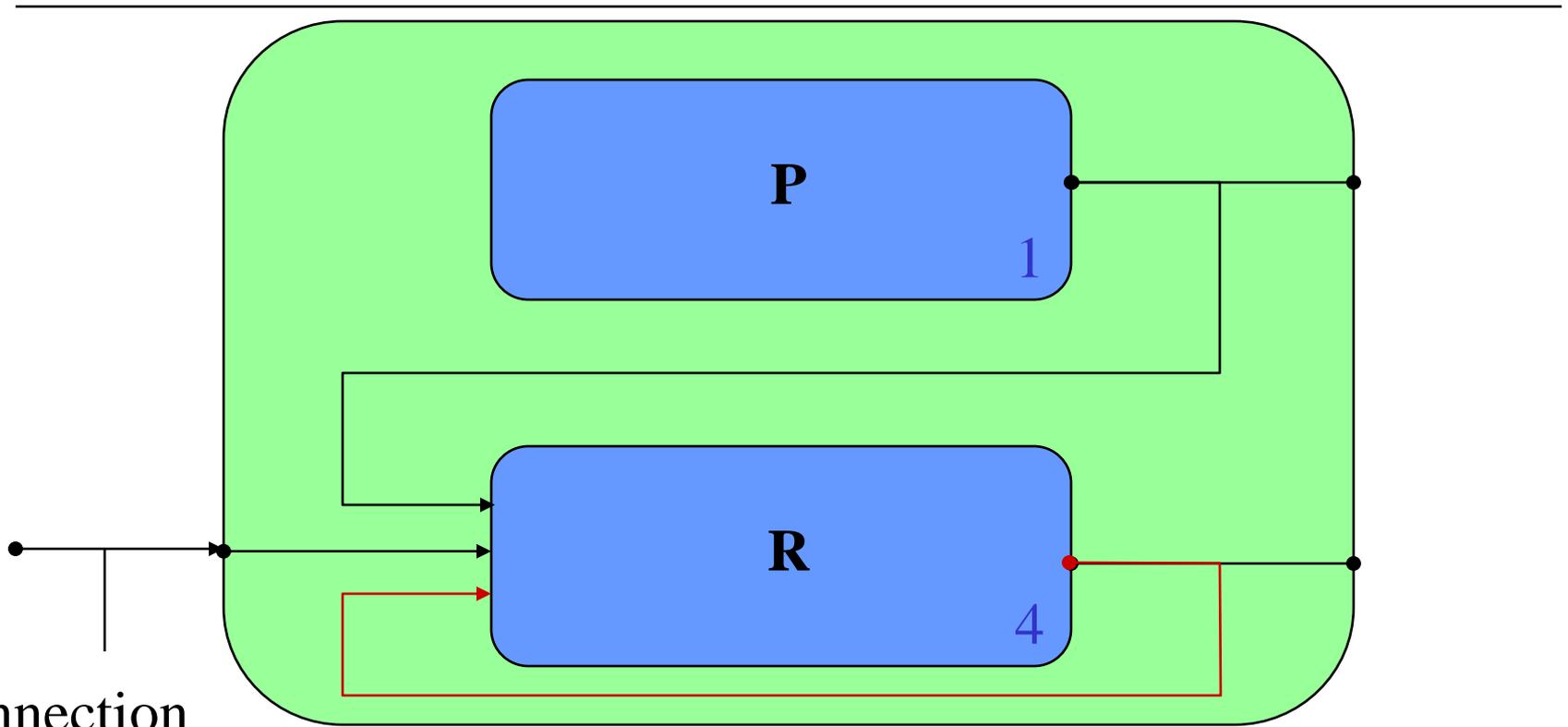
# Mode M'



# Semantics of a Mode Switch

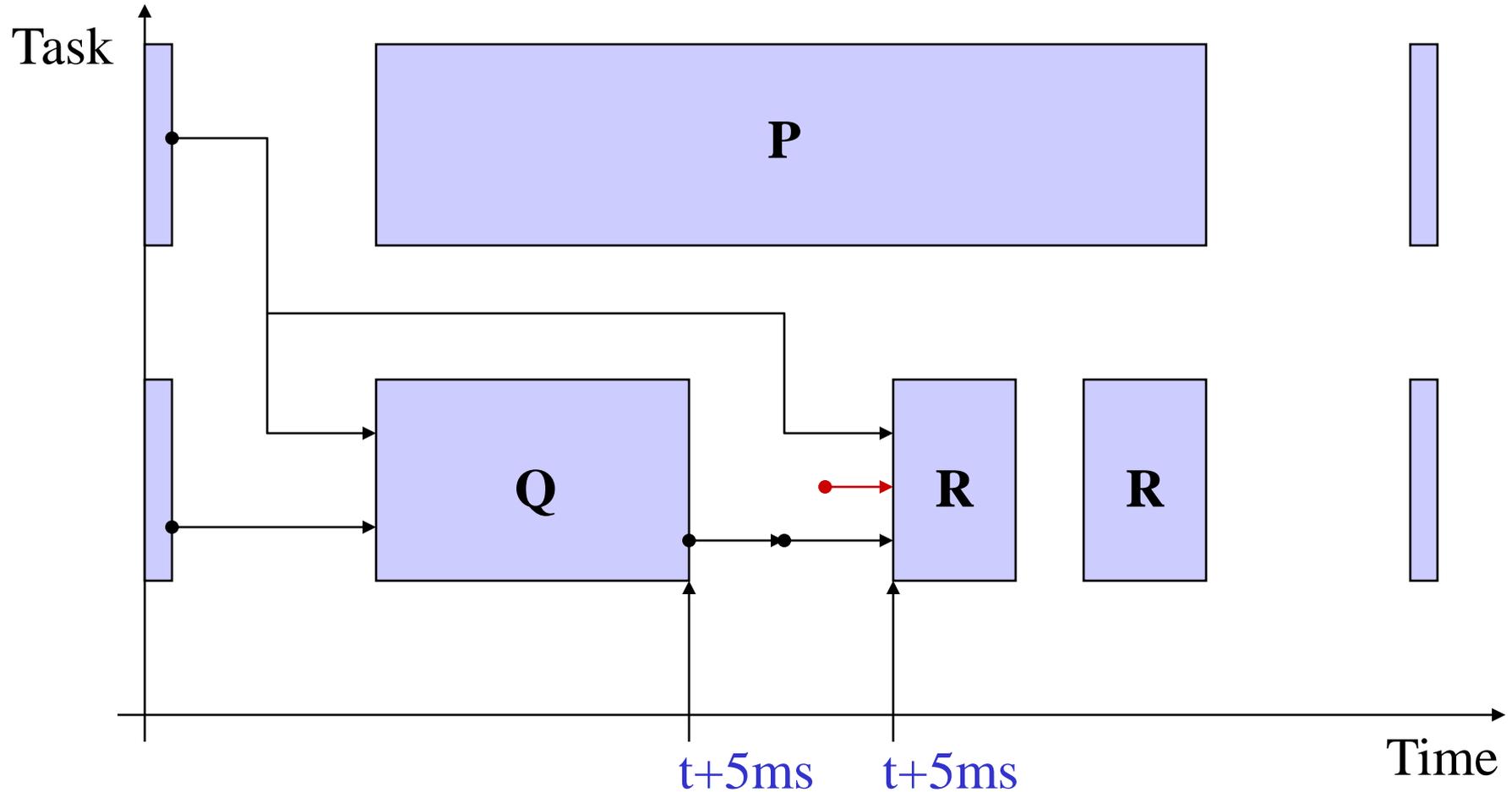


# Mode M'

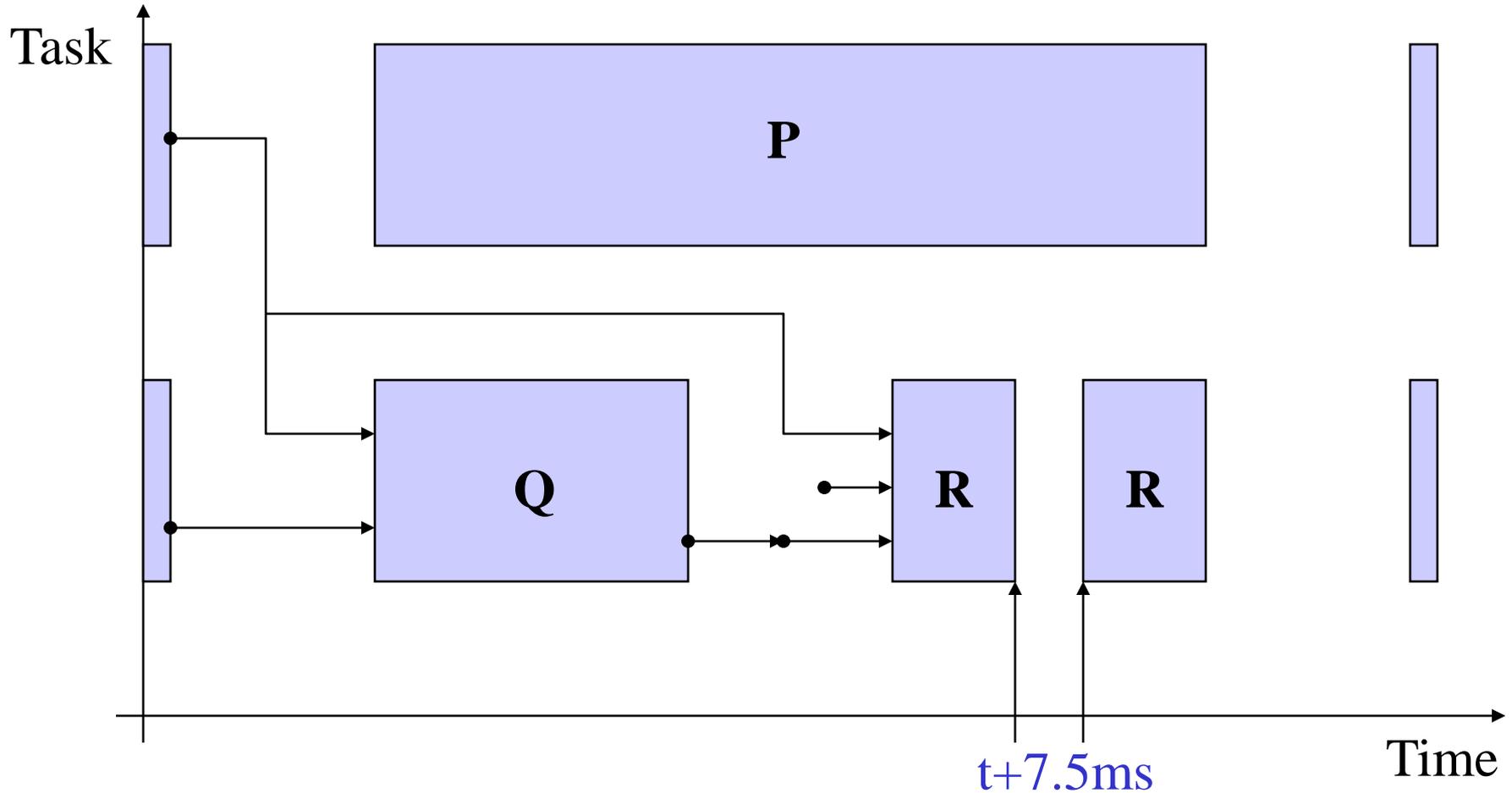


Connection

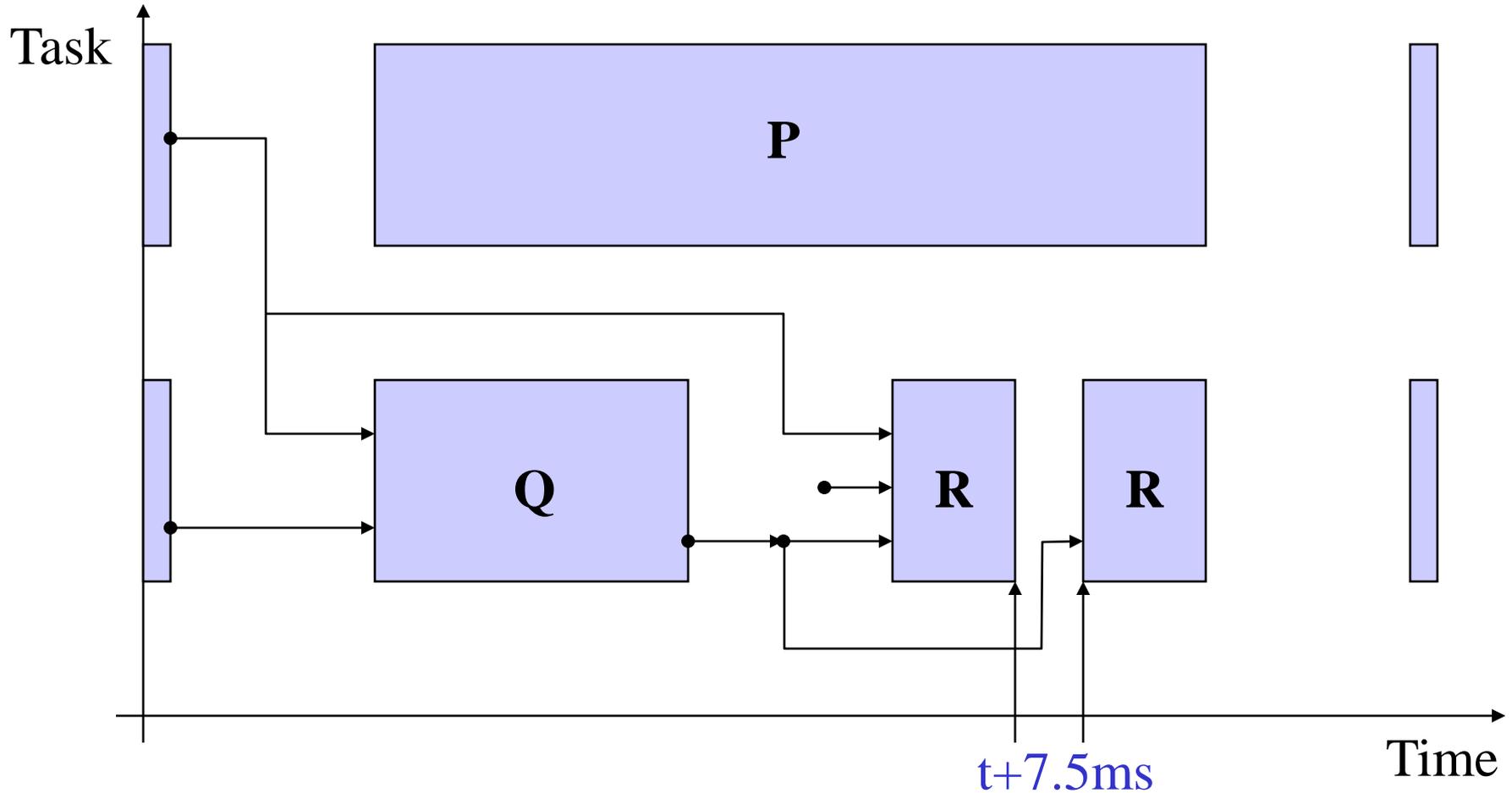
# Semantics of a Mode Switch



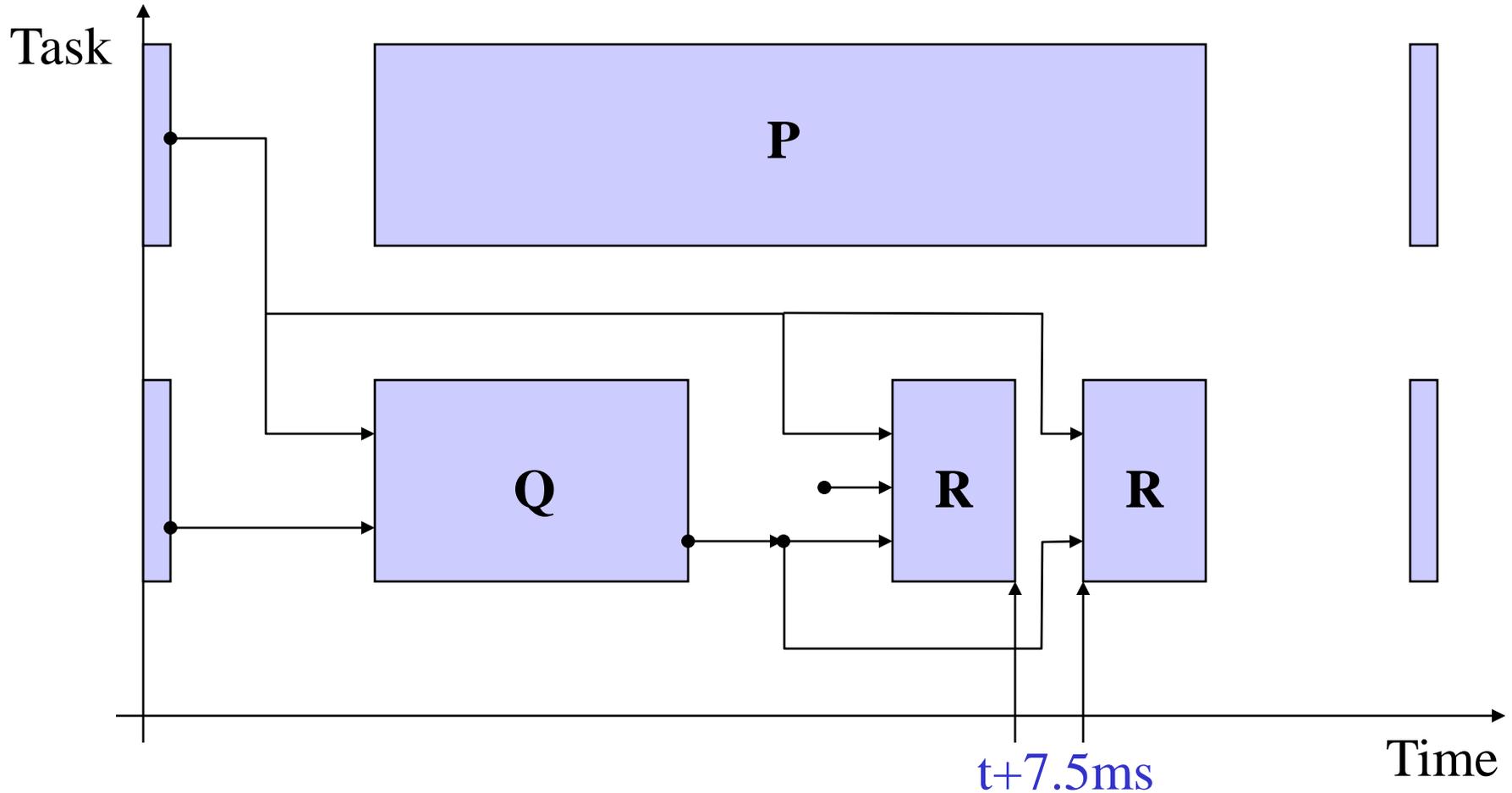
# Semantics of a Mode Switch



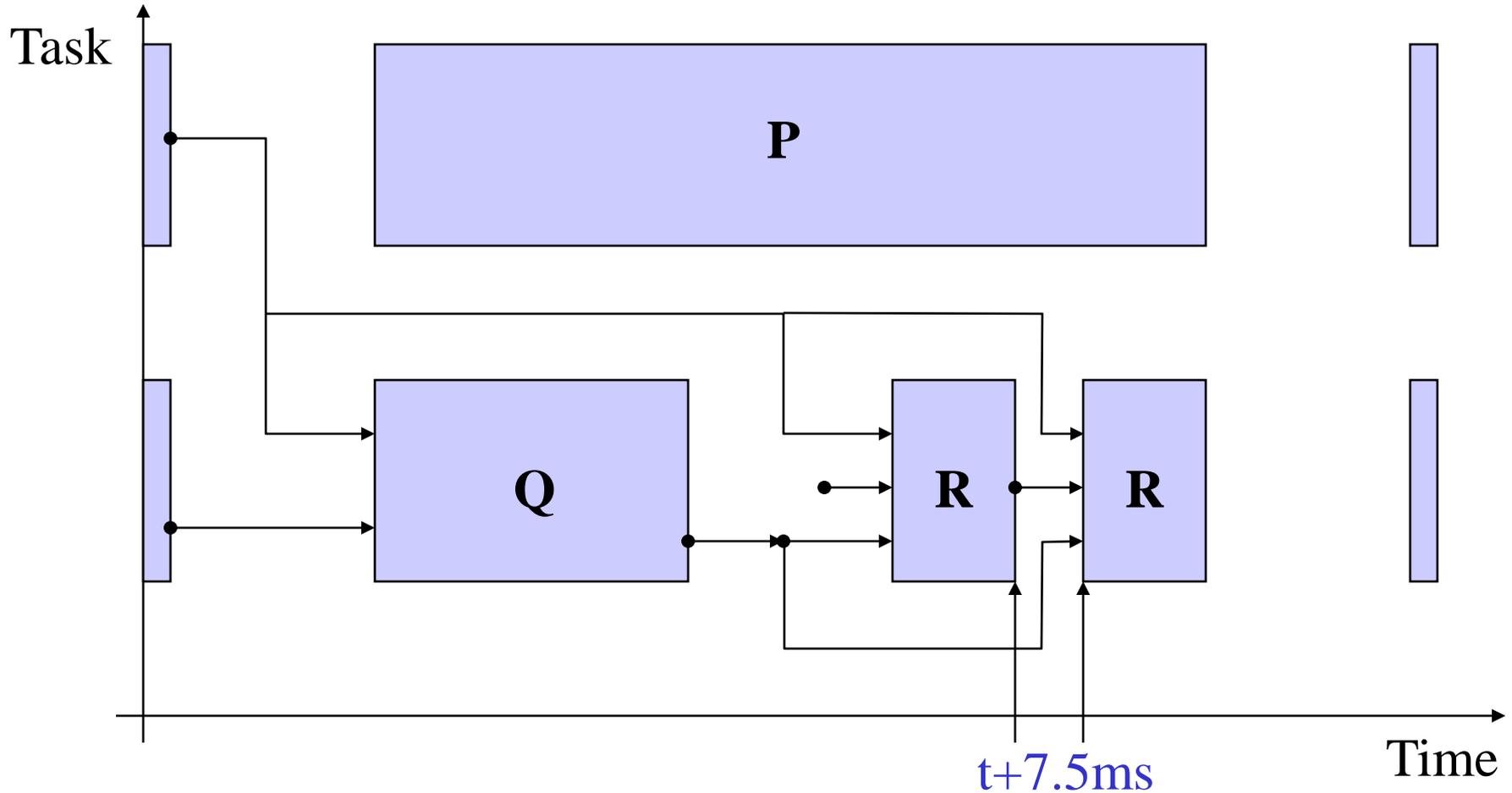
# Semantics of a Mode Switch



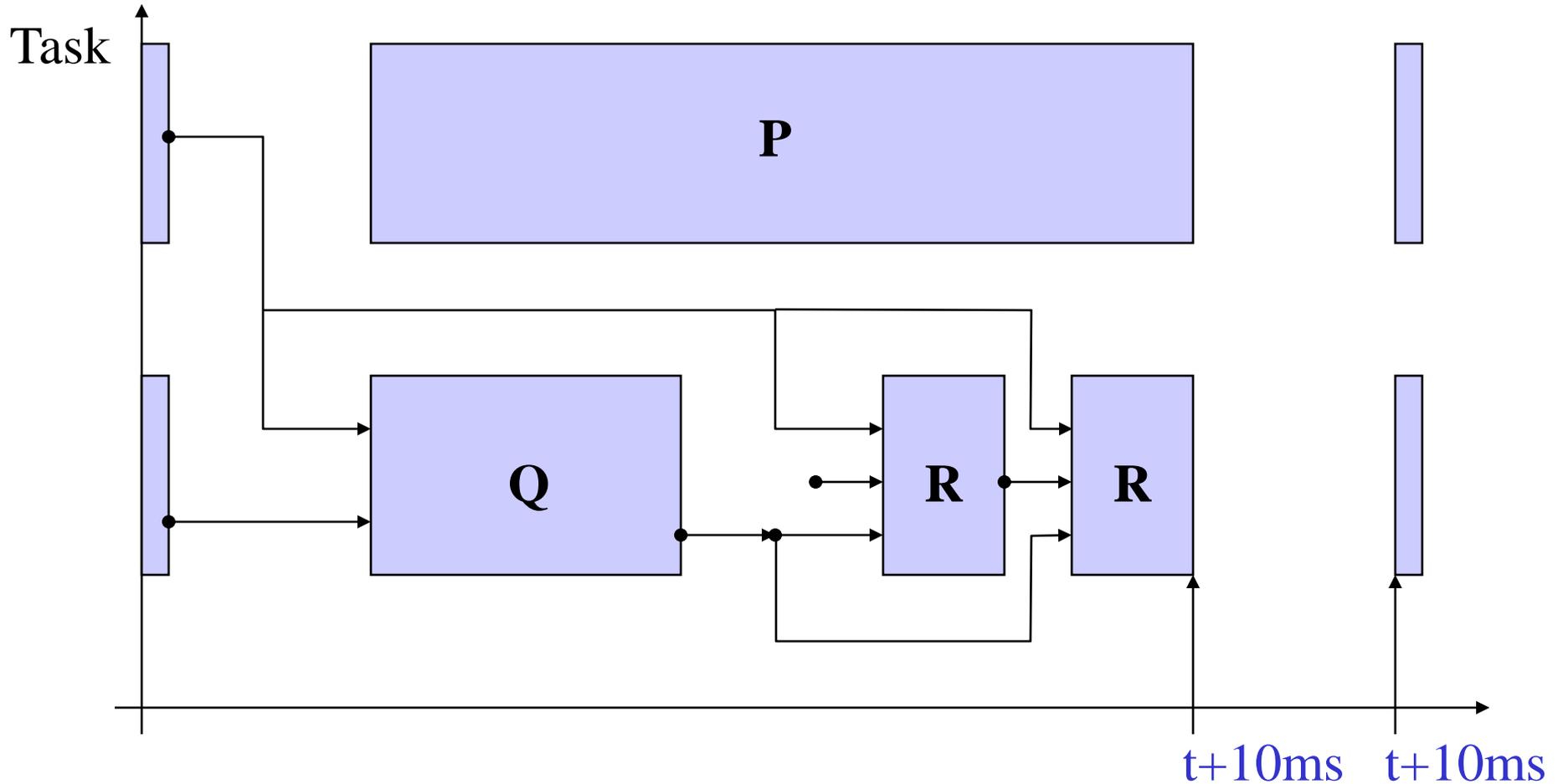
# Semantics of a Mode Switch



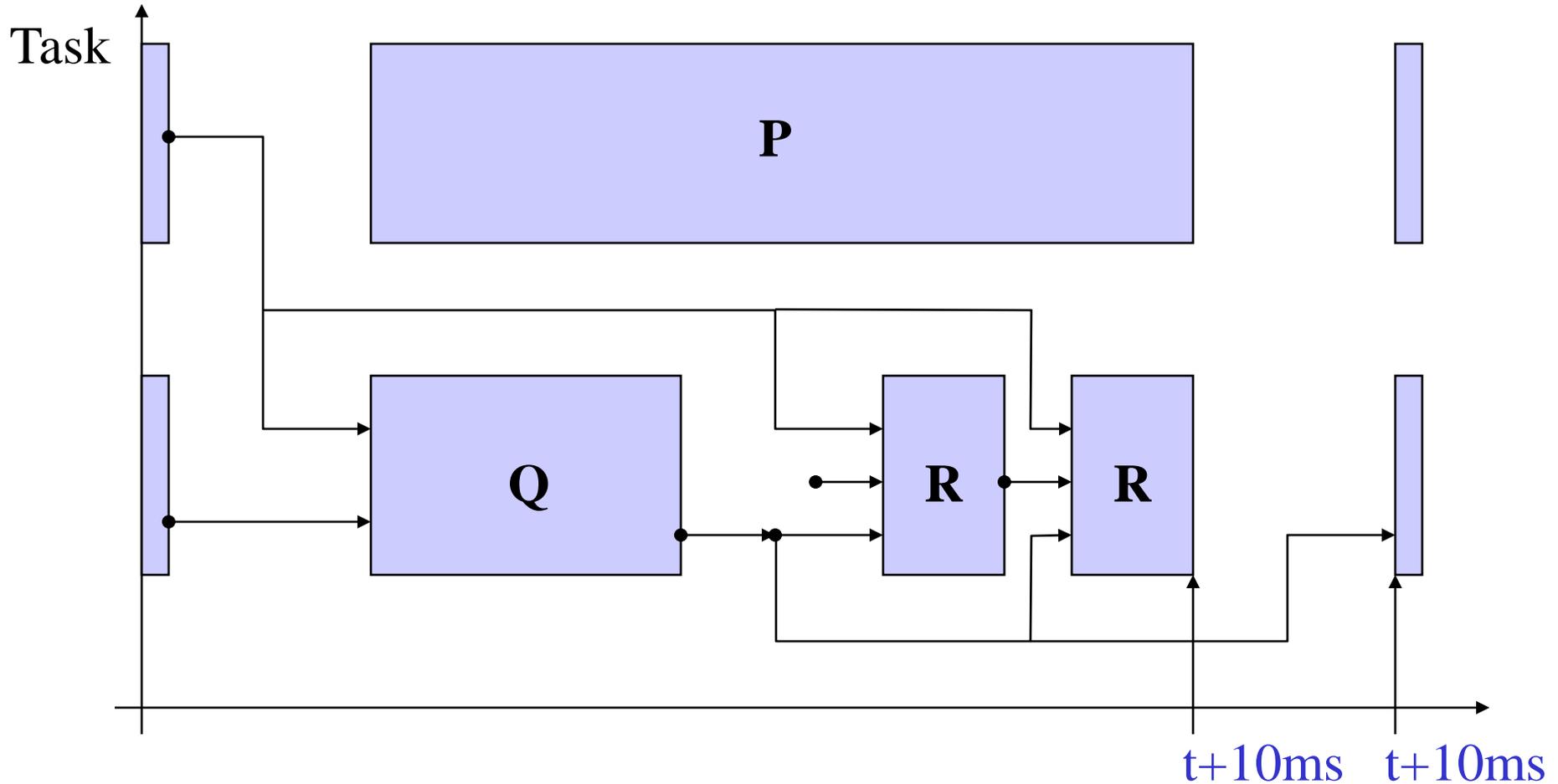
# Semantics of a Mode Switch



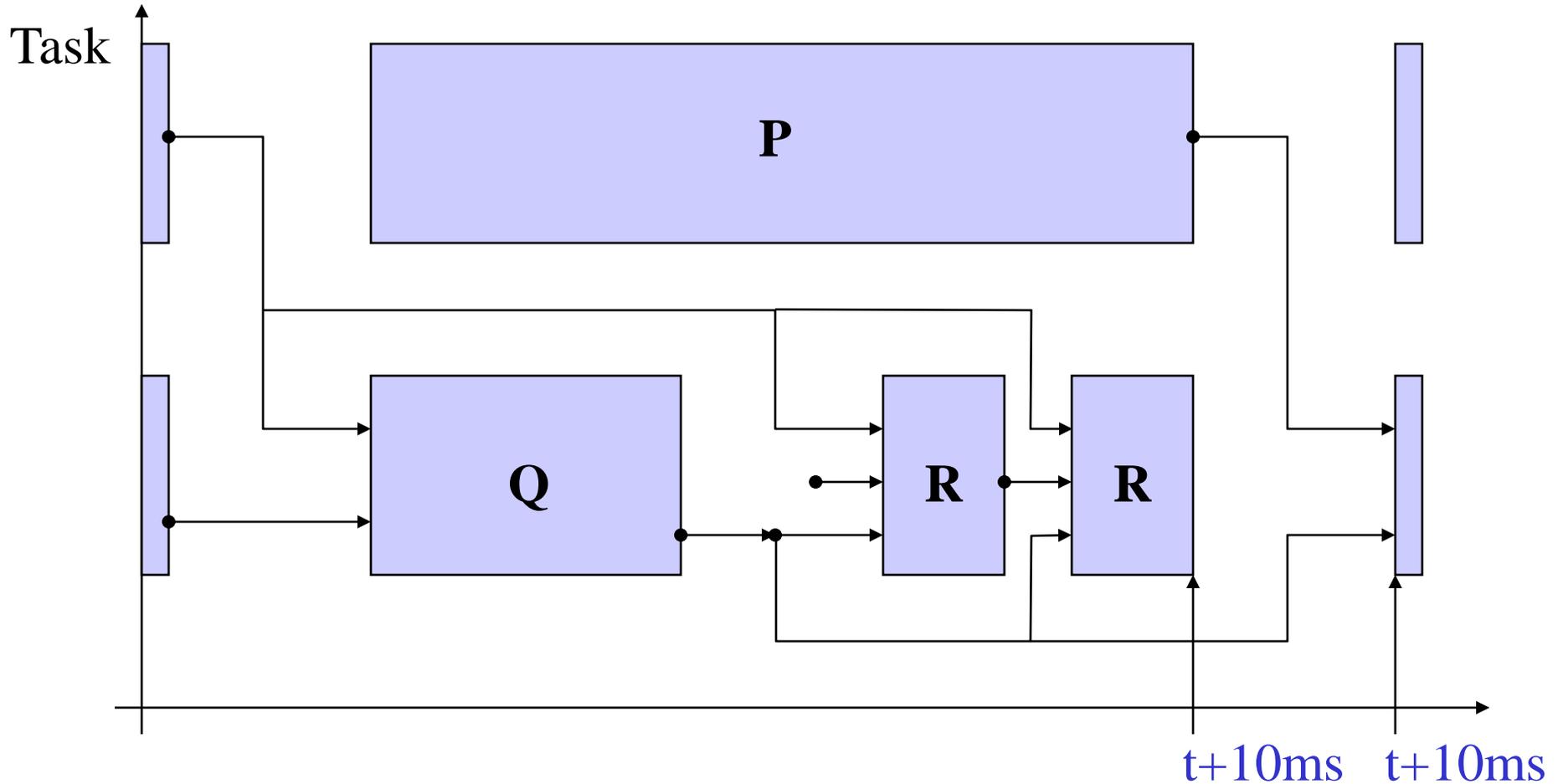
# Semantics of a Mode Switch



# Semantics of a Mode Switch



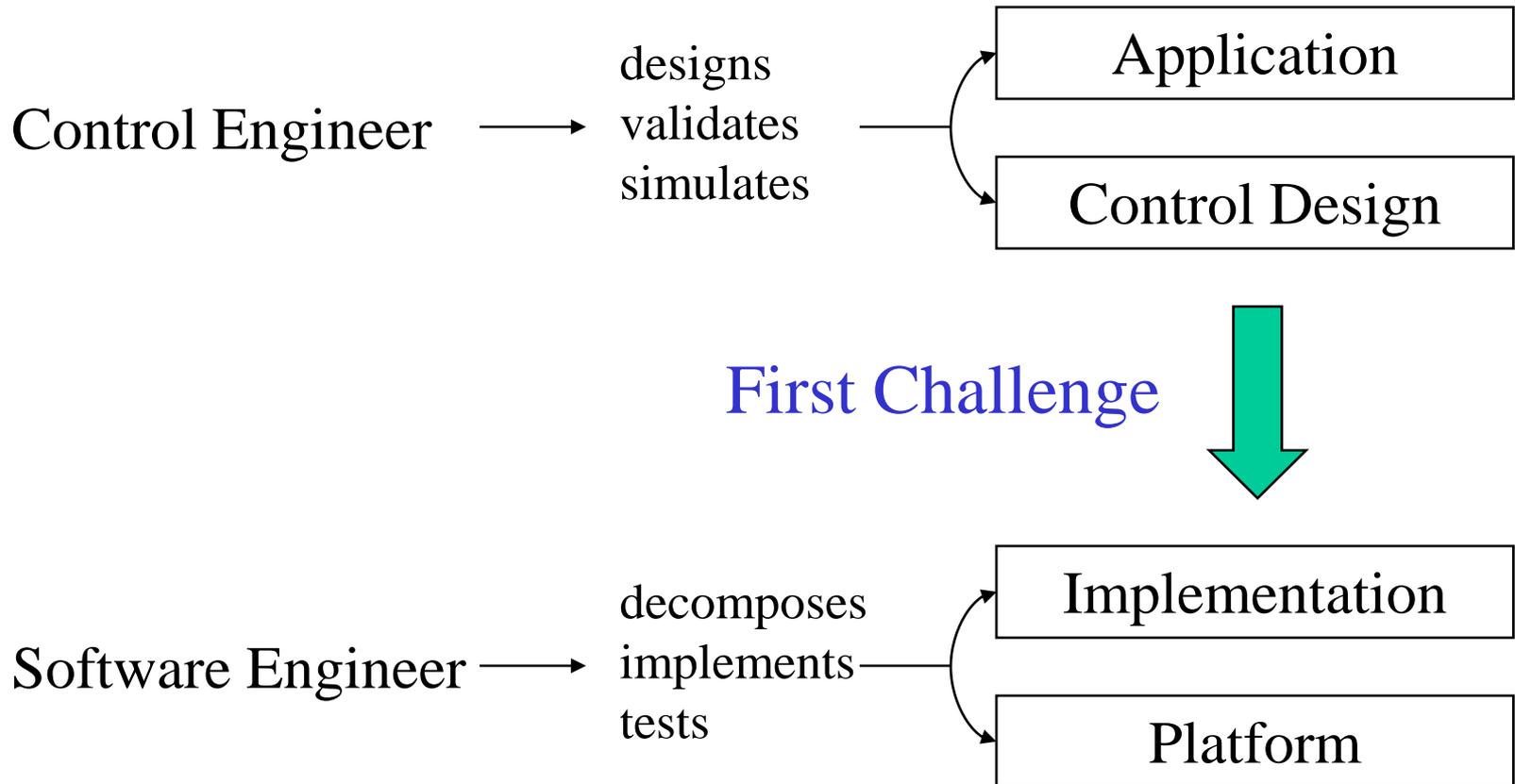
# Semantics of a Mode Switch





# Functionality & Timing

---

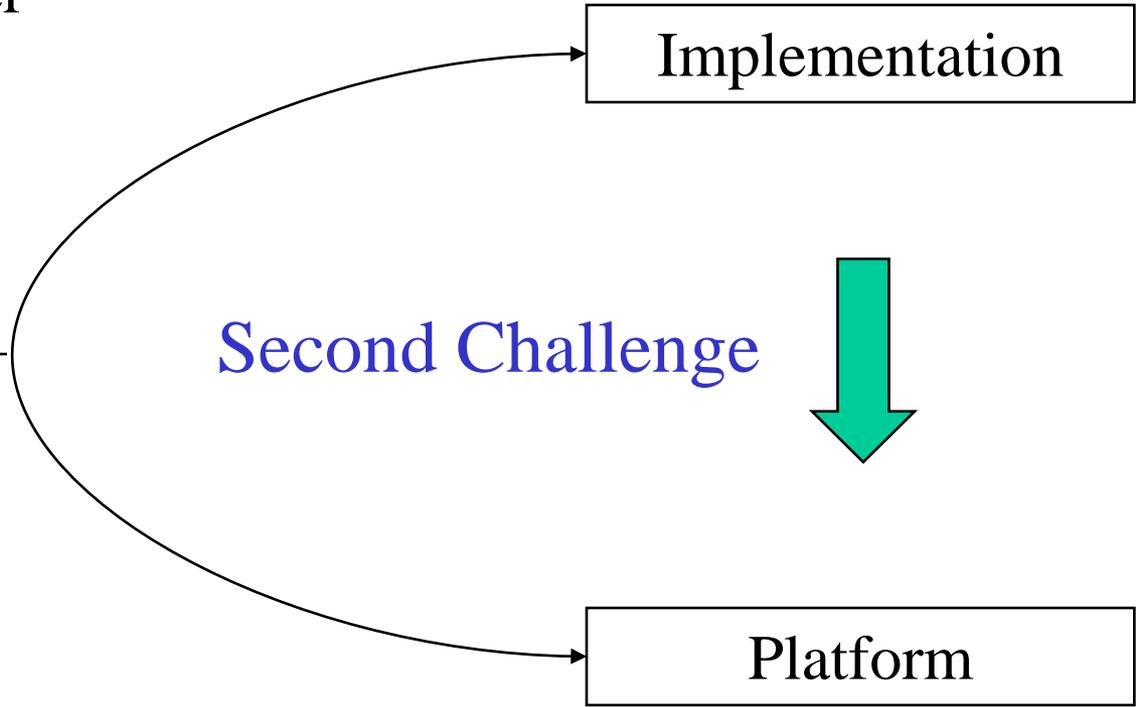


# Compilation

---

Software Engineer

↓  
decomposes  
implements  
tests



Implementation

Second Challenge

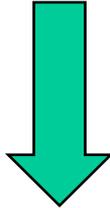
Platform

# Annotated Giotto

---

Giotto Program

Giotto Compiler



Platform

# Giotto-**A**rchitecture

---

Giotto Program

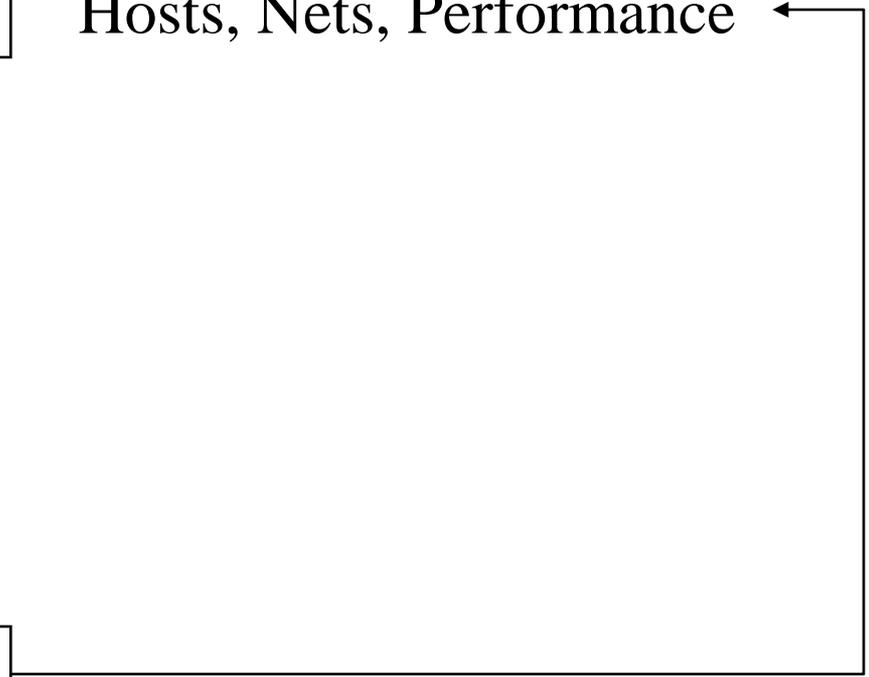
Giotto-**A** Program

Hosts, Nets, Performance

Giotto **Compiler**



Platform



# Giotto-Scheduler

---

Giotto Program

Giotto-**A** Program

Hosts, Nets, Performance

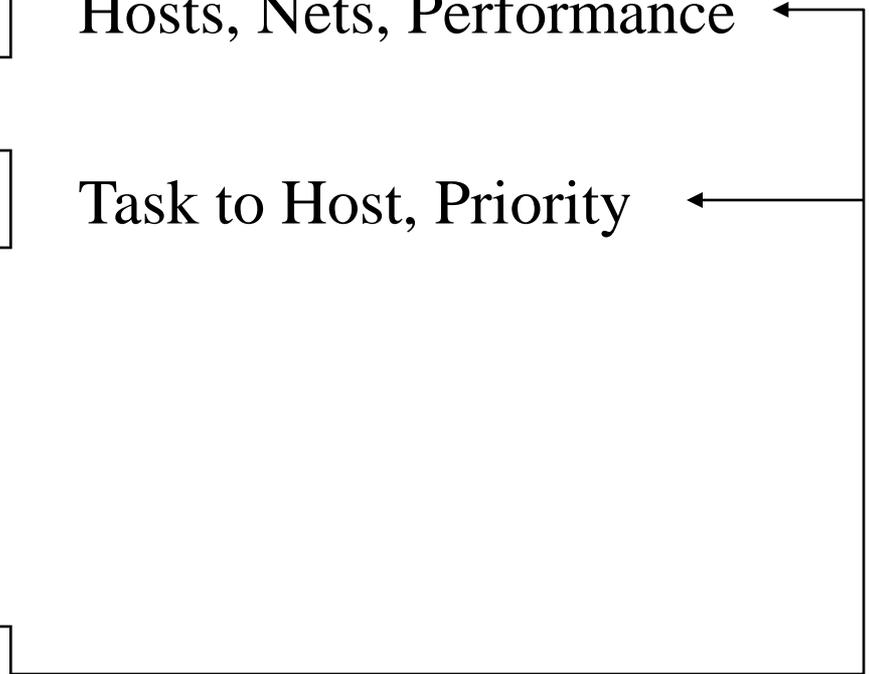
Giotto-**AS** Program

Task to Host, Priority

Giotto Compiler



Platform



# Giotto-C Communication

---

Giotto Program

Giotto-**A** Program

Hosts, Nets, Performance ←

Giotto-**AS** Program

Task to Host, Priority ←

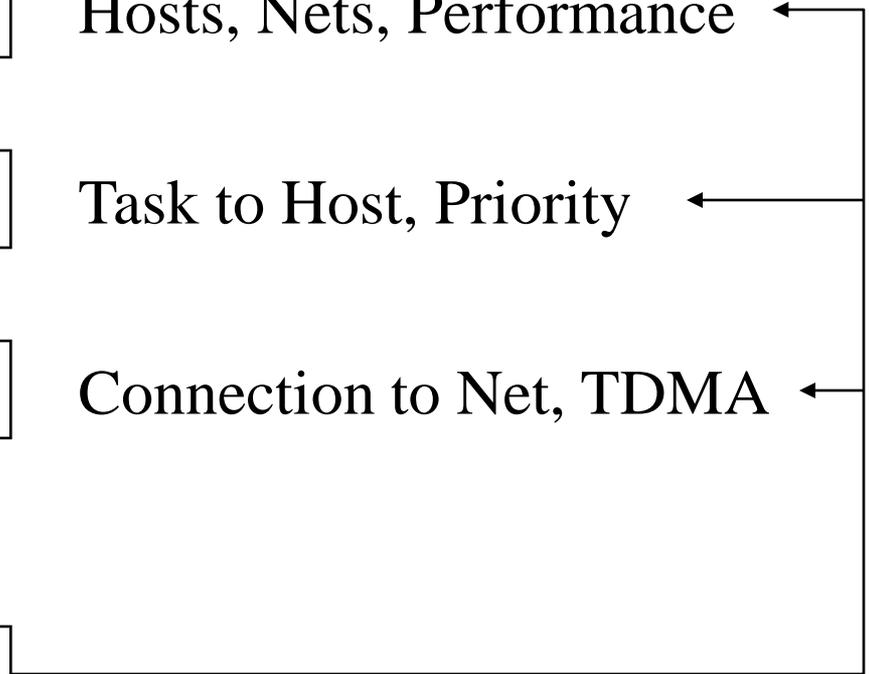
Giotto-**ASC** Program

Connection to Net, TDMA ←

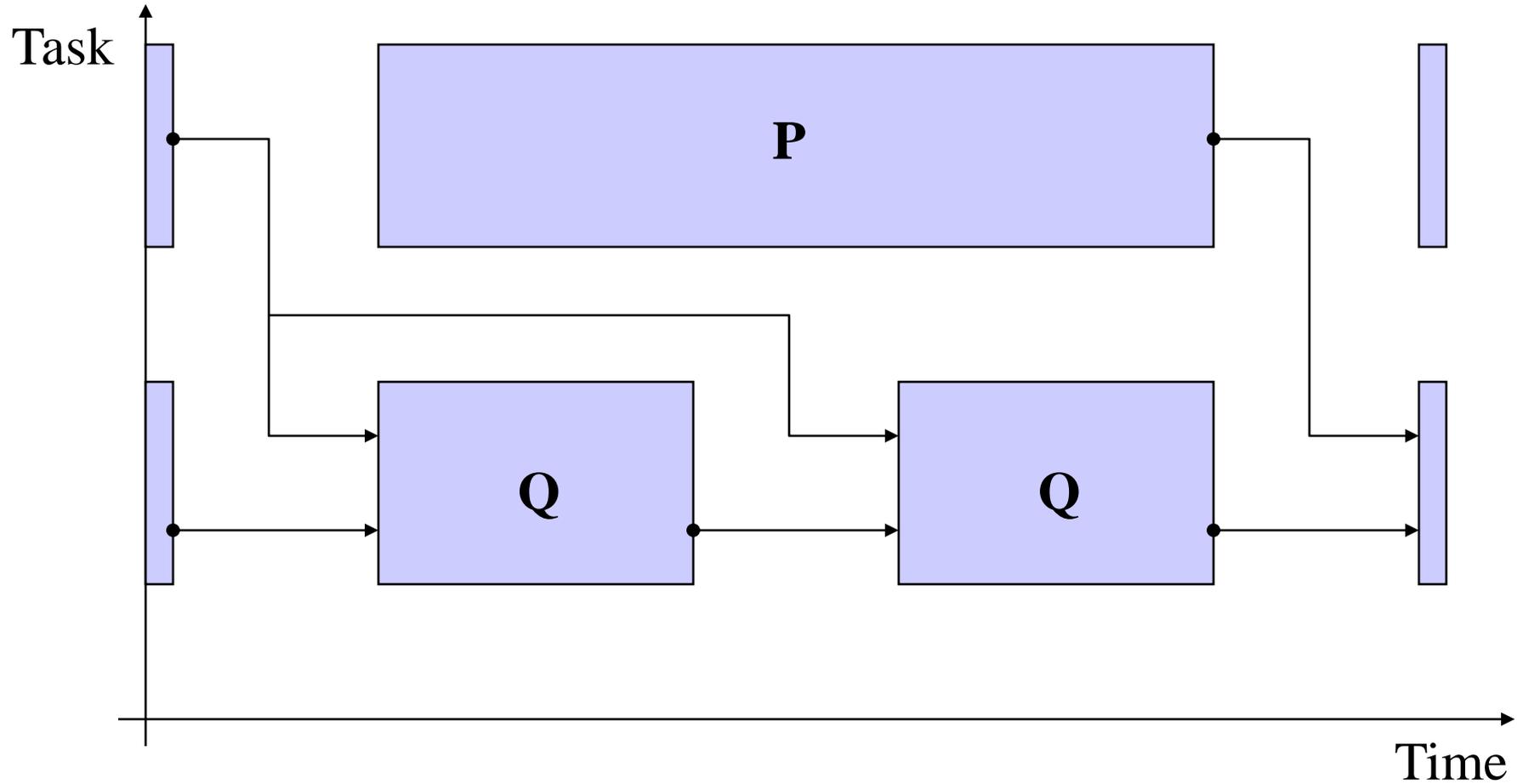
Giotto Compiler



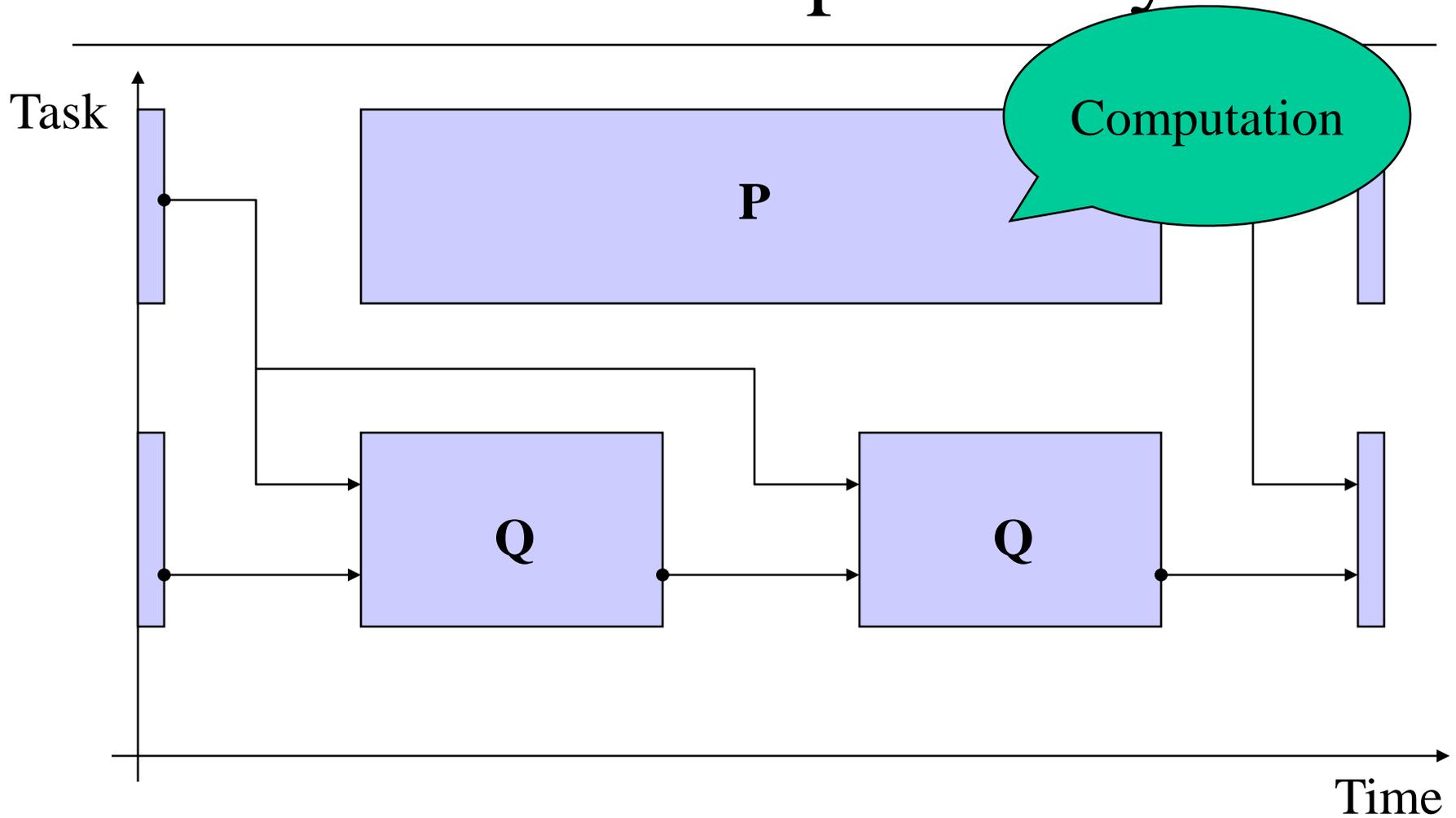
Platform



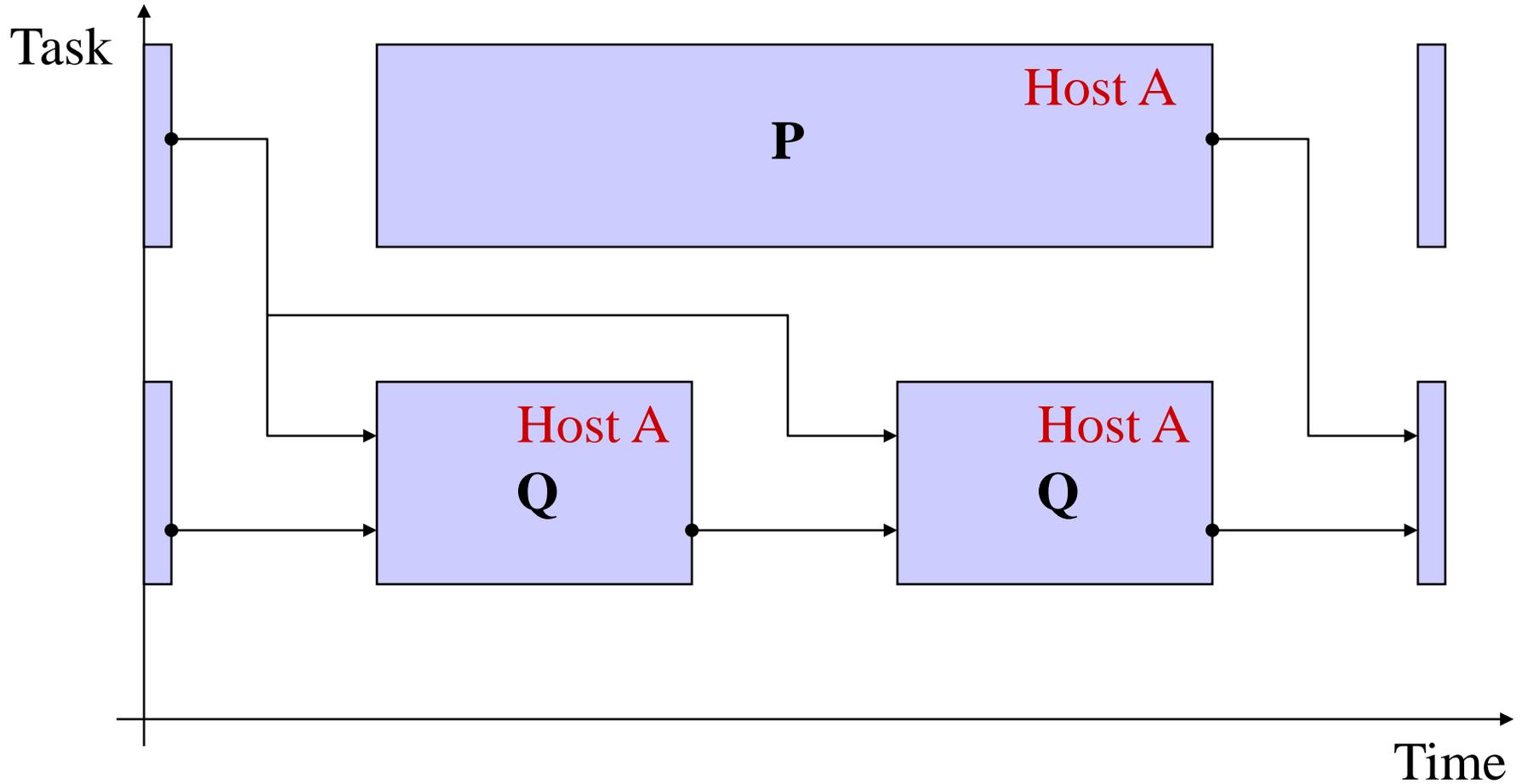
# Platform Dependency



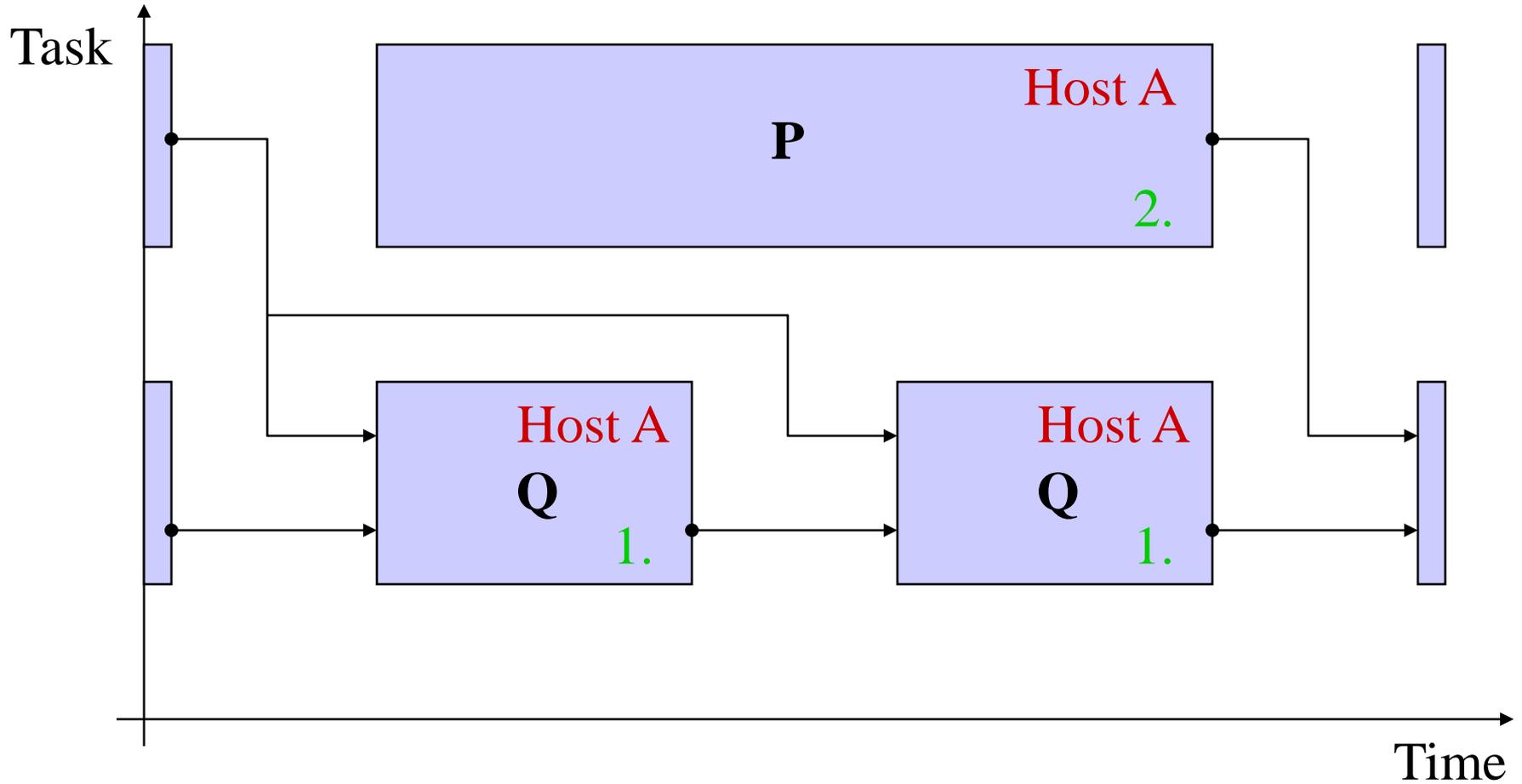
# Platform Dependency



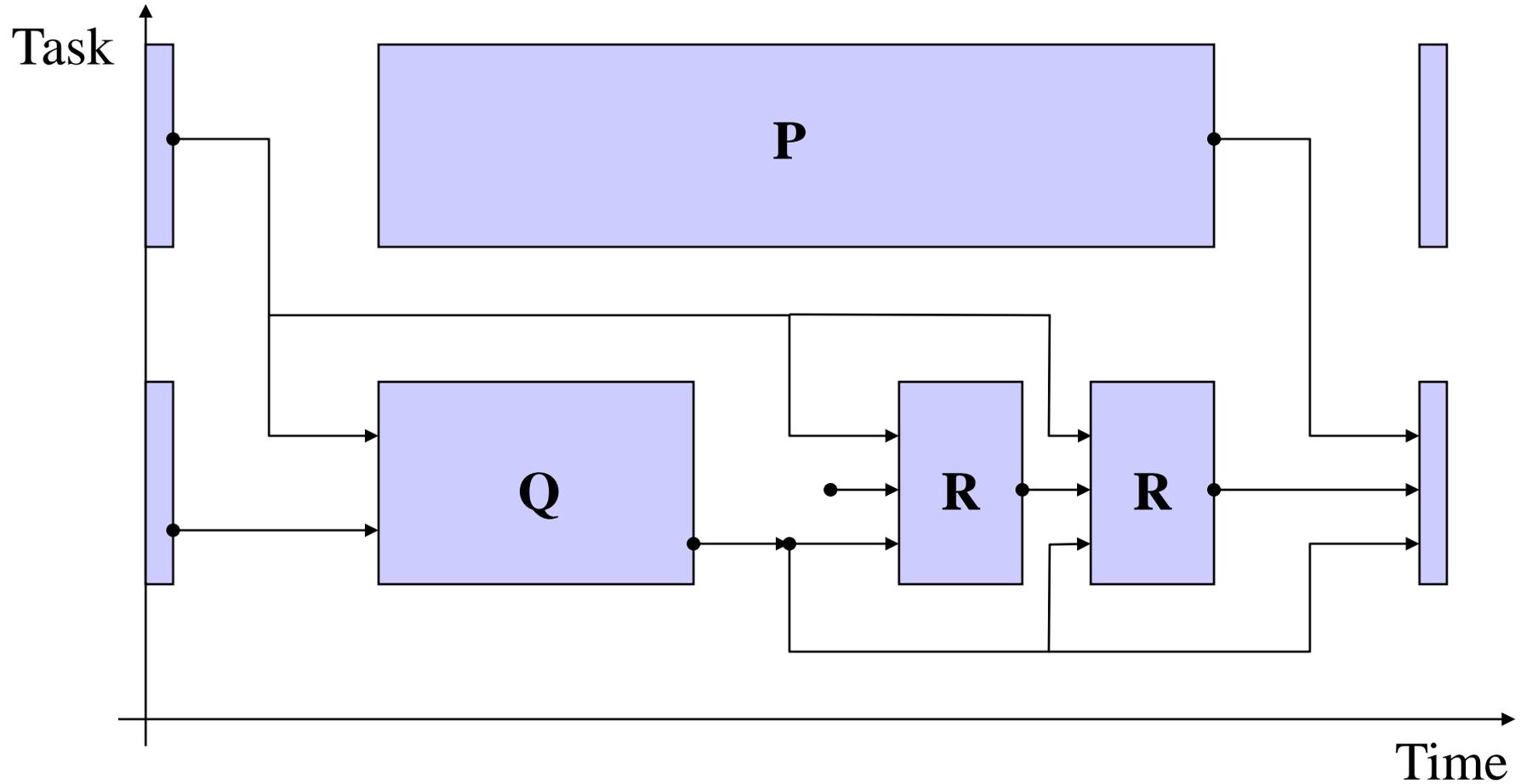
# Giotto-AS



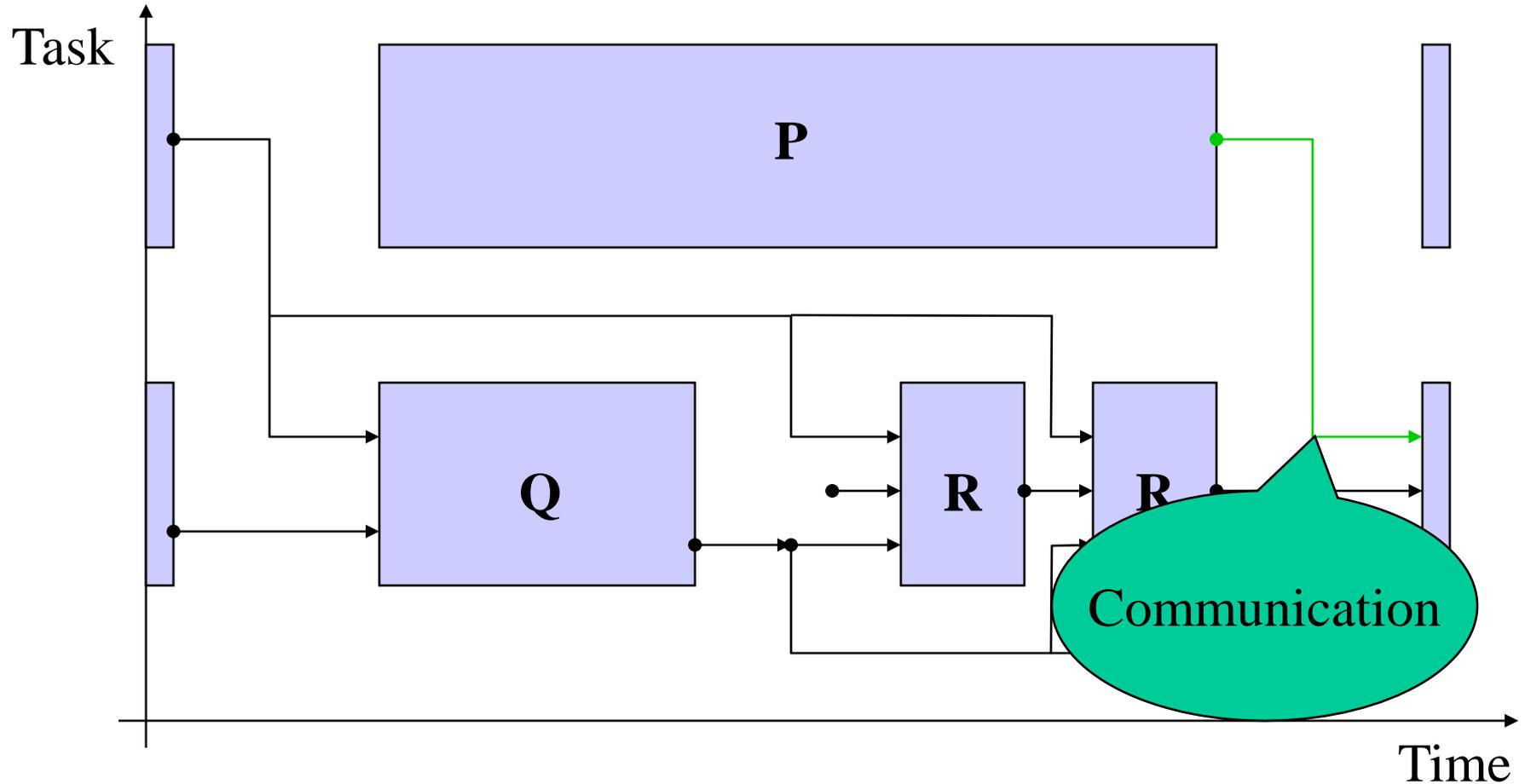
# Giotto-AS



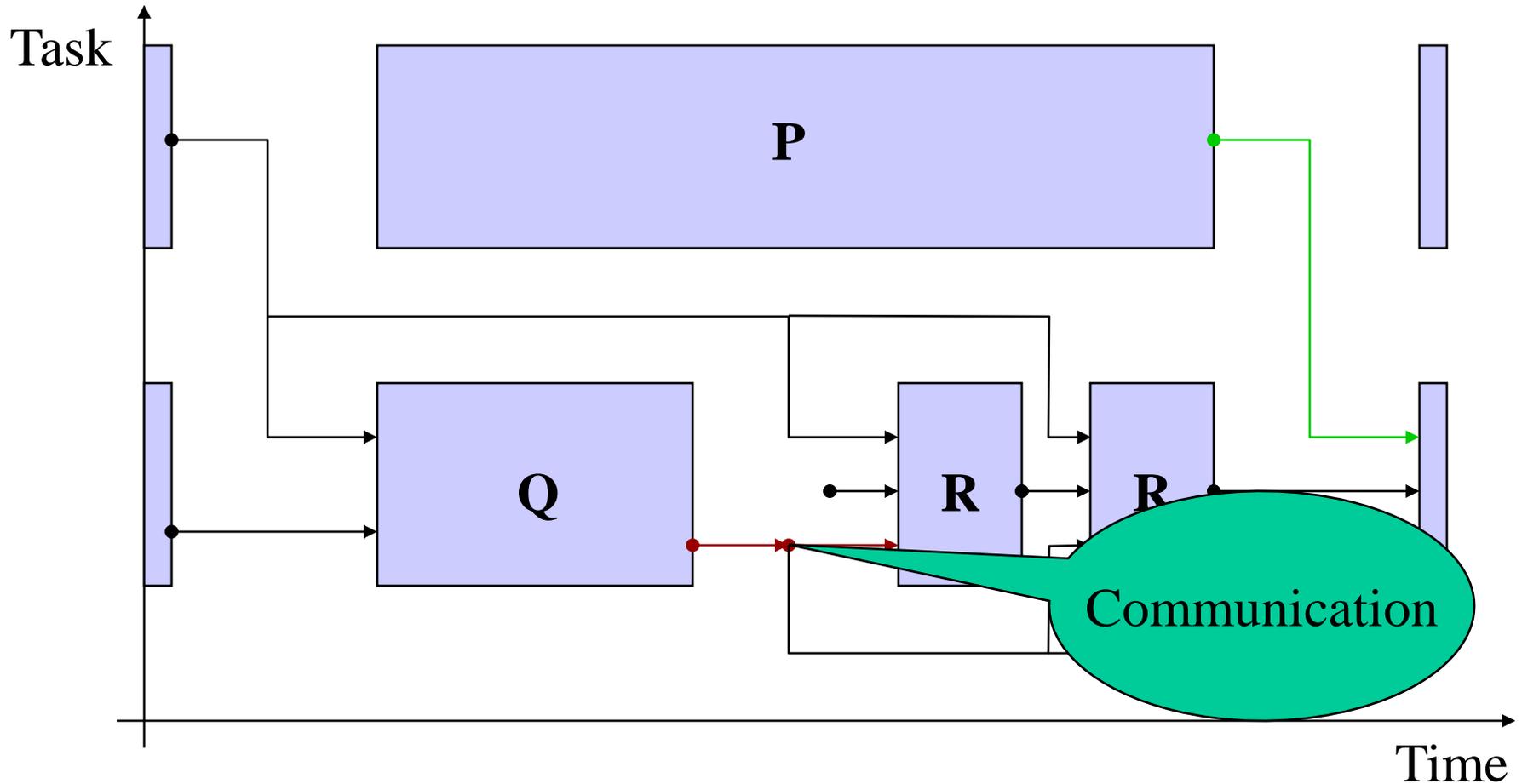
# Platform Dependency



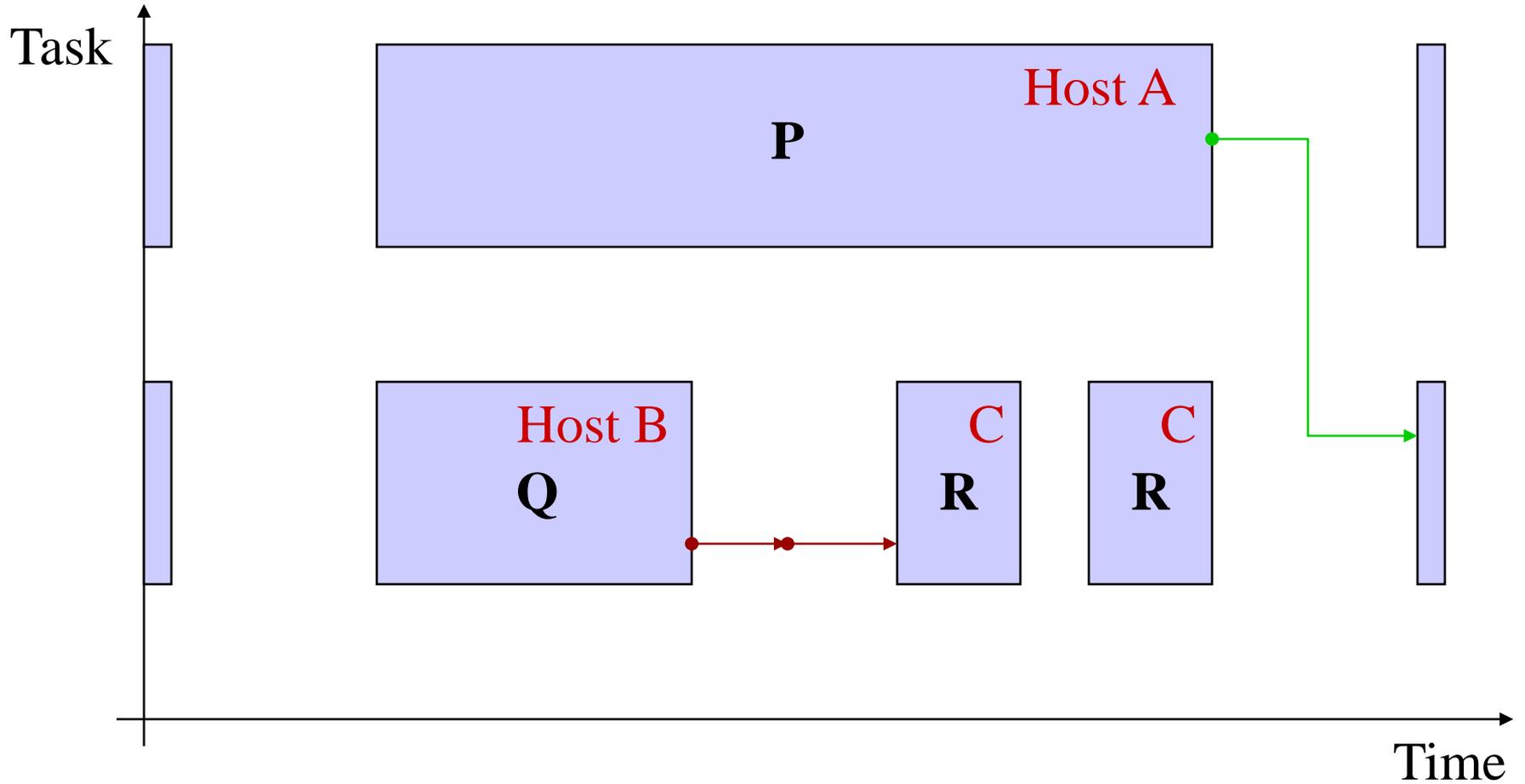
# Platform Dependency



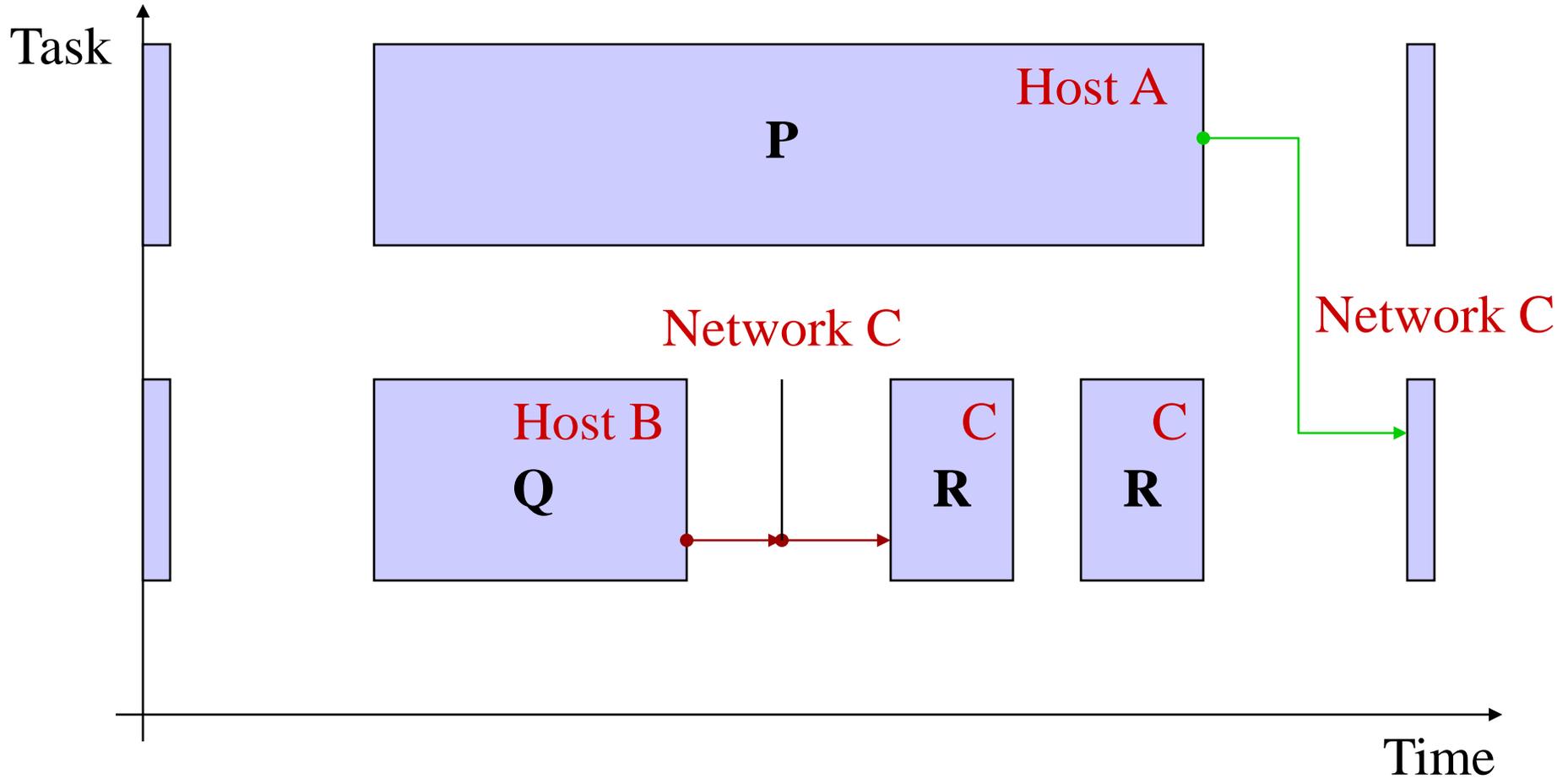
# Platform Dependency



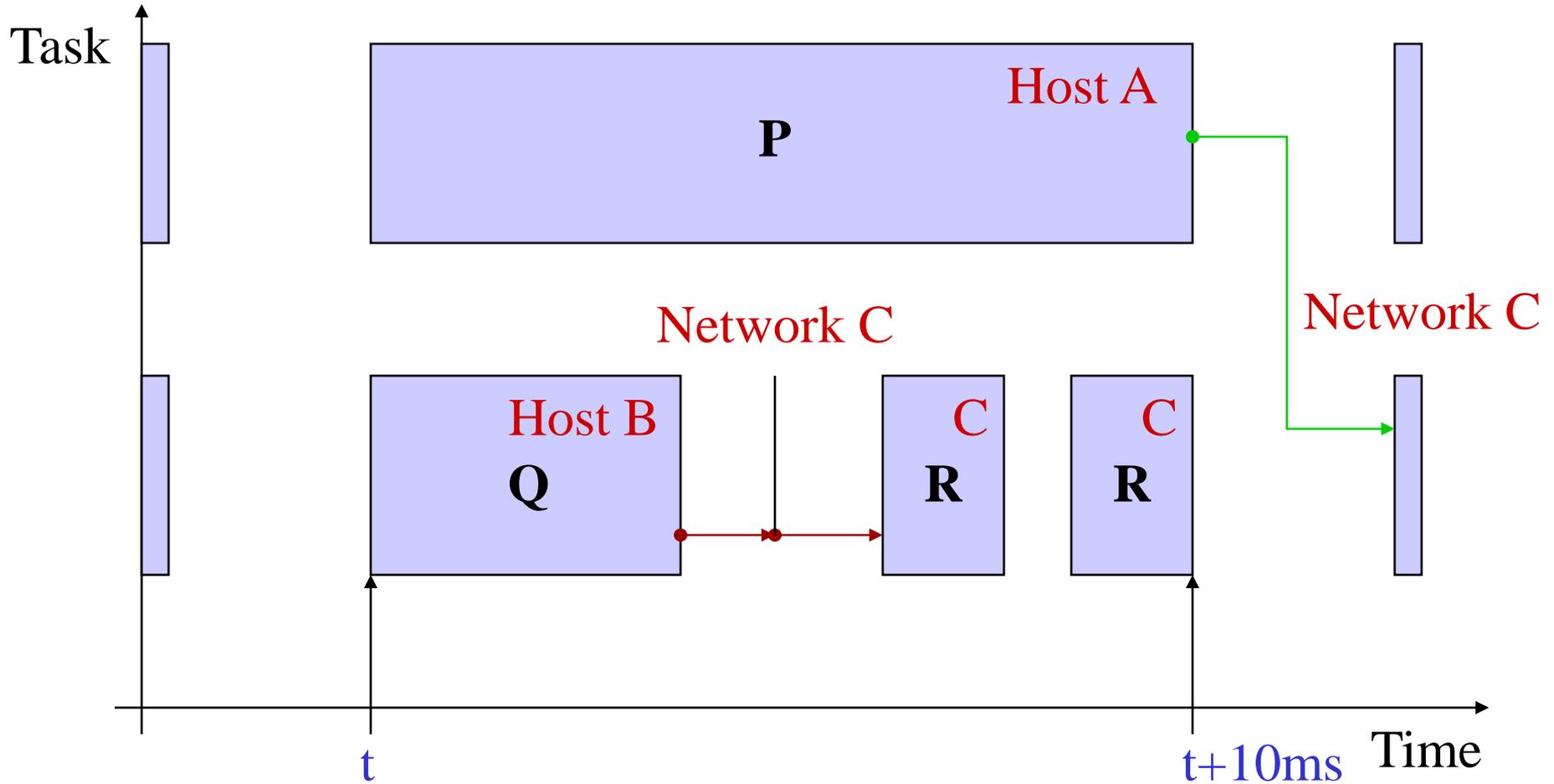
# Giotto-ASC



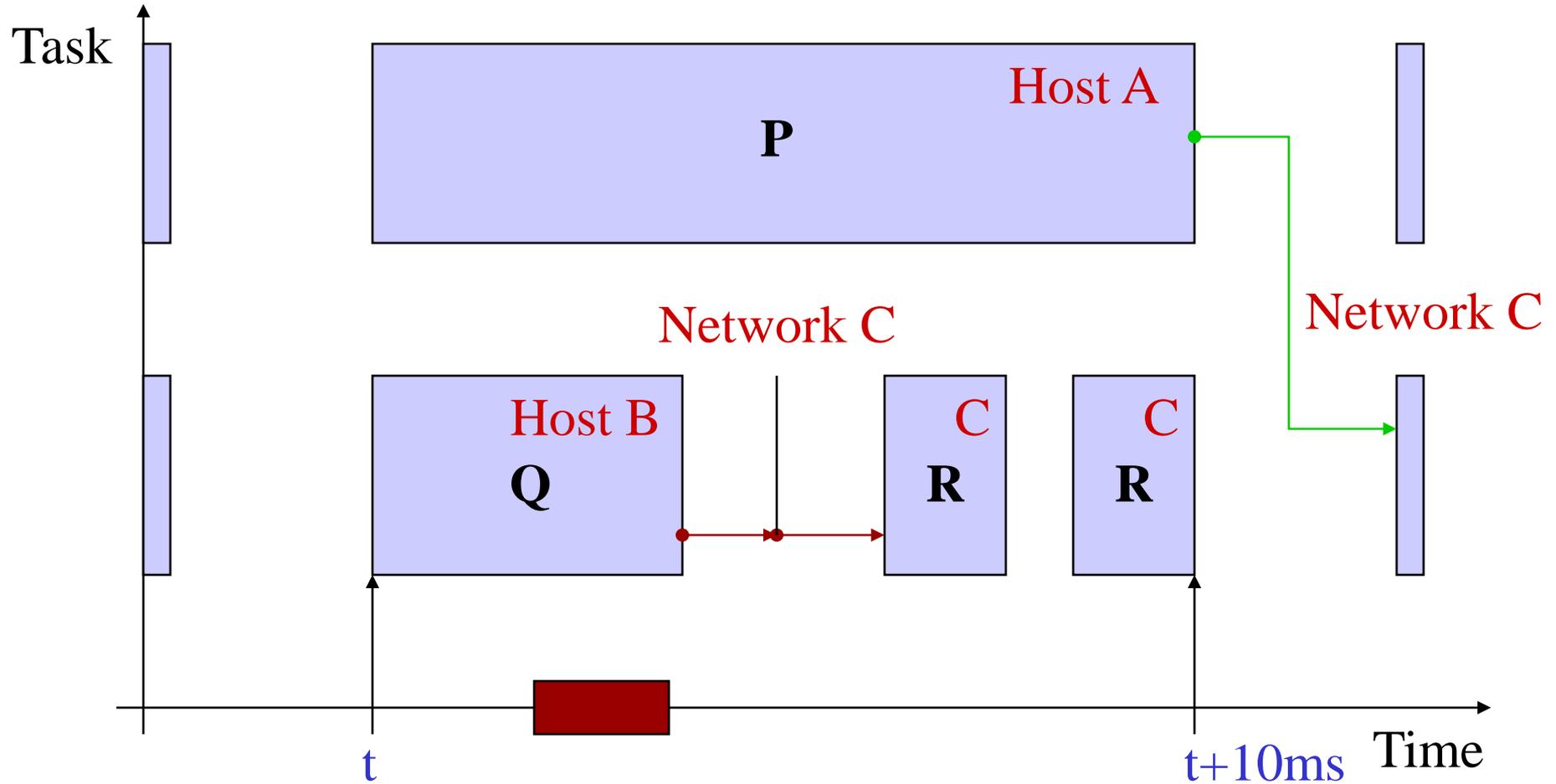
# Giotto-ASC



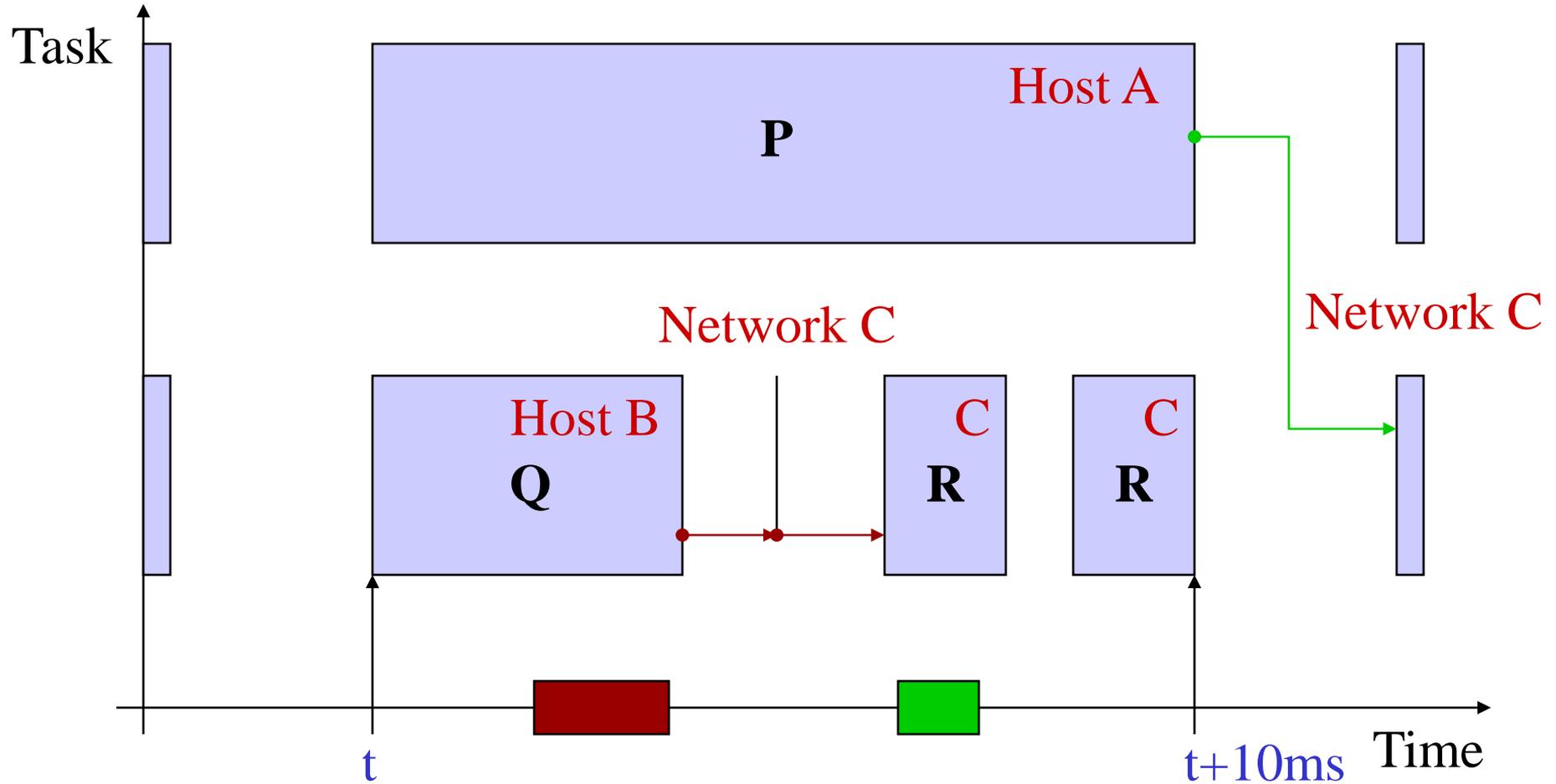
# Giotto-ASC



# Giotto-ASC

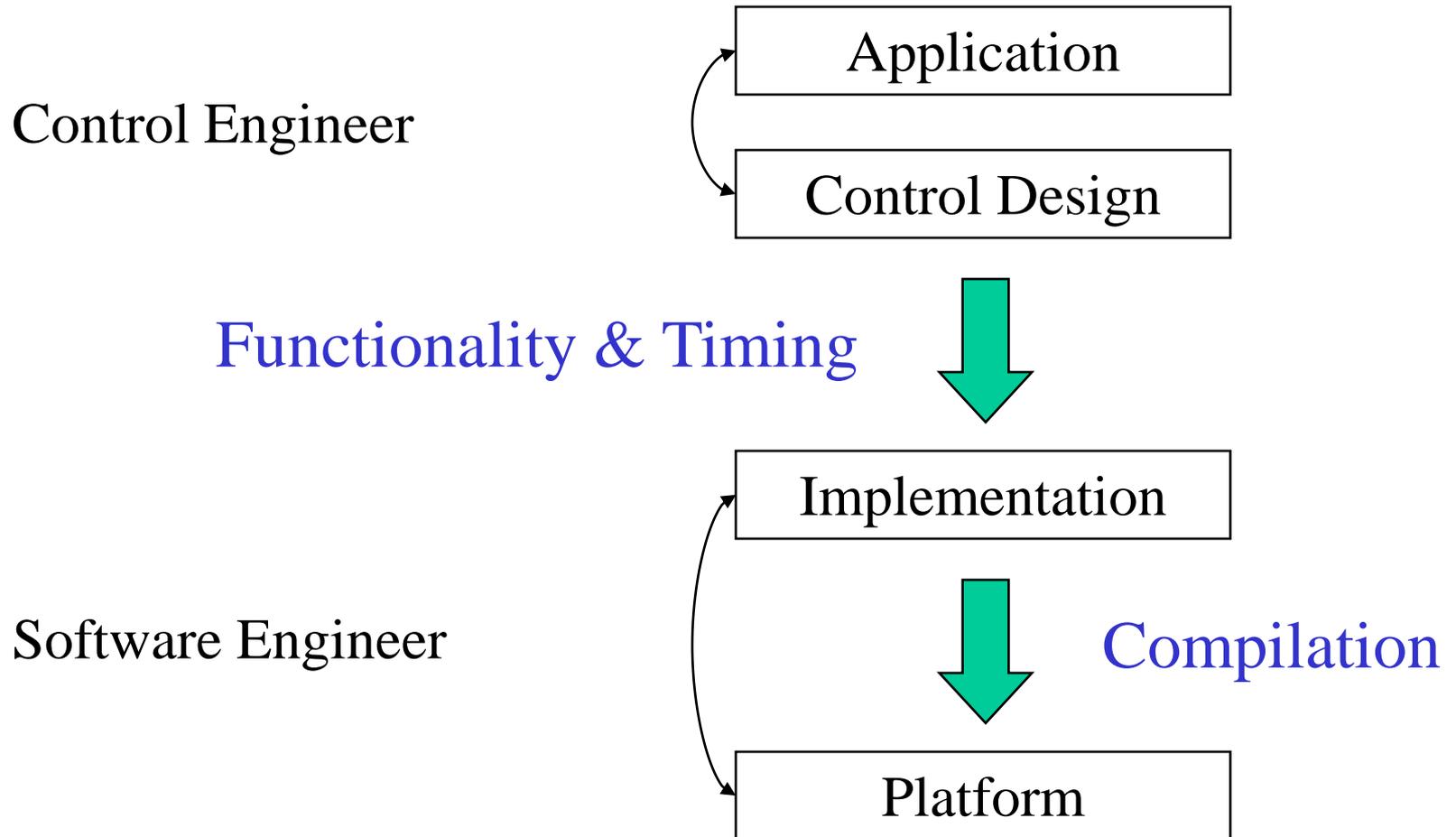


# Giotto-ASC



# Summary

---



# Demo

---

