Course 02224

Mandatory Assignment 1

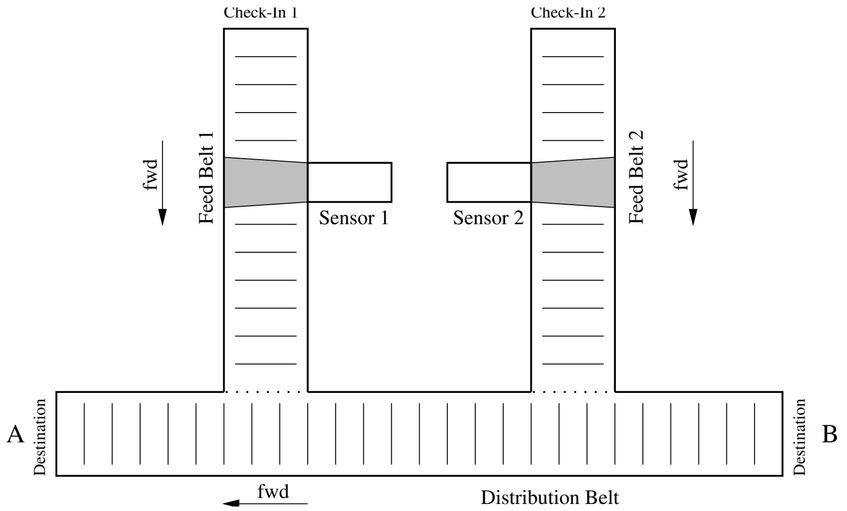
Hans Henrik Løvengreen

DTU Compute, Technical University of Denmark

Real-Time Systems Assignments: General

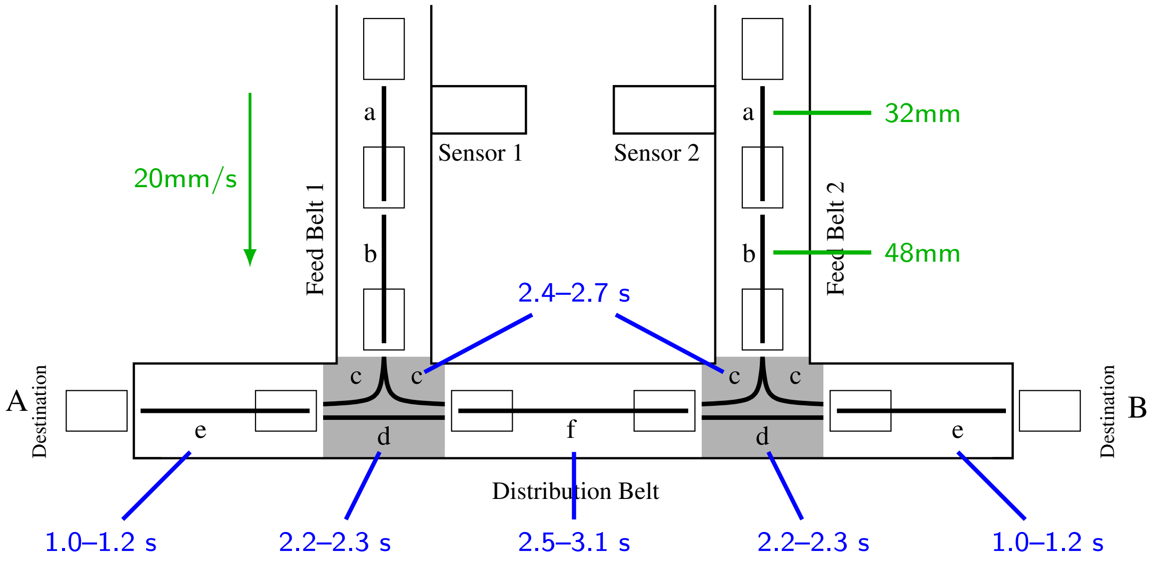
- There are two mandatory assignments:
  - Assignment 1: **Baggage Sorting Facility (simple)**
  - Assignment 2: **Baggage Sorting Facility (cont'd)** or **one of 2-3 other standard projects** or **your own modelling and analysis project**
- The assignments should be carried out in groups of 2-3 persons.
- Each group must hand in two *reports*:
  - Assignment 1: **Sunday March 26 at 23.59**
  - Assignment 2: **Sunday May 21 at 23.59**
- Groups of 3 persons are expected to make more comprehensive solutions
- The assignments and the exam all contribute to the final mark

Baggage Sorting Facility

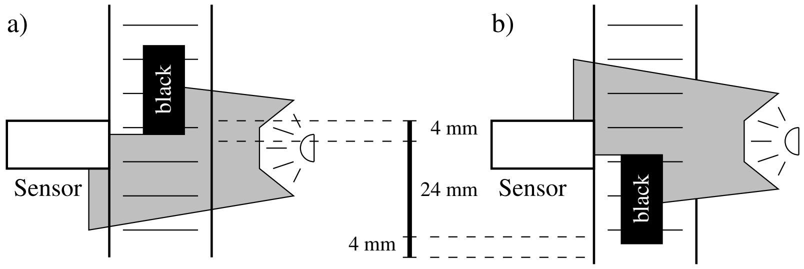


All black bags must go to Destination A and all yellow bags must go to Destination B.

Timings



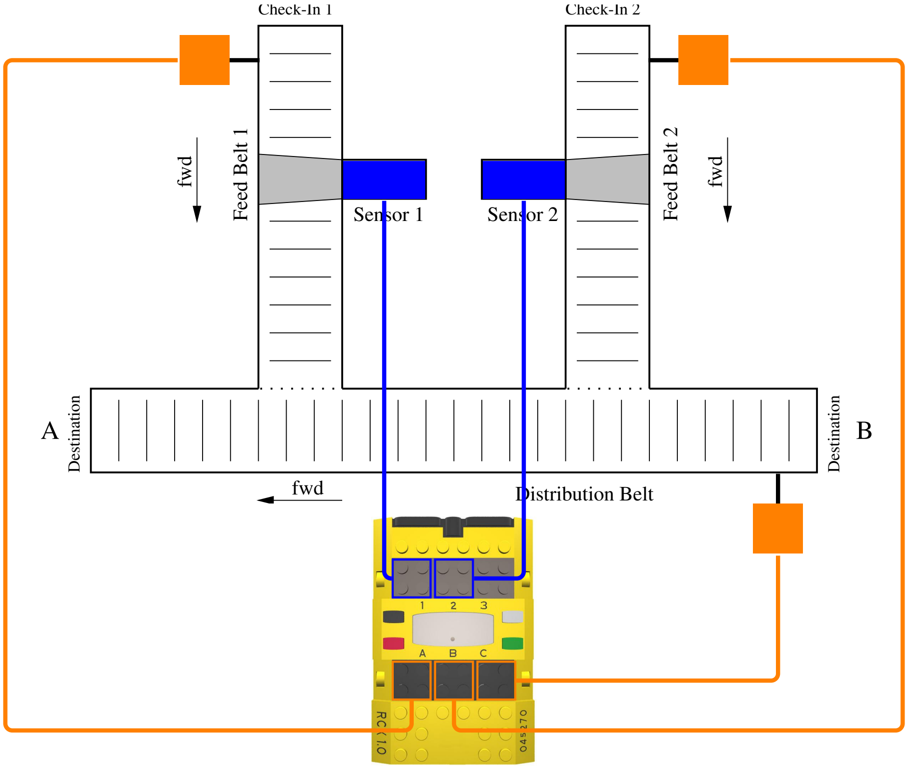
Sensors



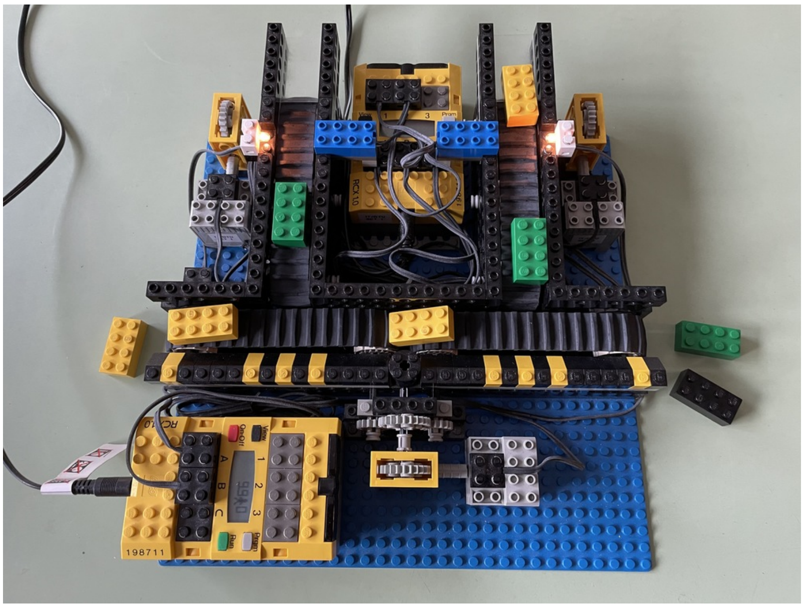
The RCX Box

- Introduced in 1998 as part of the Lego Mindstorm Construction Set
- A Hitachi 8-bit micro-controller equipped with:
  - 32 K RAM
  - 3 input ports: 1, 2, 3
  - 3 output ports: A, B, C
  - 1 bidirectional serial infra-red port
  - 1 LCD-display
  - 1 “loud-speaker”
  - Some control buttons
  - A simple “BIOS” in ROM (for loading firmware).
- Superseded by NXT (2006), EV 3 (2013) and Spike Prime (2019)
- May all be programmed in a reduced Java version: Lejos
  - RCX devices represented by objects: `Motor.A`, `Sensor.S2`
  - Threads and sychronization may be used
  - Device changes may be detected by `Poll` constructs

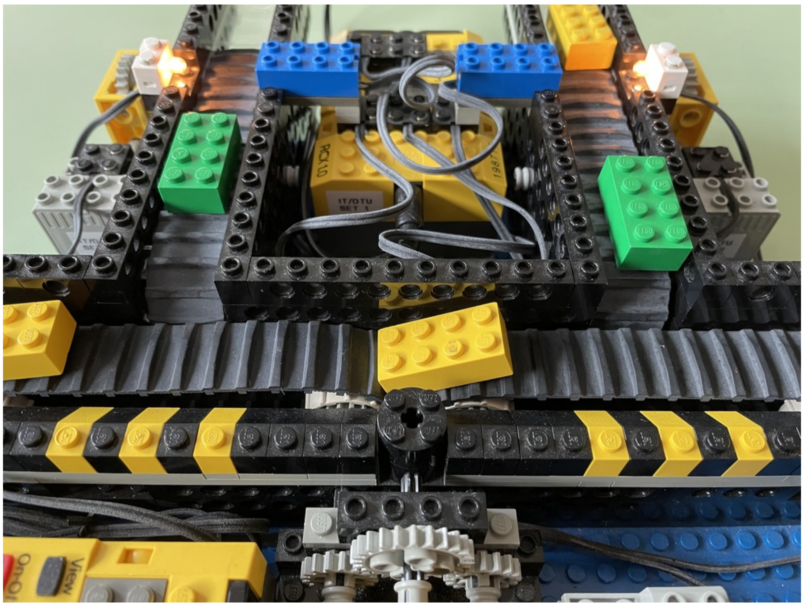
The Baggage Handling System



Baggage Handling System



Distribution Belt Junction



The Given Control Program

SingleSort.java

- Simple Java program using the Lejos framework
- Two threads — one for each feed belt
- Each thread sorts the bags arriving at its belt
- No coordination among the two!

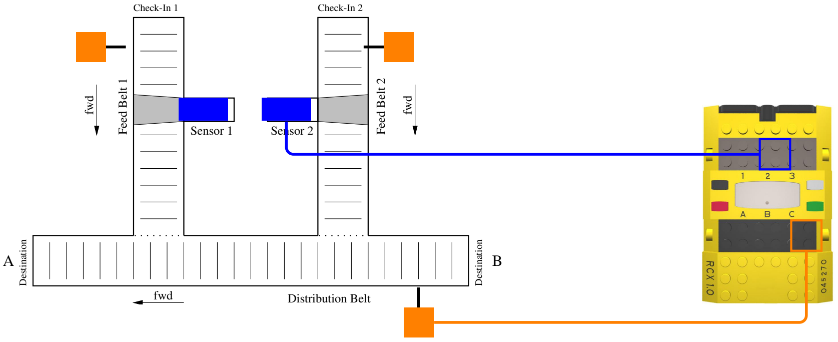
Requirements for Mandatory Assignment 1

- You should:
  1. Make a UPPAAL model of the *physical system*
  2. Make a UPPAAL model of a *given sorting strategy*
  3. *Verify safety and liveness* properties of the total system
  4. Demonstrate modelling soundness
  5. Demonstrate modelling details
  6. Determine the baggage handling *limits* of the system
  7. [Opt] Optimize the system *performance*
  8. [Opt] Find conditions for *alternating check-in*
  9. [Opt] Determine *optimal stop position*
  10. [Opt] *Green version*
  11. [Opt] Simple *double check-in* sorting
- Your work must be described and discussed in a report
- Each group member most be main contributor to one optional task
- The report should focus on interesting/challenging aspects
- The report size should not exceed 10 pages (excl. model printouts)

The Modelling Challenge

Physical

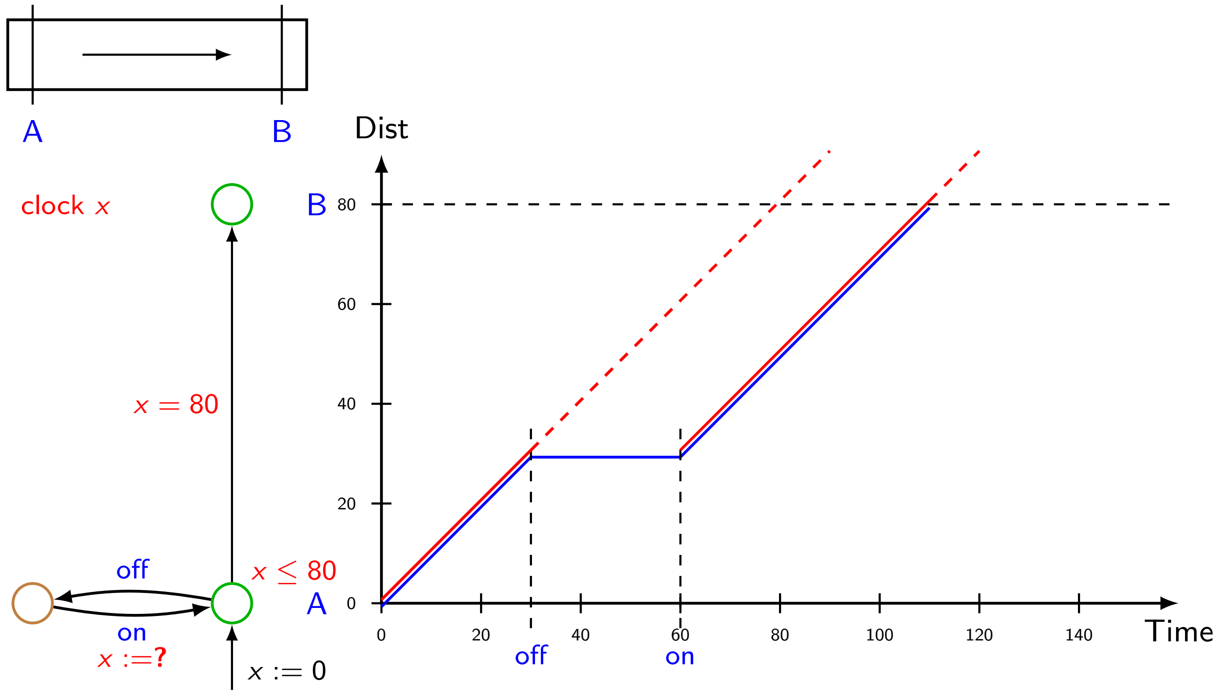
Control



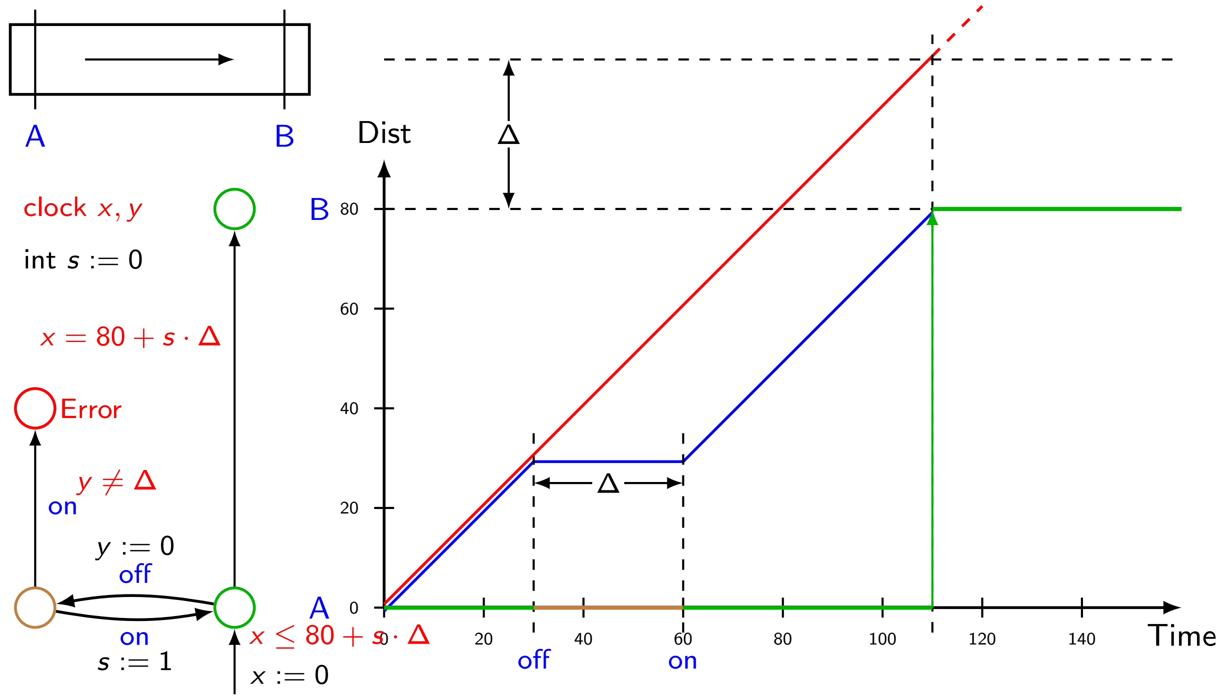
Modelling Issues: Erronous Behaviour

- The model must model the physical system separately from the control
- The physical model must not constrain the control
- The combined model must allow for faulty control
- Not all control can be adequately modelled in the physical model
- Such *erroneous control* must be detected (and avoided)

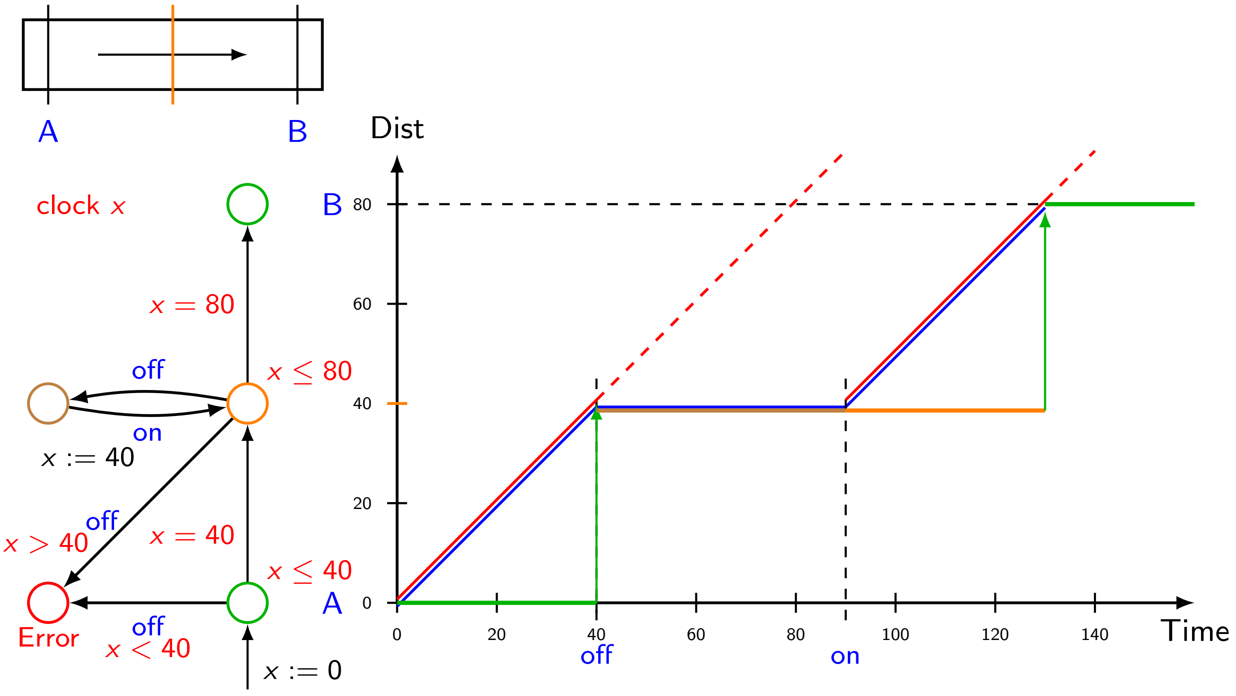
Modelling Issues: The Stop Problem



Modelling Issues: Stop Solution — Constant Stop Time



Modelling Issues: Stop Solution — Fixed Stop Position


A
B
clock x
$x = 80$
off
on
$x \leq 80$
$x := 40$
off
$x > 40$
$x = 40$
$x \leq 40$
Error
off
$x < 40$
$x := 0$
Dist
B 80
60
40
20
A 0
0
20
40
60
80
100
120
140
Time
off
on