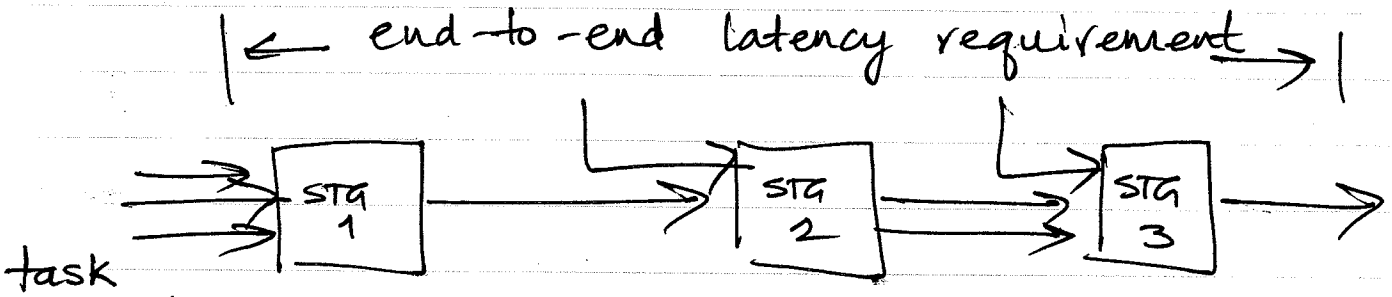
EECE 494

March 9 2011

distributed real-time systems



each stage is a separate processor.
Stages connected by some network.

When there is only one processor, we know how to compute response times, (in particular WCRT). when task arrivals are periodic.

A task is comprised of subtasks: one subtask per stage. Each subtask has its own execution time. The entire task has a period.

T_1: e_{11} – exec. time on stg. 1
e_{12} – ——— " ——— 2
e_{13} – ——— " ——— 3

] Task has a period P_1.

Stage 1 has two tasks:

T_1 : $e_{11} = 3$, $P_1 = 10$

T_2 : $e_{21} = 2$, $P_2 = 15$

What is the WCRT of all jobs of T_1 if scheduled using RM?

The response time (WCRT) is always 3.

arrival times at stg 1: 0, 10, 20, ...
" " at stg 2: 3, 13, 23, 33, ...

What about T_2?

WCRT is = 5

arrival times at stg 1: 0, 15, 30, 45, ...
arrival times at stg 2: 5, 17, 35, 47, ...

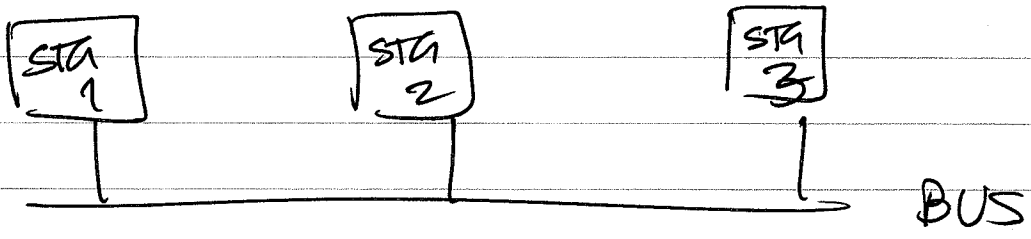
12 18 12

$\left\lceil \frac{L}{P_i} \right\rceil$ if we use period of T_2 as 12, we are safe but we overestimate the utilization of T_2.

To ensure a task is strictly periodic at a stage, ensure that it waits for its WCRT at the previous stage.
(buffer + use ~~interrupts~~ a timer to release jobs to the next stage).

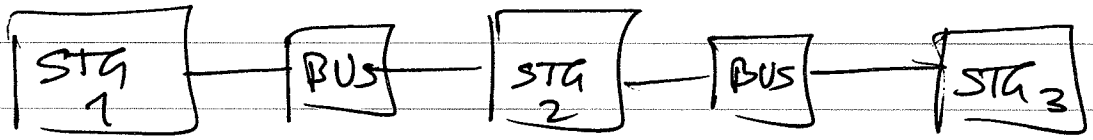
The above approach is called PHASE MODIFICATION.

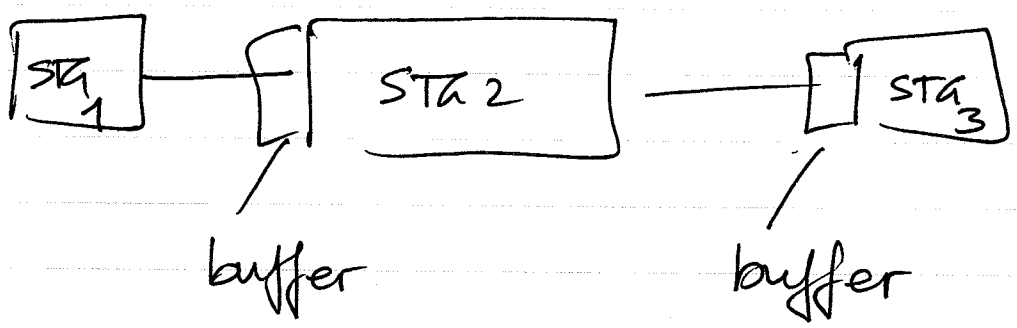
Challenges



redraw

Needs to be real-time to provide latency guarantees.





When to read from the buffer?

A) Read periodically
- Requires some time sync.
but time sync. is hard.

B) Let a_{ijk} be the arrival time of job k of task j at stage i. & let $r_{ij(k-1)}$ be the release time of the previous instance.

$$r_{ijk} = \max \{ a_{ijk}, r_{ij(k-1)} + P_j \}$$

RELEASE GUARD PROTOCOL