6SENG002W Concurrent Programming

FSP Process Composition Analysis & Design Form

Name	Helani Kathyana Waidyarathne
Student ID	2017211/w1673716
Date	2021/01/04

1. FSP Composition Process Attributes

Attribute	Value
Name	SHARED_PRINTER
Description	Models a printer machine, a technician and two students
Alphabet (Use LTSA's compressed notation, if alphabet is large.)	{ st2.aquireStudent, st2.aquireTechnician, st2.print,st2.refill, st2.releasePrinter,st3.aquireStudent,st3.aquireTechnician, st3.print,st3.refill,st3.releasePrinter,t.aquireStudent,t.aquireTechnician , t.print, t.refill, t.releasePrinter, t.wait }
Sub-processes (List them.)	STUDENT, TECHNICIAN, PRINTER
Number of States	55
Deadlocks (yes/no)	No deadlocks/errors
Deadlock Trace(s) **(If applicable)**	None

2. FSP "main" Program Code

The code for the parallel composition of all of the sub-processes and the definitions of any constants, ranges & process labelling sets used. (Do not include the code for the other sub-processes.)

FSP Program:

```
const MAX_PAPER_COUNT=3

const PRINT_MAX_COUNT=3

range PRINT_RANGE=1..PRINT_MAX_COUNT

set ALLACTIONS={aquireStudent,aquireTechnician,print,refill,releasePrinter}

||SHARED_PRINTER=(st3 : STUDENT(3)
                || st2 : STUDENT(2)
                || t : TECHNICIAN
                ||{st3,st2,t} :: PRINTER).
```

3. Combined Sub-processes

(Add rows as necessary.)

Process	Description
STUDENT	The process describes students who print documents. A student can print three documents at once.
TECHNICIAN	The process describes a technician refill the printer when paper count become zero.
PRINTER	The process act as a resource to student and technician. Printer performs print process and refill process in each requirement.

4. Analysis of Combined Process Actions

- **Synchronous** actions are performed by at least two sub-process in the combination.
- **Blocked Synchronous** actions cannot be performed, since at least one of the sub-processes cannot preform them, because they were added to their alphabet using alphabet extension.
- **Asynchronous** actions are preformed independently by a single sub-process.

Group actions together if appropriate, for example if they include indexes,
e.g. in[0], in[1], …, in[5] as in[1..5].

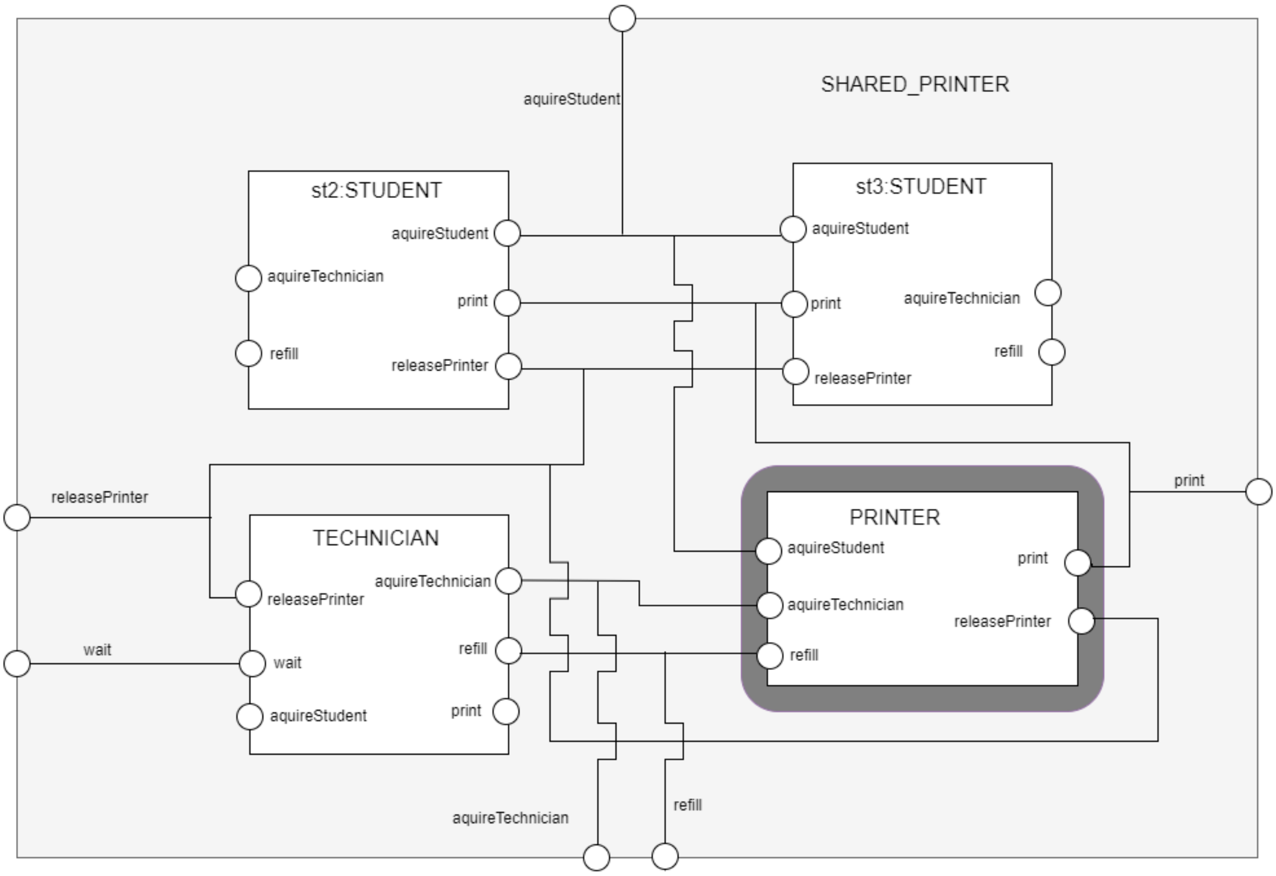
(Add rows as necessary.)

Synchronous Actions	Synchronised by Sub-Processes (List)
st2.aquireStudent,st2.print, st3.aquireStudent,st3.print	STUDENT, PRINTER
t.aquireTechnician, t.refill	TECHNICIAN, PRINTER
st2.releasePrinter, t.releasePrinter, st3.releasePrinter	STUDENT, PRINTER, TECHNICIAN

Sub-Process	Asynchronous Actions (List)
TECHNICIAN	t.wait
PRINTER	none
STUDENT	none

5. Parallel Composition Structure Diagram

The structure diagram for the parallel composition.


aquireStudent
SHARED_PRINTER
st2:STUDENT
aquireStudent
aquireTechnician
print
refill
releasePrinter
st3:STUDENT
aquireStudent
print
aquireTechnician
refill
releasePrinter
print
releasePrinter
TECHNICIAN
aquireTechnician
releasePrinter
PRINTER
aquireStudent
print
aquireTechnician
releasePrinter
refill
wait
wait
refill
aquireStudent
print
aquireTechnician
refill