Critical Application Development
TP1: ESTEREL
EMC5
2016-2017

Exercise 1: Simulation

1. Write the Esterel program simple1.strl composed of the module Mod.

```
module Mod:
   input IN;
   output OUT;
    await IN;
    emit OUT;
end module
```

compile the module for simulation using the command esterel with the option -simul

```
esterel -simul simple.strl
```

compile the generated C code

```
gcc -c -m32 simple.c
```

simulate using xes

```
xes -Xcomp -m32 simple.o
```

2. Add a loop to the program simple.strl and compile and simulate the new module

```
module Mod:
   input IN;
   output OUT;
   loop
     await IN;
     emit OUT;
   end loop
end module
```

3. compile and simulate the following module

```
module Mod:
   input IN: integer;
   output OUT: integer;
      loop
         await IN;
         emit OUT(?IN);
      end loop
end module
```

Exercise 2

Write and test using the simulator xes an Esterel module composed of an input signal named IN, and three output signals TIC, TAC and COUNTER. The signals TIC, TAC and IN are pure signals. The signal COUNTER is a valued (integer) signal. The program has the following behavior: "The signal TIC and TAC are emitted alternately after each reception of the signal IN: TIC, TAC, TIC, TAC ... "

The module is constituted of three parallels branches:

1. The first one waits for the occurrence of a signal IN and emits, using the signal COUNTER, the total number of received IN signals from the beginning of the execution,
2. The second one emits the signal TIC,
3. The third one emits the signal TAC.

Use local signals to synchronize the different branches.

Exercise 3

Write and test using the simulator xes an Esterel program (use two threads) incrementing or decrementing a data. The program is composed of the following signals:

1. The valued (integer) input signal DATE_IN.
2. The valued (integer) output signal DATA_OUT.
3. The pure input signal SWITCH.

At the beginning of the execution, at each reception of DATA_IN, the program increments the value given by DATA_IN. If the signal SWITCH is received, the program will decrement the values given by DATA_IN until the next reception of the signal SWITCH. At the reception of the signal SWITCH, the program will increment the values given by DATA_IN, and so on.

Exercise 4

The goal is to specify the behavior of a phone using an Esterel program. The program contains the pure input signals CALL, HANGUP, PICKUP, NUMBER and SECONDS, the pure output signals CALLFAILURE and BELL, and the valued (integer) output signal TIMECOMMUNICATION.

The phone has two features:

1. **Management of incoming calls**: An incoming call starts when the phone receives the signal CALL. The phone emits every one second an occurrence of the signal BELL. The seconds occurrence is modeled using the signal SECONDS.

   If no one picks up the phone (occurrence of the signal PICKUP) after 20 bells emission, then the signal CALLFAILURE is emitted and the phone waits for a new call.

   If the user picks up the phone, the phone displays at each second the total duration of the communication, until the communication terminates. A communication terminates when the user hangs up the phone (occurrence of the signal HANGUP).

2. **Management of outgoing calls**: An outgoing call starts when the phone receives the signal PICKUP. When the signal PICKUP is received, the phone waits 10 seconds during which the user can compose a telephone number.

   If the user don't give the number before the 10 seconds, the signal CALLFAILURE is emitted. The phone waits until the phone is hanged up, then can treat other outgoing calls.

If the number is given before the 10 seconds, the phone displays every second of the total duration of the communication. A communication terminates when the user hangs up the phone (occurrence of the signal HANGUP).

Exercise 5: Execution

1. Write the Esterel program simple.strl composed of the module Mod.

```
module Mod:
   input IN: integer;
   output OUT: integer;
   await IN;
   emit OUT(?IN);
end module
```

2. Write the following C program in a file simple_main.c

```
#include<stdio.h>

void Mod(void);
void Mod_reset(void);
void Mod_I_IN(int a);

void Mod_O_OUT(int a){
printf("output signal :%d \n", a);
}

int main(int argc, char* argv[]){
  int i;
   Mod_reset();
   Mod();
   for(i=0; i<10;i++){
       Mod_I_IN(i);
       printf("Esterel Program \n");
       Mod();
   }
}
```

compile the file simple.strl for execution

```
esterel simple.strl
```

compile the generated C code and the file simple_main.c

```
gcc -c simple.c
gcc -c simple_main.c
gcc simple.o simple_main.o -o simple
```

execute

```
./simple
```

3. Modify the file simple.strl as shown in the following module, and execute the new program using the same file simple_main.c used before.

```
module Mod:
   input IN: integer;
   output OUT: integer;
   await IN;
   emit OUT(?IN);
   await IN;
   emit OUT(?IN);
   await IN;
   emit OUT(?IN);
end module
```

Exercise 6

1. Download the file calcul.strl (http://www.esiee.fr/~abdedday/calcul.strl) programing a simple calculator displaying the results of operations using the signal LCD. Compile and simulate this program.

2. Download the file calcul_main.c (http://www.esiee.fr/~abdedday/calcul_main.c) and the Makefile (http://www.esiee.fr/~abdedday/Makefile). Compile, execute and test the Esterel program using the Makefile.

3. Modify the program calcul.strl by adding to the calculator a reset function. Use an input signal RESET. When the signal RESET is received, the calculator interrupt the keyboarding and displays 0. To test the new Esterel program, modify the C file calcul_main.c to permit to the user to enter a command for the reset (example: enter the value r for reset).

4. We want to record in a file all the different values emitted by the calculator. Modify the program calcul.strl such that: The program uses two new signals SWITCH and WRITE. The Esterel program has two modes: the recording function can be active or not. By defaults the recording function is not activated. When the signal SWITCH is received, the Esterel program activates the recording function. A second reception of the signal SWITCH deactivates this mode. The output signal WRITE is a valued (integer) signal. The Esterel program uses this signal to add in the recording file a new integer indicating a new display of the calculator. Modify the C file calcul_main.c to permit to the user to enter a command for the recording function (example: enter the value s for recording or to stop the recording).