Digital Systems

Validation, verification

R. Pacalet
October 26, 2018

Exercise #1: temporal logics

✓ $AXEXq$ is true for the left situation and false for the right

CTL specification of the controller

- ✓ Exercise #2: $\neg EF(req \wedge EX(\neg req \wedge EX(\neg pss)))$

CTL specification of the controller

- ✓ Exercise #2: $\neg EF(req \wedge EX(\neg req \wedge EX(\neg pss)))$
- ✓ Exercise #3: $AG(tok \wedge \neg req \Rightarrow AX(pss))$

CTL specification of the controller

- ✓ Exercise #2: $\neg EF(req \wedge EX(\neg req \wedge EX(\neg pss)))$
- ✓ Exercise #3: $AG(tok \wedge \neg req \Rightarrow AX(pss))$
- ✓ Exercise #4: $AG(req \wedge tok \Rightarrow AX(ack))$

CTL specification of the controller

- ✓ Exercise #2: $\neg EF(req \wedge EX(\neg req \wedge EX(\neg pss)))$
- ✓ Exercise #3: $AG(tok \wedge \neg req \Rightarrow AX(pss))$
- ✓ Exercise #4: $AG(req \wedge tok \Rightarrow AX(ack))$
- ✓ Exercise #5: It depends on strong/weak until:
  - ✓ With strong until (the client *will* release the bus): $AG(req \Rightarrow A(\neg pss\ U \neg req))$
  - ✓ With weak until (it can be that the client *never* releases the bus): $AG(req \Rightarrow AX(\neg req \Rightarrow AX(pss)))$

Specification of the whole system

- ✓ Exercise #6: It depends on strong/weak until:
  - ✓ With strong until (the client *will* be granted the bus): $AG(req \wedge \neg ack \Rightarrow A(req\ U\ ack))$
  - ✓ With weak until (it can be that the client is *never* granted the bus): $AG(req \Rightarrow AX(req \vee ack))$

Specification of the whole system

- ✓ Exercise #6: It depends on strong/weak until:
  - ✓ With strong until (the client *will* be granted the bus): $AG(req \wedge \neg ack \Rightarrow A(req\ U\ ack))$
  - ✓ With weak until (it can be that the client is *never* granted the bus): $AG(req \Rightarrow AX(req \vee ack))$
- ✓ Exercise #7: It depends on strong/weak until:
  - ✓ With strong until (the client B *will* be granted the bus): $AG(ackA \wedge reqB \wedge \neg reqA \Rightarrow AX(A(\neg ackA\ U\ ackB)))$
  - ✓ With weak until (it can be that the client B is *never* granted the bus): $AG(ackA \wedge reqB \wedge \neg reqA \Rightarrow AX(\neg E(\neg ackB\ U\ ackA)))$