

Finite statemachines (FSM)

Tilstandsmaskiner

Øvelse 7

Agenda

- VHDL template
- Anbefaling vedr. code-lock opgaven
- Hvordan skal tilstandsdiagrammet for Code-lock opgaven tolkes
- Anbefaling Code-lock vs. UART

VHDL template

- Der ligger en VHDL template på Blackboard



Øvelse 7: Finite State Machines in VHDL ^{ooo}

Øvelsen øger dit kendskab til brug af Mealy og Moore state machines og du får chancen for at implementere en kodelås eller et UART interface i VHDL.

Du skal løse opgave 1 samt opgave 2 eller 3. Har du energien og evnerne, så løs dem alle :-)

Du finder øvelsen her:

[Exercise 7 - Finite State Machines in VHDL \(Code Lock, UART\).pdf](#) ^{ooo}

Tre-process VHDL template fra Lærebogen er [her](#) ^{ooo}

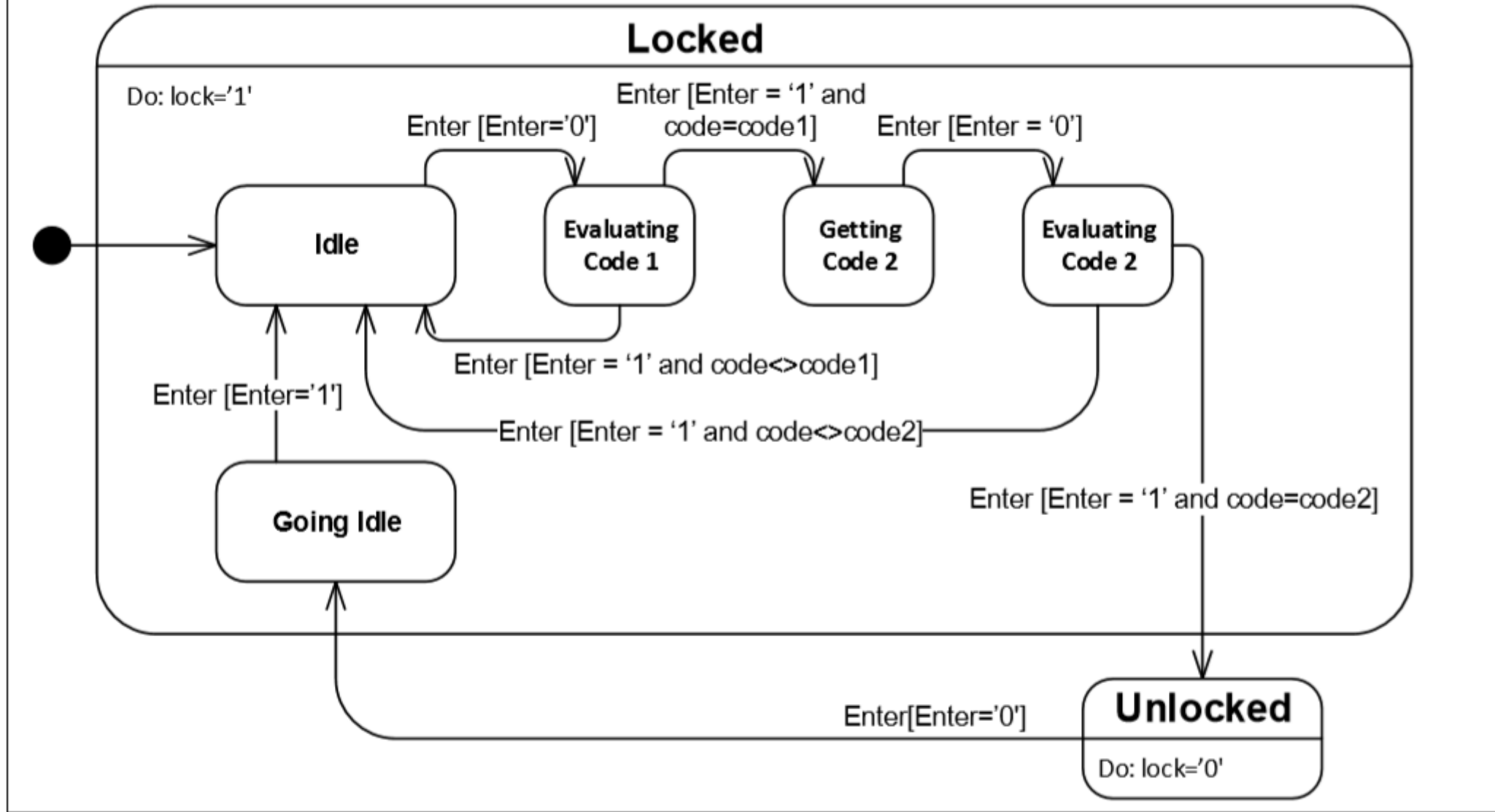


Fig. 4: Code lock simple STM.

Anbefaling vedr. Code-lock opgaven

Hierakisk FSM

```
architecture rtl of code_lock_fsm is

    -- Indre statemachine, eksekveret i "Locked" state
    type state_type_inner is (idle, ev_c1, get_c2, ev_c2, going_idle);
    -- Register to hold the current state
    signal present_state_inner, next_state_inner : state_type_inner;

    -- Ydre statemachine med "Locked" og "Unlocked" states
    type state_type_outer is (locked, unlocked);
    signal present_state_outer, next_state_outer : state_type_outer;
```

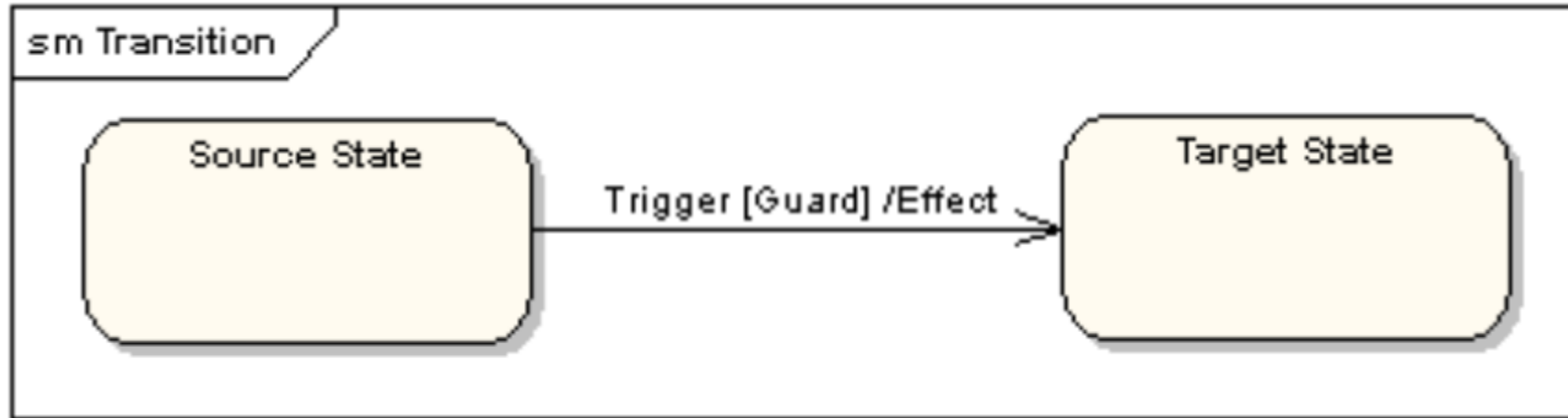
Enkelt FSM

```
architecture rtl of code_lock_fsm is

    -- Indre statemachine, eksekveret i "Locked" state
    type state_type is (idle, ev_c1, get_c2, ev_c2, going_idle, unlocked);
    -- Register to hold the current state
    signal present_state, next_state : state_type;
```

Hvordan skal
tilstandsdiagrammet for code-
lock opgaven tolkes

SysML og statediagram



"Trigger" is the cause of the transition, which could be a signal, an event, a change in some condition, or the passage of time. **"Guard"** is a condition which must be true in order for the trigger to cause the transition. **"Effect"** is an action which will be invoked directly on the object that owns the state machine as a result of the transition.

Hvordan forstår man så det her statediagram?

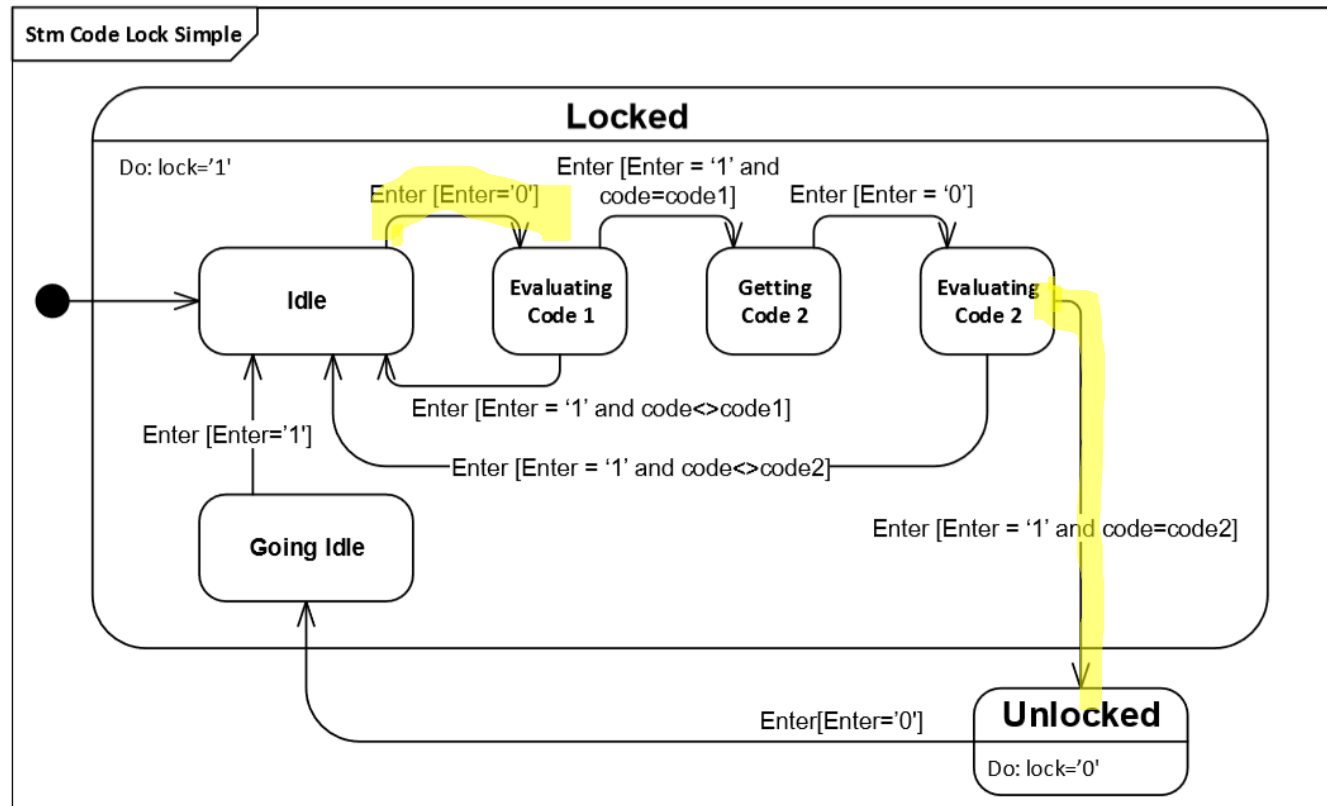


Fig. 4: Code lock simple STM.

Sådan her....

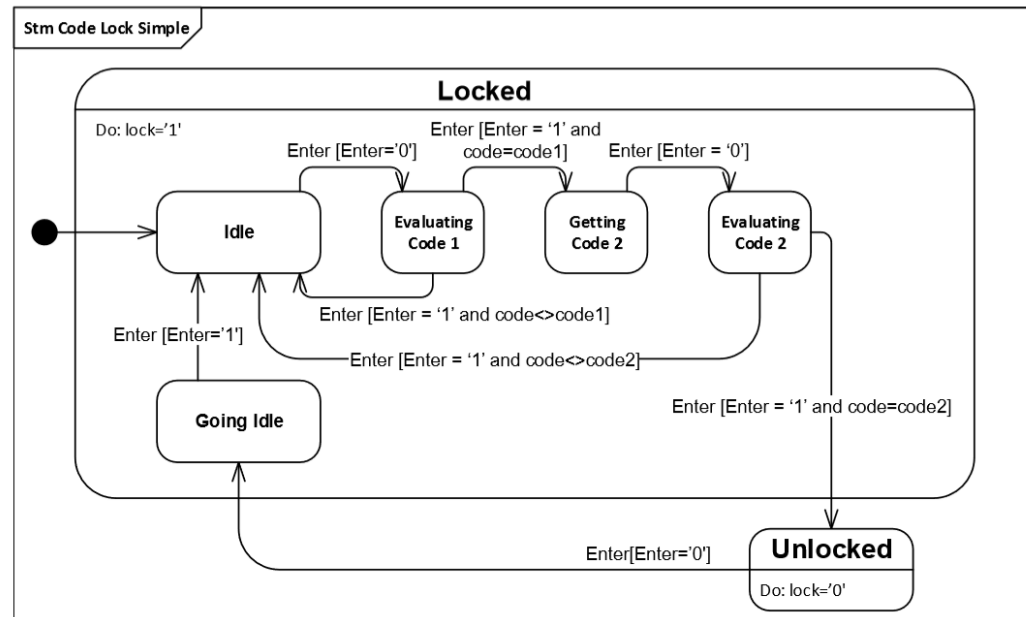


Fig. 4: Code lock simple STM.

- KEY tryk-hold, KEY slip, KEY tryk-hold, KEY slip...
 - Og teste Enter=0, Enter=1,...

Utilsigtede latches

Two screenshots of the Quartus II 64-Bit IDE showing VHDL code for a state machine and its inferred latch warnings.

Left Screenshot: VHDL Code (Mee_Moo.vhd)

```
next_state: process(present_state, a, b)
begin
  case present_state is
    when IDLE =>
      if b = '1' then
        next_state <= INIT;
      else
        next_state <= IDLE;
      end if;

    when INIT =>
      if (a = '1' and b = '0') then
        next_state <= AKTIV;
      elsif (b = '1' and a = '0') then
        next_state <= INIT;
      elsif (b = '1' and a = '1') then
        next_state <= INIT;
      elsif (b = '0' and a = '0') then
        next_state <= INIT;
      end if;

    when AKTIV =>
      next_state <= IDLE;
  end case;
end process;
end;
```

Right Screenshot: VHDL Code (Mee_Moo.vhd)

```
next_state: process(present_state, a, b)
begin
  next_state <= present_state; -- DEF. assignment

  case present_state is
    when IDLE =>
      if b = '1' then
        next_state <= INIT;
      end if;

    when INIT =>
      if (a = '1' and b = '0') then
        next_state <= AKTIV;
      end if;

    when AKTIV =>
      next_state <= IDLE;
  end case;
end process;
end;
```

Bottom Panel: Latch Warnings

The bottom panel shows a list of messages related to inferred latches. The left screenshot shows warnings for the first code version, while the right screenshot shows a message for the second code version.

Type	ID	Message
Warning	12021	Found 2 design units, including 1 entities, in source file Mee_Moo.vhd
Warning	12022	Found design unit 1: mee_moo-three_processes_with_latches
Warning	10631	VHDL Process Statement warning at Mee_Moo.vhd(153): inferring latch(es)
Warning	10041	Inferred latch for "next_state.AKTIV" at Mee_Moo.vhd(153)
Warning	10041	Inferred latch for "next_state.INIT" at Mee_Moo.vhd(153)
Warning	10041	Inferred latch for "next_state.IDLE" at Mee_Moo.vhd(153)
Warning	14026	LATCH primitive "next_state.IDLE_105" is permanently enabled
Warning	14026	LATCH primitive "next_state.INIT_90" is permanently enabled

The right screenshot shows a similar message for the second code version:

Type	ID	Message
Warning	12021	Found 2 design units, including 1 entities,
Warning	12022	Found design unit 1: mee_moo-three_processes

Fejlsøgningsmetodik

Formel procedure: hvad hjælpelærer vil spørge om

1. Hvad virker ikke?
2. Hvor kan fejlen isoleres til?
3. Hvordan vil du kunne teste at det virker?
4. (Hvad skal ellers undersøges for at rette fejlen?)

