# Labs for Module 3

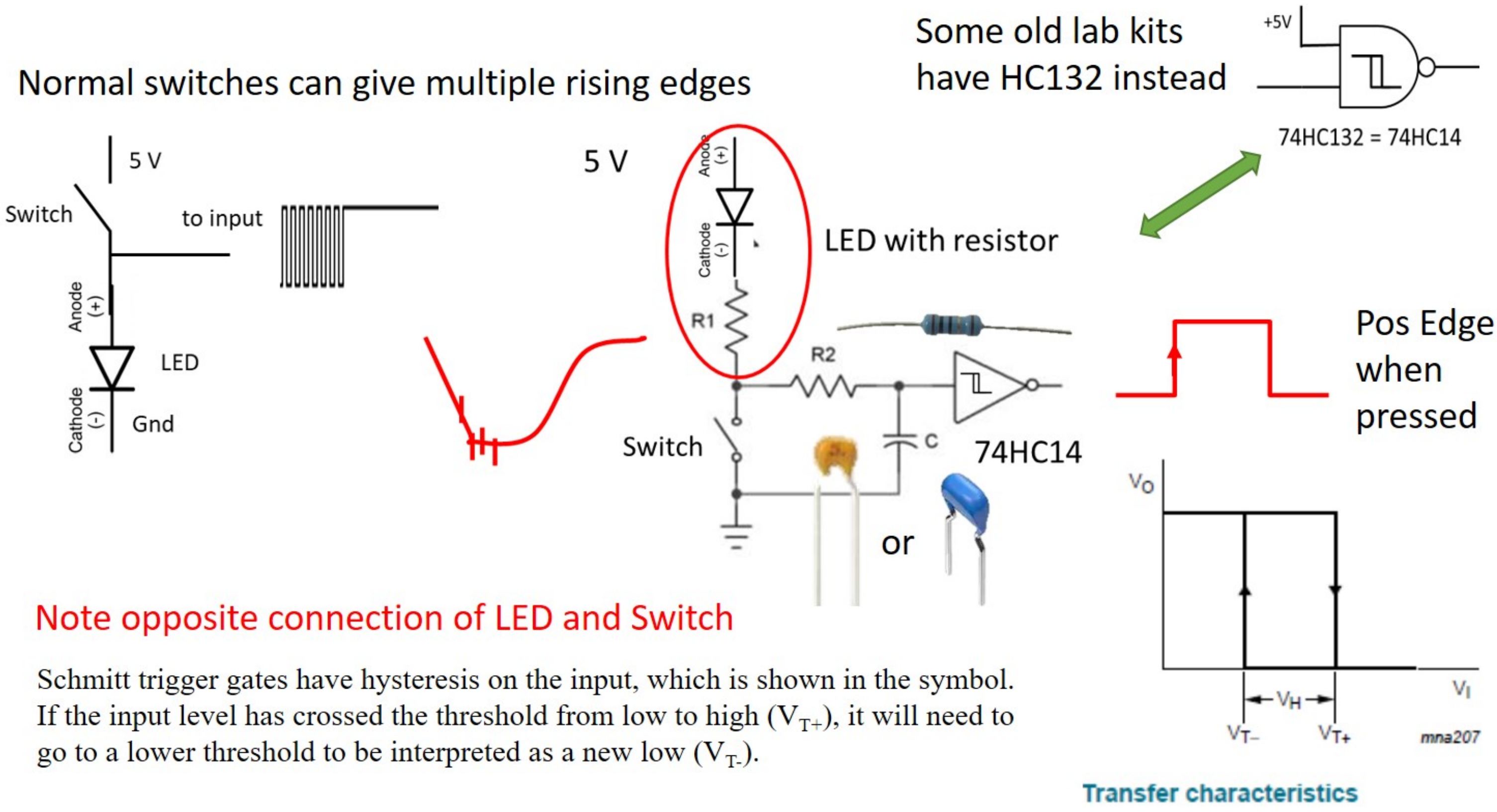
## The compulsory exercise is further down in blue, but you must build the Schmitt trigger clock

**NOTE: Don’t forget that you can use Logisim to simulate most of these circuits (not the Schmitt Triggers or oscillators with capacitors, but all Flip-Flops are available).**

## Preparation: Clock signal using a Schmitt Trigger with Hysteresis

If you just connect a wire to the CLK input of a sequential circuit it will pick up noise and generate multiple pulses, and you will not be able to test the operation properly. The pushbutton switches from module 1 and 2 will also cause extra pulses due to switch bouncing. For the tests in Module 3 you need a clock generator that supplies single clock pulses with a well-defined positive (rising) and negative (falling) edge. This can be built using the 74HC14 hex Schmitt trigger (or a 74HC132 quad NAND Schmitt trigger with one pin connected high), by adding a capacitor and a resistor, see schematic below. The resistor R1 can be the LED with built in resistor from the previous circuit, but R2 should be one of the resistors in the Lab kit. When the pushbutton switch is closed you get a clean rising edge at the output, and the signal returns to zero when the switch is released.

**Note that 74HC00 or 74HC04 can’t be used even though they have the same pinout.**



Note opposite connection of LED and Switch

Schmitt trigger gates have hysteresis on the input, which is shown in the symbol. If the input level has crossed the threshold from low to high ($V_{T+}$), it will need to go to a lower threshold to be interpreted as a new low ($V_{T-}$).

Transfer characteristics

**Explanation:** These are inverters with an input that has hysteresis, and they can be used as standard inverters too. The hysteresis is shown in the symbol, and means that if the input level has crossed the threshold from low to high ($V_{T+}$), it will need to go to a lower threshold to be interpreted as a new low ($V_{T-}$), see figure of the transfer characteristics from the 74HC14 datasheet.

## *Testing Latches (suggested exercises, may use ordinary pushbuttons)*

- Build and test an SR-latch using two NOR gates
- Build and test an S'R'-latch using two NAND gates
- Build a gated SR-latch using four NAND gates
- Build and test a D-latch by adding an inverter to the gated SR-latch

See Lecture 9 slides and recording for help

## *Testing Flip-flops (suggested exercises, use Schmitt trigger clock)*

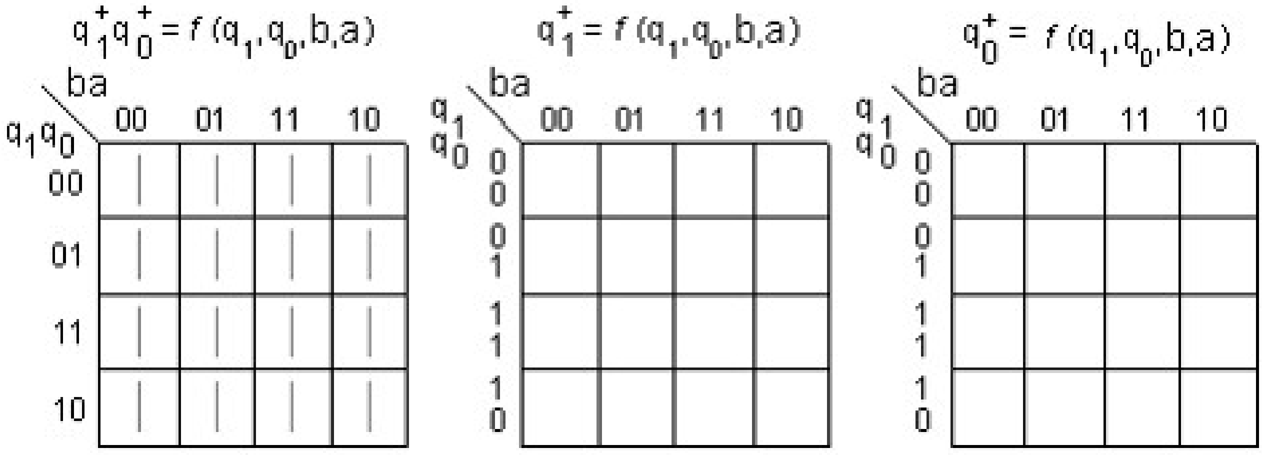
- Connect and test the JK' Flip-Flops (74HC109, connect Preset and Clear to 5 V, note K is inverted)
- Make a T Flip-Flop from the 74HC109
- Make a D Flip-Flop from the 74HC109
- Connect and test the D Flip-Flops (74HC74, connect Preset and Clear to 5 V)
- Can you make a T Flip-Flop from the 74HC74 and an XOR? MUX?
- Connect Q' (not Q) to D to make a 1 bit binary counter

See Lecture 9 slides and recording for help

## Module 3 task to report in the written Lab Report and demonstrate

- Find the state diagram corresponding to your Month and Date of Birth in the appendix **FSM Calendar 2022 (https://canvas.kth.se/courses/36215/files/5931531?wrap=1)** ↓ **(https://canvas.kth.se/courses/36215/files/5931531/download?download_frd=1)** . Make a note of its number.
- If you have nr 1, 6 or 9 in the FSM calendar you will need to add set circuitry to reach all states. These three have isolated states that you can only reach if you start in them. Use a separate button or clock signal that can ground the ~PRE pin(s) on the 74HC74.
- Draw the state table in the format below.
- Make K-maps to simplify the expressions for $q_1^+$ and $q_0^+$
- Hint: For $q_1^+$ use a minimized **SOP** form with an AND-OR network
  Hint: a three input OR should suffice but you need some three or four input ANDs
- Hint: For $q_0^+$ use a minimized **POS** form with an OR-AND network, circle “0” groups
  Hint: a three input NOR (OR / NOR combination) should suffice.
- You may use other ways to solve the problem than SOP/POS, for instance a **MUX** (74HC253).
- Check your K-map results using Logisim.
- Build the state machine using two D Flip-flops (74HC74) and any logic gates from the design kit (there should be enough if you follow the instructions above). Make sure the ~PRE and ~CLR inputs are connected high (5V).
- **No output logic is needed; just connect LEDs to $q_1$ and $q_0$.**

- **Connect LEDs to $q_1^+$ and $q_0^+$ for debugging.**
- **Connect a single pulse generator pushbutton to the clock input of both DFFs and test the state machine operation (use the Schmitt trigger clock above).**
- Check your FSM operation with Logisim.
- Take a photo of your working circuit.
- Demonstrate this circuit to a Lab Assistant.



$q_1^+q_0^+ = f(q_1,q_0,b,a)$

| $q_1q_0$ \ ba | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | \| | \| | \| | \| |
| 01 | \| | \| | \| | \| |
| 11 | \| | \| | \| | \| |
| 10 | \| | \| | \| | \| |

$q_1^+ = f(q_1,q_0,b,a)$

| $q_1$ $q_0$ \ ba | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 0 | | | | |
| 0 1 | | | | |
| 1 1 | | | | |
| 1 0 | | | | |

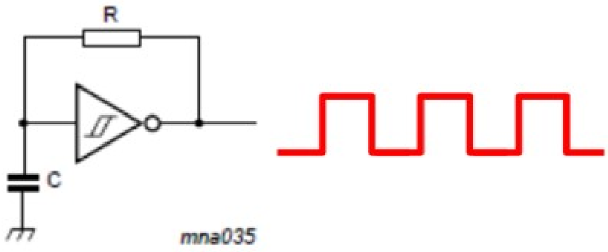
$q_0^+ = f(q_1,q_0,b,a)$

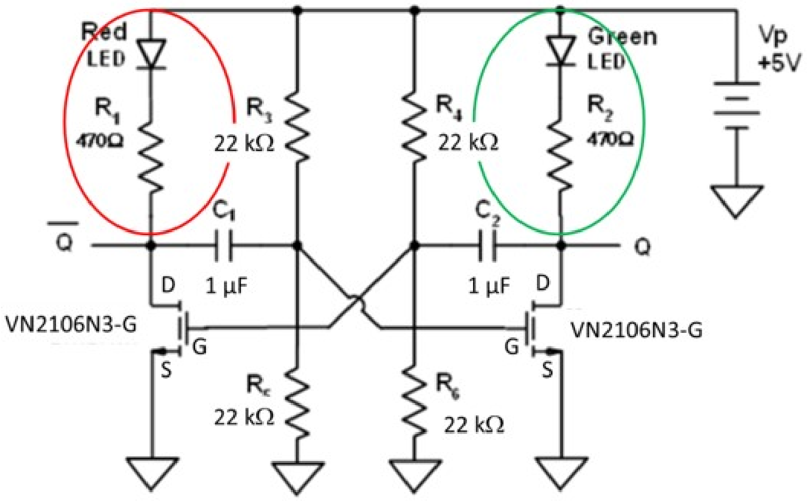
| $q_1$ $q_0$ \ ba | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 0 | | | | |
| 0 1 | | | | |
| 1 1 | | | | |
| 1 0 | | | | |

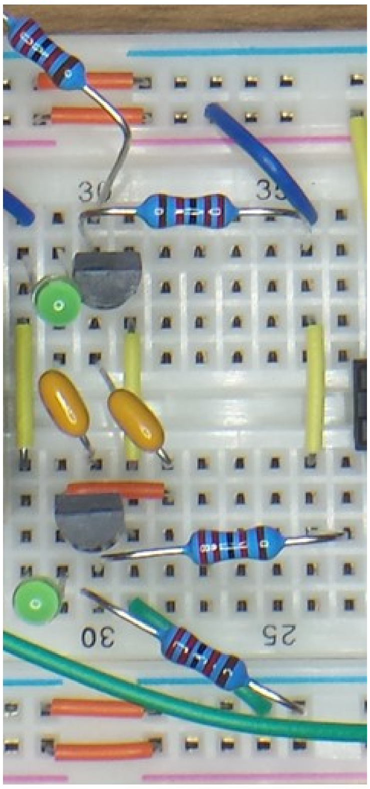
# **Building and testing free running clock generators / oscillators*

** = extra exercise if you are finished with everything else*

- Connect a 74HC14 or 74HC132 Schmitt trigger with a capacitor and a resistor to make an oscillator/free running clock generator
- If you connect an LED it will flicker and you can't determine the blinking speed. By connecting it to the 74HC161 binary counter you can determine its frequency, since each successive output stage has half the frequency.
- Build the astable multivibrator with two NMOSFETs and other components, and test it in the same way (this one is a little trickier).
- Try to capture the signals with the logic analyzer from.


R
C
mna035


Red LED
R1 470Ω
R3 22 kΩ
R4 22 kΩ
Green LED
R2 470Ω
Vp +5V
C1
C2
1 µF
1 µF
Q̅
Q
D
D
G
G
S
S
VN2106N3-G
VN2106N3-G
Rc 22 kΩ
R6 22 kΩ


30
35
30
25