

UART Regex

Should match valid UART frames in a sequence of ones and zeros

"111110101010100111111111" "10010011001100100011111" "11111010101010101111111"

The regular expressions should match once in the first test sequence, twice in the second and never in the third

Enable native scrolling

1/7

UART Regex

The regex consists of the following parts:

```
0 // Match exactly the start bit
[01]{8} // Match on the eight data bits
[01] // Match on the parity bit (withou checking it)
1 // Match exactly on the stop bit
```

There is no need to check for the presence of a second stop bit as we are already in the accept state

The complete solution is:

```
0[01]{8}[01]1
```

2/7

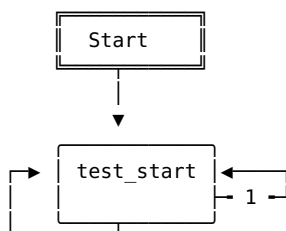
UART Regex

To get the actual data bits out of the regular expression we can optionally group them into a capturing group.

```
0([01]{8})[01]1
```

3/7

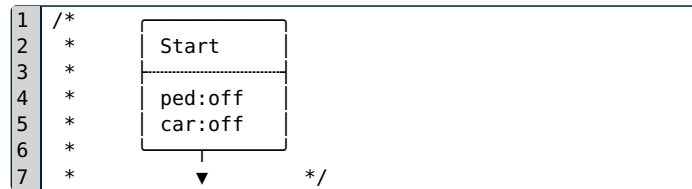
UART FSM



...

4/7

Traffic light FSM



The FSM diagram is included as a comment in the [source code](#)
The machine is designed to be a [moore machine](#), every state explicitly states the output values and following state

5/7

Traffic light implementation

```
1 int isctn_step(int state_old)
2 {
3     switch(state_old) {
4     case ISCTN_START:
5         tl_car(TL_OFF);
6         tl_pedestrian(TL_OFF);
7         state_new= ISCTN_PED_GREEN;
8         break;
```

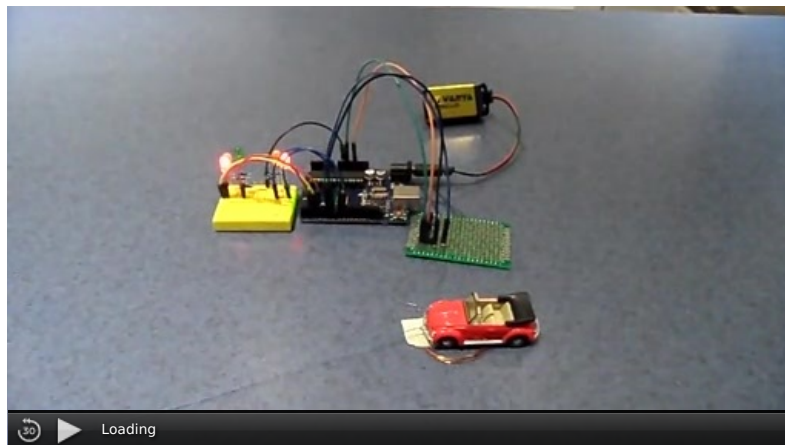
The `isctn_step` function is the so called transfer function, it takes the current state of the machine and outputs the following state based on the inputs and the time

A pure state machine can not use delays in the transfer function

6/7

Traffic light implementation

As a little gimmick the implementation uses an inductive coil in the "road" to detect cars



The [source code](#) contains a schematic of the circuit used

7/7