

Topic: Coach button
Author: Søren Haxvig
Date: 2023.07.25
To: OEM Partners
Purpose: Demonstrate how to use the logic engine to activate a program manually by pushing a button connected to a sensor decoder.

Button input:

The sensor decoder setup has a setup to catch the button press. The setup of the IO configuration is explained at the end of the document.

Control of program activation:

The button push shall start a program manually via a POST action. The controller only allows a program to be started once so the logic needs to prevent the program from being started more than one time in case the button is hit multiple times within a short time. The current logic engine version doesn't give access to the state of a program meaning it is not possible to read if the program is already activated. Thus, the solution is made with a delay of 300 seconds between activations. This time shall be set to minimum the length of the program activated.

Variables:

	Id	Description	State	Type	Source id	Source sub id	Value
	1	Two wire state	Ok	Tw state	0	0	2
	2	Push button value	Ok	IO value	679088867	1	0
	3	Program state	Ok	Local	1	1	0
	4	Program number	Ok	Local	0	0	4

Variable #1 is the state of the 2-wire. It is important to make sure the 2-wire is operational while reading and reacting on sensors.

Variable #2 is the input from a sensor decoder.

Variable #3 is a manually/ timer controlled of the state of the program.

Variable #4 is the program to activate.

The coach program is made with a variable for easy modifications on the variable page or via the API. Using the API:

```
PATCH /le/var/4/value=4
{"result": "ok"}
```

Block:

▼ if: \$1 == 2 && \$2 && !\$3

log info: Program \$4 activated by push button

action: POST /we/program/\$4/start

set: \$3 = 1

▼ while: \$2

sleep: 1

▼ if: \$1 == 2 && \$3

sleep: 300

set: \$3 = 0



The first “if” statement checks that the 2-wire is operational, and the button is active, and the program is not already activated.

The program is activated, and the state is set true.

The while block waits until the button is released.

The 2nd “if” block checks the program is activated and waits 300 second.

Sensor decoder setup:**IO configuration:**

Protocol	mk3	
Type	input	
Description	Push button	
Expression to modify interrupt value stored in object	x	
x gets replaced by value		
Secondary expression, usage depends on context	0	
x gets replaced by value		
Reset to value to 0 after X seconds without interrupt, 0 = indefinitely	0	

GENERAL	INTERRUPTS	TIMINGS
Input power source		
12V		
Input type		
Switch		
Sensor operation mode		
Continuous mode, constant on		
Current control	1.15	mA
23		
Digital trigger level		
1/4		
Digital debounce time	488	us
2		

GENERAL

INTERRUPTS

TIMINGS

Interrupt rate limit (min. time between interrupts)

1

s

Minimum value change (since last interrupt) for interrupt to be sent

0

The setting above will update the value every 1 second if there is a change.

Send interrupt for selected values

☒ Measurement Value☐ Measurement Minimum Value☐ Measurement Maximum Value☐ Digital count☐ Milliseconds since last digital input☐ Attention

This setup ensures the sensor decoder sends interrupt when the reading changes.