

Topic: Fill water tank
Author: Søren Haxvig
Date: 2023.07.14, revised 2023.08.01
To: OEM Partners
Purpose: Demonstrate how to use the logic engine to control filling a water tank based on measured level.

Level input:

The sensor decoder setup has an expression to convert the input signal to a level. The setup of the IO configuration is explained at the end of the document.

Control of water fill:

The tank is supposed to be filled when level drops below a certain value and continue until it has reached the top level. The control must account for fluctuations in the reading due to variations in the surface caused by e.g. wind.

Variables:

	Id	Description	State	Type	Source id	Source sub id	Value
	1	Two wire state	Ok	Tw state	0	0	2
	2	Pressure Transducer value	Ok	IO value	679086867	1	653
	3	Well Pump value	Ok	IO value	671086775	1	0
	4	Timer	Ok	Local	0	0	0
	5	Count till time	Ok	Local	0	0	0
	6	Refill trigger level	Ok	Local	0	0	400
	7	Refill to level	Ok	Local	0	0	430
	8	Trigger delay (seconds)	Ok	Local	0	0	60

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. In this example the conversion via the IO configuration is made to convert the 4-20mA signal into 1/10th of an inch (See conversion setup at the end of this document).

Variable #3 is the output decoder controlling the well pump.

Variable #4 is a timer used to make sure the sensor reading is stable before doing any actions.

Variable #5 is a counter for the fill time.

Variable #6 is the refill trigger level.

Variable #7 is the refill to level.

Variable #8 is the trigger delay.

The 2 level indicators and trigger delay are made with variables for easy modifications on the variable page or via the API. Using the API:

```
PATCH /le/var/6/value=400
{"result": "ok"}
PATCH /le/var/7/value=430
{"result": "ok"}
PATCH /le/var/8/value=60
```

```
{"result": "ok"}
```

Block:

▼ if: \$1 == 2 && ((\$2 && (\$2 < \$6)) \$3)	↑ ↓ + ✎ 🗑
▼ if: \$3 && !\$4	↑ ↓ + ✎ 🗑
log info: Well Pump manually activated at \$2 inch	↑ ↓ + ✎ 🗑
set: \$4 = \$8	↑ ↓ + ✎ 🗑
▼ while: \$2 && (\$2 < \$6) && !\$3 && \$4	↑ ↓ + ✎ 🗑
set: \$4 = \$4 - 1	↑ ↓ + ✎ 🗑
sleep: 1	↑ ↓ + ✎ 🗑
▼ if: !\$3 && !\$4	↑ ↓ + ✎ 🗑
action: POST /u/671088775/1/on	↑ ↓ + ✎ 🗑
log info: Well Pump started at \$2 < \$6 inch	↑ ↓ + ✎ 🗑
set: \$5 = 0	↑ ↓ + ✎ 🗑
▼ while: \$2 && (\$2 < \$7) && \$3	↑ ↓ + ✎ 🗑
sleep: 1	↑ ↓ + ✎ 🗑
set: \$5 = \$5 + 1	↑ ↓ + ✎ 🗑
set: \$4 = \$8	↑ ↓ + ✎ 🗑
▼ while: \$2 && (\$2 >= \$7) && \$3 && \$4	↑ ↓ + ✎ 🗑
set: \$4 = \$4 - 1	↑ ↓ + ✎ 🗑
sleep: 1	↑ ↓ + ✎ 🗑
set: \$5 = \$5 + 1	↑ ↓ + ✎ 🗑
▼ if: !\$2	↑ ↓ + ✎ 🗑
log warning: Unable to read level from pressure transducer!	↑ ↓ + ✎ 🗑
▼ if: \$1 != 2	↑ ↓ + ✎ 🗑
log info: Two-wire disabled	↑ ↓ + ✎ 🗑
▼ if: (\$1 == 2) && !\$3 && (\$5)	↑ ↓ + ✎ 🗑
log info: Well Pump manually deactivated \$2 in \$5 sec	↑ ↓ + ✎ 🗑
set: \$5 = 0	↑ ↓ + ✎ 🗑
▼ if: \$3 && !\$4	↑ ↓ + ✎ 🗑
action: POST /u/671088775/1/off	↑ ↓ + ✎ 🗑
log info: Well Pump stopped at \$2 > \$7 in after \$5 sec	↑ ↓ + ✎ 🗑
set: \$5 = 0	↑ ↓ + ✎ 🗑

The first “if” statement checks that the 2-wire is operational, and the level is below the trigger level, or the well pump is already activated.

The first while block checks that the level stays below the limit for one minute.

The 3rd “if” block activates the well pump.

The 2nd while block loops until the level is above the refill level.

The 3rd while block checks that the level stays above the refill to level.

Last “if” block closes the well pump.

Automatic run of a tank fill:

Timestamp	Severity	Type	Data
Tue, 01 Aug 2023 12:47:44	Info	le::user::log	logStr: Well Pump stopped at 441 > 430 in after 592 sec
Tue, 01 Aug 2023 12:47:44	Info	io::off	addr: 671068775, idc: 1, verify: 1, initialLoad: 2, initialLineVoltage: 41621, finalLoad: 2, finalLineVoltage: 41621, errCode: 0, errDetail: 0
Tue, 01 Aug 2023 12:37:49	Info	le::user::log	logStr: Well Pump started at 312 < 400 inch
Tue, 01 Aug 2023 12:37:49	Info	io::on	addr: 671068775, idc: 1, holdTime: 0, verify: 1, ping: 0, force: 0, switchParams: , initialLoad: 1, initialLineVoltage: 41621, finalLoad: 2, finalLineVoltage: 41608, verifySum: 2599, errCode: 0, errDetail: 0

Manual run/intervention of a tank fill:

The logic shown above allows the user to start or stop the well pump manually from the Two-wire/IO section. If it is started and the level is above the refill to, then it will automatically stop after 60 seconds. Likewise, if it is stopped and the level is below the refill trigger, then the well pump will be started again after 60 seconds. Example:

Timestamp	Severity	Type	Data
Tue, 01 Aug 2023 12:59:02	Info	le::user::log	logStr: Well Pump started at 281 < 400 inch
Tue, 01 Aug 2023 12:58:00	Info	le::user::log	logStr: Well Pump manually deactivated 277 in 117 sec
Tue, 01 Aug 2023 12:56:03	Info	le::user::log	logStr: Well Pump started at 277 < 400 inch
Tue, 01 Aug 2023 12:54:30	Info	le::user::log	logStr: Well Pump stopped at 441 > 430 in after 60 sec
Tue, 01 Aug 2023 12:53:28	Info	le::user::log	logStr: Well Pump manually activated at 441 inch

Sensor decoder setup:**IO configuration:**

Protocol	mk3	
Type	input	
Description	4-20mA transducer (tenths of an inch)	
Expression to modify interrupt value stored in object	$1384.8 * (x - 4000) / 16000$	
x gets replaced by value		
Secondary expression, usage depends on context	x	
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		
4-20mA		
Sensor operation mode		
Continuous mode, constant on		
Current control	24.05	mA
Analog processing		
0-20 mA, value in uA, steps of 50		
Analog high measure level	uA	
20000		
Analog low measure level	uA	
0		

The expression converts the 4000-20000uA to 0-1384.8 1/10th inches.

The Input power source of 12V might differ depending on the actual sensor.

GENERAL	INTERRUPTS	TIMINGS
Interrupt rate limit (min. time between interrupts)		
5 s		
Minimum value change (since last interrupt) for interrupt to be sent		
0		

The setting above will update the value every 5 seconds regardless of there is any change. A minimum uA change could be defined to avoid too many interrupts on the 2-wire if the sensor fluctuates.

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.