

Uitnodiging Fgg 11 dec. 2021

Verslag van de werkgroepen:

- 1) Status maas netwerk**
- 2) Project Forth Works initiatief**

Lezingen:

- 1) noForth button versie-2**

Willem Ouwerkerk

- 2) Project Forth Works voorbeeld**

Willem Ouwerkerk

<https://project-forth-works.github.io/>

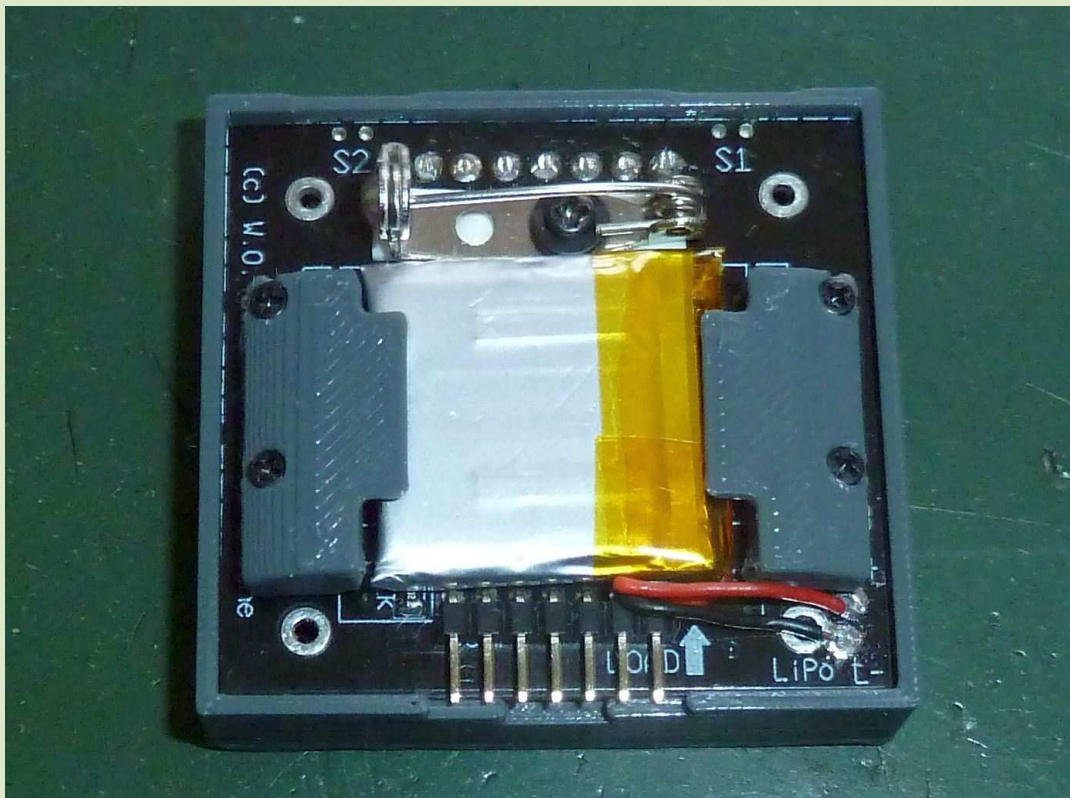
- 3) Snelste Route op een netwerk**

Henny Luijkx

SPI drivers in actie op noForth button 2



Voorkant met OLED display



Speld, accu en laad/programmeer connector

Project Forth Works – Bit array

Vandaag lopen we door de de bit-array data structuur heen, zoals gepubliceerd via P.F.W.

Bit array data structure

Bit array idea

To store many on/off events or a present/not present note, we could use an array of bit flags to save space.

Implementation

The implementation assumes a byte addressed machine. The number of bits is rounded up to the number of cells needed to store the flags. The cell size in bytes is used and multiplied by 8 to calculate the number of bits in a cell.

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0

This is a one cell array on a machine with 16-bit cell size. Three of the 16-bits are high. Note that only 4 cells (8 bytes) are required to keep track of 64 flags.

A note on `?ABORT` noForth uses the name of the word in which `?ABORT` is included as the error message. In the example code below, the message "Error from LOC" is printed. This is useful for adding clear error messages to small embedded systems. It is also an extra motivation to use meaningful names :)

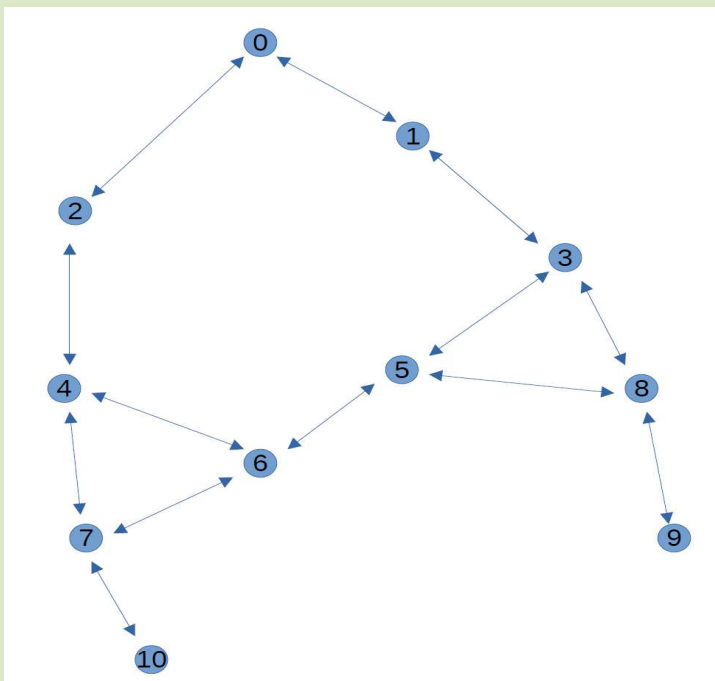
De snelste route in een draadloos netwerk van nrf24 modules.

Communicatie in een draadloos netwerk is afhankelijk van de topologie. Modules die zich dicht genoeg bij elkaar bevinden kunnen rechtstreeks communiceren. Communicatie met modules die zich verder weg bevinden gaat via andere modules.

In het voorbeeld kan module 0 direct met module 1 en 2 communiceren. Voor de overige modules communiceert module 0 via module 1 en/of 2. Hoe het netwerk voorbij deze modules eruit ziet weet module 0 niet. Hierdoor weet een module niet langs welke weg het snelste met een andere module te communiceren.

Om onnodige communicatie te voorkomen is een routetabel geïntroduceert. De tabel ziet er als volgt uit:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH	VH



De eerste regel is de module waarmee wordt gecommuniceert. De tweede regel geeft via (V) welke module deze bereikbaar is en hoeveel modules (H) worden gepasseerd om de gewenste module te bereiken.

Hoe de tabel en een tabel voor een alternatieve route wordt opgebouwd wordt in de lezing verder uitgelegd.