Daisy Chain Protocol Development in Force-Guiding Particle Chains for Shape-Shifting Displays

Developmet Board Implementation

Raoul Rubien
rubienr@sbox.tugraz.at

Institute for Technical Informatics
Graz University of Technology

5th August 2016

What this session covers

physical network structure
MCU selection
development board

Underlying work

Force-Guiding particle Chains for Shape-Shifting Displays[1]

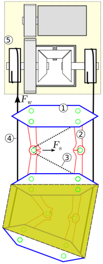


Figure 1: *particle chain*

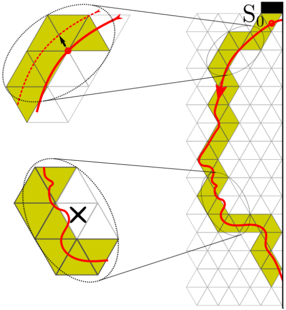


Figure 2: *folding a shape*

chain is stretched in natural state
shape Memory Alloy used as actuator (3)
joints unlocked by actuators (2)
force F_s folds chain

Approach & limitation

current particle implements 1-Wire via power supply wires
energy must be buffered before communication starts
power must be switched off/on
automatic chain position detection is costly

Idea

decouple communication from power supply (4)
using a daisy-chain protocol, and
actuator wires (3)

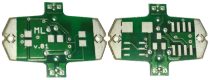


Figure 3: *particle PCB*

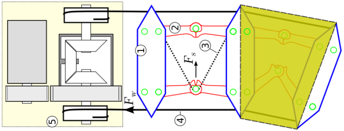


Figure 4: *particle chain*

Project extent

particle board development
easy accessible test points and transmission wires
flexible and fast network assembly

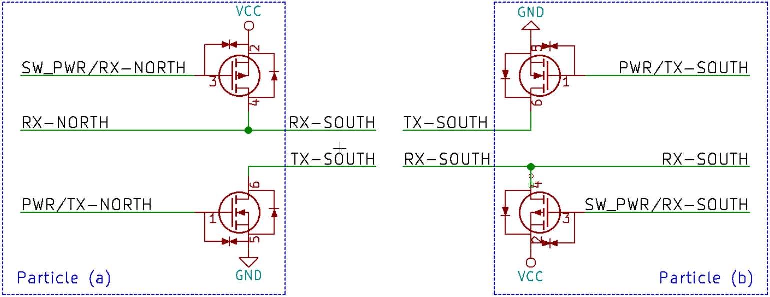
Project constraints

reasonable low level MCU
minimize number of components on particle PCB
communication:
- exploit SMA wires
- decouple from power supply

small MCU package in final productive particle
single communication entry point to the network

Network approach - I

exploit actuator wires
also for communication



Network approach - II

linear network

daisy chained participants

Advantages

+ simple to implement
+ no media access control
+ no loops
+ no dynamic routes

Disadvantages

- error detaches segment
- no recovery for segment

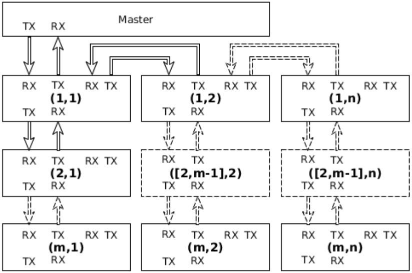


Figure 5: *network topology*

MCU requirements - capabilities

- three separate external interrupts
- self programmable EEPROM
  - for firmware replication (future work)
- package size
  - small package for productive particle
  - bigger package for development board

MCU memory requirements - upper bound estimation

Flash

expected max. firmware size:	$\sim 4k$ SLOC
estimated object code bytes per SLOC [2]	$\sim 4.0B$
flash usage estimation:	
$4k * 4B =$	**$\sim$16kB**

SRAM

tx/rx buffers: 3 ports, 8byte	
$3 * 8B * 2 =$	$16B$
Manchester code decoding buffer	
with 2 flank time stamps per bit	
$3 * 8 * 2 * sizeof(uint16\_t)B * 0.75 =$	$576B$
other global variables	$200B$
stack: max. 50 nested void function	
calls with $\sim (1 * uint8\_t)$ argument	
$50 * (1 + 2)B =$	$150B$
SRAM estimation:	**$\sim$950B**

Candidates

candidates are all ATTiny20 family MCUs having

$\geq 16kB$ flash and

$\geq 1kB$ SRAM

Comparison of used MCUs

	ATTiny20 (proof of concept)	**ATTiny1634**
# pin change int.	sufficient	sufficient
EEPROM	no	yes
flash	$2kB$	$16kB$
SRAM	$128B$	$1kB$
small package	$3mm \times 3mm$	$4mm \times 4mm$
alternative pkg.	no	yes, SOIC

The prototype

- not satisfying new requirements
- too small/unhandy for development

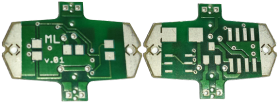


Figure 6: *prototype*

Version 1.0

linear chain of development particles
+ not mounted in chain mechanics
- but still time consuming assembly

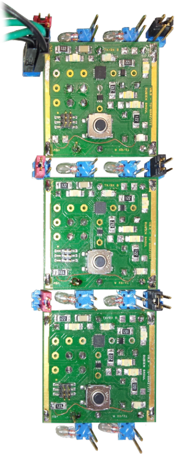


Figure 7: *Version 1.0*

Version 1.1

Advantages

+ repetitive design
+ configurable network shape

Disadvantages

- costly soldering
- expensive connectors
- one faulty particle breaks whole PCB

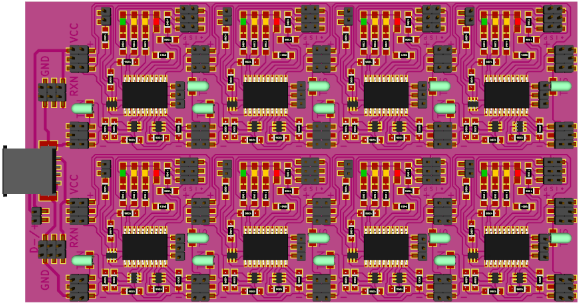


Figure 8: *Version 1.1 - particle array PCB*

Version 1.21

configurable network dimension
easy extensible network
faulty particles can be replaced
cheaper
higher particle density

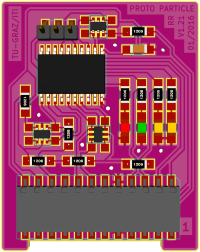


Figure 9: *pluggable particle*

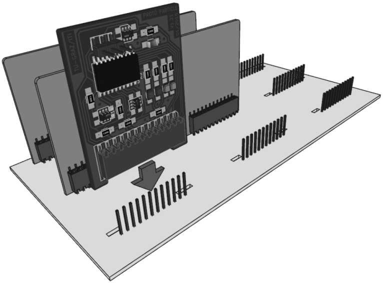


Figure 10: *grid board*

Results

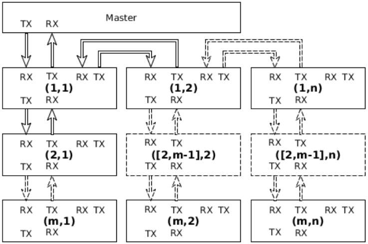


Figure 11: *network structure*

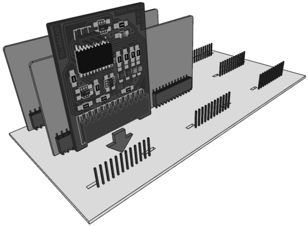
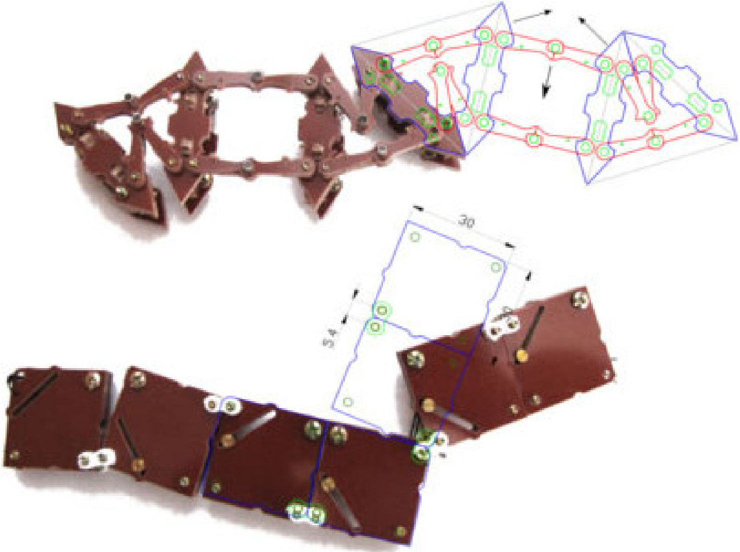


Figure 12: *pluggable particle module*


30

Future work

hardware
- simplify development board
- enhance grid board

communication protocol
- Physical Layer: implement coding
- Data Layer: fault detection
- Network Layer: self enumeration, addressing,
  - synchronization, clock compensation
  - task scheduling (TDM of communication and tasks)

runtime compensation of RC-oscillator discrepancy

firmware replication: customize boot loader

M. Lasagni and K. Römer, "Force-guiding particle chains for shape-shifting displays," *CoRR*, vol. abs/1402.2507, 2014.

J. Ganssle, *Embedded systems*.
Amsterdam Boston: Elsevier/Newnes, 2008.