

Modernizing a Pinball machine

Foreword

- Some ideas taken from BLEnky
- Focus on general principle

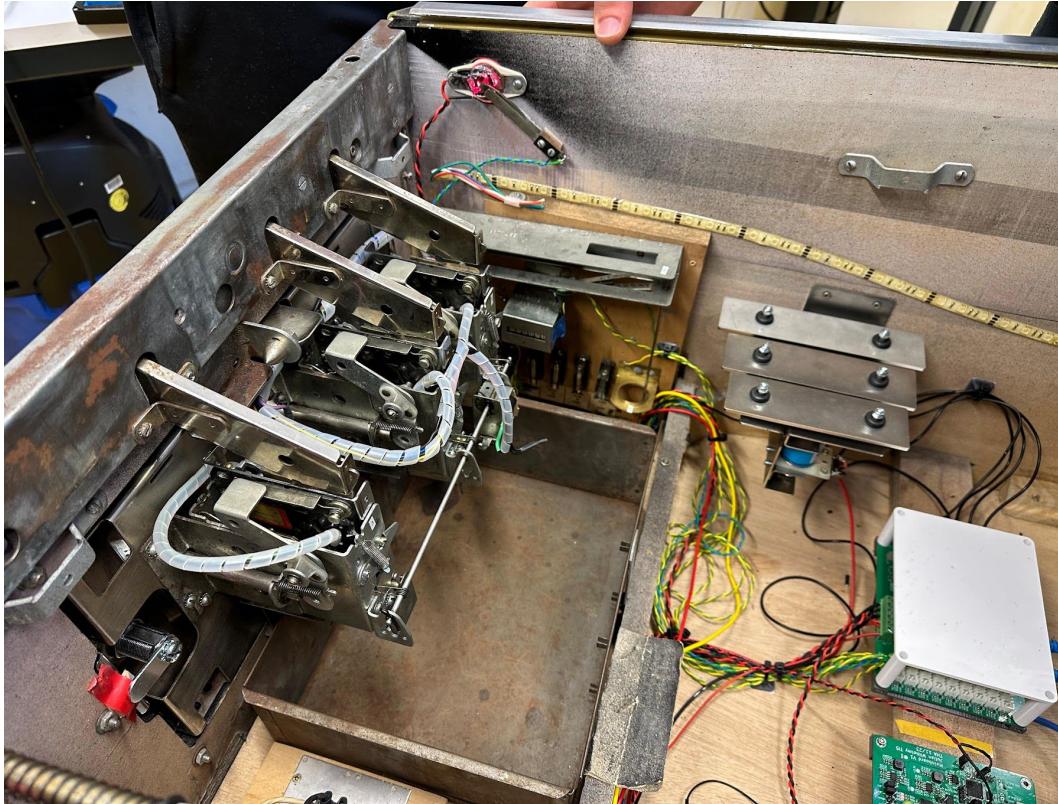
Machine in question



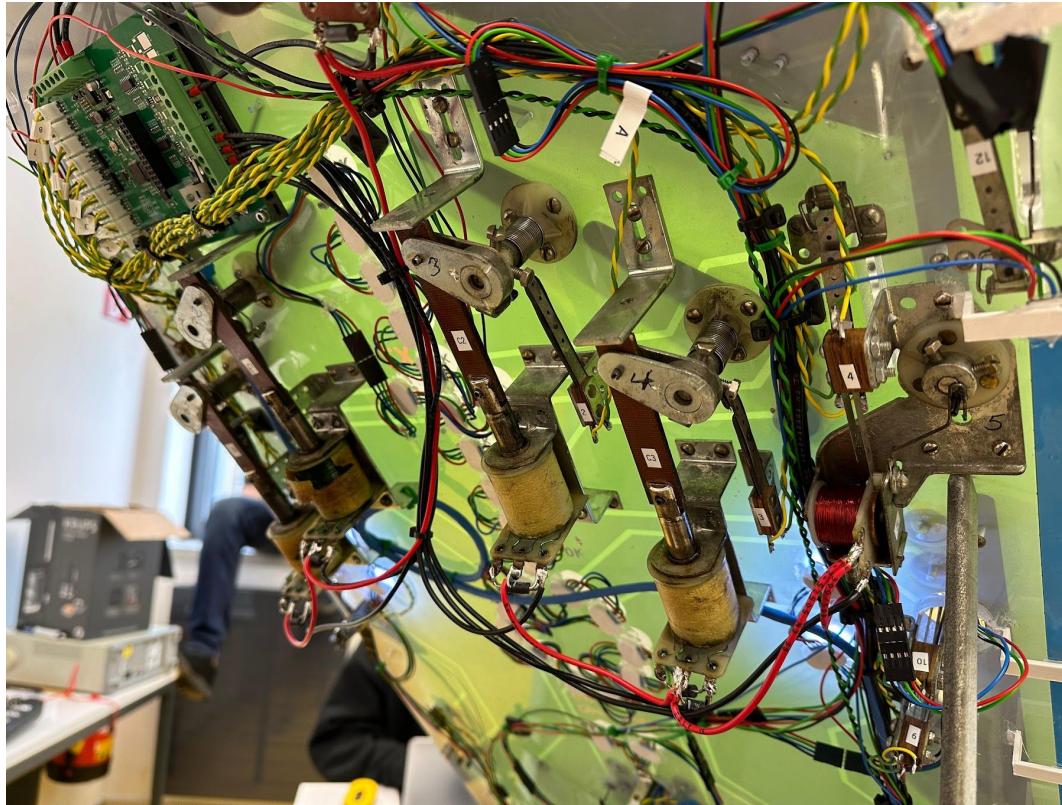
Sensors



Sensors



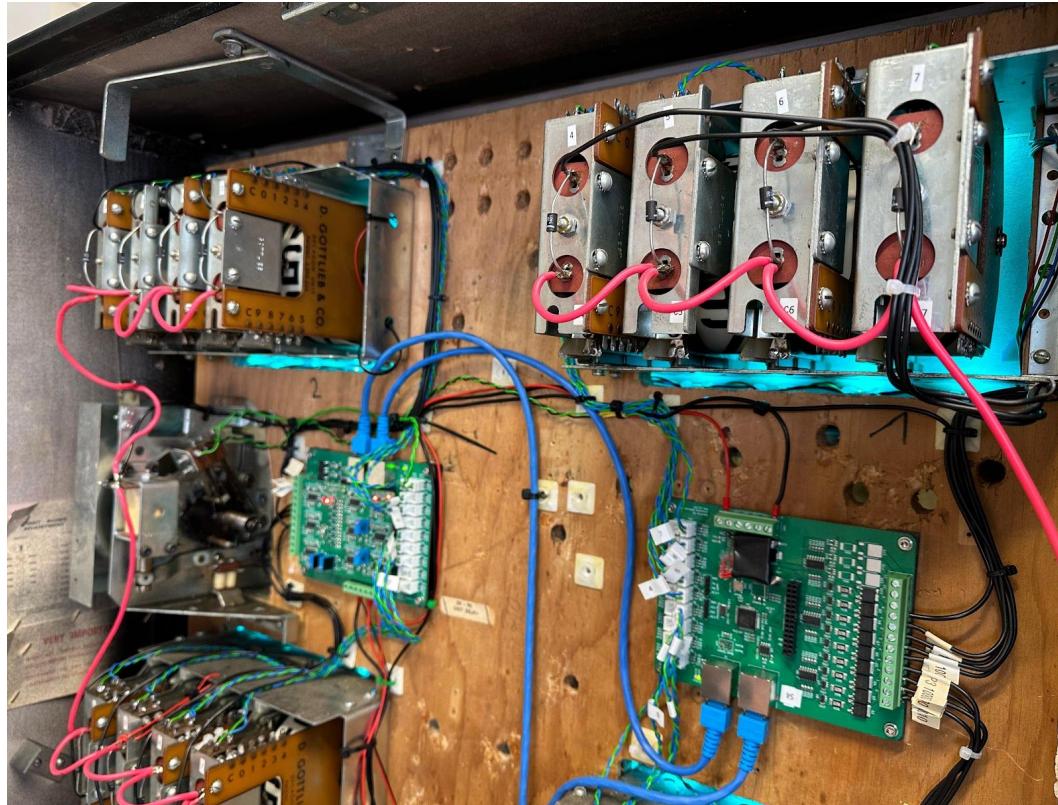
Actuators



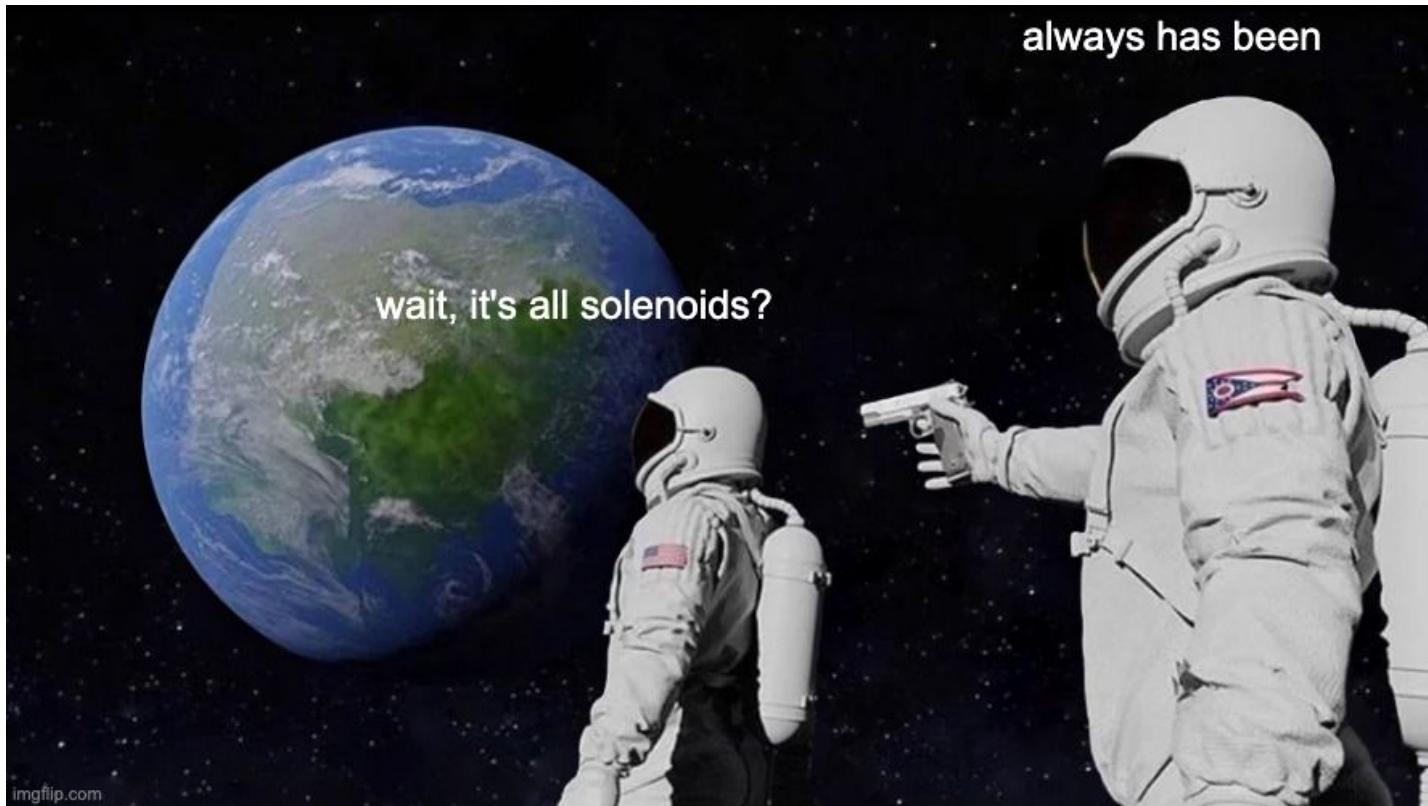
Actuators



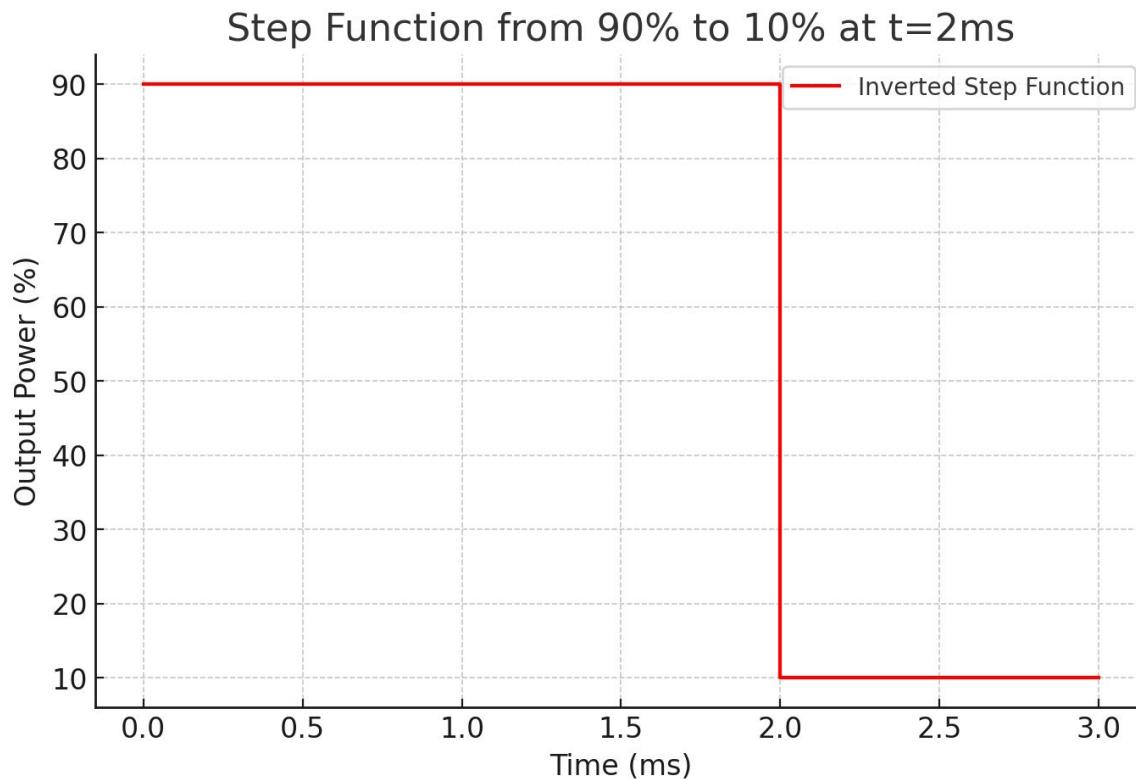
Counters



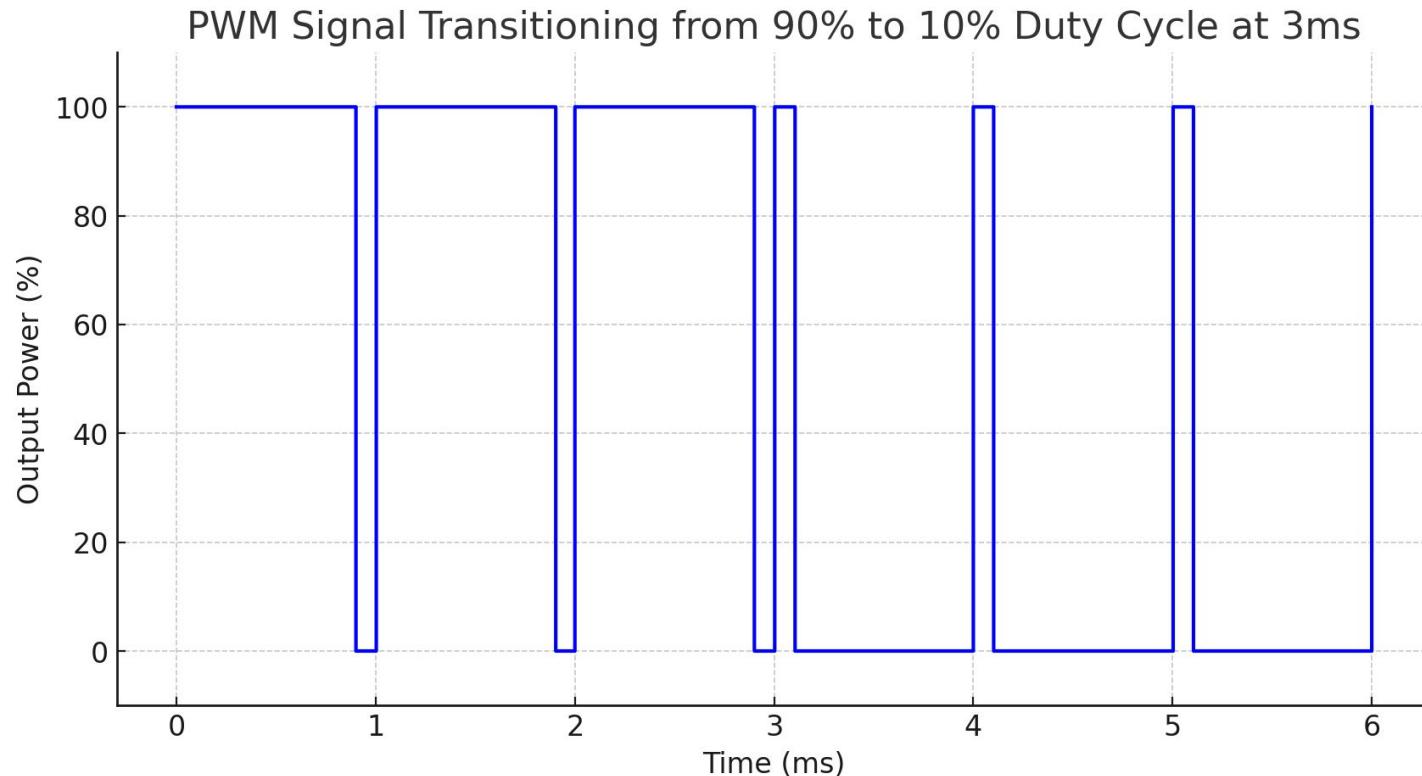
Actuators, takeaway



Actuators, technical



Actuators PWM

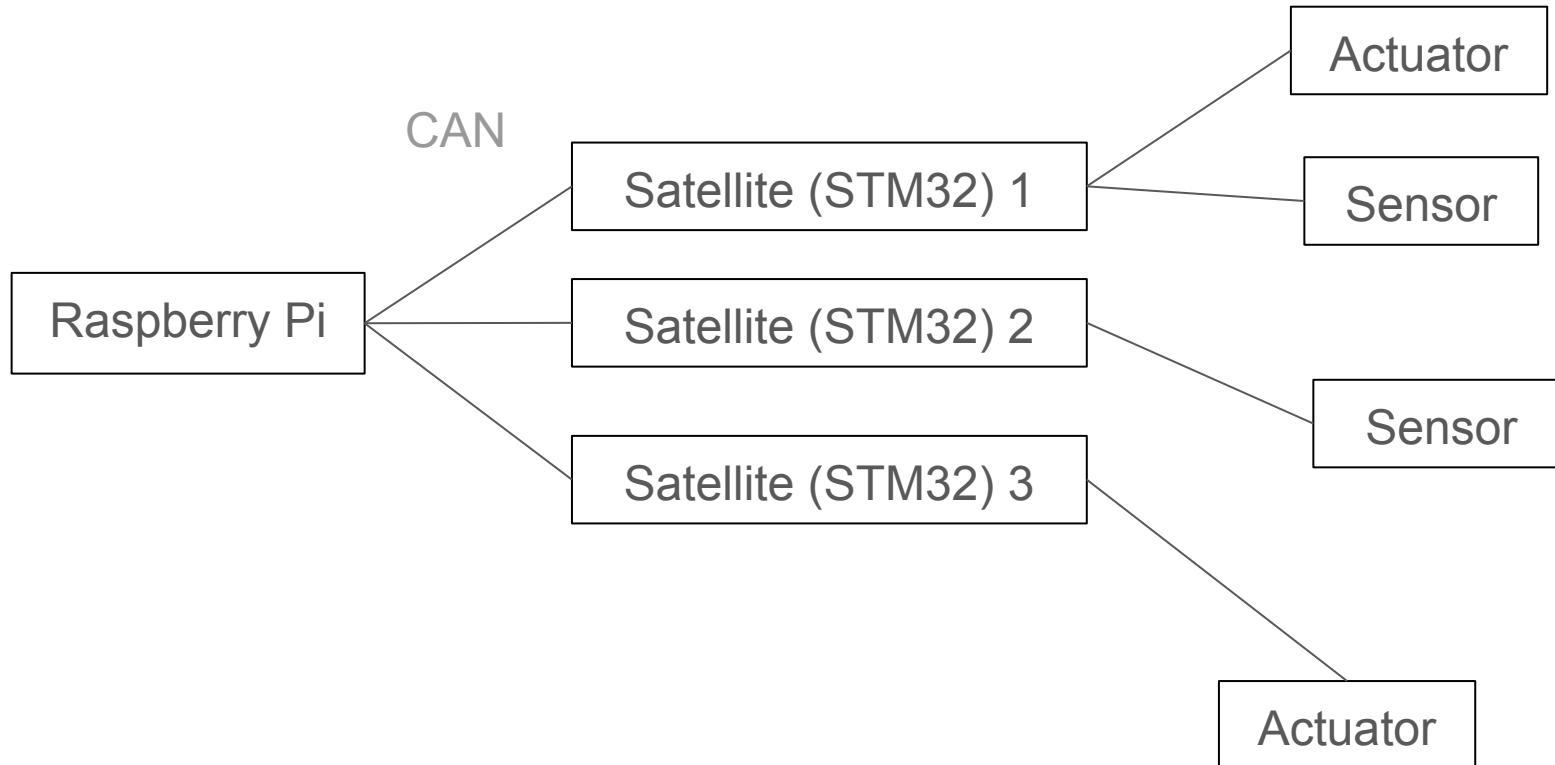


Implementation

```
analogWrite(PIN_OUTPUT, 900);  
delay(3);  
analogWrite(PIN_OUTPUT, 100);
```

→ 5 different codebases

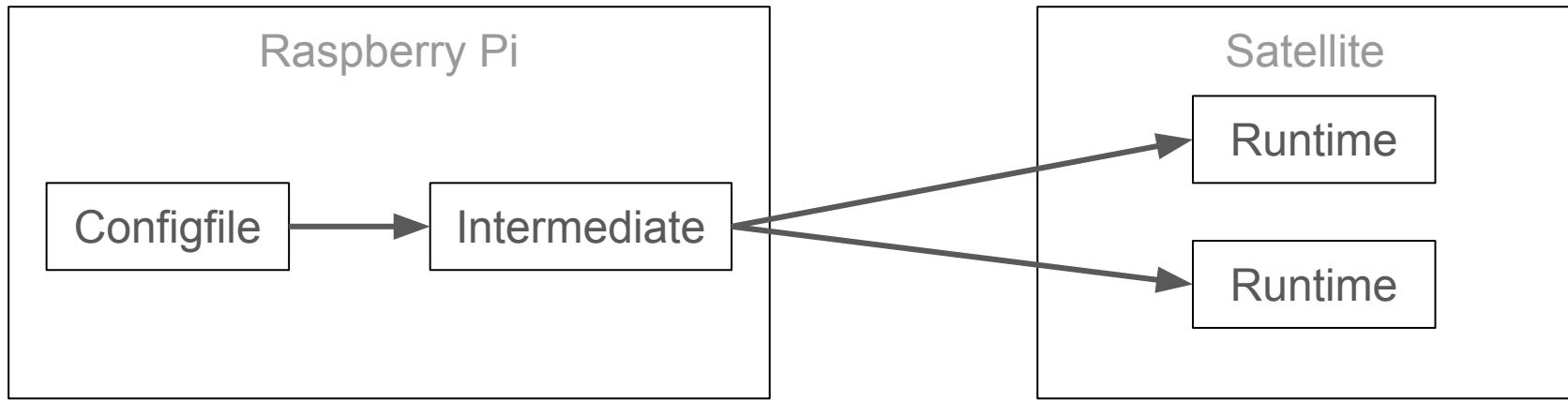
Infrastructure



Idea

1. Satellites programmed once
2. Single codebase
3. ~~Code~~ configuration
4. Config held centrally
5. Config updateable

Idea, low level control



CAN bus

- Differential pair
- Multi-master
- Collision detection/avoidance
- Short packets

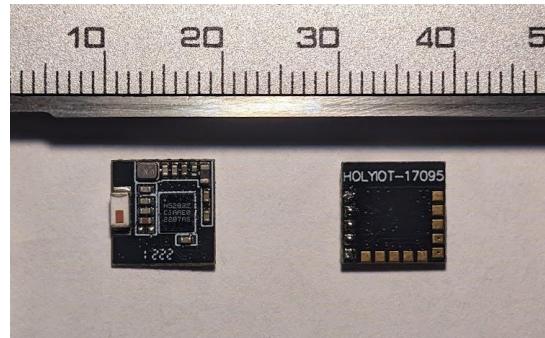
CANopen

- Object dictionary
 - Collection of “variables”
- Profiles
 - CIA 401: “Generic I/O device profile”
 - Our own: intermediary data
- Heartbeats
- ...

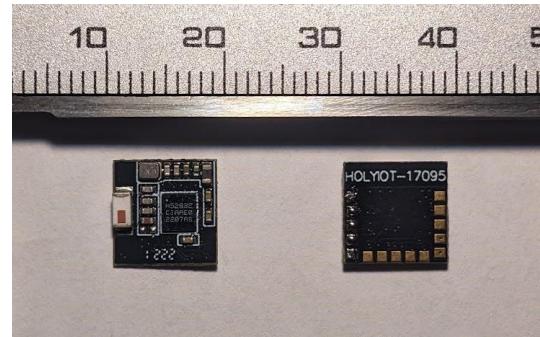
Item configuration

```
flippers:  
    id: flipper_left_outer  
    type: microcode  
    code: solenoid  
    duty_cycles: [[80, 30], [20]]  
    analog_channel: 0  
    position:  
        x: 0.29  
        y: 0.31
```

BLEnky



BLEnky



Pin 17

Disabled ▾

Pin 18

Output ▾

Digital ▾

Default low ▾

Not invert ▾

Pin 19

Input ▾

Pullup ▾

Invert ▾

BLEnky the second

Digital outputs

18

19

20

21

Digital inputs

5

6

7

BLEnky, again

Output sequence

Repetition count (0 = infinite)

Send sequence to device

delay (ms):

100

$\pm \mp \times$

delay (ms):

100

$\pm \mp \times$

delay (ms):

1000

$\pm \mp \times$

18 19 20 21

18 19 20 21

18 19 20 21

gpioASM

```
1      label start  
2  
3      write_digital 11111  
4      sleep_ms 500  
5      write_digital 00000  
6      sleep_ms 500  
7  
8      jump start
```

gpioASM

```
1      label start  
2  
3      write_digital 11111  
4      sleep_ms 500  
5      write_digital 00000  
6      sleep_ms 500  
7  
8      jump start
```

More commands:

- jump_match label 1-0
- sleep_match -00

gpioASM generation

```
1  |   flippers:
2  |     id: flipper_left_outer
3  |     type: microcode
4  |     code: solenoid
5  |     duty_cycles: [[80, 30], [20]]
6  |     analog_channel: 0
7  |     position:
8  |       x: 0.29
9  |       y: 0.31
10 |     can:
11 |       id: 4
```

gpioASM generation

```
1  flippers:  
2      id: flipper_left_outer  
3      type: microcode  
4      code: solenoid  
5      duty_cycles: [[80, 30], [20]]  
6      analog_channel: 0  
7      position:  
8          x: 0.29  
9          y: 0.31  
10     can:  
11         id: 4
```

```
1  label start  
2  
3  {% if check_input %}  
4  jump_match_all exit {{ '-' * gpio.index }}{{ (gpio.state + 1) % 2 }}  
5  {% endif %}  
6  
7  {% if wait_for_input %}  
8  sleep_match_all {{ '-' * gpio.index }}{{ gpio.state }}  
9  {% endif %}  
10  
11  {% for duty_cycle in duty_cycles %}  
12  
13  {% if analog_channel is defined %}  
14  write_analog_channel_{{ analog_channel }} {{ duty_cycle[0] }}  
15  {% endif %}
```

gpioASM generation

```
1  flippers:  
2      id: flipper_left_outer  
3      type: microcode  
4      code: solenoid  
5      duty_cycles: [[80, 30], [20]]  
6      analog_channel: 0  
7      position:  
8          x: 0.29  
9          y: 0.31  
10     can:  
11         id: 4
```

```
1  label start  
2  
3  {% if check_input %}  
4  jump_match_all exit {{ '-' * gpio.index }}{{ (gpio.state + 1) % 2 }}  
5  {% endif %}  
6  
7  {% if wait_for_input %}  
8  sleep_match_all {{ '-' * gpio.index }}{{ gpio.state }}  
9  {% endif %}  
10  
11  {% for duty_cycle in duty_cycles %}  
12  
13  {% if analog_channel_is_defined %}  
14  write_analog_channel_{{ analog_channel }} {{ duty_cycle[0] }}  
15  {% endif %}
```

gpioASM infrastructure

```
1  label start
2  write_analog_channel_0 80
3  sleep_ms 30
4  write_analog_channel_0 20
5  label exit
6  exit
```

gpioASM infrastructure

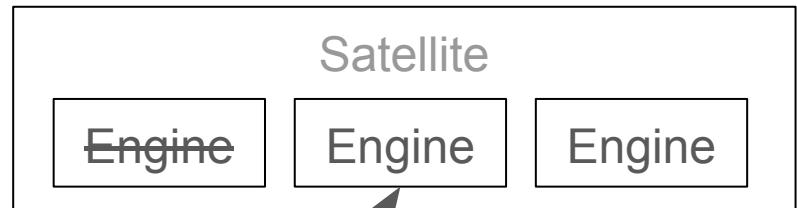
```
1  label start
2  write_analog_channel_0 80
3  sleep_ms 30
4  write_analog_channel_0 20
5  label exit
6  exit
```

A diagram illustrating the mapping of assembly code to memory. A black rectangular box on the left contains the assembly code. A thin black arrow originates from the bottom right corner of this box and points to a white rectangular box at the bottom. This bottom box contains the memory address `008000105000201e101400c0`. The entire assembly code box has a thin gray border.

008000105000201e101400c0

gpioASM infrastructure

```
1  label start  
2  write_analog_channel_0 80  
3  sleep_ms 30  
4  write_analog_channel_0 20  
5  label exit  
6  exit
```



008000105000201e101400c0

DBus interface

ItemRegistry

`org.freedesktop.DBus.Properties`

Get

Gets an items state

DBus interface

ItemRegistry

`org.freedesktop.DBus.Properties`

Get

Gets an items state

Set

Sets an items state

DBus interface

ItemRegistry

`org.freedesktop.DBus.Properties`

Get

Gets an items state

Set

Sets an items state

GetAllItems

Flat items list

Python gamecore

```
@engine.on_item_state_changed('button_left')
def handle(item_id, item_state):
    engine.item_proxy.flipper_left_outer = item_state
    engine.item_proxy.flipper_left_inner = item_state
```

Python gamecore

```
@engine.on_item_state_changed('button_left')
def handle(item_id, item_state):
    engine.item_proxy.flipper_left_outer = item_state
    engine.item_proxy.flipper_left_inner = item_state
```

```
engine.item_follow(
    'flipper_right_outer',
    'flipper_right_inner',
    item_source='button_right'
)
```

Website



Frontend



HTTP-DBus gateway



ItemRegistry

DBus ↔ HTTP gateway

A screenshot of a web browser window displaying a DBus introspection interface. The address bar shows the URL `localhost:5000/static/html/index.html`. The page content is a JSON-like structure representing DBus interface information.

```
{
    "path": "/",
    "interface": "org.freedesktop.DBus",
    "methods": [
        {
            "name": "Hello",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                }
            ],
            "return_type": "string"
        },
        {
            "name": "RequestName",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                },
                {
                    "name": "bus_name",
                    "type": "uint32"
                }
            ],
            "return_type": "boolean"
        },
        {
            "name": "ReleaseName",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                }
            ],
            "return_type": "boolean"
        },
        {
            "name": "StartServiceByName",
            "parameters": [
                {
                    "name": "service",
                    "type": "string"
                },
                {
                    "name": "bus_name",
                    "type": "uint32"
                }
            ],
            "return_type": "boolean"
        },
        {
            "name": "UpdateActivationEnvironment",
            "parameters": [
                {
                    "name": "env",
                    "type": "a{,string,string,}"
                }
            ],
            "return_type": "boolean"
        },
        {
            "name": "NameHasOwner",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                }
            ],
            "return_type": "boolean"
        }
    ],
    "signals": [
        {
            "name": "NameAcquired",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                }
            ]
        },
        {
            "name": "NameLost",
            "parameters": [
                {
                    "name": "name",
                    "type": "string"
                }
            ]
        },
        {
            "name": "ServiceActivating",
            "parameters": [
                {
                    "name": "service",
                    "type": "string"
                }
            ]
        },
        {
            "name": "ServiceDeactivated",
            "parameters": [
                {
                    "name": "service",
                    "type": "string"
                }
            ]
        }
    ],
    "interfaces": [
        {
            "name": "org.freedesktop.DBus.Properties",
            "methods": [
                {
                    "name": "Get",
                    "parameters": [
                        {
                            "name": "interface_name",
                            "type": "string"
                        },
                        {
                            "name": "property_name",
                            "type": "string"
                        }
                    ],
                    "return_type": "variant"
                },
                {
                    "name": "GetAll",
                    "parameters": [
                        {
                            "name": "interface_name",
                            "type": "string"
                        }
                    ],
                    "return_type": "dict<string,variant>"
                },
                {
                    "name": "Set",
                    "parameters": [
                        {
                            "name": "interface_name",
                            "type": "string"
                        },
                        {
                            "name": "property_name",
                            "type": "string"
                        },
                        {
                            "name": "value",
                            "type": "variant"
                        }
                    ],
                    "return_type": "void"
                }
            ],
            "signals": [
                {
                    "name": "PropertiesChanged",
                    "parameters": [
                        {
                            "name": "interface_name",
                            "type": "string"
                        },
                        {
                            "name": "changed_properties",
                            "type": "dict<string,variant>"
                        },
                        {
                            "name": "invalidated_properties",
                            "type": "list<string>"
                        }
                    ]
                }
            ]
        }
    ],
    "constants": [
        {
            "name": "NAME_INVALID",
            "value": -1
        },
        {
            "name": "NAME_HAS_OWNER_CHANGED",
            "value": 1
        },
        {
            "name": "NAME_OWNER_CHANGED",
            "value": 2
        },
        {
            "name": "NAME_OWNER_DIED",
            "value": 3
        },
        {
            "name": "NAME_LOST",
            "value": 4
        },
        {
            "name": "NAME_REQUESTED",
            "value": 5
        },
        {
            "name": "NAME_RELEASED",
            "value": 6
        },
        {
            "name": "NAME_SERVICE_ACTIVATED",
            "value": 7
        },
        {
            "name": "NAME_SERVICE_DEACTIVATED",
            "value": 8
        }
    ],
    "params": [
        {
            "name": "bus_name",
            "type": "string"
        },
        {
            "name": "name",
            "type": "string"
        },
        {
            "name": "env",
            "type": "a{,string,string,}"
        },
        {
            "name": "service",
            "type": "string"
        }
    ],
    "errors": [
        {
            "name": "org.freedesktop.DBus.Error.NameHasOwnerChanged",
            "message": "The name has been taken by another process"
        },
        {
            "name": "org.freedesktop.DBus.Error.NameInUse",
            "message": "The name is already in use"
        },
        {
            "name": "org.freedesktop.DBus.Error.NameInvalid",
            "message": "The name is invalid"
        },
        {
            "name": "org.freedesktop.DBus.Error.NameOwnerDied",
            "message": "The name owner has died"
        },
        {
            "name": "org.freedesktop.DBus.Error.NameReleaseFailed",
            "message": "The name release failed"
        },
        {
            "name": "org.freedesktop.DBus.Error.ServiceActivating",
            "message": "The service is activating"
        },
        {
            "name": "org.freedesktop.DBus.Error.ServiceDeactivated",
            "message": "The service is deactivated"
        }
    ],
    "properties": [
        {
            "name": "NameOwner",
            "type": "string"
        },
        {
            "name": "NameRequestRefusedCount",
            "type": "uint32"
        }
    ],
    "interfaces_by_name": {
        "/org/freedesktop/dbus": "org.freedesktop.DBus"
    }
}
```

Result

- Robust
- Flexible
- Latency of around 30ms
- Reusable
- Lots of engineering

Outlook

- Idea validated
 - Centralized control of decentralized infrastructure
 - Configuration / intermediate

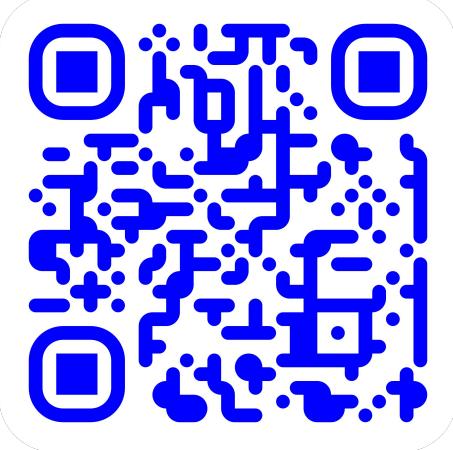
Outlook

- Idea validated
 - Centralized control of decentralized infrastructure
 - Configuration / intermediate
- BLEnky
 - Platform agnostic (beyond Nordic)
 - Transport agnostic (beyond BLE)
 - Protocol standardization (Packet based)

Sources

- <https://pinball-factory.com/products/king-rock-flipper>
- <https://www.deko-design-klein-shop.de/185317-Glockenspiel-Chime-Unit-fuer-WILLIAMS-Flipper-anschlussfertig>

Portfolio



- Projects
- Presentations (incl. this one)
- Guides
- Failures