Ceedling

- **support files** - additional C files needed for tests but not for project (not added to project src).

Hints

- Orgranize code into very small functional modules.
- Make separate `.h` for each `.c` module.
- Try to create separate `.h` for types and separate for function prototypes (it helps later with mocking)
- Always wrap HAL or other lower parts using some thin wrappers

add .c file

When `ceedling` fails to pickup automatically .c file it can be added to given test by

```
TEST_FILE("source_file_to_compile.c")
```

cross includes = deep linking

Also `ceedling` cannot automatically pick `.c` files when included .h file depends on another `.h` file. All dependent includes needs to be added manually in test `.c` file.

This is desired behavior because `ceedling` gives you control how to treat additional dependeny headers. Perhaps you should break dependency chain by including mocked header.

```
# This will include 20 another headers :)

#include "cpu_hal.h"

# To prevent this:
#include "mock_cpu_hal.h
```

common includes

```
:cmock:
  :includes:
    - <stdbool.h>
    - <stdint.h>
```

extern keyword

```
WARNING: No function prototypes found!
```

By default cmock will ignore extern function (is not mocking them). To enable mocking of extern functions:

```
:cmock:
  :treat_externs: :include
```

volatile function parameters

```
void myfn(custom_t *m, volatile uint32_t *var);

error: conflicting types for ...
```

[CMock chokes on volatile function parameters #135](#)

```
The C99 standard recommends against using volatile as function call
parameters and as return values,
stating that it is undefined behavior (because it is implementation-
specific) and therefore should
not be depended upon.
```

- **Workaround1:** is to use macro for volatile and redefine it to nothing when TEST.
- **Workaround2:** is to use typedef volatile uint32_t my_volatile_type;

enable valgrind

```
:tools:
  :pre_test_fixture_execute:
    :executable: valgrind
    :arguments:
      - --track-origins=yes
      - --leak-check=full
      - --show-leak-kinds=all
      - --errors-for-leak-kinds=all
      - --error-exitcode=10
      - ${1}
:plugins:
  :enabled:
    - command_hooks
```

From:
<https://niziak.spox.org/wiki/> - **niziak.spox.org**

Permanent link:
https://niziak.spox.org/wiki/programming:c:ceedling

Last update: **2021/04/22 16:55**