# Rainbow FW unit test plan

## Package or subsystems

Rainbow FW structure is below:



The final realization is separated into the below parts:

1. EEP subsystem, which store the EEP data;
2. Flash subsystem, which stores the data log;
3. T\_unit, unified interfaces of the data;
4. Canopen communication;
5. Local communication
6. Shell & real-time print
7. Behavior {Fluid control, Schedule, Measure..etc}

## Which needs to be tested and also the goal

The subsystem EEP and Flash are proposed to be common component which could be reused.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Subsystem | Unit-test | Subsystem integration | Code review | Static analysis | Comments |
| EEP | Branch Coverage | Y | Y | Y | The source code is based on the IIC interfaces;  Realize the stub function about HW interfaces and branch coverage is required. |
| Flash | Branch Coverage | Y | Y | Y | The source code is based on the SPI interfaces;  Realize the stub function about HW interfaces and branch coverage is required. |
| T\_UNIT | None | None | Y | Y | This is the core parts with “defensive programming” well integrated. This parts is tested in run-time always. |
| Canopen | None | None | None | None | Use the commercial code, the functionality is tested in system integration |
| Local Communication | None | None | Y | Y | Quite simple which uses the T\_UNIT interfaces directly, which could be tested in black box (system integration). |
| Shell | None | None | Y | Y | The basic realization is reused. The function just simple use the interfaces of T\_Unit. |
| Behavior | None | None | Y | Y | This is realized with too many different tasks and only trigger the simple behavior. Which could be tested when validation. |

## Test report