Blue Clover Devices

Test Plan Reference

PLT

0.4.8

2019-03-08

Revision History

Revision History	
Revision 0.4.8	2019-03-08
Renamed to Test Plan Reference	
Revision 0.4.7	2019-03-08
Split off from System Manual	

Table of Contents

Preface

Document describing the test suite definition for use with the Production Line Tool.

1. Test Suite Reference

1.1 Test Suites

Examples:

A minimal test suite, with a single test item, scanning a bar code.

```
title: "LY10-PLT demo: Scan"
suite:
  - ident: SCAN-T1
    title: Scan MAC address
    steps:
      - command: scan MAC_ADDRESS
```

1.1.1 Test Suite Structure

PLT test suites are encoded as YAML text files, starting with the test suite title.

```
title: "LY10-PLT demo: Scan"
```

The title is followed by the suite: section, containing the test items in the test plan.

1.1.1.1 Test Items

Test items are identified with an ident: line, can contain a descriptive title, and should contain one or more test item steps.

During test execution, all test item steps need to complete successfully for the test item to succeed.

1.1.1.2 Test Item Steps

Test item steps consist of command: or uartcmd: blocks. A Test Item can contain multiple Test Item steps.

1.1.2 Basic Example

A basic In-Circuit Test suite for the PLT demo board.

```
title: "v0.1.8 (Green)"
suite:
  - ident: ICT-T1
    title: Identify DUT
    steps:
      - command: identify nRF52
  - ident: ICT-T2
    title: Erase nRF52 with FT2232H Mini Module
    steps:
      - command: erase nRF52
  - ident: ICT-T3
    title: Program LY10-DEMO-BOARD FW nRF52 with FT2232H Mini Module
    steps:
      - command: program nRF52 s132_nrf52_6.0.0_softdevice.hex,ly10-demo-fw-
0.1.8.hex,none
  - ident: ICT-T4
    title: BLE discovery
    steps:
      - command: bledis %BLEMAC% 30  # Wait up to 30 seconds for BLE discovery to
complete
```

1.2 Test Commands

1.2.1 bledis - Test BLE Discovery

Establishes a BLE connection to the DUT and discovers GATT services.

Usage:

```
bledis %BLEMAC%|<name> [timeout] [minRSSI]
```

Argument	Description
name	GAP name advertised by DUT. %BLEMAC% to specify the DUT's BLE MAC address instead.
timeout	Timeout, in seconds.
minRSSI	RSSI treshold (optional)

Example: BLE discovery of identified BLE HW MAC address.

```
- ident: ICT-T1
  title: Identify DUT
  steps:
    - command: identify nRF52
- ident: ICT-T2
  title: BLE Discovery
  steps:
    - command: bledis %BLEMAC% 30 -60
```

1.2.2 erase - Erase DUT Flash

Erase DUT MCU on-board flash.

Usage:

```
erase <target>
```

Argument	Description
target	Target to erase.

Example: Erase STM32L4 on-board flash.

```
- ident: ICT-T1
  title: Erase
  steps:
  - command: erase STM32L4
```

1.2.3 extflash_write - Write Peripheral Flash

Write DUT periperal flash.

Usage:

```
extflash_write <port> <filename>
```

Argument	Description
port	UART port
filename	Firmware Element Filename

1.2.4 identify - Identify DUT

Identify DUT MCU and/or RF peripherals.

Usage:

```
identify <target>
```

Argument	Description
target	Target to identify

1.2.5 operator - Operator Test

Instruct operator to perform a manual test step.

Usage:

```
operator <message>
```

Example: Instruct operator inspect housing.

```
- ident: ICT-T1
  - ident: FAT-T1
    title: Visual Inspection (manual)
    steps:
    - command: operator "Inspect Housing"
```

1.2.6 program - Program DUT

Erase and Program DUT MCU on-board flash.

Usage:

```
program <target> <bootFw>,<appFw>,<parts>
```

Argument	Description
target	Target to program.
bootFw	Bootloader Firmware Element
appFw	Application Firmware Element
parts	Partition Table Firmware Element

1.2.7 scan - Scan Barcode

Scan a barcode using USB-attached barcode scanner.

Usage:

```
scan <format>
```

Argument	Description
format	Format of code to scan

1.2.8 sleepms - Delay

Temporarily suspend test suite execution.

Usage:

```
sleepms <duration>
```

Argument	Description
duration	Duration, in milliseconds

1.2.9 uart - Send and Extract UART response

Extract data from UART.

Usage:

```
uartcmd: uart <port>
  [[expect: <expectText>]
   [extract: <extractText>
    extractKey: <extractKey>]]
  [send: <sendText>]
```

Argument	Description
port	UART port
expectText	Text to expect, prior to extraction
extractText	Regular expression to extract
extractKey	Key in which to store extracted text
sendText	Text to send prior to extaction

Example: extract ICCID from cellular modem.

```
- uartcmd: uart MCU
  expect: "+CCID:"
  extract: "CCID: (\\d{20})\r\n"
  extractKey: ICCID
  send: "AT+ICCID\r\n"
```

1.2.10 uartAwait - Await UART response

Wait for a specific UART response.

Usage:

```
uartAwait <port> <seconds>
```

Argument	Description
port	UART port
seconds	Time to await response, in seconds

Example:

```
- command: uartExpect MCU Pressed
- command: operator "Press button"
- command: uartAwait MCU 1
```

1.2.11 uartExpect - Set expectation for uartAwait

Set a UART response to wait for with uartAwait.

Usage:

```
uartExpect <port> <expect>
```

Argument	Description
port	UART port
expect	String to expect with subsequent uartAwait command

Example:

```
- command: uartExpect MCU Pressed
- command: operator "Press button"
- command: uartAwait MCU 1
```

1.2.12 uartReadTimeout - Test if UART is not transmissing

Test if nothing is received from UART.

Usage:

```
uartReadTimeout <port> <seconds> [<sendText>]
```

Argument	Description
port	UART port
seconds	Number of seconds to wait for incoming data
sendText	Text to send before waiting

Example: Test if modem is shut down.

```
- command: uartReadTimeout MCU 1 "AT"
```

1.3 Targets

target	Description
DA14580	Dialog DA14580 RFSoC
ESP32	Espressif ESP32 RFSoC (JTAG)
ESP32_HomeKit	Espressif ESP32 RFSoC (JTAG; HomeKit)
nRF52	Nordic nRF52 RFSoC (SWD)
nRF52_DevKit	Nordic nRF52 RFSoC (USB, JLink)
STM32F4	ST STM32F4xx MCU (SWD)
STM32F4_DevKit	ST STM32F4xx MCU (USB, JLink)
STM32L4	ST STM32L4xx MCU (SWD)
STM32L4_DevKit	ST STM32L4xx MCU (USB, JLink)