Open STDF

Read RFT

No

Ready for test:

Force read

Start

Vac off

Vac on

Read ID:



Just test at current temp:

4 of 15

Currently testing unit:

15

Units in tray (optional):



-40, 0, 25, 50, 80, 115

Temperatures to loop:

**Lot processing**

Start Test

Close STDF

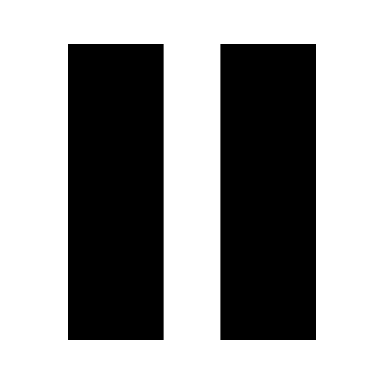
Last result:

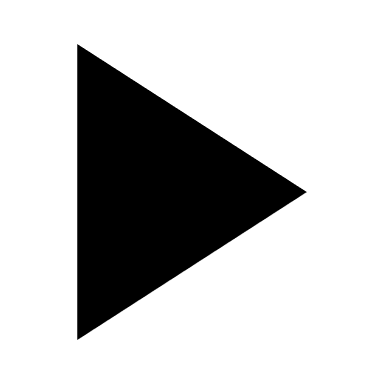
Will either be Green or red, and hbin/sbin

STDF filename:

C:/misc/path\_to\_stuff/example\_1.stdf

**Tester**





Press button to read ID. ID string will be displayed here. Read only box

**Handler**

**Temp unit**

Current Temp:

Set point:

85.1C

85.0C

**Basics**:

This should control the handler, temp unit, and tester. I should define the basic state machine(s) and the ala-carte buttons/functions.

**Controls**:

**Temp unit**: (These take forever (like 2s each), so show “working” or some other indication something is happening.)

* read current – little R/O text box for 4 digits of temperature.
* set setpoint -- little R/W text box for 4 digits of temperature.
* stop/go. – with indication which is active.
* Force read button.

**Handler**:

* Read ID (alive check) – show the ID string
* Write bin1
* Write bin2
* Read RFT…. This is more of an event that will need triggered. Not needed in the alacarte version, but will be needed for the state machine.
* Vac on
* Vac off
* Ready for test indication

**Tester**:

* Set datalog filename/location. Open file. Use an OOB file picker. This returns the name/string and I pass to the command.
* Actually open/start the file, with the above information.
* Close current datalog file
* Start test.
* Last result (big green/red HBIN/SBIN)

**Lot running**:

* Lot info—pop up a form with lot #, user, starting serial?
* List of temps. Just a comma list.
  + Checkbox to “just test”. Use the current temp setting and just test everything once and bin it.
* Units in tray text entry box. Optional, but allows a kind of progress bar.
* Optional unit count entry (you can enter in the # of units that are in the tray)
* Unit number/out of unit count for progress
* State machine runs with selected feedback.
* State machine outline below
* Maybe show yield?

**Research/to do:**

* Event handling for the RTS. See pyvisa page.
* Automate datalogging commands. For some reason it wasn’t working last time.
* Tester commands for start test, read result.
* Tester set variables? Need a way to communicate not blowing fuses and such. Setting lot number, starting serial number, etc.
* Event for testing complete? Or just poll?
* Support for programming serial numbers. This is a limited time thing, for unprobed die, but that will be common for prototyping.

**Lot processing state machine:**

Get lot info (optional)

(here I assume the program is loaded, and it is otherwise ready. In theory the OI could load it too, but this isn’t really for operators)

Window saying load the program.

Window saying hit the handler start button (or use the relay control).

Window saying that if you are testing, enable that in the handler control (2/1 bins). If characterizing, set to 3/3.

Wait for event: RFT from handler.

If (just test at current temp): (requires handler 1/2 mode)

Start test on tester

Poll every 2s or so to see if the test is done. (or see if an event is possible)

Ask tester what the result was.

Send appropriate bin1/bin2.

Wait for event: RFT from handler

Else there is a temp list:

For each temp in temp\_list:

Set temp = this temp

Set a tester variable with the temp value and print to datalog?

Poll temp unit for ready.

Start test on tester.

Poll every 2s or so to see if the test is done. (or see if an event is possible)

Ask tester what the result was. (maybe?)

Set temp =25C to “safe” it.

Send appropriate bin1/bin2

Wait for event: RFT from handler

**Upon an X/close:**

Set temp to 25C

Send a close stdf command? I think its harmless

Vac off?