Stand-alone Qapps Installation Procedure

(To be used outside Qualcomm network)

Shared by: Stephen Fu (Advantest)

07-April-2016

Agenda

- Setup an environment to run Qapps
  - Perl Modules
  - System Environment Variables
- Install TCU driver package
- Qappmodel package
  - Test program generation
  - Execute input library
  - Hand test
  - Handler test
- Run test program with Qapps

Setup an environment to run Qapps

ADVANTEST

Perl Modules:

Qapps include Perl scripts to launch GUI and handle post data processing, which refer to the following Perl modules:

```
#!/pkg/qct/bin/perl
use Tk 800.00;
use POSIX;
use DBI;
require Tk::FileSelect;
require Tk::DialogBox;
require Tk::Dialog;
require Tk::LabEntry;
require Tk::ROText;
require Tk::Balloon;
require Tk::NoteBook;
use File::Basename;
use Data::Dumper;
use FindBin qw($RealBin);
```

Check if all these perl modules are installed on tester system, if not, Suggest to use CPAN module to do auto installation one by one.

Example:

```
cpan> install Data
Running install for module Data
Running make for G/GR/GROMMIER/Text-Editor-Easy-0.01.tar.gz
Fetching with LWP:
  ftp://ftp.perl.org/pub/CPAN/authors/id/G/GR/GROMMIER/Text-Editor-Easy-0.01.tar.gz
Fetching with LWP:
  ftp://ftp.perl.org/pub/CPAN/authors/id/G/GR/GROMMIER/CHECKSUMS
Checksum for /root/.cpan/sources/authors/id/G/GR/GROMMIER/Text-Editor-Easy-0.01.tar.gz ok
Text-Editor-Easy-0.01/
Text-Editor-Easy-0.01/lib/
```

Setup an environment to run Qapps

System Environment Variables:

Environment Variables are used in application model, they are used for datalogging purpose:

```
-----------------------------
-- GET GENERAL INFORMATION --
-----------------------------
* PROGRAM                = GET_TEST_INFO (testprog);
* PROGRAM_REV            = GET_TEST_INFO (test_rev);
* PROGRAM_DESC           = GET_TEST_INFO (test_desc);
* START_TIME             = GET_TEST_INFO (date_time);
* DEVICE_REV             = GET_TEST_INFO (dev_rev);
* USERNAME               = GET_ENV(USER);
* HOSTNAME               = GET_ENV(HOST);
* HOME                   = GET_ENV(HOME);
* OS                     = GET_ENV(HOSTTYPE);
* DEVICE_DIR             = GET_ENV(PWD);

-- STDF KEYWORDS --
* JOB_NAM                = CONST_INPUT({PROGRAM});
* JOB_REV                = CONST_INPUT({PROGRAM_REV});
* SETUP_T                = CONST_INPUT({START_TIME});
* NODE_NAM               = CONST_INPUT({HOSTNAME} {OS});

-- THERMAL STREAM AND LOOPS VARIABLES --
* THERMAL_LOOPCOUNT      = CONST_INPUT(1);
```

Check if these system environment variables: USER, HOST,HOME,HOSTTYPE,PWD are created on workstation, if not, set up these variables correctly.

Example:

```
[sfu@sgpws06 dummy]$ env |grep PWD
OLDPWD=/home1/sfu/c_wfu/dummy/testprog
PWD=/home1/sfu/c_wfu/dummy
[sfu@sgpws06 dummy]$ env |grep HOME
UNO_JAVA_JFW_JREHOME=file:///opt/java1.7_x86_64/jre
JAVA_HOME=/opt/java1.7_x86_64
HOME=/home1/sfu
```

Create HOSTTYPE:

```
[sfu@sgpws06 dummy]$ env |grep HOSTTYPE
[sfu@sgpws06 dummy]$ export HOSTTYPE=linux
[sfu@sgpws06 dummy]$ env |grep HOSTTYPE
HOSTTYPE=linux
[sfu@sgpws06 dummy]$
```

Install TCU driver package

ADVANTEST

TCU is used when doing temperature testing, the driver need to be installed on tester system for the communication between TCU and tester.

To install TCU driver, just untar the attached TC_SCPI_1_7_1.tar.gz and copy all the files/folders under TC_SCPI_1_7_1 to /usr/share/Sensata/tc_scpi/.

```
[root@sgpws06 TC_SCPI_1_7_1]# pwd
/home1/sfu/TC_SCPI_1_7_1
[root@sgpws06 TC_SCPI_1_7_1]# ls -rtl
total 20
drwxrwxr-x 2 sfu sfu 4096 Apr  6 11:30 lib
drwxrwxr-x 2 sfu sfu 4096 Apr  6 11:30 inc
drwxrwxr-x 2 sfu sfu 4096 Apr  6 11:30 doc
drwxrwxr-x 2 sfu sfu 4096 Apr  6 11:30 bin
-rwxrw-r-- 1 sfu sfu  293 Apr  6 13:14 install.sh
[root@sgpws06 TC_SCPI_1_7_1]#
```

Or you can use install.sh to install driver, check the install.sh before running the shell script, make sure the install_dir =“/usr/share/Sensata/tc_scpi/”

```
install_dir="/usr/share/Sensata/tc_scpi/"
mkdir -p $install_dir
```

Qappmodel package

Download the attached qappmodel package: qappmodel_4p7.tar.gz and put it under device folder ../device/applicat, Untar the package here.

```
[sfu@sgpws06 applicat]$ pwd
/home1/sfu/c_wfu/dummy/applicat
[sfu@sgpws06 applicat]$ ls -rtl
total 300
-rwxrwxr-x 1 sfu sfu     99 Mar 23 08:23 qappmodel_handtest_stdf_4p7.app
-rwxrwxr-x 1 sfu sfu     94 Mar 23 08:23 qappmodel_handtest_4p7.app
-rwxrwxr-x 1 sfu sfu     98 Mar 23 08:23 qappmodel_handler_stdf_4p7.app
-rwxrwxr-x 1 sfu sfu     93 Mar 23 08:23 qappmodel_handler_4p7.app
drwxrwxr-x 2 sfu sfu   4096 Apr  5 09:21 bkfolder
-rw-r--r-- 1 sfu sfu 272425 Apr  5 09:23 qappmodel_4p7.tar.gz
drwxrwxr-x 6 sfu sfu   4096 Apr  6 13:17 qappmodel_4p7
-rwxrwxr-x 1 sfu sfu   1718 Apr  6 17:00 qapp_install_local.pl
[sfu@sgpws06 applicat]$ cd qappmodel_4p7
[sfu@sgpws06 qappmodel_4p7]$ ls
applicat  EventFormatter  exec_input  perl
[sfu@sgpws06 qappmodel_4p7]$
```

The package include handtest and handler test application models, execute input lib, perl scripts, EventFormatter lib.

The *qapp_install_local.pl* is used to created test program here.

Qappmodel package

Test program generation:

qapp_install_local.pl can be used to generate handtest and handler test program.

```
[sfu@sgpws06 applicat]$ pwd
/home1/sfu/c_wfu/dummy/applicat
[sfu@sgpws06 applicat]$ ls
bkfolder                qappmodel_4p7          qappmodel_handler_4p7.app        qappmodel_handtest_4p7.app
qapp_install_local.pl   qappmodel_4p7.tar.gz   qappmodel_handler_stdf_4p7.app   qappmodel_handtest_stdf_4p7.app
[sfu@sgpws06 applicat]$ ./qapp_install_local.pl -i testflow.ttf
Testflow: testflow.ttf
Creating
        ../testprog/testflow_handtest.tp
        ../testprog/testflow_handler.tp
        ../testprog/testflow_handtest_rhel5.tp
        ../testprog/testflow_handler_rhel5.tp
        ../testprog/testflow_handtest_rhel5-64.tp
        ../testprog/testflow_handler_rhel5-64.tp
Done
[sfu@sgpws06 applicat]$
```

Totally 6 test program are generated including 3 handtest and 3 handler test, they are used on the different tester system:

testflow_handtest.tp/testflow_handler.tp ----------RedHat 3
testflow_handtest_rhel5.tp/testflow_handler_rhel5.tp----------RedHat 5/32bits
testflow_handtest_rhel5-64.tp/testflow_handler_rhel5-64.tp-------RedHat 5/64bits

Qappmodel package

ADVANTEST

Test program generation:

Handtest test program, leave the Hp_lib empty as no handler is used, Pdi_lib can support the communication with TCU.

```
[sfu@sgpws06 testprog]$ more testflow_handtest_rhel5-64.tp
hp93000,testprog,0.1
Testflow:     testflow.ttf
Userproc:
Waferdescr:
Applicdata: qappmodel_handtest_4p7.app
Pdi_lib: ../applicat/qappmodel_4p7/exec_input/linux5-64
Hp_lib:
Dataform:
```

Handler test program, must specify the Hp_lib as handler is used in this case. The Hp_lib is depended on selected handler. User need to specify the Hp_lib before starting handler testing.

```
[sfu@sgpws06 testprog]$ more testflow_handler_rhel5-64.tp
hp93000,testprog,0.1
Testflow:     testflow.ttf
Userproc:
Waferdescr:
Applicdata: qappmodel_handler_4p7.app
Pdi_lib: ../applicat/qappmodel_4p7/exec_input/linux5-64
Hp_lib:
Dataform:
```

manually add `Hp_lib: /opt/hp93000/soc/PH_libs/GenericHandler`

Qappmodel package

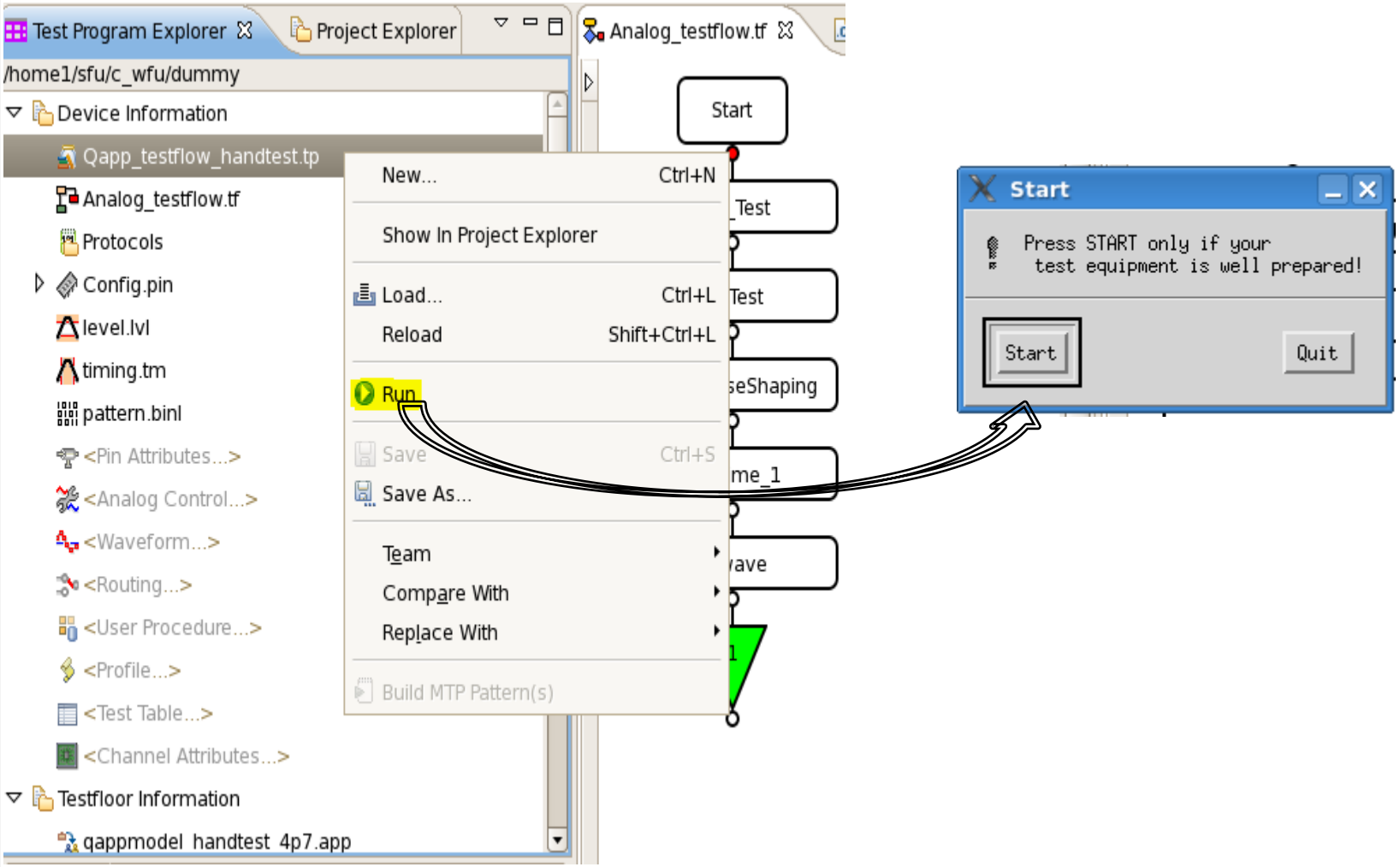
Execute input library:

qappmodexec is executable binary file and built output from C source code. The binary file have different version (rehat 3, rehat 5/32bits and rehat 5/64bits), depending on the building system environment.

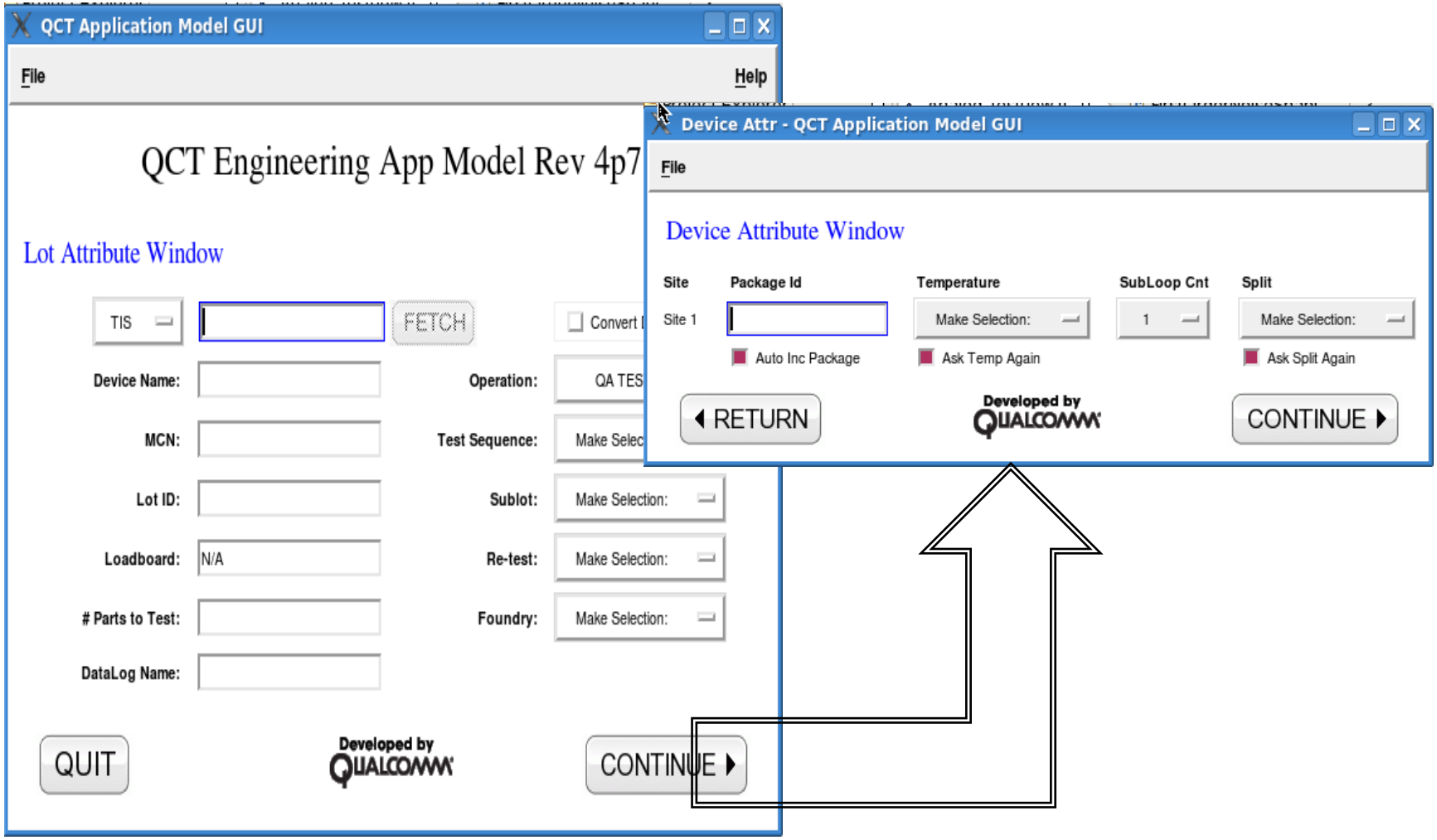
```
[sfu@sgpws06 qappmodel_4p7]$ pwd
/home1/sfu/c_wfu/dummy/applicat/qappmodel_4p7
[sfu@sgpws06 qappmodel_4p7]$ ls
applicat  EventFormatter  exec_input  perl
[sfu@sgpws06 qappmodel_4p7]$ cd exec_input
[sfu@sgpws06 exec_input]$ ls -rtl
total 12
drwxrwxr-x 2 sfu sfu 4096 Apr  6 13:17 linux5-64
drwxrwxr-x 2 sfu sfu 4096 Apr  6 13:17 linux5
drwxrwxr-x 2 sfu sfu 4096 Apr  6 13:17 linux
[sfu@sgpws06 exec_input]$ cd linux5-64
[sfu@sgpws06 linux5-64]$ ls -rtl
total 220
lrwxrwxrwx 1 sfu sfu     48 Apr  6 13:17 Ci -> /opt/hp93000/soc/hp83000/prod_env/bin/ci_program
-rwxrwxr-x 1 sfu sfu 220882 Apr  6 13:22 qappmodexec
[sfu@sgpws06 linux5-64]$ file qappmodexec
qappmodexec: ELF 64-bit LSB executable, AMD x86-64, version 1 (SYSV), for GNU/Linux 2.6.9, dynamically linked (uses shared libs),
[sfu@sgpws06 linux5-64]$ cd ..
[sfu@sgpws06 exec_input]$ cd linux5
[sfu@sgpws06 linux5]$ ls
Ci
[sfu@sgpws06 linux5]$ cd ../linux
[sfu@sgpws06 linux]$ ls
Ci
[sfu@sgpws06 linux]$
```

* Currently only Redhat 5/ 64 bits executable inplut library is available, so you can only run testflow_handtest_rhel5-64.tp/testflow_handler_rhel5-64.tp on Redhat 5/64 bits system.

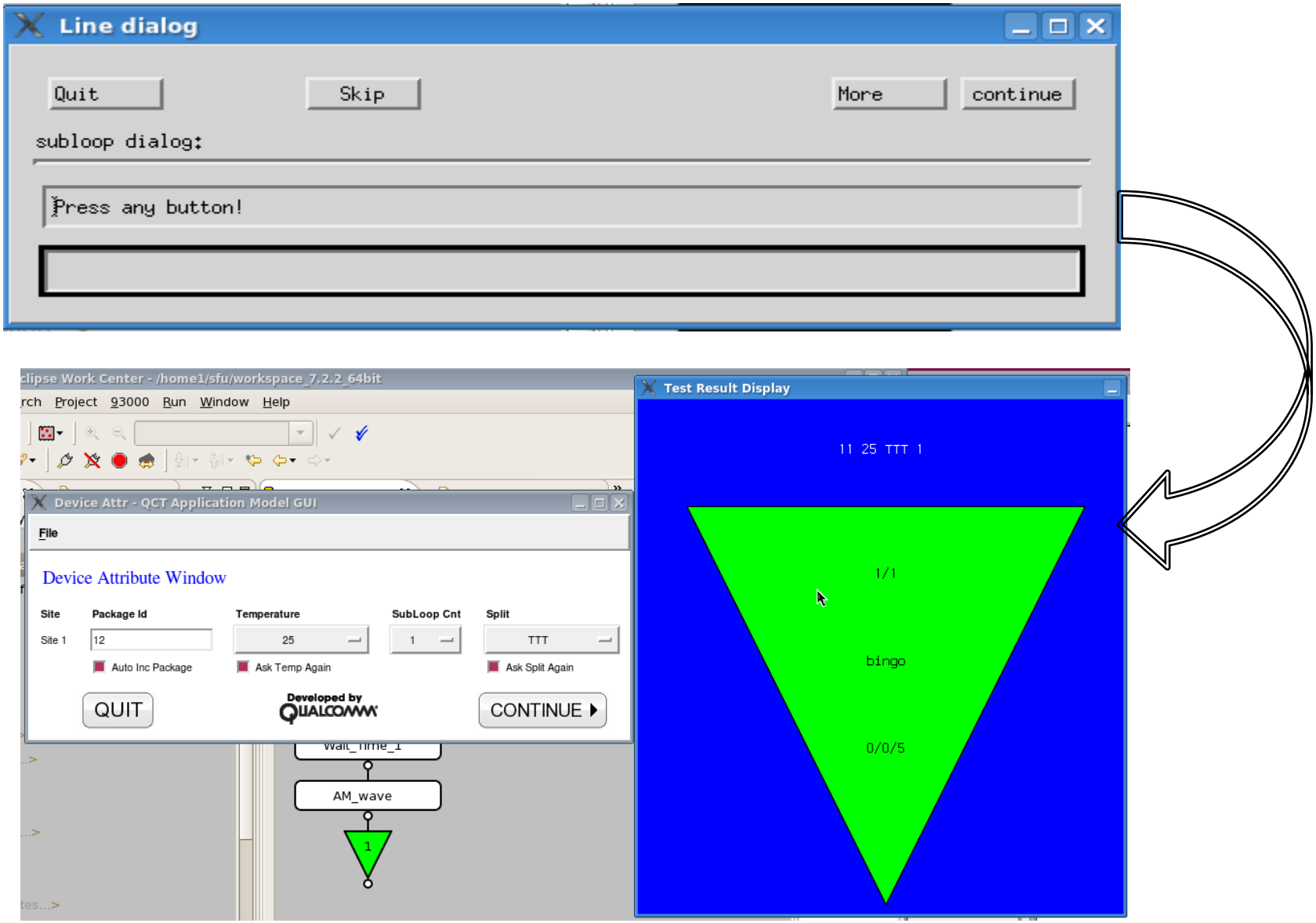
Run test program with Qapps



Run test program with Qapps



Run test program with Qapps



THANK YOU!