ATLAS ITk Upgrade
Strip Detector Quality Control

Documentation and user manual

LabVIEW code for Sensor QC

Christoph Klein

10th March 2020

Contents

1 Getting started

Setting up the hardware and starting the LabVIEW scripts

2 Common VIs

3 IV measurements

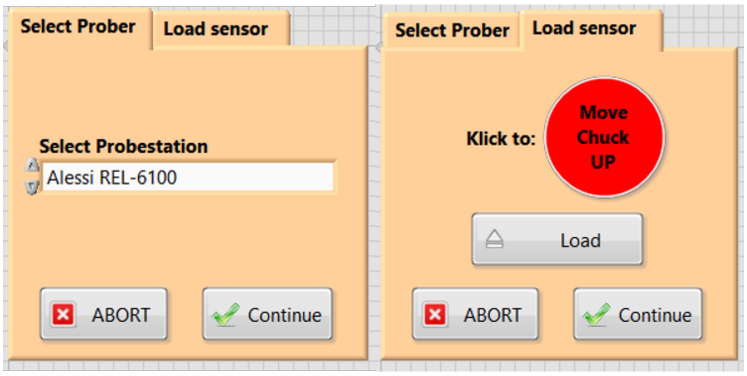
4 CV & C_{is} measurements

5 R_{is} measurements

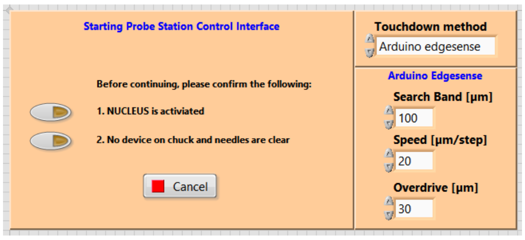
6 Striptests

Prober selection

Upon running the striptest VI, the probe station selection automatically opens, which also includes the selection of some parameters e.g. for the edgesense. Currently there are 2 probe stations implemented, the Alessi REL-6100 and the SemiProbe. The initialisation VIs are shown below in Fig. 1. After the probe station is selected and initialised, the open VI is meant to facilitate loading of sensors by being able to move the chuck to a pre-defined load position and back to the centre, as well as allowing the chuck to be moved between the *Separate* and *Contact* position.



(a)



(b) (c)

Figure 1: (a): VI for the selection of probe station and loading of DUT. (b,c): Initialisation VIs for the Alessi and SemiProbe probe station, respectively.

Loading a probe plan and aligning the sensor

The next step includes the loading of the probeplan, either the alignment of chuck in theta or the rotation of the probeplan coordinates to fit the chuck rotation, and the option to test the loaded coordinates and alignments by jumping to the AC measurement pads.

The user has to select a probe plan used as a map of probe points in the strip test by clicking on the "New" button. Care has to be taken to set the correct number of probe card channels or select the "Single Needle" Mode first, since the coordinates included in the probe plans are *enabled* for movement according to the number of needles used. The coordinates loaded can be checked in the window which opens upon selecting a probe plan.

After confirming the probe plan selection the jig has to be aligned and a reference on the sensor surface has to be chosen. For this purpose move the cross-hair to corresponding top and bottom fiducials and read the coordinates with the respective buttons. In the process of doing this the chuck moves automatically to the last coordinates used for the next field to be readout. After alignment the coordinates of the reference point are readout similarly.

Alternatively, if the jig is already aligned and/or the reference point is the same, the respective buttons can be selected to skip these steps. This is especially useful if only a new reference point has to be selected without the sensor being exchanged or if the measurement is restarted with the same sensor.

Completing the alignment procedure leads back to the probe plan initialisation VI. The probe plan can be exercised by jumping between strips corresponding to the loaded probe plan. Simply select the respective region and strip number from the list and press the "Jump" button.

important: Neither is the edgesense activated yet nor is the movement limited to the strips enabled according to the number of probe needles! Therefore keep the platen raised and/or the chuck lowered while exercising the probe plan. Use this function to test the layout of the jig in order to prevent collisions from happening during strip test.

Special features using a probe card

The main reason to use a probe card is the opportunity to measure multiple strips without having to move the chuck to a new position for each new strip. Hence the strip test procedure was edited in order to minimise the time needed for the full strip test. Since hot-switching at 100 V can yield difficulties due to the RC behaviour of the cables and the probe card itself and resulting in current spikes originating from discharge processes while switching the card channels, these effects have to be accounted for with a reasonable settling time.

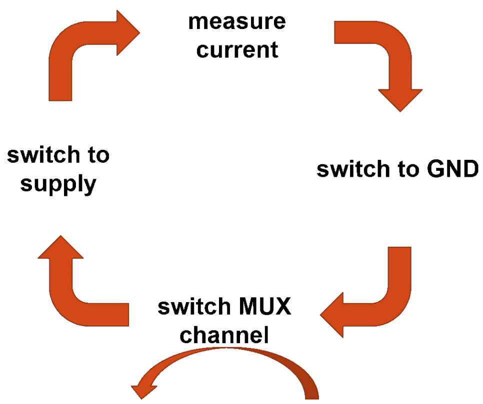


Figure 2: Schematic of pinhole measurements used for the *Rolling Current Readout*.

Due to the two 8x2 multiplexers leading to 4 simultaneously usable probe card channels, a specific readout procedure has been devised in order to minimise the time necessary for the pinhole measurement. According to the scheme displayed in Fig. 2 there are 4 steps in the current measurement used to determine pinholes. Having one MUX output present in each step at any given time during the measurement allows for the necessary delay to be reduced, since the switching processes and the SMU readout itself introduce a "natural" delay, during which the current of channels, which are switched to high voltage/GND, are allowed to settle.

The probe card also provides two touchdown needles in addition to the 32 signal channels used for measurement. These needles can be used in a custom-made touchdown feedback using make or break contact configuration, i.e. one needle touches down on the sensor surface in alignment with the measurement needles, whereas the second one loses the contact to the first needle upon touching the surface. The edgesense circuit uses the break contact configuration in conjunction with a *Schmitt trigger* on a digital input pin of the Arduino to provide the necessary touchdown information. The circuit is shown in Fig. 3 The reference

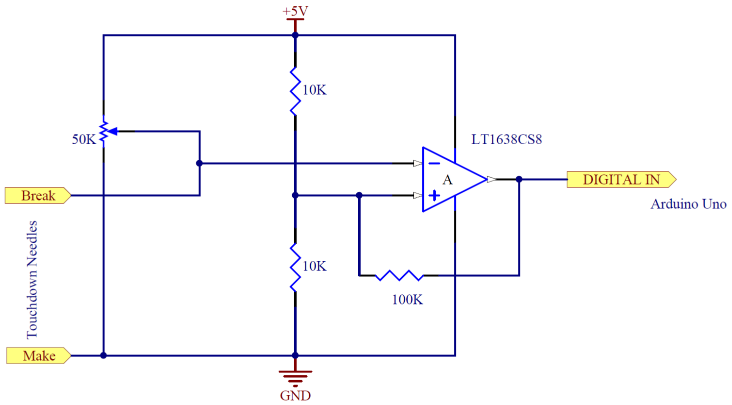


Figure 3: Circuit diagram of the Schmitt trigger used in conjunction with the Arduino for touchdown feedback

voltage is set to 2.5 V with the Arduino providing the asymmetric supply +5V/GND for the amplifier. The make needle is pulled to GND potential resulting in the amplifier sourcing +5 V without surface contact and 0 V upon touchdown due to the break needle being at floating potential.

For the edgesense to work, the initial z coordinate – i.e. the chuck height at the previous position – has to be known, as well as the search band, defining the limits around the initial z within which the new z value is searched, the step size, and the overdrive. The general concept of the edgesense is as follows:

1. when moving to a new position, the chuck is lowered to the *Separate* state
2. upon arriving at the target x and y coordinates, the chuck is risen to the lower limit of the search band, i.e. the initial z minus half the search band
3. the output of the Arduino is checked for contact
4. if no contact is established, the chuck is moved up by the step size
5. the last two steps are repeated until either a contact has been established or the upper limit of the search band is reached
6. the chuck is moved up by the overdrive value to ensure a good contact for all needles
7. in the case of no contact until the previous step, the Arduino is checked one last time and if the result is still negative, an error message is shown; if there is a contact, the z coordinate minus the overdrive is saved for moving to the next position

important 1: In the single needle mode the Arduino edgesense cannot be used, however, it can still be used as a temperature and light sensor.

important 2: Please be considerate about the number of probe card needles. The VIs are made with the intention of using a probe card with 32 needles. A probe card with a number of needles other than a multiple of 4 might not work in combination with the rolling current readout. If the number of needles does not work with the sensor specs, the VI will give a warning.

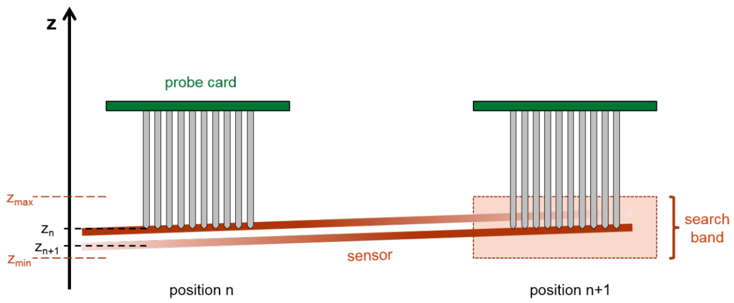


Figure 4: Schematic of the touchdown procedure.

Striptest setup

Similar to all other measurement VIs, firstly, a prompt opens which requires the user to enter the DUT information. The user is selected in the striptest VI itself, where the DUT information are displayed. For the striptest setup itself as shown in Fig. 5, options are to select the regions to be tested and their respective range.

> **important:** When selecting the measurement range, please keep in mind that when using a probe card only multiples of the number of probe card channels are enabled for movement. As such even when selecting a smaller strip range, more/fewer strips can end up being measured due to the restricted movement when using a probe card.

If not all parts of the striptest procedure can be done at the same time, e.g. due to the lack of a switching matrix, the included tests can be selected with the checkboxes in the lower right corner.

The default values for the actual measurement parameters – sensor bias, LCR meter frequency, required fraction of good strips for passing, and the pinhole measurement parameters – are those from the QC document. Like for the other measurements, they are read from a text file when running the VI and can be changed either for a single run during the setup (changes need to be confirmed with the "Configure Measurement" button) or for all future runs by changing the text file.

The hardware used can be selected from the dropdown selection and configured in the *HardwareConfiguration* VI which opens when clicking the "Configure Hardware" button.

The striptest will commence with the "Continue" button and everything will be aborted and the running VIs stopped with the "Quit" button, similar to the other measurements.

Striptest run

Continuing from the striptest setup, the measurement procedure starts.

> **important:** Make sure the hatch of the probe station is closed.

The first step of the strip test is to bias the DUT in steps of 10 V to the value given in the strip test setup. After the subsequent initialisation and trimming of the LCR meter, a reference measurement for the pinhole test is conducted without touching the DUT. If the compliance of the bias supply (not the SMU) is exceeded, the strip test is aborted automatically.

Upon completion of these steps the actual strip test is executed for every strip given in the test setup. The measured current, capacitance, and resistance, as well as the type of strip defect is continually being displayed – automatically for the active region, for the other regions if selected. Additionally, statistical information consisting of the total number of

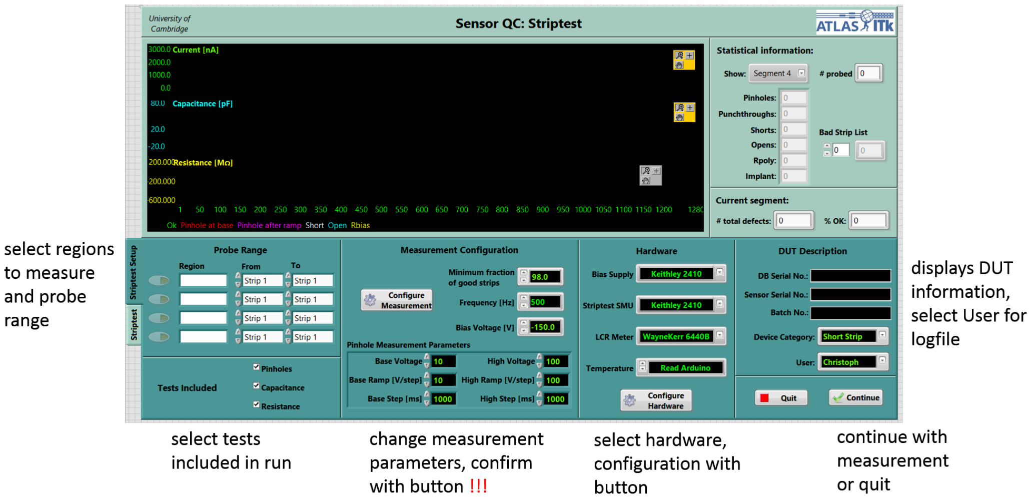


Figure 5: Striptest VI front panel during setup.

probed strips, the strip defect list of the selected segment. The ratio of good strips and the total number of defects in the segment currently being measured is shown at all times.

important 1: While the capacitance, and resistance thresholds are taken from the technical specifications and enforced depending on the type of sensor, the current threshold can be adjusted if necessary (currently hardcoded in the LV block diagram). The current threshold should be adjusted with regards to the leakage current observable in the pinhole test, but generally pinholes and punchthroughs can be easily identified by the exceedingly high current compared to the baseline.

important 2: The rolling current readout can be enabled/disabled via the respective button. If a probe card is used for the measurement, it is enabled by default. It cannot be enabled in single needle mode.

important 3: If all occurring defects should be re-tested automatically or the bias ramped down after the test (and potentially the automatic re-test), the corresponding buttons can be enabled to toggle this.

After the strip test is finished, the user can choose to retest defect strips (or let them be retested automatically) by pressing the "ReCheck" button. Similarly to the actual strip test, current, capacitance, and resistance of the currently (re-)tested region are shown and subsequently updated for the repeated measurement. The measurement data and the defect list can be either updated automatically or the changes have to be confirmed for each retested strip.

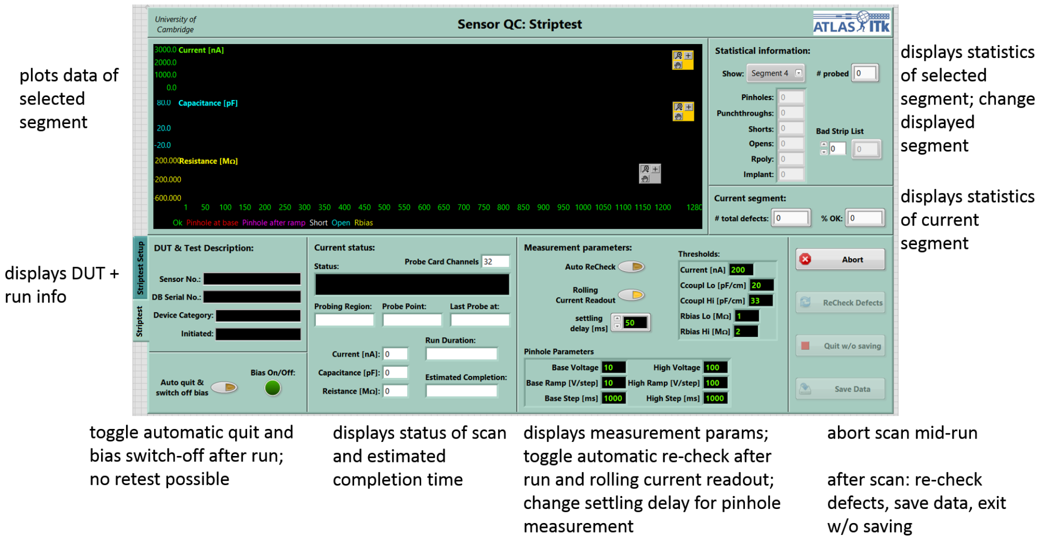


Figure 6: Striptest VI front panel during scan.

7 Including new instruments

(a) Open LabVIEW project "InstrControl.lvproj". Don?t have any other VIs open or editing of the project may be blocked. The project lists all instrument classes present.

(b) Right-click "My Computer", select "New →Class". Options upon right-clicking may look slightly different depending on the LV version and installed add-ons. Give your class a name. Choose one which allows easy identification of the instrument it is being used for.

(c) Right-click new class, select "Properties →Inheritance" and change inheritance accordingly.

(d) Right-click new class, select "New →VI for Override". VIs with * must be overwritten; all others can be, if necessary. Usually, only VIs which require procedures specific for each instrument or need to be unique – and not just differ in GPIB commands used – are denoted as "override required".

(e) "InstrumentSetup" global: add default properties cluster. Add instrument (name, address, type) in GPIB address cluster. If new instrument is not a GPIB instrument, use NI-VISA address, but still give name + type like other GPIB instrument.

(f) Add instrument with relevant parameters in "HardwareConfiguration.vi". Not all parameters which can be configured are also relevant. Select only those necessary to be configured by the user, all others can be hardcoded in the instrument initialisation VI.

(g) Add instrument in "InstrumentSelection.vi" with correct type + name. The Instrument selection VI is important to create an object with the correct address from the instrument classes and store that newly created object in the corresponding measurement configuration global. E.g. during IV scans the selected supply from the specific instrument daughter classes will be saved as the more generic "Supply" class object from its parent class in the configuration cluster to be referenced during the measurement script.

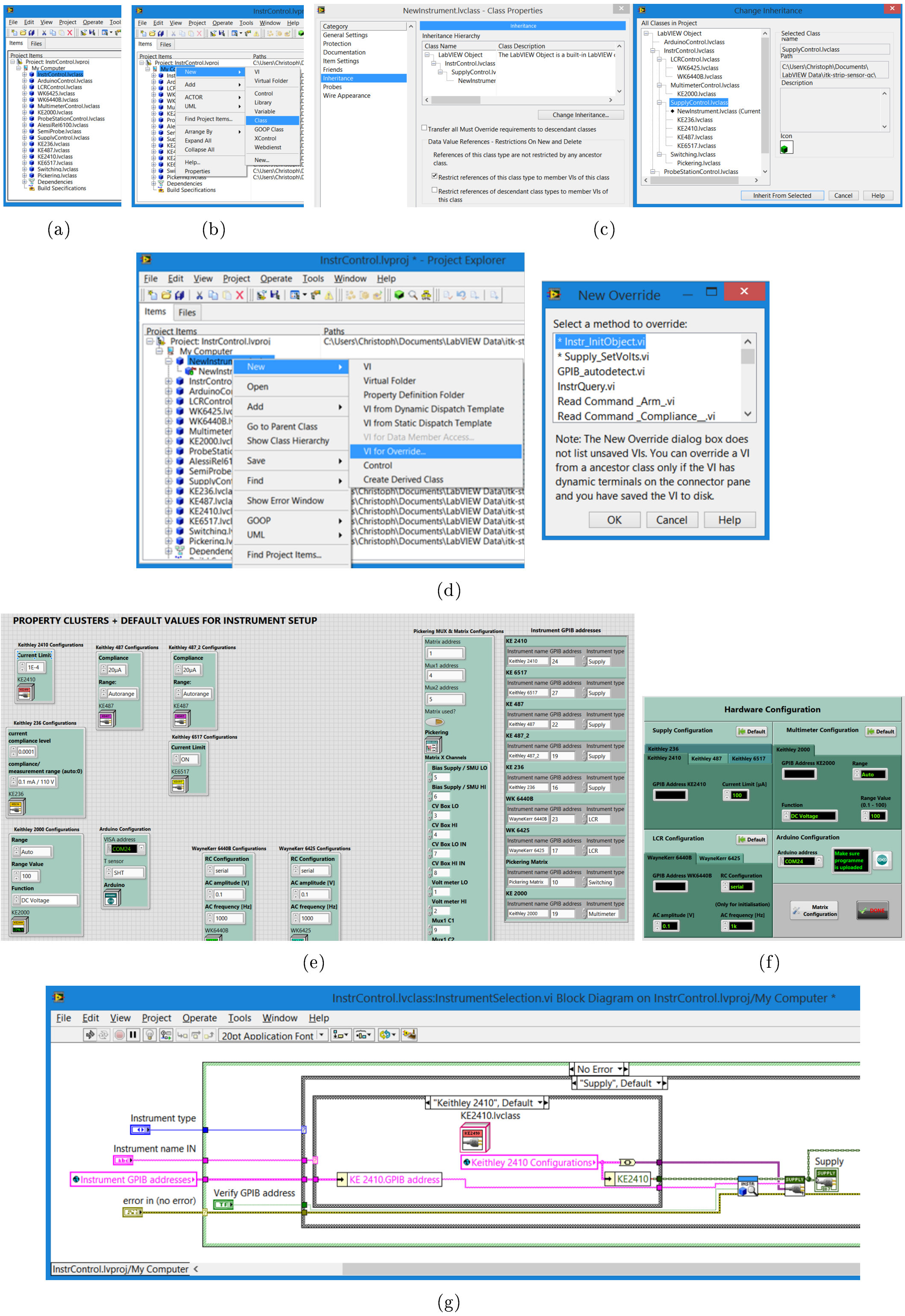


Figure 7: Steps to include a new instrument in the instrument control class structure.