A 'Universal' Life-Test System for Electromechanical Relays.

Brian.J.Frost CEng MIET, Stuart.J.Hobday BEng MIET

Applied Relay Testing Ltd

Dorset, England

Abstract **– The field of Electrical Contact Switching brings together physics, material technology and accumulated experience and remains a very analogue science. Real-world monitoring of contact switching is vital in determining the best materials and construction for a relay device and to ensure quality in today's demanding market place. Although it is simple in principle to cycle a relay with a specific contact load and observe total failure, designers today require much more insight into trends allowing them to better predict failures in contact use, in application areas that have not been fully tested. Applied Relay Testing has been performing research and development into a 'universal life test system' which would provide information about contact voltage and current, contact timing and device operate / release voltage actually DURING the life-test, showing each of these parameters in a graphical form and allowing their trends to be observed. This paper will describe the measurement techniques involved and the results obtained, illustrating how this test system brings a unique insight into the life-time of the modern electromechanical relay.**

Keywords: Life-Test, Electromechanical Relays, Contact Timing, Operate/Release Voltage, Contact Resistance.

I. INTRODUCTION.

It started with an idea for a flexible relay life-test system [2]. Our company had first produced life-test equipment for telecom relays back in the mid 90's with the RT96, a test system that quickly established itself within the low to mid-sized relay power band and with very specific telecoms resistive, cable and inductive loads [1]. More often now though, in addition to conventional low power and signal relay life testing there are demands for high-power relay and switch testing, but at lower cost.

We decided that we needed a way of making a life-test system that met the following design goals:

- Simple to understand and configure.
- Minimal hardware.
- Operates over a wide cycle rate range.
- Handles high or low power devices.
- Measures contact voltage drop (CVD), contact resistance (CR) and Stick voltage (VLOAD).
- Timing and Operate / Release Voltage to be optional.
- Flexible test limits.

The flexibility to apply the system to both small (fast) and large (slow) relays would normally come at a price yet we realised that users of the system should only need to pay for capability that they require. Quickly we identified that a very modular design made up of building blocks would enable delivery of this capability and still be tied closely to cost.

II. BASIC CONCEPTS AND PROGRAMMING

Most relay life-test environments can be closely related to the diagram of a 12V (e.g. automotive) life-test on a single contact as shown in Fig. 1.

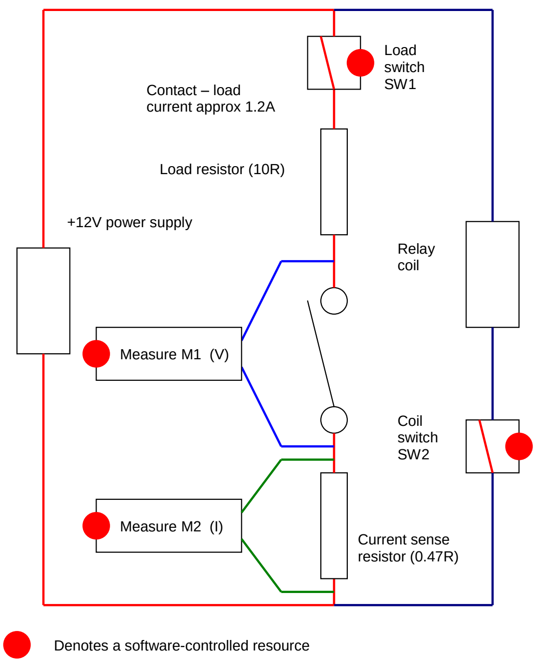

Figure 1. Simple high-power life-test circuit, e.g. automotive.

We quickly realised that we could put together this circuit using two main electrical building block resources:

- Voltage measurement inputs (M1 and M2)
- Semiconductor power switches (SW1 and SW2).

With hardware modules that implement this capability and flexible software we go further to make almost any measurement on any relay, even parametric measurements such as operate and release voltage or timing measurements. To prove this concept we built a simple prototype system using modular electronics coupled to the two types of I/O shown in Fig. 2 below.

Figure 2. Prototype Life-Test Hardware

The hardware is coupled to a standard PC which runs the Life-Test software. The I/O modules control relay coils and load switching and monitor contact voltage drops. Each contact input is capable of measuring AC or DC in the microvolt range (for the closed state) and up to tens of volts to monitor the open contact state.

Although this I/O is more 'relay-specific' than off-the-shelf PC I/O cards, so far we have provided little more than has been constructed as 'home-built' equipment by relay manufacturers. Our goal was to go further to provide a flexible programming environment that would harness the power of the I/O modules without requiring detailed programming or hardware knowledge, thus ideally a technician would be able to program the system.

The key to providing this simplicity was to realise that such a technician would only need to wire contact inputs and switches into a circuit such as Fig. 1 and enter simple cycle-based commands to tell the system how to work with these resources. The rest of the control and data processing could be a pre-written within the 'relay cycling environment' therefore no formal knowledge is required for use.

As an example of this simple setup for (say) a 10 Hz cycle rate measuring CR, the User should only need to tell the system the following actions:

At time	Action to perform
0 ms	Turn coil on
20 ms	Start monitoring CVD
30 ms	Stop monitoring CVD
-	CR := CVD/1.2
50 ms	Turn coil off
70 ms	Check CVD > 90% of VLOAD

For a life-test to be performed, this series of steps are simply repeated as many times as cycles are required.

It turns out that programming these kind of 'steps' is a familiar operation to those used to working with programmable logic controllers and visual programming languages, and we used this idea to create a simple cycle-based programming environment as shown in Fig. 3 below:

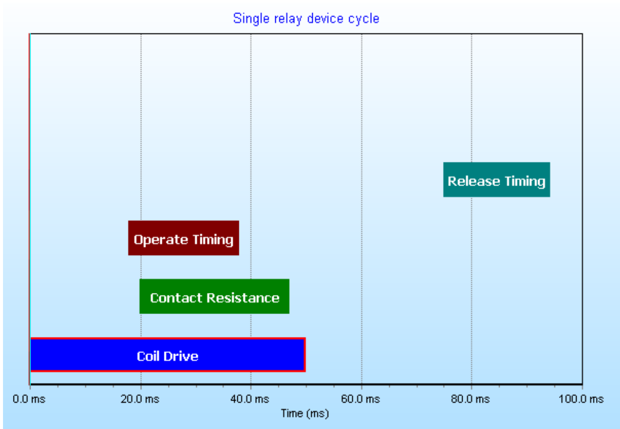

Figure 3. Programming Cycle Time slots

This display represents a single relay cycle – shown above as taking 100ms (a cycle rate of 10 Hz). Within this cycle the User is able to place one or more 'time slots' (shown as the coloured blocks) for example 'Coil Drive' or 'Contact Resistance'. Each time slot is simply a definition for a start and stop time together with the ability to program an event that will be executed at those time points. By entering suitable hardware control commands into these events, the tester hardware can be made to perform its testing in exactly the way that the test circuit requires.

For example the relay coil is energised by programming the start event of the 'Coil Drive' time slot as shown in Fig. 4:

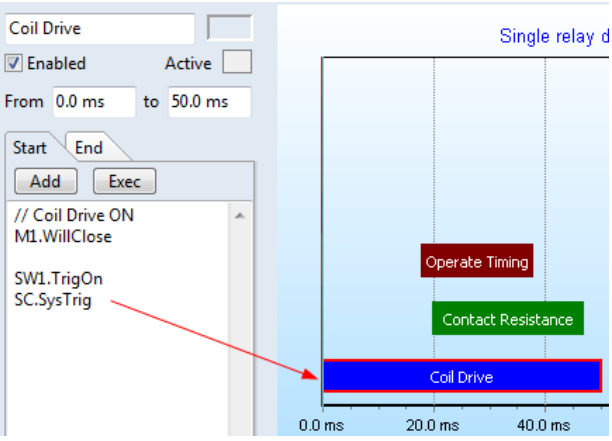

Figure 4. Actions at the start of the Coil Drive Time Slot

The three programming lines shown are executed at cycle time 'zero', i.e. the start of the cycle, and perform the following actions:

Action	**Description**
M1.WillClose	Tells the contact measure input that it will be monitoring a 'closed' contact.
SW1.TrigOn	Prepare the SW1 (coil drive) switch to close on a trigger pulse.
SC.SysTrig	Send/issue the trigger pulse.

After these actions, the coil will be energised and the relay contacts start to transfer. At the end of the coil drive time slot (50ms), similar actions are executed as shown in Fig. 5. These will remove the coil drive for the remainder of the cycle.

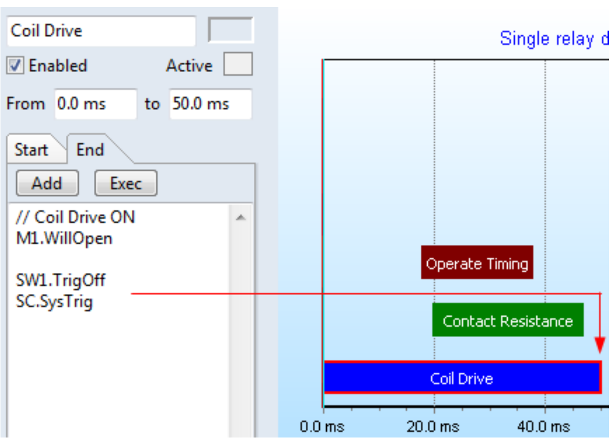

Figure .5 Actions at the end of the Coil Drive Time Slot

With this simple programming concept we now have a flexible way of controlling the relay coil and its cycling.

III. CONTACT VOLTAGE DROP AND CONTACT RESISTANCE MEASUREMENT.

So far, the actions shown have all been 'outputs'. To monitor the relay contact we use additional time slots to define time points for the monitoring parameters that the hardware already knows how to measure. For example the 'Contact Resistance' time slot defines activity at 20ms (start) and 47ms (stop). The two events are programmed to start and stop the voltage measurement integration as shown in Fig. 6 and Fig. 7:

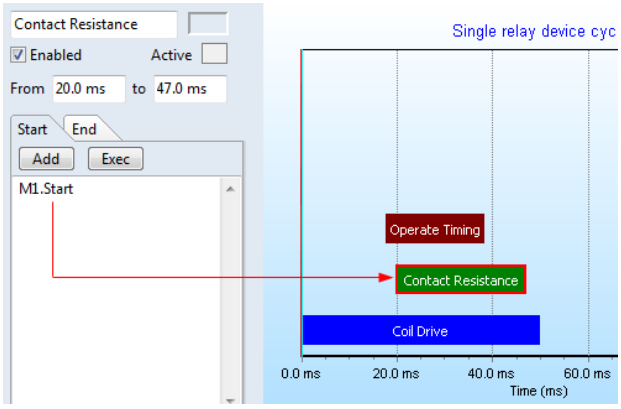

Figure 6. The Contact Resistance time slot START event

At 20ms (Fig 6) the Contact Resistance time slot instructs the measure input M1 to 'start' its voltage integration. (M1 will be connected across a specific relay contact).

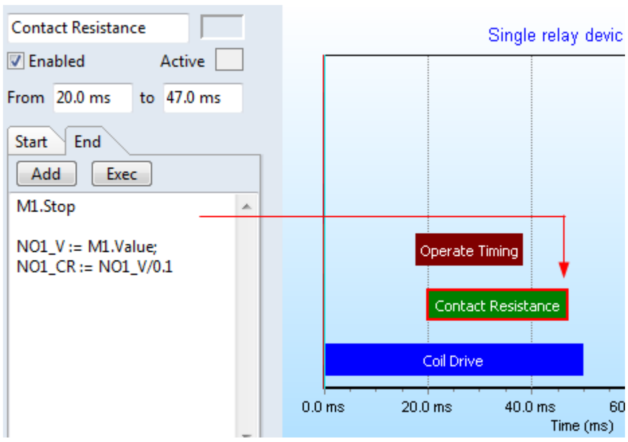

Figure 7. The Contact Resistance time slot STOP event

At 47ms (Fig 7) the Contact Resistance time slot instructs the measure input M1 to 'stop' its voltage integration – after this point, the result of the voltage integration (the contact voltage drop or CVD) is available to the system and is held ready for reading. The time duration between the start and stop is flexible and will determine signal quality and stability of the measurement. If it is an AC signal, this duration will typically be a whole number of line cycle periods.

The next two lines define the results to be displayed and a summary of these actions is shown below.

Action	**Description**
M1.Stop	Tells the contact measure input top stop its voltage integration.
NO1_V:= M1.Value	Transfer the contact voltage drop result to a life-test 'result sheet' called 'NO1_V'
NO1_CR:=NO1_V/0.1	Use the contact voltage drop to calculate the contact resistance and transfer this to a life-test 'result sheet' called 'NO1_CR'

The lines with the ':=' assignment are simple to code but very powerful in the system because for each assignment, a 'result sheet' is created which knows how to display potentially millions of values (one per cycle), compare them with programmed limits and stop the cycling if a failure occurs.

For example, the action of multiple cycle executions of the line 'NO1_V:= M1.Value' is to create a unique display of contact voltage drop such as Fig. 8 below which builds over time:

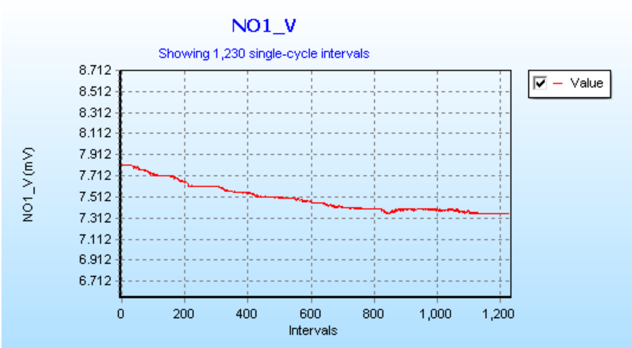

Figure 8. A result sheet from the assignment NO1_V:= M1.Value

This result sheet plots numbers of test cycles at the bottom against contact voltage in millivolts at the left.

The final program line 'NO1_CR:=NO1_V/0.1' is a simple 'Ohms-law' calculation to derive contact resistance from contact voltage drop, using the known 0.1A test current. The assignment of this calculation to 'NO_CR' creates a second result sheet as shown in Fig. 9:

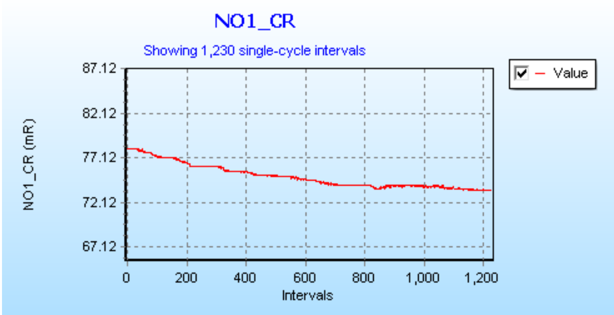

Figure 9. A result sheet from the assignment NO1_CR:=NO1_V/0.1

Note that an alternative way of obtaining contact resistance could be to make TWO measurements, one for contact voltage drop (as above) and a second for contact current (measured across a current shunt resistor). This would be a minor change to the syntax replacing the '0.1' with another measured input value for the shunt resistor voltage.

IV. CONTACT TIMING.

Using the same concepts of time slots and flexible events it turns out that measuring and reporting 'contact timing' can be implemented in a very similar way to that of measuring contact voltage drop.

The 'Operate timing' time slot shown in Fig. 10 below need have no 'start' event programmed, only a 'stop' – chosen to be at 38ms – enough time for the contacts to transfer into the operated state. The event programming shows three lines, each of which is a simple assignment for transferring specific hardware measurements into unique result charts: 'time to first edge', 'number of bounces' and 'bounce time'.

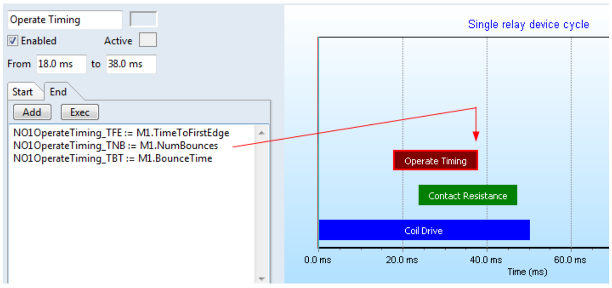

Figure 10. Creating three result sheets for the hardware timing results of the contact.

When the cycling starts, the execution of the event creates charts and data just like we saw previously for contact resistance – examples are shown in Fig. 11 and Fig. 12. Note that implementing the measurement of release time values is identical but in the latter half of the relay cycle.

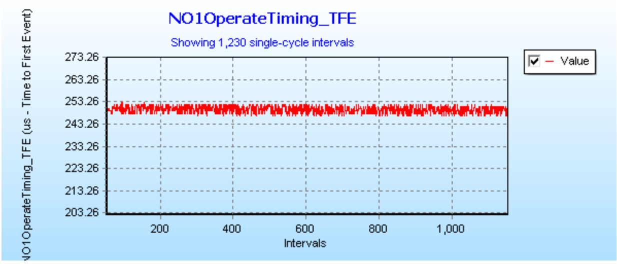

Figure 11. Operate time chart by contact cycles.

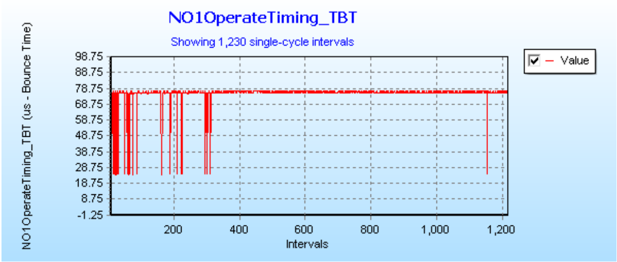

Figure 12. Bounce time chart by contact cycles.

Already we can see the benefit of this approach to obtaining life test data. The charts for timing (operate time and bounce time) can be directly compared over life with those for CVD and CR, thus any change in the device behaviour over life will show up on any relevant chart leading to a level of insight much greater than obtained with only simple timing test made before and after a life-test.

V. USING 'INTERVALS' FOR TESTING AND REPORTING.

The charts displayed above show some 1,000 cycles of test results but in a real world life-test there are usually many more cycles, possibly millions. Each cycle must be tested and yet the charts need only show between 100 and 1,000 data points to convey useful information about performance and trends. This can be achieved by the use of test 'intervals' where chart points are only plotted at the end of each interval rather than for every cycle. Because each cycle is still measured and checked, there are actually three values now to be plotted – min, max and mean for each interval. A typical chart showing interval testing is shown in Fig. 13:

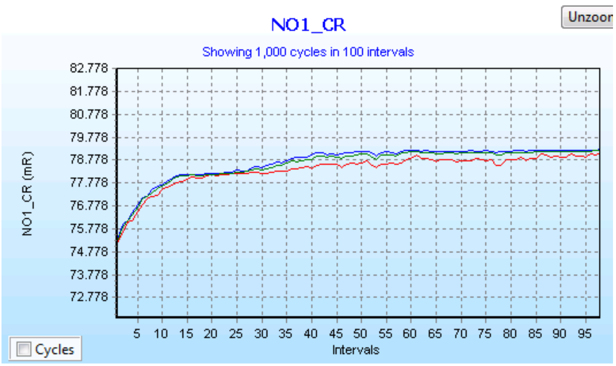

Figure 13. A chart showing the Min, Max and Mean interval results

In this chart the 100 intervals shown are distilled from 1,000 actual test cycles with each interval representing 10 cycles. You can still clearly see useful trend data even with 100 points instead of 1,000. You can see the contact of the tested reed relay heating up under its 100 mA load during the first 15 intervals.

VI. OPERATE AND RELEASE VOLTAGE.

To measure operate and release voltage, the interval method of testing is used. A typical method for measuring the operate and release voltage of a relay is to ramp all coil voltages up and down whilst monitoring the contact states. Because this takes some 10-30s to complete it cannot be done on a single-cycle basis but is ideal for testing after each interval, which depending upon the user configuration can take from several minutes to several hours.

To implement the operate and release voltage test at the end of each interval two simple lines are inserted into the 'fixed event' 'interval after end' as shown in Fig. 14:

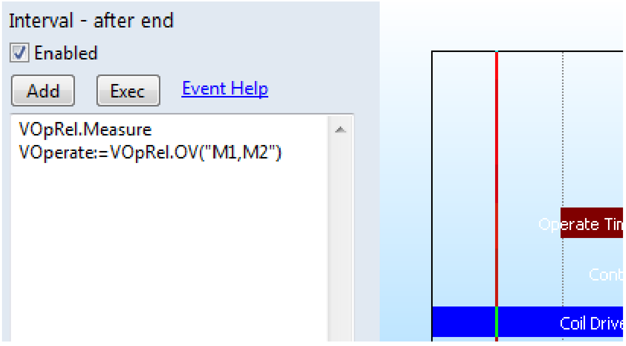

Figure 14. Programming the system to perform an Operate / Release Voltage test.

When a cycle interval completes the relay cycling pauses and this event is 'executed'. The first line requests the test to be performed, measuring all contacts in parallel to obtain the operate and release voltage value for each contact. The second line obtains the stored result for named contacts and assigns them to a result chart display (there is another configuration page where the ramp parameters are defined once for the entire test).

A typical output from this part of the test is shown in Fig. 15.

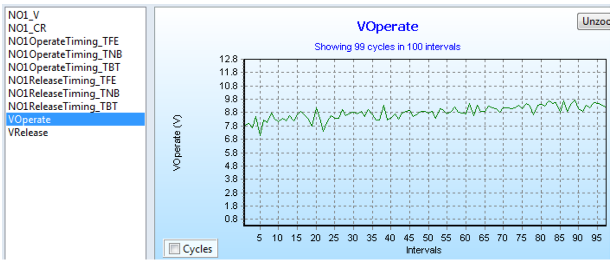

Figure 15. Operate Voltage chart by test intervals.

The release voltage results are shown in Fig. 16.

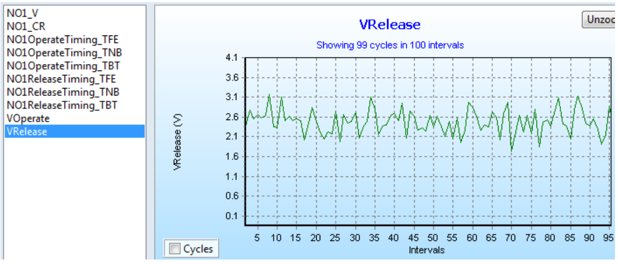

Figure 16. Release Voltage chart by test intervals.

You can see from the charts that there is a trend in the operate voltage slightly upwards throughout the test. This may indicate an increase in the relay mechanical pivot friction due to wear – an insight that may not have been as clear if the operate and release voltage was simply measured at the beginning and end of the life test.

VII. FAILURE DETECTION.

So far, all data has been shown in the form of line charts, but each result sheet can display its numerical data as a line chart, a summary of values (min, max etc) and a distribution chart as shown in Fig. 17.

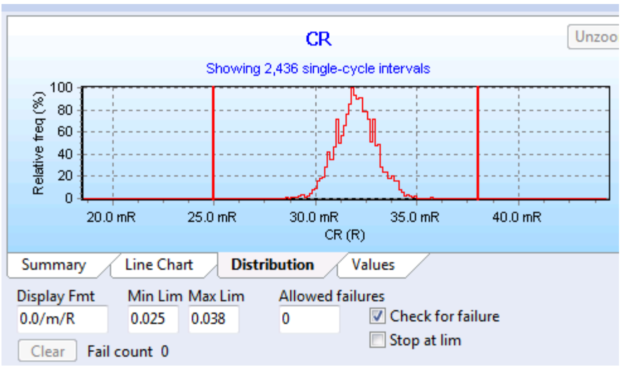

Figure 17. Each result can display its data in various ways.

In addition, each result sheet has settings for its display format and limits. Other controls allow it to stop the cycling after a specified number of failures. Because these controls are unique to each result sheet, significant flexibility is obtained, for example to program the test to stop on a key change in one specific measured parameter.

VIII. SIGNAL CONDITIONING AND PROTECTION.

Although it is possible to connect real-world relay contact signals directly into the measurement inputs of the test system we chose to use an architecture where signals first pass through a signal conditioning module. This provides inputs which are fully differential with a common-mode range of up to +/- 400V. An additional benefit of applying this signal conditioning is that cable runs back to the measurement controller have little effect on actual measurement performance since we are emulating the situation that exists with an active oscilloscope probe where wiring between probe and measurement is far less sensitive to pick-up – a boon when constructing physically large test systems with high cable currents.

The result of this architecture is to create a system that can be 'overlaid' on to an existing high-power life test setup and with a few simple connections, a complete life-test capability created, all with little more than with typical oscilloscope connections.

IX. TESTING COMPLEX DEVICES.

Because the cycle programming of input measurements and output switching is very flexible the architecture can be used to program the testing of more complex devices such as that shown in Fig. 18.

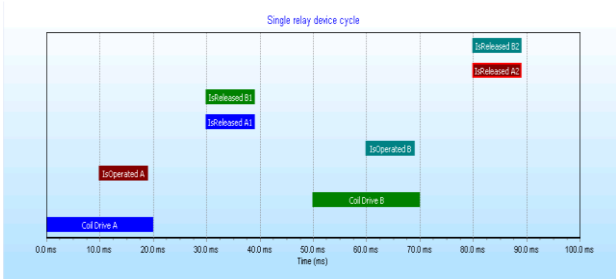

Figure 18. Testing a more complex device.

Here, the device is a special relay that has two devices within the same package – relay 'A' and relay 'B'. This style of relay is actually a dual changeover style and is often found driving an automobile window motor in each direction. Testing this device requires that each device is tested individually and that there is no interaction between them. The flexibility of the cycle programming allows for this very easily within the one relay cycle.

X. DERIVING NEW RESULTS FROM MEASURED DATA.

The ability of the architecture to produce 'derived' results using expressions is highly useful since it permits you to make additional device tests at no cost from your existing measurements. Consider the case where a relay has two parallel contacts and it is known that the matching between the contacts is an important measure of manufacturing quality. You will already be measuring and checking the basic contact resistance of each contact as part of a normal test, but to monitor the matching of the contacts you can simply introduce a new result sheet based on an expression for deriving the contact resistance difference (CRDiff) between the two measured contact resistance values:

$$\text{CRDiff} := \text{CR2} - \text{CR1}. \quad (1)$$

The CRDiff' now has its own result sheet together with its limits, so you are now able to program a test that stops on a 'matching failure' of (say) 3mR as well as on a basic contact CR failure that exceeds (say) 25mR.

XI. CONCLUSION

This paper has introduced a new architecture of life-test equipment using a very modular set of hardware resources backed by cycle-based test software. Simple programming expressions adapt the system to the exact requirements of the test in the manner of a Programmable Logic Controller. Further insight can be provided by the inclusion of timing and operate/release voltage measurements on a cycle by cycle or interval basis throughout the entire life-test.

XII. ACKNOWLEDGEMENT.

The Authors wish to thank the customers of Applied Relay Testing Ltd and its engineering staff for helpful advice and assistance in the preparation of this paper.

XIII. REFERENCES.

[1] B.J.Frost "A new generation of test equipment" *Proc. of the 45th relay conference, NARM, April 1997*

[2] S.J.Hobday "Bringing insight into the analysis of relay life-test failures" *Proc. of the 51st relay conference, NARM, April 2003*