



User's Guide to Flexdata 2

v12

**System Control Platform for Integrated Data Capture and
Analysis**

Easy Experiment Setup

Test Management

Data Analytics

Cloud Storage

Remote Collaboration

Table of Contents

User's Guide to Flexdata 2	1
Table of Contents	2
Preface - Original Flexdata	3
Introduction - Flexdata 2.....	4
Flexdata 2 Installation	5
FT232H Installation for ConnectBox.....	7
Flexdata 1 Quick-Start Guide.....	9
Flexdata Algorithms	14
Flexdata DR Motor Move Block.....	31
Flexdata Variables	33
Flexdata Load Cell.....	37
Flexdata Continuous Measurements.....	44
Flexdata SmartStop	55
Flexdata Action Replay	65
Flexdata ConnectBox.....	69

Preface - Original Flexdata

The original Flexdata platform ("FD1") was designed and conceived to meet the needs of easily connecting popular and often used mechanical testers, measuring instruments and data visualization tools dedicated for printed electronics lab automation applications. There was a growing emphasis for easy and powerful data analytics to enable more efficient and rapid prototype development, training, and commercialization certification.

Flexdata includes Plug and Play for control and synchronization of various mechanical testing tools, measuring instruments (resistance, temperature, humidity), and remote video surveillance. Test machines from Yuasa Systems include DMLHB, DMLHP, and DR11MR/4 motor drives with jigs that fold, stretch, flex, and twist samples. Instruments are Keithley DMM6500, DAQ6510 (with 7710 Multiplexer), and Hioki RM3545-2 (with Z3003 multiplexer). Cameras are USB webcams.

Flexdata also provides Smart Operation including the (re)use of complex testing patterns designed by test algorithms to run repetitive or similar testing conditions for multiple motors at any time. Import/export of data files for legacy experimental data ingestion facilitates collaboration.

System integration includes compatibility with temperature and humidity probes (designed for hostile integration), web and advanced optical camera synchronization (Logitech Webcams, IDS Cameras)

Integrated cloud infrastructure enables real-time automatic backup of data and images that facilitates multi-site, multi-location collaboration. If preferred, FD can be used on a PC with private/local storage.

Introduction - Flexdata 2

Flexdata has been updated with a new version FD2. While all the features of FD1 are retained, many new features have been added.

Flexdata Load Cell has been added. The Unipulse Tension / Compression Load Cell UNCLB is used to allow measurement of the Load on samples under Stretching and Compression. The Flexdata Load Cell can be used with Yuasa ST and FS-N Jigs.

Continuous Measurements are now possible. With FD1 it was necessary to stop the test machine to make measurements. With FD2 measurements can be made continuously throughout a test. This feature is especially useful when measuring Load which changes during each test machine reciprocation cycle.

SmartStop has been added. SmartStop allows FD2 to use measured data to change the direction of a test. The test algorithm can set maximum, or minimum values that if exceeded can cause the test to pause. This gives the Operator a chance to examine the sample once a critical threshold has been reached. The Operator can decide to continue the test or to stop the test.

Action Replay has been added. This feature can be helpful when examining a sample following an endurance test. If the sample at the end of the test has changed from what it was at the beginning of the test, the problem, and when the problem occurred, might be discoverable by examining snapshots taken at the measurement periods during the test. Flexdata Action Replay can play back the snapshots at high speed, like a movie, to make the discovery quicker and easier.

ConnectBox enables engineers to add other accessories or proprietary hardware to the Flexdata platform. The external hardware will use their own programming to customize their test requirements. FD2 now makes that possible by use of the Flexdata ConnectBox.

Flexdata 2 Installation

FD2 installation is the same as FD1 installation which is covered in the Flexdata Quick-Start Guide according to this chart from the Quick-Start Guide.



Bayflex Solutions Flexdata Control Package
test machines + control + data capture & analytics + cloud storage + remote viewing

bayflex
solutions

Steps to set up Bayflex Flexdata

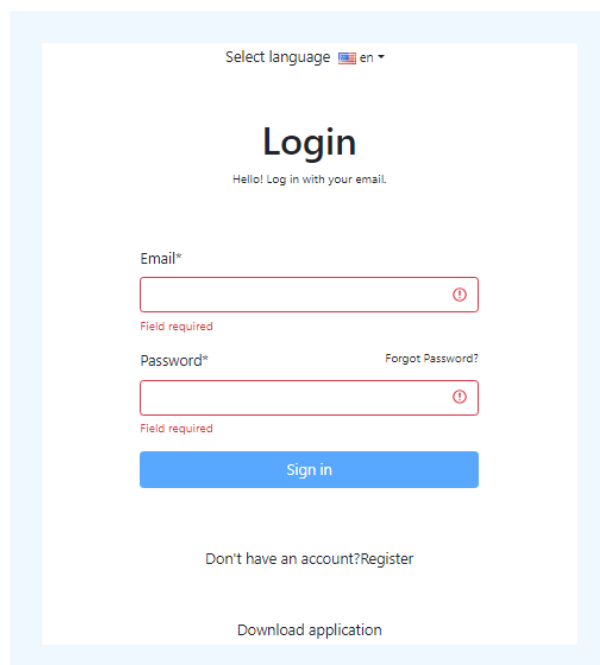
First Time Installation

1. Download and Install on the Laptop
 - Virtual Instrument Software Application (VISA)
 - Flexdata App
2. Set IP Address and Subnet Mask
 - Motor Unit
 - Laptop
3. Connect Motor Units and Laptop to the LAN with RJ-45 Cables
4. Connect Meters to the Laptop with USB cables
5. Connect Cameras to the Laptop
6. Run Flexdata App
7. Set up Equipment -- Motor Units, Meters, Cameras

To download the FD2 app, open the FD2 Cloud with a browser at:

<https://flexdata.bayflectechnologies.com>

You will see the following screen:



Select language en

Login

Hello! Log in with your email.

Email*

Field required

Password* [Forgot Password?](#)

Field required

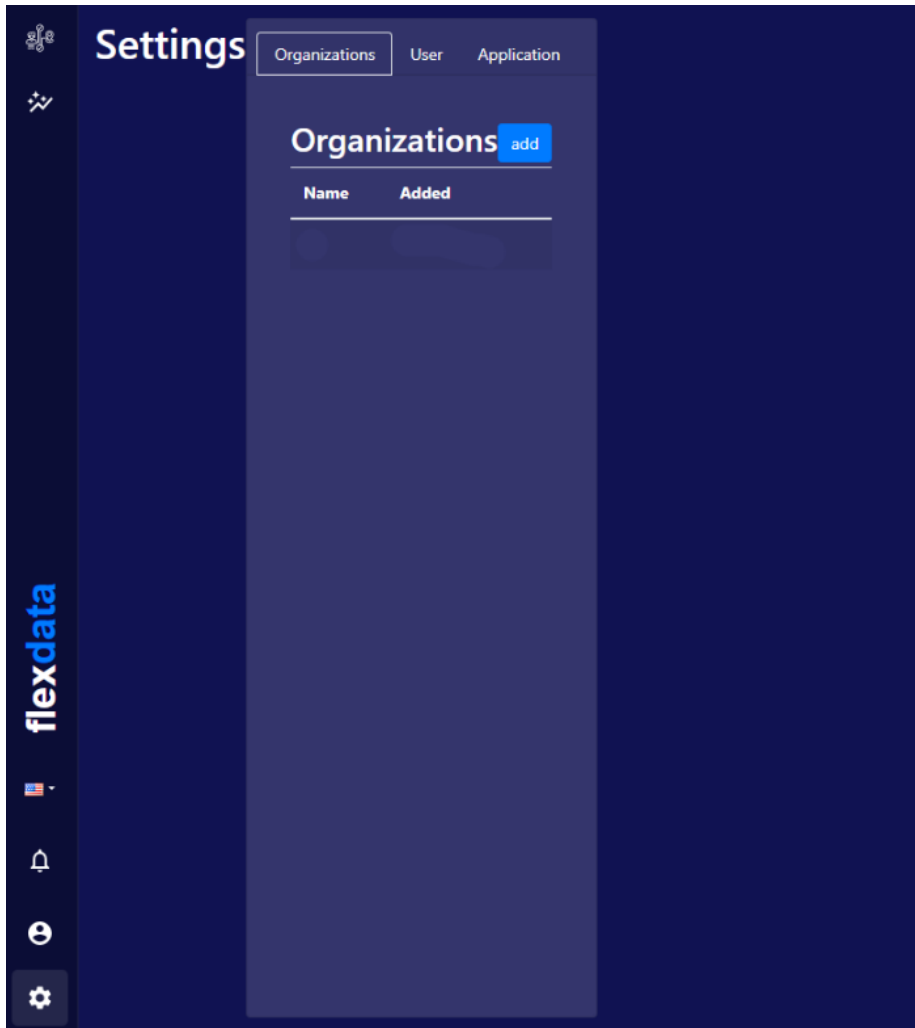
[Don't have an account? Register](#)

[Download application](#)

Click on "Register" beside "Don't have an account?". Register using your email address and select a password for your own use.

You will get an email to confirm registration. Clicking on the link in the email will give you a Confirmation with another link "Go to main page", click on that link.

You should see the following screen with your "Name" listed and the "Added" date.



If you already have an account, logon and go to the Settings Page.

Click on "Application" and you will be able to download the FD2 alpha version "2.0.0-1.22.exe".

If upgrading from FD1 to FD2, Equipment Templates may need to be updated. As an example, the Keithley DAQ6510 in FD1 had 10 multiplexer channels, in FD2 there are 20 multiplexer channels and two multiplexers. Simply delete existing templates that need updating and set up new templates. The FD1 "default" template should be deleted for a new FD2 "default".

One other download and installation, "Zadig", will be required for "ConnectBox".

FT232H Installation for ConnectBox

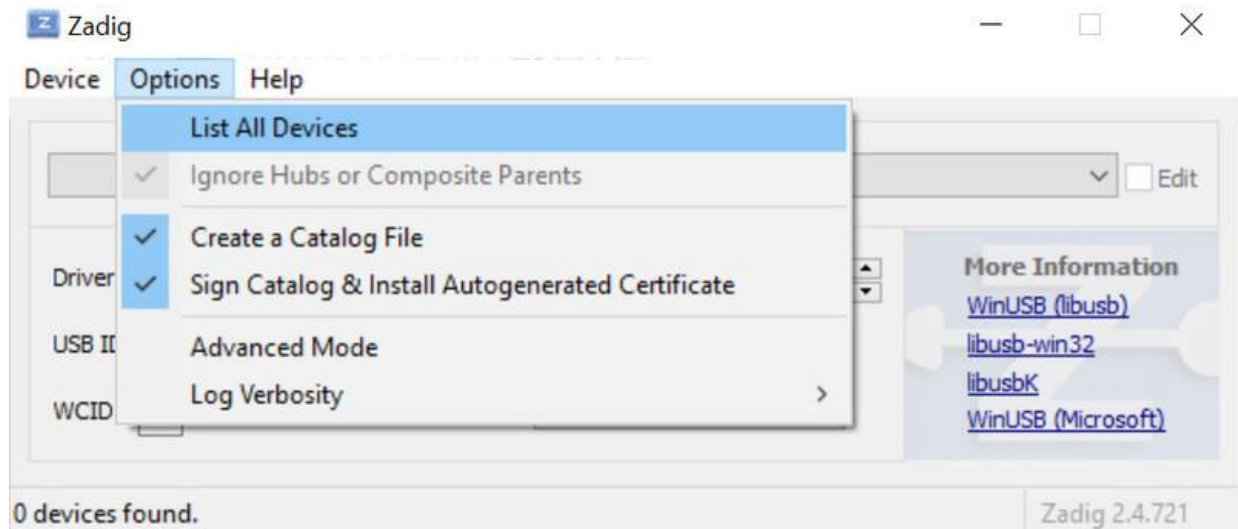
The Zadig website is at: <https://zadig.akeo.ie/>

You can download Zagid-2.8.exe at:

<https://github.com/pbatard/libwdi/releases/download/v1.5.0/zadig-2.8.exe>

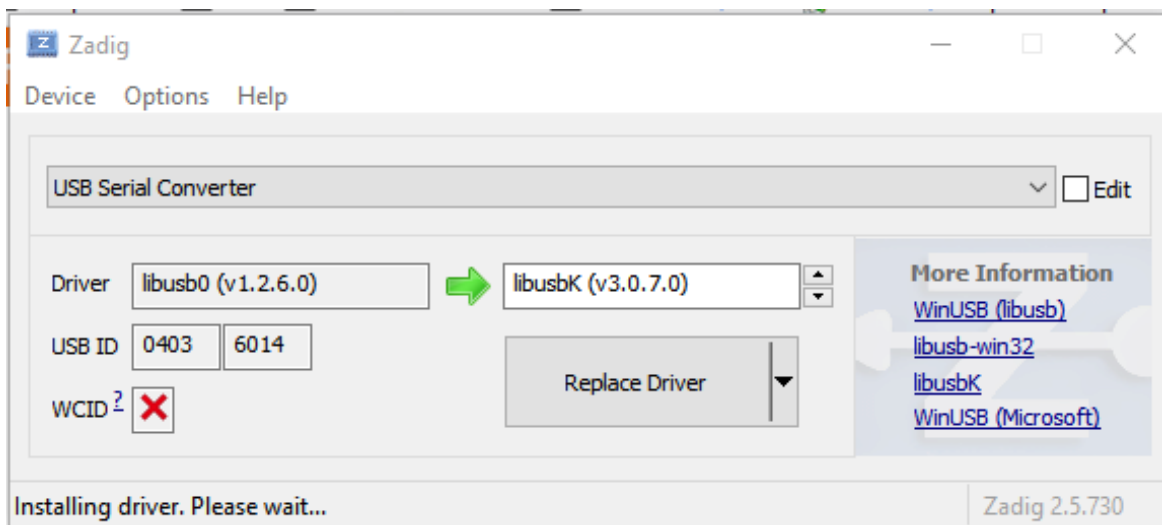
Connect the ConnectBox to the Laptop.

Run the Zadig tool. Click the **Options** menu and select the **List All Devices** item below:



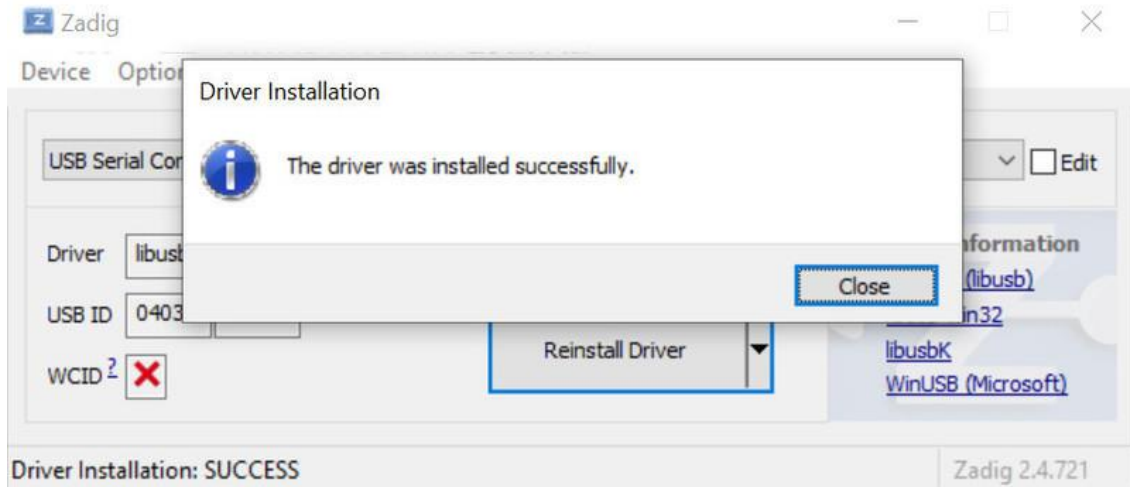
The list box of devices should populate with many devices. Select the USB Serial Converter device shown below.

Note: Make sure to select the device with Driver equal to FTDIBUS and USB ID equal to 0403 6014! If you pick the wrong device you might accidentally uninstall another device's driver and make it inoperable.

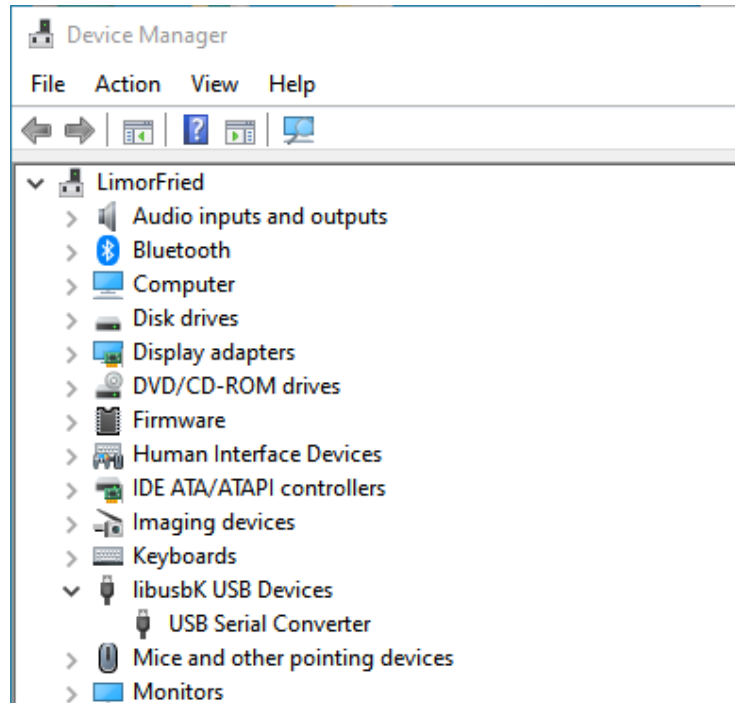


Click the up/down arrows on the driver select box to the right of the green arrow and select the **libusbK** driver as shown above.

Click the Replace Driver button to replace the FTDI driver with the libusbK-based driver. After the driver replacement finishes you can close Zadig tool.

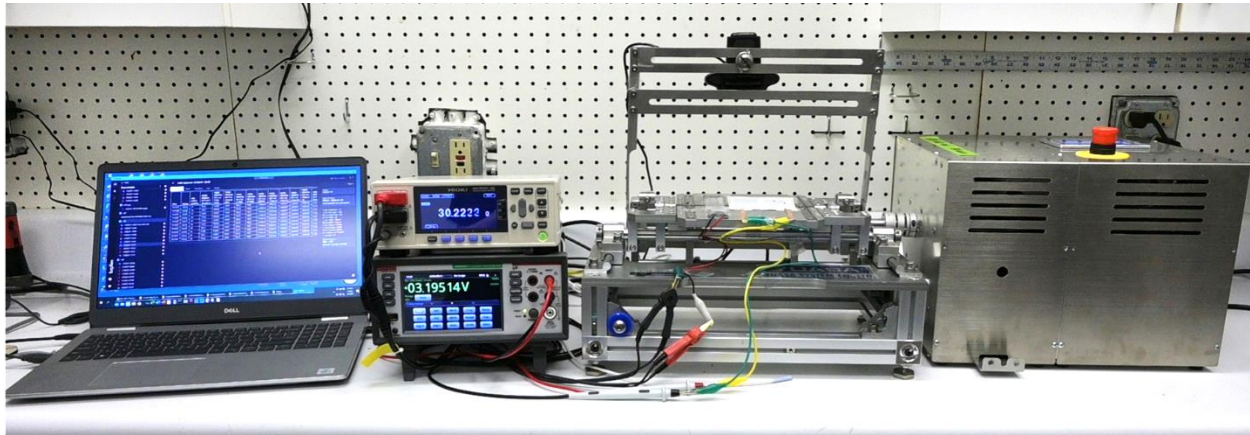


To check that the driver was successfully replaced, open Device Manager from Control Panel or searching in the Start menu. You should see a new top level node **libusbK devices** and the **USB Serial Converter** underneath it as shown below.



Flexdata 1 Quick-Start Guide

The Flexdata 1 Quick-Start Guide has been written to supplement the [Flexdata Setup Video](#). The Quick-Start Guide is brief but contains essential information to run the first Flexdata test. See the [Video Link](#) for more detailed information, and for more information on designing test algorithms.



First Time Installation

1. Download and Install on the Laptop
 - 1.1 Virtual Instrument Software Application (NI-VISA)
 - 1.2 Flexdata App
2. Set IP Addresses and Subnet Masks
 - 2.1 Motor Units
 - 2.2 Laptop
3. Connect Motor Units and Laptop to the LAN with RJ -45 Cables
4. Connect Meters to the Laptop with USB cables
5. Connect Cameras to the Laptop with USB cables
6. Run Flexdata App
7. Set up Equipment in Flexdata -- Motor Units, Meters, Cameras
8. Initial Test Setup

Flexdata Equipment Compatibility

Yuasa Motors	Yuasa Test Jigs	Meters
DMLHP	CS	Hioki RM3545-2
DR11MR	FS	Hioki Terminal Box
DMLHB	ST	Yuasa Termnal Box
DLDM111LH	P150	Keithley DAQ6510
TCDM111LH	FT	7700 Multiplexers
Motors with Ethernet Port	TW	Keithley DMM6500
	FR	Keithley 2100
	and many others	Keithley 2110

First Time Installation

1. Download and Install on the Laptop

1.1. NI-VISA Setup File [Video Link 4:15](#)

- 1.1.1. Go to <https://www.ni.com/downloads> using your Browser.
- 1.1.2. Search for "NI-VISA".
- 1.1.3. Download the setup file to your "Downloads" directory (or directory of your choice).
- 1.1.4. Run file "ni-visa_20.0_online_repack3.exe" in your "Downloads" directory (downloaded version might have different filename).
- 1.1.5. Accept default setup, accept license terms, and continue the installation until it is complete.

1.2. Flexdata Setup File [Video Link 5:09](#)

- 1.2.1. Go to <https://flexdata.bayflectechnologies.com/downloads> using your Browser.
- 1.2.2. Click on "Download".
- 1.2.3. Download the setup file to your "Downloads" directory (or directory of your choice).
- 1.2.4. Run file "Flexdata Setup.exe" in your "Downloads" directory (downloaded version might have different filename).
- 1.2.5. At the end you will be given an option to open Flexdata, decline the option. You will open Flexdata later.

2. Set IP Addresses and Subnet Masks [Video Link 6:02](#)

2.1. Motor Units

- 2.1.1. "DM" Motor Unit default IP Address is 192.168.0.1 and can be changed. The default Subnet Mask is 255.255.255.0.
- 2.1.2. To run "DM" on LAN, change IP Address and Subnet Mask to match LAN (e.g., if LAN is 192.168.2.1, perhaps use 192.168.2.151 and higher for Motor Units). Use the settings page on the Motor Unit Touch Panel to change the IP Address.
- 2.1.3. "DR" Motor Unit IP Address is 192.168.10.1. It cannot be changed and it cannot be run on a LAN.

2.2. Laptop

- 2.2.1. With "DM" Motor Units on a LAN, perhaps use DHCP for Laptop. Laptop **must** be able to connect to the Internet for the Cloud.
- 2.2.2. With "DM" Motor Units using a single RJ-45 Cable between Motor Unit and Laptop, set Laptop IP Address to match the Motor Unit default IP Address 192.168.0.1 (perhaps use 192.168.0.10). Laptop **must** have WiFi to connect to the Internet for the Cloud.

2.2.3. With "DR" Motor Units, you must use a single RJ-45 Cable between the Motor Unit and the Laptop. Change the Laptop IP Address to match the Motor Unit default IP Address 192.168.10.1 (perhaps use 192.168.10.10). Laptop **must** have WiFi to connect to the Internet for the Cloud.

3. Connect Motor Units and Laptop to the LAN with RJ-45 Cables

3.1.1. If using single RJ-45 Cable, connect Motor Unit directly to the Laptop. Laptop **must** have WiFi to connect to the Internet for the Cloud.

4. Connect Meters to the Laptop with USB Cables

5. Connect Cameras to the Laptop with USB Cables

6. Run Flexdata App

6.1.1. Installation should have put a "Flexdata Link" on your Laptop's desktop.

6.1.2. Double-click on the Flexdata Link.

6.1.3. You may be required to Login. If you have a Flexdata Cloud account, simply do a Login. If you do not have a Flexdata Cloud account, you can Register by providing a Login ID and Password.

7. Set up Equipment in Flexdata - Motors, Meters, Cameras *Video Link 11:33*

7.1. Install Motors in Flexdata

7.1.1. Click on Equipment page.

7.1.2. Click on DM or DR Motor Unit page.

7.1.3. Click on "add..."

7.1.4. Type in Motor Unit's IP Address.

7.1.5. Click on Connect.

7.1.6. Set up Template as desired. The Template sets the start conditions for a test.

7.1.6.1. For DMLHP motors, no special comments on the Template.

7.1.6.2. For DMLHB or DLDM or TCDM motors the Template can include Slow-Down Mode values.

7.1.6.3. For DR11MR motors, the Template can include "Test angle +" (or "+SP"), "Test angle -" (or "-SP"), and "Jog position" (or "Stop Position"). These are start values, they do not change during the test.

7.1.7. Install other Motor Units as desired.

7.2. Install Meters in Flexdata

7.2.1. Click on Equipment page.

7.2.2. Click on Keithley page or Hioki page.

7.2.3. Click on "add..."

- 7.2.4. Flexdata will find the meters.
- 7.2.5. Select the desired Serial Number.
- 7.2.6. Click on Connect.
- 7.2.7. Set up Template as desired.
- 7.2.8. Install other Meters as desired.

7.3. Install Cameras in Flexdata

- 7.3.1. Click on Equipment page.
- 7.3.2. Click on Camera page.
- 7.3.3. Flexdata will find the cameras.
- 7.3.4. Select the desired camera.
- 7.3.5. Click on Connect.
- 7.3.6. Set up Template as desired.
- 7.3.7. Install other Cameras as desired.

8. Initial Test Setup

8.1. Design Algorithm [*Video Link 21:20*](#)

- 8.1.1. This Quick-Start Guide will help you set up a simple experiment, you will need to design your own more complex experiments. For other suggestions, see:
[*Video Link 29:11*](#) [*Video Link 39:11*](#) [*Video Link 45:36*](#)
- 8.1.2. Click on Experiment page.
- 8.1.3. Click on "Create new experiment".
- 8.1.4. Click on "Templates" tab.
- 8.1.5. Select a Template (typically the first of the two suggestions).
- 8.1.6. Select other Blocks as desired from "Device" tab (to add second meter, for example)
- 8.1.7. Move the Blocks into a single stack, in the proper order, for the desired experiment.

8.2. Set Blocks for desired equipment and templates

- 8.2.1. Select a Motor Unit with the first Device drop-down arrow and the preferred Motor Unit Template from the second drop-down arrow on the same Device block. The Algorithm might suggest a device and template, change as desired.
- 8.2.2. Select Meters and Meter Templates with other Device drop-down arrows.
- 8.2.3. Select Cameras and Camera Templates with other Device drop-down arrows.
- 8.2.4. Add Device blocks as needed. Move the Device block into the desired location in the Experiment Algorithm.

8.3. Set "repeat" => Number of Measurements during Test

8.3.1. Click on the number inside the Repeat Block to set the number of measurements to be made during the test.

8.4. Select Motor to "move" during Test

8.4.1. Click on the drop down list to select the motor to be used.

8.5. Set Motor "speed"

8.5.1. Click on the speed number to set the motor speed to be used in the experiment.

8.6. Set "position" => Cycles per Measurement Period

8.6.1. Click on the position number to set the number of cycles per measurement.

8.7. Save the Experiment, type in Title for the Experiment

8.7.1. A title is required, if you leave this page before saving, the Algorithm will be lost.

8.8. Run First Test [*Video Link 24:43*](#)

8.8.1. Click on "Execute".

8.8.2. If there are errors in the Algorithm, you will receive a notice.

8.9. Monitor in Cloud [*Video Link 27:35*](#) [*Video Link 42:16*](#)

8.9.1. Go to "<https://flexdata.bayflextechnologies.com/experiments>" using your Browser. This is the Bayflex Solutions Cloud.

8.9.2. The Chrome Browser was used during the design of Flexdata.

8.9.3. MS Edge Browser has been tested and found to work properly on the Flexdata Cloud.

8.9.4. If the live camera is not showing on the Cloud, make sure that the browser allows autoplay.

version 9 - 2 June 2021

Flexdata Algorithms

1. Introduction

Flexdata is a Cloud based controller for Yuasa Endurance Test Machines that allows control of the test machines, data capture and analytics, and Cloud storage with remote viewing of the test in progress. Flexdata is compatible with the Keithley DAQ6510 data acquisition meter (front panel and two 7710 multiplexers), the DMM6500 digital multimeter (front panel only), and Hioki RM3545-2 resistance meter (front panel and two Z3003 multiplexers). Flexdata offers easy experiment setup, test management, and data analytics.

An experiment in Flexdata is based on an algorithm that the Operator designs. A number of already-designed algorithms exist that the Operator can select. Using an algorithm from a library may require that the Operator adapt it to the particular experiment that is to be conducted. An algorithm is a .xml file that contains all the instructions to run an experiment. The xml file can be imported into Flexdata and can be exported from Flexdata. This allows other users to easily incorporate new algorithms into their Flexdata library.

This Application Note will explain how algorithms work, the blocks that make up the algorithms, how test machines and meters are set up in the algorithms, and how the algorithm actually controls an experiment. Algorithms using a Torque Meter and Triggers will not be included in this Application Note but will be the subject of a separate Application Note.



Figure 1. Flexdata Experiment Using a PC Laptop, Hioki Resistance Meter, Keithley Digital Multimeter, Yuasa Clamshell Test Jig with Logitech Camera Mounted on a Bridge, and a Yuasa DMLHP Motor Drive Unit

2. Simple Flexdata Algorithms

Figure 2 is a very simple algorithm that controls the DR Motor Drive Unit, the Keithley DAQ6510 Digital Acquisition Meter, and a Logitech C920e Webcam.

There are seven blocks in the algorithm, arranged vertically, and only one block in each line. Flexdata executes the algorithm the same way, one line at a time. Flexdata will not move to the next line until the current line is complete. It is a serial process, do the first command entirely, then do the second line entirely, etc. When all lines have been fully executed, the experiment ends.

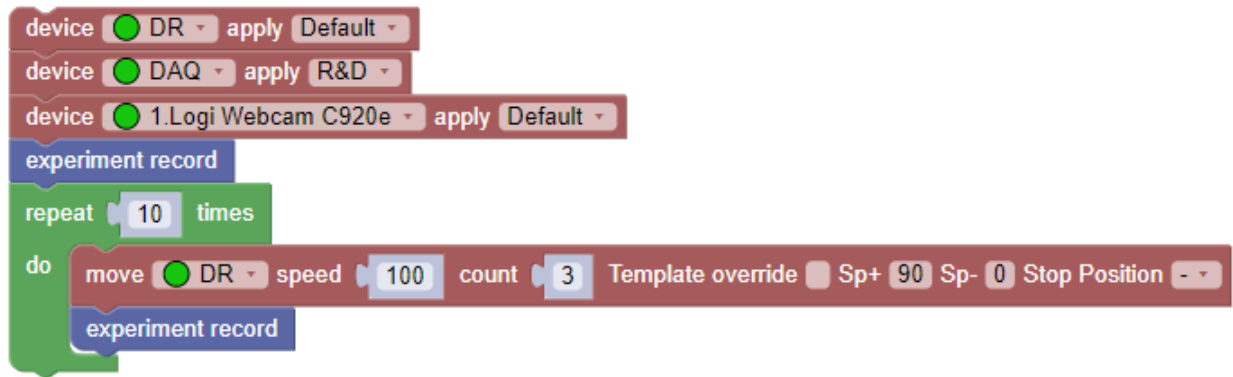


Figure 2. Simple Flexdata Algorithm

The first block, the brown "device" block, is required to instruct Flexdata to use the DR Motor in the experiment. Other motors could be specified, but for this experiment the DR will be used. The second window in the block instructs Flexdata to use the DR's "Default" Template to control the motor. The template is used to set up the motor for different conditions and will be explained later.

The second "device" block is required to instruct Flexdata to use the Keithley DAQ6510 in the experiment. Other meters could be specified, but for this experiment the DAQ will be used. The second window in the block serves the same purpose as the second window in the first block, it instructs Flexdata to use the DAQ's "R&D" Template.

The third "device" block is required to instruct Flexdata to use the Logitech C920e webcam in the experiment. As before, the second window instructs Flexdata to use the C920e's "Default" Template.

The fourth block, the blue "experiment record", instructs Flexdata to make all the measurements specified in the Templates for all the measuring devices, in this case the DAQ and the C920e.

The next block, the green "repeat" block, instructs Flexdata to use the commands inside the loop for 10 successive loops. Those commands are the brown "move" block and another "experiment record".

The "move" block contains seven different windows. The first window is to designate the motor. Only motors listed in the "device" blocks can be selected. The second window specifies the motor speed that is to be used. If the motor is a DR, it will be degrees per second. For other motor drive units, it will be reciprocations per minute. The next window "count" is to tell Flexdata how many reciprocations will be run in this specific "move" block. The remaining windows have meaning only for the DR. The next window, "Template override" will be explained later. The next two windows, "Sp+" and "Sp-" are used to set the end points of the motor for every cycle. The final window "Stop Position" is used to tell the DR to stop at SP+ or Sp- or at the midpoint of Sp+ and Sp- when the total "count" of the "move" has been reached. The cycle end points (Sp+ and Sp-) and Stop Position can be set remotely with the DR, but not with the other motor drive units. The end points and Stop Position for the other motor drive units, depending on the specific unit, must be set from the motor's Touch Panel, or manually through a window port on the motor drive unit. The analog motor drive units do not have a changeable Stop Position. The "Template override" window is used to tell the DR to ignore the values in the "Sp+" and "Sp-" and Stop Position windows and use the values that are on the DR Template that is being used for the experiment.

The final block, the second blue "experiment record" block, instructs Flexdata to make all the measurements specified in the measurement "device" templates.

The experiment will then consist of:

- 1) connecting to the equipment specified in the "device" blocks
- 2) making measurements on each of the measuring devices
- 3) running the motor for three cycles
- 4) making measurements on each of the measuring devices
- 5) repeating #3 and #4 an additional nine times.

Figure 3 is an algorithm that is included in Flexdata to allow the use of archival ECP files. Prior to Flexdata, ECP was a system to control Yuasa motors and take measurements. The ECP ProgramData file, a .csv file, was used to specify the motor speed and number of cycles at that speed. At the conclusion of the current number of cycles, measurements would be made, then the next sequence would begin. The file is specified in the "iterate CSV" window. To select the file, click on "Options" under the "Title" on the Flexdata screen. Just above the algorithm click on the window to the right of "Files" and then browse until the correct "csv" file is located.



Figure 3. Algorithm to Use Archive ECP ProgramData.csv Files

The only difference in the Figure 2 Algorithm and the Figure 3 Algorithm is the use of the "iterate CSV" loop in place of the "repeat" loop.

3. Explanation of the Blocks in Flexdata Algorithms

3.1. Flexdata Algorithms — Device Block

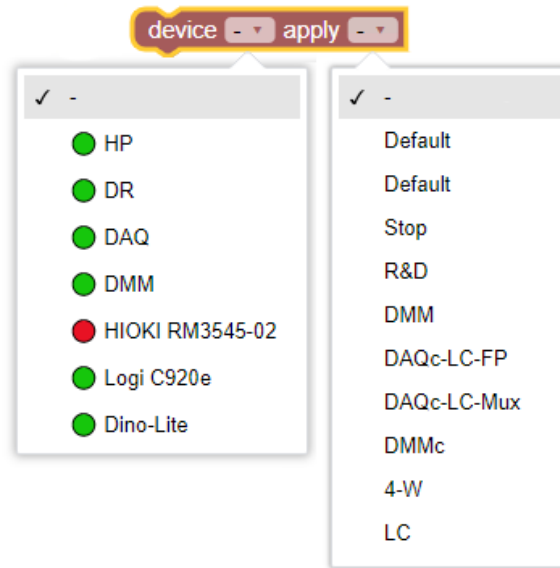


Figure 4. Flexdata Device Block

"device" — Device blocks are required for all equipment used in the experiment. Device blocks are used for motors and meters and cameras. Clicking on the down arrow in the "device" window will show all the equipment that has been hooked up and installed. Every piece of equipment used in an experiment must have its own individual Device block. Clicking on the down arrow in the "apply" window will show all the possible templates for the device that was selected in the first window. Device blocks must come first in the algorithm.

3.2. Flexdata Algorithms — Meter and Camera Device Blocks

1. **get measure** DAQ config on
2. **Load calibration** DAQ config on
3. **Start measurements** DAQ
4. **Stop measurements** DAQ
5. **snapshot** 0.Logi Webcam C920e

Figure 5. Flexdata Meter and Camera Device Blocks

1. "get measure" — Will measure the channel specified in the "on" window (selected using the down arrow) using the meter specified in the first window and the template in the "config" window (selected using the down arrow).

2. "Load calibration" — Will use the meter specified in the first window to measure the output voltage on the Load Cell on the channel specified in the "on" window using the template specified in the "config" window.
3. "Start measurements" — Will start continuous recording using the meter specified in the window. This block will be explained in Section 3.11 in this Application Note.
6. "Stop measurements" — Will stop continuous recording of the meter specified in the window. This block will be explained in Section 3.11 in this Application Note.
6. "snapshot" — Will take a snapshot using the camera specified in the window.

3.3. Flexdata Algorithms — Motor Device Blocks

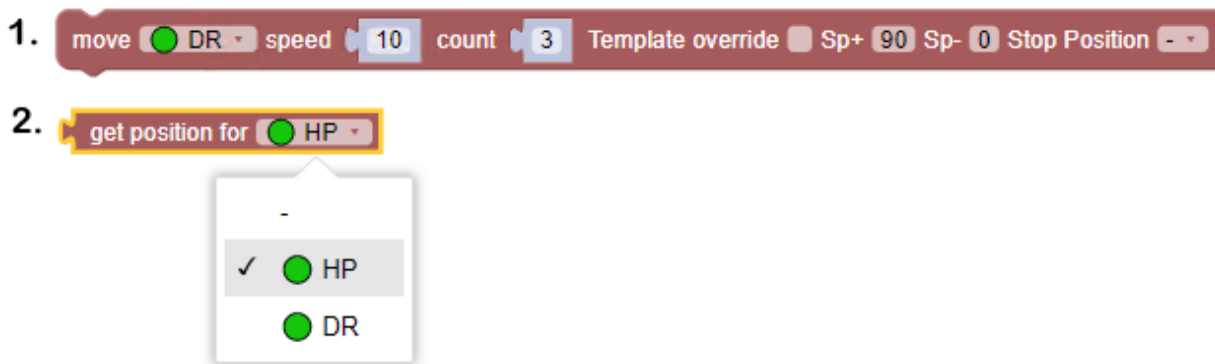


Figure 6. Flexdata Motor Device Blocks

1. "move" — Will start running the motor specified in the first window at the speed specified in the "speed" window for the number of cycles specified in the "count" window. The DR motor will use the "Sp+" and "Sp-" and "Stop Position" specified in the template that was given in the DR device block if the "Template override" window has a check. If no check, the "Sp+" and "Sp-" and "Stop Position" specified in the "Sp+" and "Sp-" and "Stop Position" windows will be used. If using "Template override" the Operator should confirm that the template values are the values desired for the experiment, and they may be different from the values in the "Sp+" and "Sp-" and "Stop Position" windows. This "move" block can be used by the DMLHP or DMLHB or DLDM111LH or TCDM111LH motors and only the windows "speed" and "count" will be used. The "Template override" and "Sp+" and "Sp-" and "Stop Position" windows will not be used.
2. "get position" — Will get the current position, or count, of the motor specified in the window.

3.4. Flexdata Algorithms — Variable Blocks

"Variables" — Flexdata allows the Operator to define Variables based on measured values from meters and values set by other blocks or combinations of blocks. Both the "Humidity" and "Load" parameters used in meter templates in Flexdata are internal variables based on a measured voltage. Variable Block choices are shown in Figure 7.

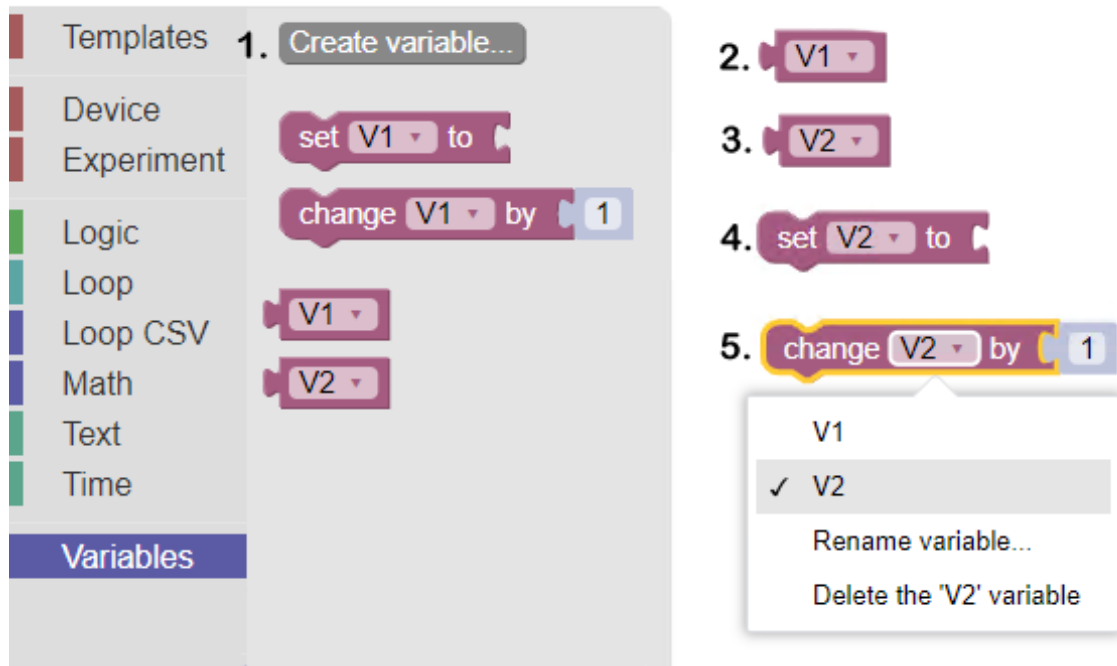


Figure 7. Flexdata Variable Blocks

A Honeywell sensor is used for Humidity. The equation that calculates the humidity is based on the equation:

$$V = M \cdot v + B$$

where V is the relative humidity, M is a constant dependent on the specific sensor, v is the measured voltage, and B is a second constant dependent on the specific sensor.

A Unipulse tension/compression load cell is used for Load. Load is exactly the same equation where V is the Load in Newtons, M is a constant depending on the specific sensor, and B is a second constant dependent on the specific sensor.

1. "Create variable" — Clicking on "Create variable" opens a window to name the variable.
2. "V1" — A variable created and named "V1".
3. "V2" — A variable created and named "V2".
4. "set" — This block is used to define the variable. It could be the value obtained by one of the meters and could be voltage, resistance, or any of the measured parameters used in Flexdata. In that case the block used on the right side of "set" would be the meter block "get measure".
5. "change" — This block is similar to the "set" block, it simply allows a change in the value of the variable. The variable will be changed by the value in the numerical window, and in the case of the value shown in the Figure, incremented by "1".

3.5. Flexdata Algorithms — Loop Blocks

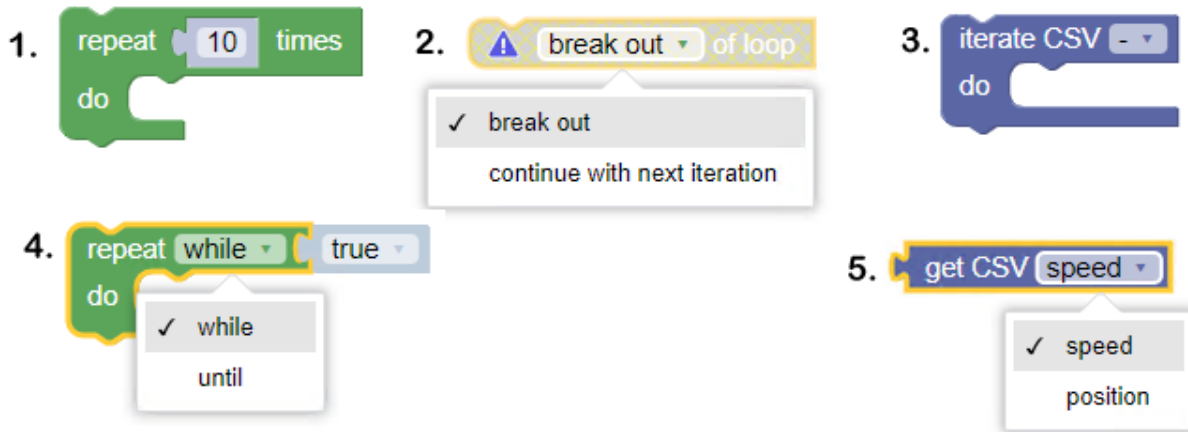


Figure 8. Flexdata Loop Blocks

1. "repeat" — The "repeat" block was introduced in the algorithm in Figure 2. Whatever blocks are inside the "repeat" loop will be executed serially, one at a time, until the total number of loops has reached the value in the window.
2. "break out" — The "break out" block is used to immediately move out of the loop and go to the next block below the "repeat" loop. A typical use will be to check for some measurement to be a specified value and then stop that portion of the experiment.
3. "iterate CSV" — The "iterate CSV" loop was introduced in the algorithm in Figure 3. This loop behaves the same as the "repeat loop" except the instructions on speed and count come from an archival CSV file used in an earlier control program ECP. The window after "CSV" is for entry of the "ProgramData.csv" filename.
4. "repeat while" or "repeat until" — This is a conditional loop where the number of repeats is unknown, it will depend on the condition that is set in the loop using math or logic blocks.
5. "get CSV speed or position" — This block is used in the "iterate CSV" loop to read the programmed motor speed and motor count from the ProgramData.csv file.

3.6. Flexdata Algorithms — Experiment Blocks

Experiment Blocks are shown in Figure 9.

1. "experiment record" — The "experiment record" is typically used multiple times in an algorithm. There must be at least one "experimental record" in an algorithm. When this block is executed, a new line will appear in the Flexdata Table listing time, iteration number, motor speed and angle and position, all the values from measurements specified in the meter templates, requested variables, snapshot filenames, and any notes Flexdata adds. A request to pause an experiment will result in a Note.

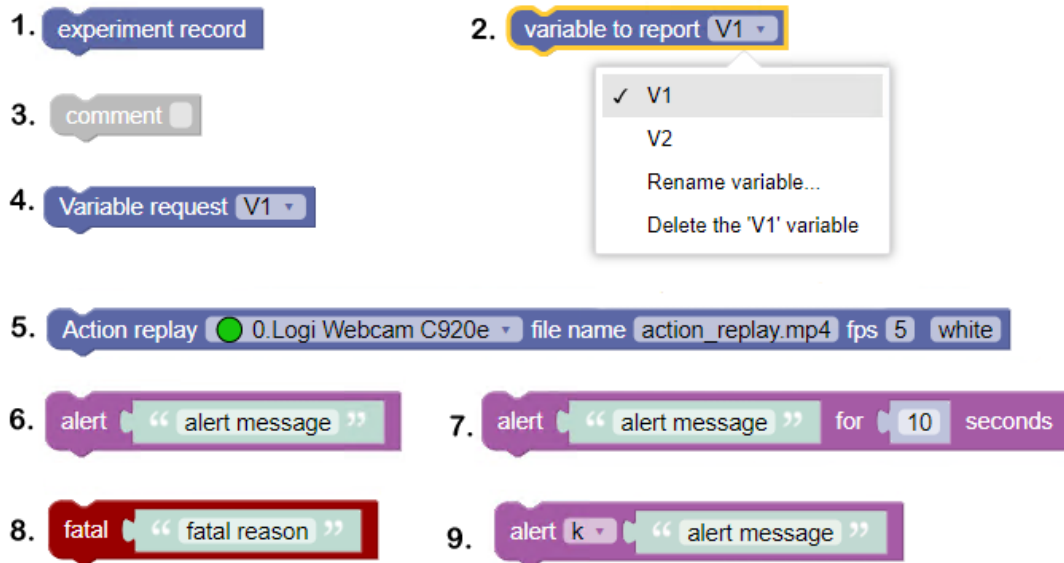


Figure 9. Flexdata Experiment Blocks

2. "variable to report" — This block will cause Flexdata to add that Variable in a new column in the Flexdata Table.

3. "comment" — A comment block can be added anywhere in an algorithm. It is often used to explain the algorithm at the beginning, and to explain specific blocks inside the algorithm. This block does not affect the algorithm, it is simply for information.

4. "variable request" — This block will cause Flexdata to pause and wait for the Operator to type in a value for the specified Variable. After typing the value, the Operator must click on "Continue" for Flexdata to resume the experiment.

5. "Action replay" — "Action replay" is used to playback a sequence of snapshots like a movie. This block will be explained in detail in Section 3.13 in this Application Note.

6. "alert" — The "alert" box is used to alert the Operator that some event has occurred. The alert will be at the same spot in the experiment as it is in the algorithm. It might be used to pause the experiment at a predetermined time, or to pause when an event has happened based on measured values. For the experiment to resume, the Operator will have to click on "Continue". If desired, the Operator can terminate the experiment by clicking on "Stop".

7. "alert for seconds" — This alert is similar to the prior "alert" except the pause will be for the specified number of seconds.

8. "fatal" — This alert is similar to the prior "alert" except the experiment will be fully terminated when this alert happens.

9. "alert variable" — This alert will pause Flexdata and display the value of the specified variable. Flexdata will remain paused until the Operator clicks on "Continue".

3.7. Flexdata Algorithms — Math Blocks

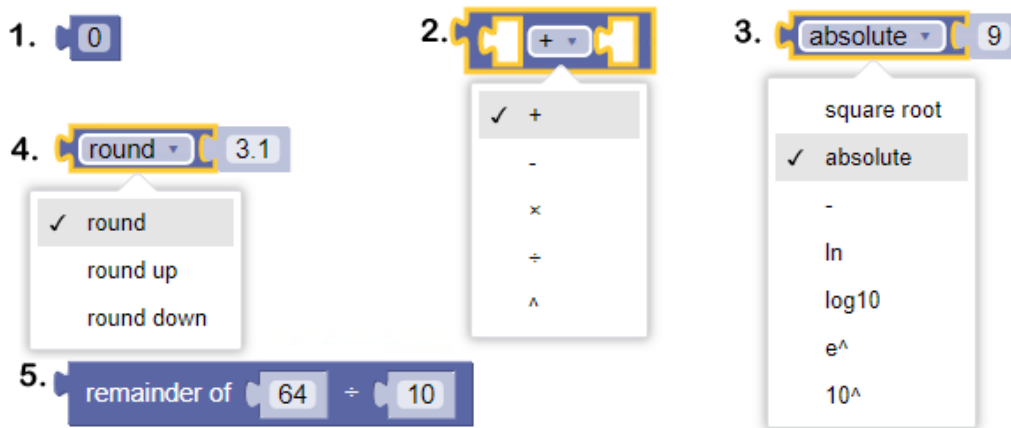


Figure 10. Flexdata Math Blocks

1. "0" — This block allows the Operator to introduce specific number values.
2. "+ -" — This block is a math operator that will combine the values in the first and third windows according to the selected math operator.
3. "absolute" — This block will perform a math operation on the value in the inserted right side window according to the selected math operator.
4. "round" — This block will round the value in the inserted right side window according to the selected round procedure. The "round" is with respect to whole numbers. For example, "round" the value "3.1" will produce "3". And "round" the value "0.31" will produce "0".

3.8. Flexdata Algorithms — Logic Blocks

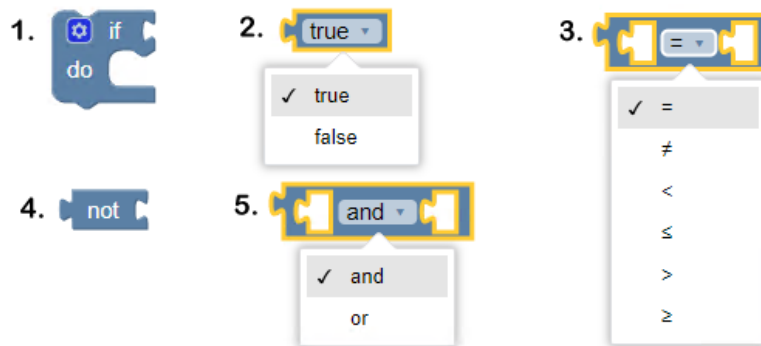


Figure 11. Flexdata Logic Blocks

1. "if / do" — This block will examine the condition specified in the block that has been inserted into the "if / do" opening on the right. If that condition is met, the "if / do" will execute the

commands in the "if / do" loop. If that condition is not met, Flexdata will execute the next command after the "if / do" loop.

2. "true / false" — Used to meet conditions.
3. "equality" — This block is used to set a condition between the two parameters in the two windows, whether equal, or not equal, or greater than, etc.
4. "not" — Reverses the condition.
5. "and / or" — Logic of both or either conditions.

3.9. Flexdata Algorithms — Text and Time Blocks

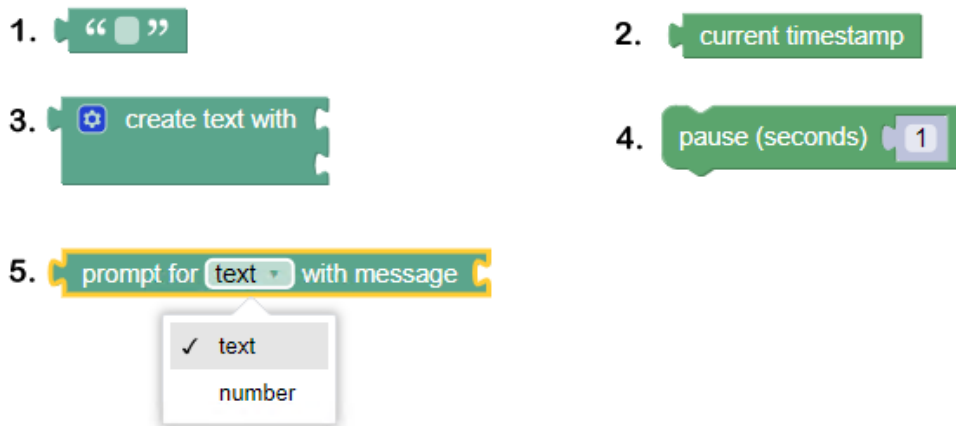


Figure 12. Flexdata Text and Time Blocks

3.10. Flexdata Algorithms — Example of Round, Math, and Variable Blocks

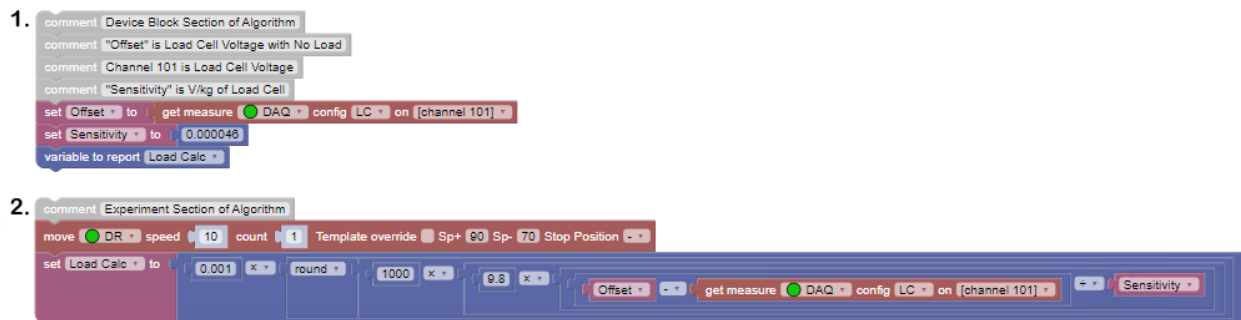


Figure 13. Rounding Example with Math and Variable Blocks

Figure 13 is an example of a rounding operation with multiple math operators and variables. This example is to calculate the Load in Newtons on a Load Cell.

The Load equation is: $Load = 9.8 \cdot (Offset\ Voltage - Measured\ Voltage) / Sensitivity$

Section "1" of Figure 13 would appear at the beginning of an algorithm, it is to define the "Offset" voltage and "Sensitivity" of the Load Cell. The "Offset" voltage is the measured voltage of the Load Cell when no load is being applied. The "Sensitivity" of the Load Cell would have been determined prior to the experiment by applying a 1 kg load to the Load Cell and measuring the voltage difference from the "Offset" voltage. Section "2" of Figure 13 would appear inside the experiment portion of the algorithm. In this example, the Motor would be moved from the no load position to some loaded position. Then the Load Cell voltage would be measured using the "get measure" block in the final command of Figure 13 and that voltage would be subtracted from the "Offset" voltage. That difference would be divided by the "Sensitivity" shown at the end of the block. That would give the Load in units of kilogram force. That value would be multiplied by "9.8" to convert the load from kilogram force to Newtons. Because rounding is to the nearest whole number, that value would be rounded to two decimals by first multiplying by "1,000", then rounded to the nearest whole number, then divided by "1,000" (or multiplied by "0.001"). One might expect to use "100" to round to two decimals, but in this case the units of "Sensitivity" were volts per kilogram force instead of Newtons, a difference of a factor of 10.

3.11. Flexdata Algorithms — Continuous Recording Blocks



Figure 14. Continuous Recording Blocks

Flexdata can make continuous measurements (actually digital values periodically acquired from sampling the signal). When continuous measurement is being used, the normal "experiment record" meter measurement is suspended.

3.12. Flexdata Algorithms — ConnectBox Blocks

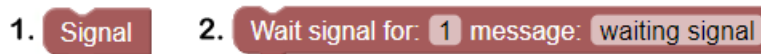


Figure 15. Flexdata ConnectBox Blocks

1. "Signal" — This block is used by ConnectBox to start external measurements. The ConnectBox green LED will turn ON and the ConnectBox Terminal #1 will go HIGH.

2. "Wait signal" — This block will cause Flexdata to wait until the signal "external measurements complete" is returned to ConnectBox Terminal #2. The signal must be a LOW voltage returned from external software, or by pressing the ConnectBox Push Button if the Operator manually made the external measurements, or by timing out based on the time value in seconds in the first window. The "complete signal" must remain LOW until the "Signal" is no longer HIGH. The second window can be used for a message to be displayed when the "Signal" begins.

3.13. Flexdata Algorithms — Action Replay Block



Figure 16. Flexdata Action Replay Block

"Action replay" is used to playback a sequence of snapshots like a movie. This block must be placed early in the algorithm, along with the device blocks. The first window specifies the camera that will be used. In the second window, the Operator can specify the mp4 file name to be used. The rate of playback can be specified in the "fps" window. The final block allows the Operator to set the color of the Action Replay overlay.

4. Flexdata Templates

Templates are not algorithms, but will be included in this Application Note because they are intricately linked to the algorithms.

4.1. Meter Templates

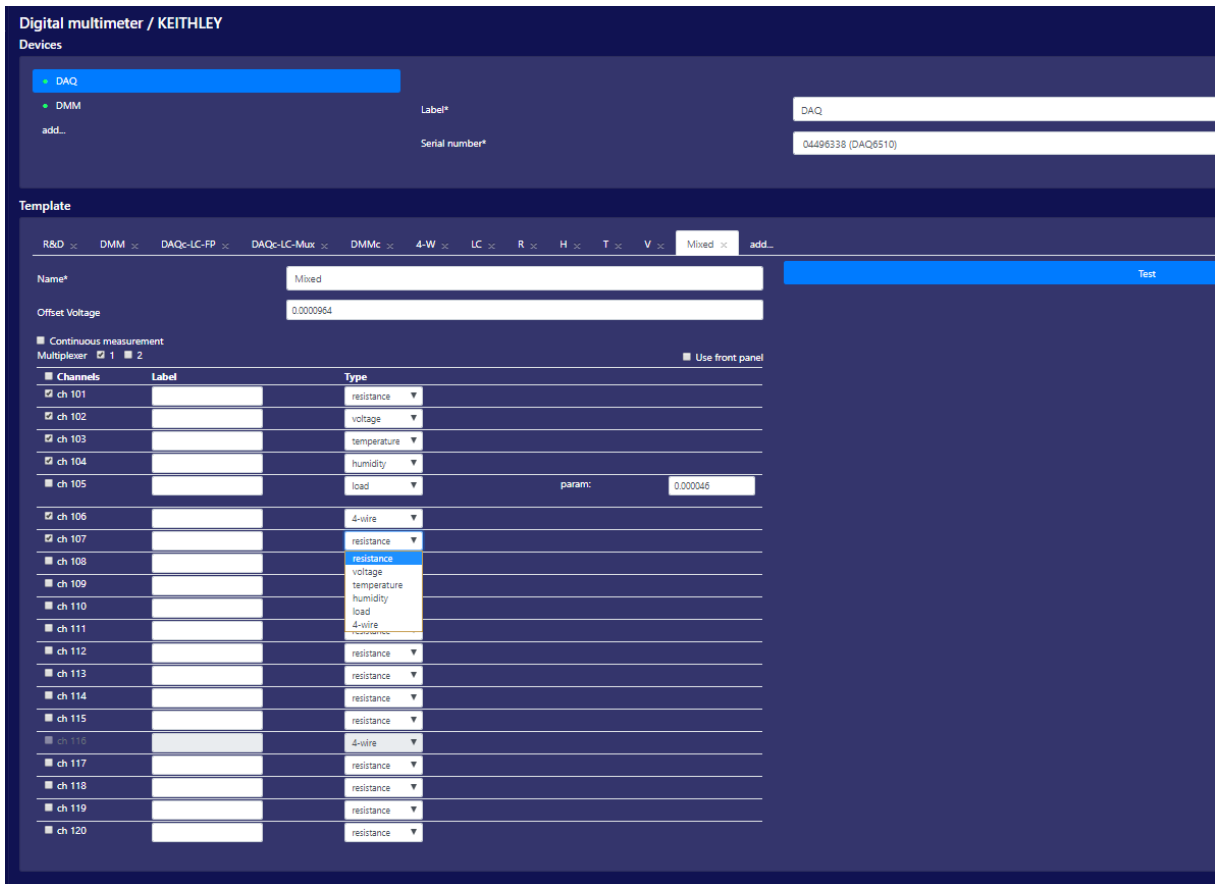


Figure 17. Template of Keithley DAQ6510 with 7710 Multiplexer Installed

Figure 17 is the Keithley DAQ template that is used in the algorithms. At the top left corner is the selection for the DAQ or DMM template. On the top right the Operator can give a name to the meter (Label*), the simple name "DAQ" is in use here. The default is the entire meter name. Below that is the means for selecting the specific meter with a drop down arrow that is not visible in this image, it is offscreen to the right. When clicking on that down arrow, all possible meters will be listed and the Operator selects the desired meter.

Below the meter selection is the list of Templates that have been defined by the Operator. Below the list are the characteristics for that specific Template. The Template in use is named "Mixed" because it has a mix of test parameters for illustration. Looking lower in the "Channels" column at the row "ch 107", and in the "Type" column, and by clicking on the down arrow, the complete list of test parameters is shown — a choice of "resistance", "voltage", "temperature", "humidity", "load", or "4-wire". Whichever parameter is selected will cause the DAQ to measure that type of parameter value. For purposes of illustration, ch 101 has been set to resistance (2-wire), ch 102 to voltage, ch 103 to temperature (type k thermocouple), ch 104 to humidity (using a Honeywell sensor and an equation to convert the voltage that is actually being measured to humidity in percent), ch 105 to load (using a Unipulse sensor and an equation to convert the voltage that is actually being measured to load in Newtons), and ch 105 to 4-wire resistance (which also uses ch 116 which is greyed out because it is in use). Only the channels with a check will be used in the Mixed Template.

Below the Template Name is a window that currently shows "0.0000964". This window is used to hold the offset voltage of the load cell which is on ch 105. To get that value, the selection parameter of ch 105 is changed to "voltage", then the Operator clicks on "Test" in the right portion of the Template. This will cause Flexdata to use the DAQ to measure the voltage on ch 105 and display that value under "Test". The Operator then copies that value into the window for the offset voltage. Then the ch 105 selection is changed back to "load" before the experiment is run. To the right of the "load" selection is a window with the title "param:". This is to be set to the Sensitivity of the specific Load Cell (the difference voltage, measured prior to the setup of the experiment, obtained by placing a 1 kg weight on the sensor, measuring the voltage, and subtracting from that the no-load voltage). That window currently shows "0.000046" which means that a 1 kg weight caused the voltage to change by 0.000046V or 46 μ V when the Sensitivity was measured. After applying the offset voltage value and the Sensitivity value, the Operator could then change the motor position to place a load on the sample, and then click on "Test" again and that value would be the load that is being applied to the sample at that moment.

Other choices on the DAQ Template page are shown just above the Channels. By placing a check in the "Continuous measurement" box, the DAQ will be set to make continuous measurements instead of measurements only during an "experiment record". Placing a check in Multiplexer 1 will use the top 7710 multiplexer in the DAQ. Placing a check in Multiplexer 2 will use the lower 7710 multiplexer. Both can be used in which case both would have a check. To the right is another check box "Use front panel". This will cause Flexdata to look only at the front panel when making measurements. Note that there is a push button on the front panel of the DAQ that must also be pressed manually to change between front panel measurements and multiplexer measurements.

Just above the list of channels is a box "Channels". That is an easy way to select "all channels or "no channels". For example, if the Operator is going to use only a small number of channels, it would be easiest to click in the "Channels" box until none of the channels are selected and then click on the specific channels that are desired. Or if most channels are going to be used, click on the "Channels" box until all channels are selected and then click on the channels that will not be used.

One other choice the Operator has is to type in Labels to identify the channels which will be shown on the Flexdata Table during an experiment.

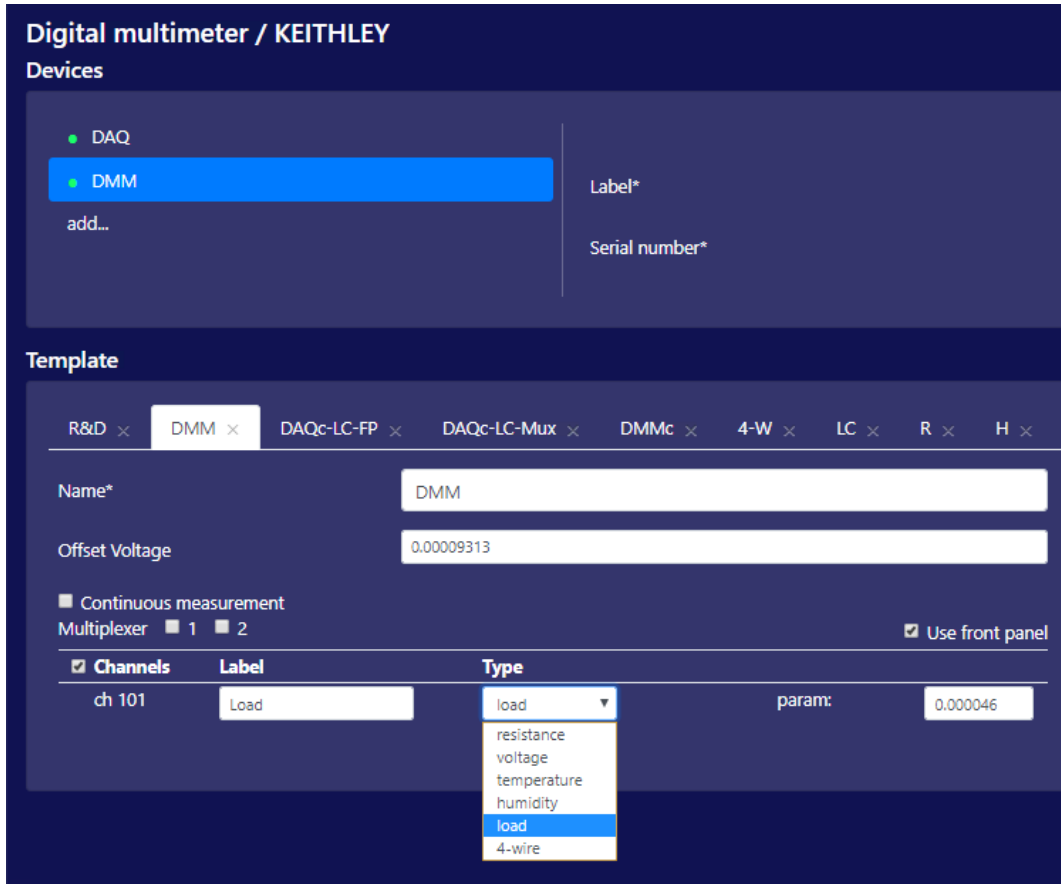


Figure 18. Template of Keithley DMM6500

Figure 18 is the DMM Template which is much simpler than the DAQ Template because DMM multiplexers are not used in Flexdata. The same Templates appear on the Template choice line. The Load windows for offset voltage and sensitivity will appear. Only "front panel" can be selected, there are no choices for multiplexers or channel numbers, only ch 101 can be used. "Continuous measurement" can be selected or not.

Figure 19 shows the top right corner of the meter template page. There is a drop down box to "Reconnect" the meter or to "delete" the meter. Just below the selections is the drop down arrow to select which connected meter is to be used.



Figure 19. Template of Keithley DAQ Top Right Corner

4.2. Motor Templates

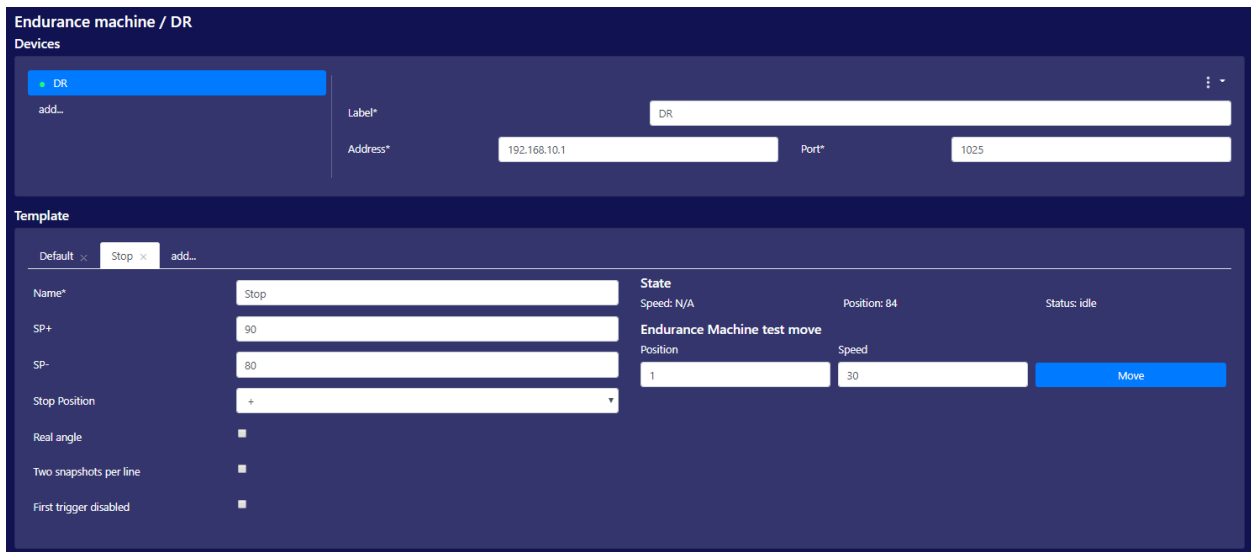


Figure 20. Template of DR Motor

As was the case with meter templates, motor selection is done in the top right portion of the template. Selection of motors is done by using the IP address of the motor. The DR is mandated to have an IP address of 192.168.10.1. A typical Laptop address would be 192.168.10.10. The Port for the DR is 1025.

There are not many items to be selected for a template. Choices are in the bottom right quadrant. First is "Sp+". The DR is setup with degrees, and 90° is common for jigs. Next is "Sp-". The setup in this template was using a DR Linear Converter and the jig was a stretching jig, so the angle change was not great, it represents about 3 mm of stretching. Next is "Stop Position" and it can be "+" or "0" or "-". The choice is selected from the pull down arrow on the right of the window. This is to set the DR to stop at "Sp+" or "Sp-" or halfway between.

There are check boxes for "Real angel" and "Two snapshots per line" and "First trigger disabled".

The bottom right quadrant has motor information of Speed and Position and Status. The motor can be tested by clicking on "Move". The motor will move one cycle using the speed set in the window.

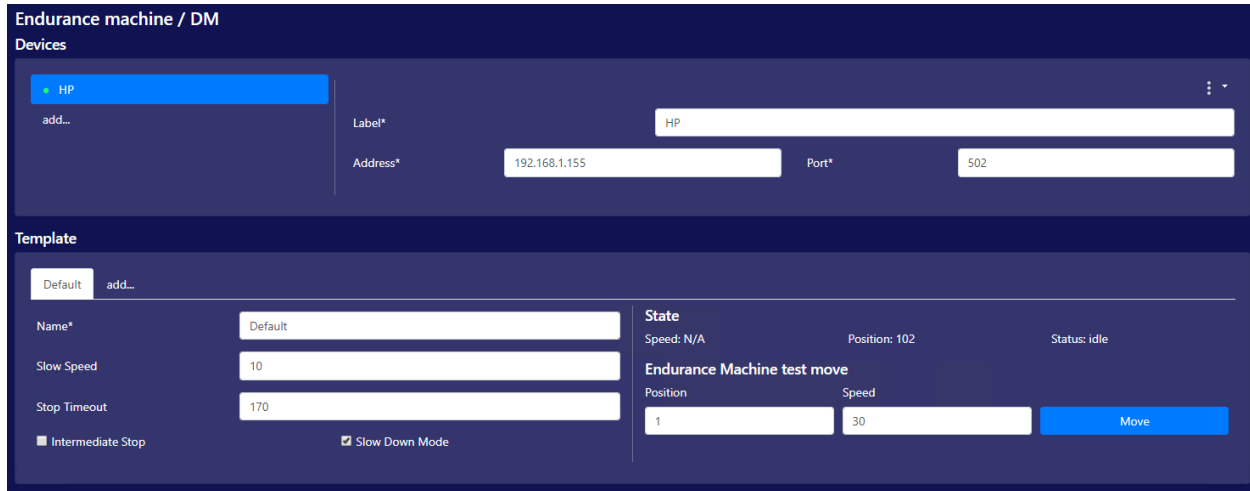


Figure 21. Template of HP Motor

The HP Template is simpler than the DR Template. Motor selection is the same. The HP does allow the Operator to change the IP address. Type the desired IP address into the window and select Port 502, then Connect to the motor.

The template choices in the bottom left are not for the DMLHP but for the DMLHB or DLDM111LH or TCDM111LH. Those motors are analog, and these parameters allow some control over the stopping position of the motor. Slowing the speed to about 10 rpm causes the motor to stop almost the moment the command is given, rather than coasting. "Stop Timeout" is used to continue power to the motor for the number of milliseconds in the window, and this can be used to cause the motor to stop at the other end of the cycle. "Intermediate Stop" will cause the motor to stop completely, then slowly move to the position set by the "Stop Timeout". "Slow Down Mode" causes the motor to run at the "Slow Speed" on the final cycle.

The bottom right corner has the same information as the DR including the possibility of causing the motor to move one cycle at the designated speed.

4.3. Camera Templates

Camera Templates are typically left on default because there is nothing to control in the camera. But there are exceptions. Normally the Dino-Lite template is left on default. Some cameras may have adjustments for lighting, scanning, and resolution.

As with the meters, selection of installed cameras is made at the top of the page. Any adjustable parameters will be in the bottom left quadrant. One difference with the camera templates is that

the camera video will be displayed in the bottom right quadrant. This is not a live camera view, click on "Apply template" to get a snapshot.

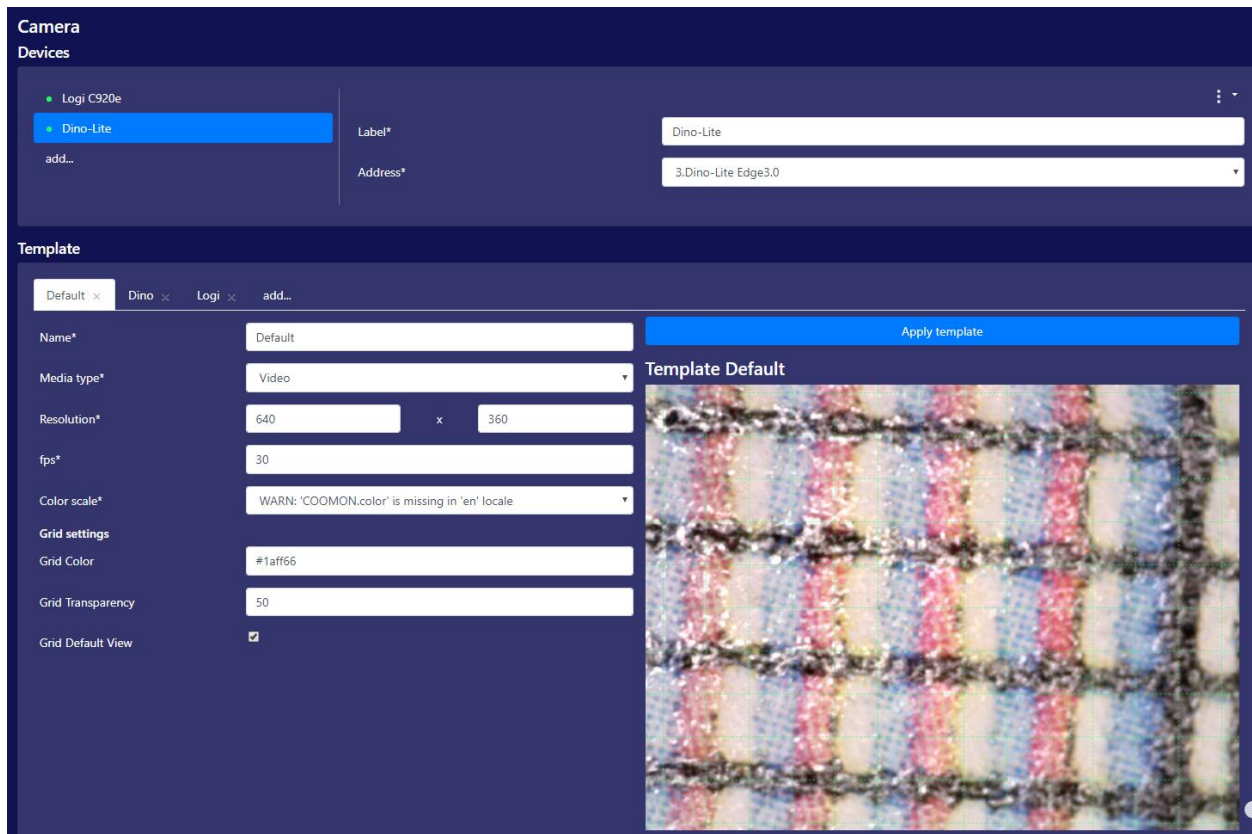


Figure 22. Template of Dino-Lite Camera

Flexdata DR Motor Move Block

The FD2 Motor Move Block can control more functions of the DR11MR than it can control the DLMHP, this is because the DR has more remote commands. This Application Note will explain how to use the Motor Move Block with the DR to get test results that are not possible with the HP. Figure 1 is the basic "move" block.

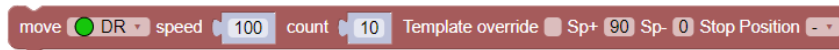


Figure 1. The FD2 Motor Move Block

The first window is for motor selection. This could be any of the motors that are compatible with Flexdata.

The second window is to set the motor speed. In the case of the DR this will be in degrees per second. In the case of the HP, it will be degrees per second if the HP is in the rotary drive mode, and it will be millimeters per second if the HP is in the linear drive mode.

The third window is to set the number of moves or reciprocations the motor will make. In the basic block, as shown here, that would mean that the motor would go back and forth between Sp+ and Sp- for 10 reciprocations.

The remaining windows are only for the DR.

Sp+ and Sp- set the end positions of the motor during operation. For the values shown, Sp+=90° and Sp-=0°, the motor would go back and forth between 90° and 0°.

"Stop Position" tells the motor at which end of the cycle the stop should be made. If Stop Position is "+" the stop will be at Sp+. If Stop Position is "-" the stop will be at Sp-. If Stop Position is "0" the stop will be halfway between Sp+ and Sp-.

The "Template override" check box, if checked, will cause the Sp+ angle and Sp- angle and Stop Position specified in the motor's template to be used. If not checked, the values in the windows in the "move" block will be used. If using "Template override" the Operator should confirm that the template values are the values desired for the experiment, and they may be different from the values in the "Sp+" and "Sp-" and "Stop Position" windows.

The first three windows will work any motor, but only the DR can have the other values set by Flexdata. With the DMLHP the other values must be set for the entire experiment using the HP Touch Panel. For the DMLHB the values for Sp+ and Sp- are set manually through a port on the motor drive unit, and the DMLHB cannot change the Stop Position, it always stops in the same position.

Figure 2 shows an algorithm that can be used with the DR to make measurements at both ends of the reciprocation cycle. This is not possible with the HP without constant manual Operator intervention using the Touch Panel. This technique of measuring at both ends of the

reciprocation cycle can be valuable when the data is expected to be different under the two different conditions, and the data is desired from both ends.

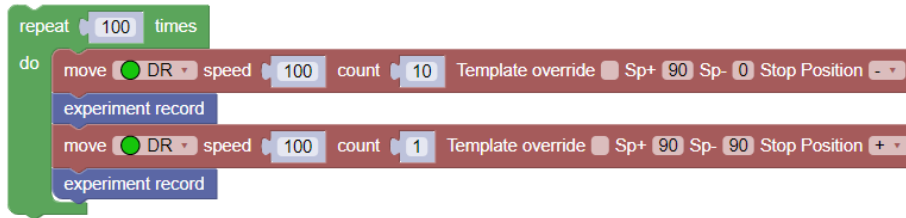


Figure 2. Blocks used to measure at both Sp+ and Sp- with the DR.

There are two motor move blocks used in Figure 2. The first is for the multiple cycles of the test. In the example shown, the DR would move back and forth between 0° and 90° for 10 complete cycles and then stop at the end specified by the Stop Position, in this case at Sp- because Stop Position is set for "-". Then the measurements would be taken. Then the DR would make one move from 0° to 90°. Both Sp+ and Sp- are specified to be 90° and Stop Position is set to "+" because the motion will be going in the Sp+ direction. If a "-" or "0" were selected, the motor might stop at 89.9° instead of stopping at 90°. Then those steps would happen again and again until the 100 iterations specified in the "repeat" loop are reached.

The technique used in Figure 2 that causes the DR to stop a second time in the reciprocation cycle can be used to make multiple stops for measurements in the reciprocation cycle.

The example in Figure 3 would stop and make 7 measurements every measurement cycle. In this case, the test would run for 100 cycles then stop to make measurements at 0° and then at 30° and at 60° and at 90° and then reverse and make measurements at 60° and at 30° and again at 0°. The stop positions do not need to be equal distances. This test would be 999 more iterations of this 100 cycle run with multiple measurements. In each of the "move" blocks, the Stop Position should be set to "+" if the motor is moving in the Sp+ direction and "-" if the motor is moving in the Sp- direction.



Figure 3. Multiple Stop Measurements with the DR

Flexdata Variables

Define New Parameters from Calculations on Measured Values

The number of parameters that can be measured with Flexdata is limited by the test meter and by the test meter parameters that are recognized by Flexdata. For example, the Keithley DAQ6510 can measure multiple types of parameters, but only resistance, voltage, and temperature are recognized by Flexdata. The latest version of Flexdata has broadened that with Humidity and Load. Those two measurements are actually voltage measurements, but Flexdata applies equations to convert those voltages to Humidity or Load. The latest version of Flexdata also allows the Operator of Yuasa Test Machines to create new Variables that can be calculated from resistance or voltage or temperature data. The test algorithm will make the appropriate measurements and convert those values to the new parameter as defined by the Variable.

Variable Blocks

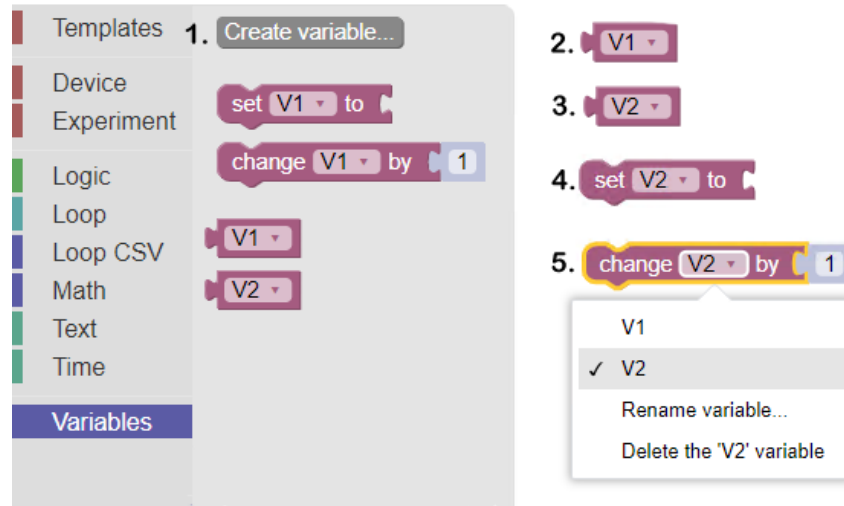


Figure 1. Flexdata Variable Blocks

"Variables" — Flexdata allows the Operator to define Variables based on measured values from meters and values set by other blocks or combinations of blocks.

1. "Create variable" — Clicking on "Create variable" opens a window to name the variable.
2. "V1" — A variable created and named "V1".
3. "V2" — A variable created and named "V2".
4. "set" — This block is used to define the variable. It could be the value obtained by one of the meters and could be voltage, resistance, or any of the measured parameters used in Flexdata. In that case the block used on the right side of "set" would be the meter block "get measure".
5. "change" — This block is similar to the "set" block, it simply allows a change in the value of the variable. The variable will be changed by the value in the numerical window, and in the case of the value shown in the Figure, incremented by "1".

Both the "Humidity" and "Load" parameters used in meter templates in Flexdata are internal variables based on a measured voltage. In this Application Note those parameters will be examined in detail. Both "Humidity" and "Load" are based on the equation

$$y = M * x + B$$

where M is the slope of the linear equation and B is the vertical axis offset. For example, if $x=0$, then $y=B$. The vertical axis crossing of the equation will be where $y=B$. If $y=0$, then $x=-B/M$. The horizontal axis crossing of the equation will be $x=-B/M$.

The Honeywell sensor HIH-400 is used by Flexdata for Humidity. The equation that calculates the humidity is:

$$RH\% = 32.26 * V_{out} - 25.8$$

where RH% is the relative humidity in percent, V_{out} is the voltage output of the HIH-400, and 25.8 is a constant. These values can be found in the HIH-400 datasheet.

Using Variables, the following set of blocks would calculate humidity from the Honeywell sensor. The meter "device" block must be used at the top of the algorithm. The Variable would be "RH%". The "set RH%" block defines the Variable "RH%" and it is according to the equation. To have RH% as a column in the Flexdata Table, the block "variable to report" would be required for RH%. This block must be in the "device" portion of the algorithm. Then during the test, the block "set RH%" would be required to calculate RH%, and the block "get measure" would be required for the meter to make a measurement to get the HIH-400 voltage.

Four "math" blocks would be used, the first to represent "M" in the equation with value of 32.26, the second to represent "B" in the equation with value of 25.8, the third to have the multiplication by 32.26, and the fourth to subtract 25.8. Figure 2 is the resulting set of blocks that would appear in the algorithm. Comment blocks are used in Figure 2 to represent locations where other blocks would likely appear.

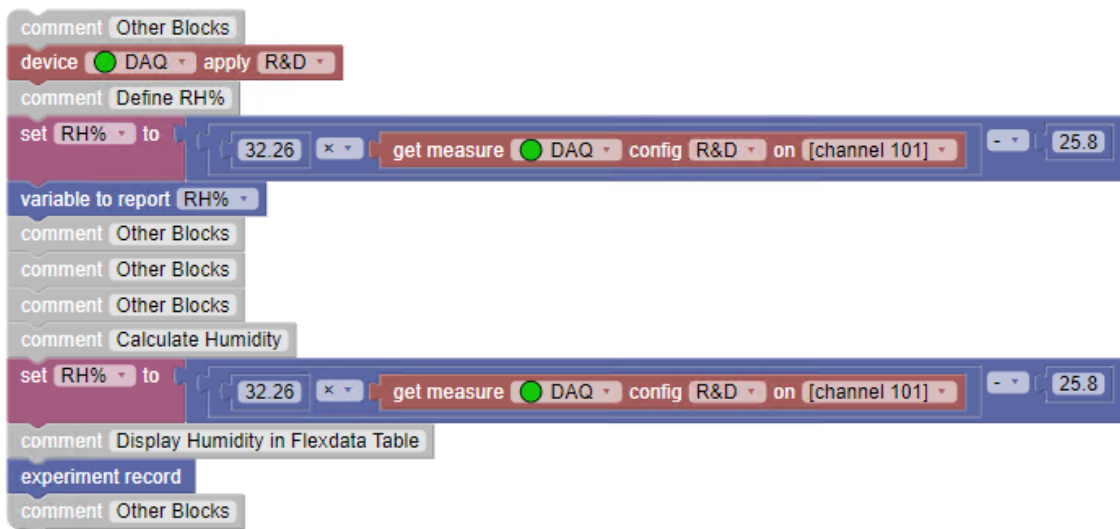


Figure 2. Using a Variable to Report RH% in Flexdata Table

The Unipulse UNCLB Tension / Compression load cell is used by Flexdata for Load. Load is exactly the same form of equation:

$$Load = 9.8 * (V_{offset} - V_{out}) / 0.000046$$

where 9.8 is the factor to convert kilogram-force (kgf) to Newtons (N). The value 0.000046 is the "Sensitivity" of the specific Load Cell used and is the voltage change of the Load Cell output when the load is changed by 1 kilogram (kg). How to get this value is covered in the "Load Cell

Application Note". Voffset is the UNCLB output voltage when no load is applied. This needs to be measured in each test setup.

Using Variables, the following set of blocks would calculate Load from the Unipulse sensor. The meter "device" block must be used at the top of the algorithm. The Variable would be "Load". The "set Load" block defines the Variable "Load" and it is according to the equation. To have Load as a column in the Flexdata Table, the block "variable to report" would be required for Load. This block must be in the "device" portion of the algorithm. Then during the test, the block "set Load" would be required to calculate Load, and the block "get measure" would be required for the meter to make a measurement to get the UNCLB voltage.

Four "math" blocks would be used, the first to represent the factor "9.8" in the equation, the second to represent "0.000046" in the equation, the third to multiply by 9.8, and the fourth to subtract the measured voltage from the offset voltage. Figure 3 is the resulting set of blocks that would appear in the algorithm. Comment blocks are used in Figure 3 to represent locations where other blocks would likely appear.



Figure 3. Using a Variable to Report Load in Flexdata Table

One more example will be given. In the Flexdata Table, temperature is reported as degrees Celsius. Suppose the value in Fahrenheit is required. It can be obtained from the same type of equation:

$$F = C * 5 / 9 + 32$$

The block in Figure 4 would give that value.

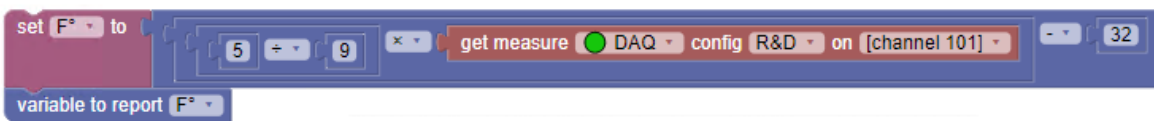


Figure 4. Using a Variable to Change Celsius to Fahrenheit

Flexdata Load Cell

Measure Tension or Compression

Yuasa Test Jigs like the ST or FS-N can be outfitted with a Unipulse Tension/Compression Sensor which can be used for Tension and/or Compression. The Fixed Clamp on the Jig is replaced with a modified Load Cell Clamp that includes the Unipulse Sensor. The Load Cell Clamp can be fitted with a Sample Clamp for tension, or a Sample Pusher for compression.

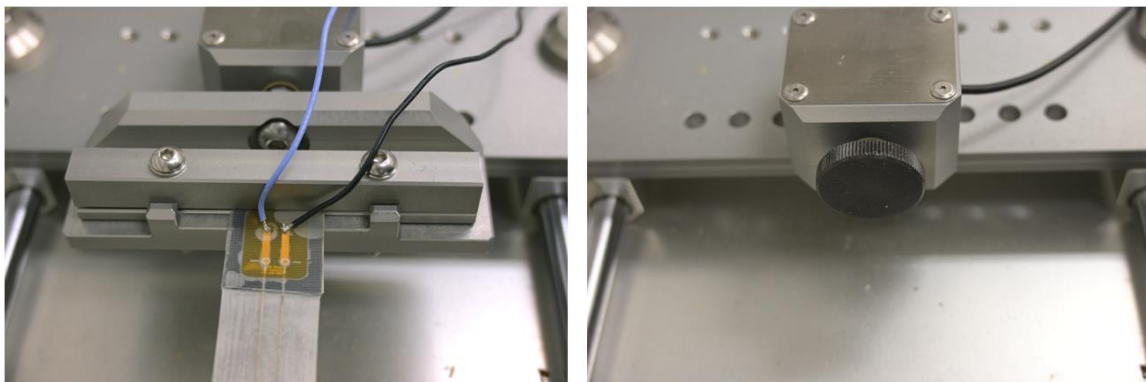


Figure 1. Load Cell Clamp, Sample Clamp on Left, Sample Pusher on Right

Example 1 — Stretching with the Load Cell Clamp

A Yuasa Test Machine consisting of the DR4 Motor plus a Linear Converter Attachment driving the FS-N Jig can be used to stretch samples. A test using the Sample Clamp on the Load Cell Clamp will demonstrate an algorithm that measures tension at both the no-load position and the maximum load position. A similar algorithm can be used with the DMLHP Motor, but only one end of the cycle can be measured when using the DMLHP. Normally this would be the maximum load position.

Example 2 — Compression with the Load Cell Clamp

A Yuasa Test Machine consisting of the DR4 Motor plus a Linear Converter Attachment driving the FS-N Jig can be used to compress samples. A test using the Sample Pusher on the Load Cell Clamp will demonstrate an algorithm that measures tension at both the no-load position and the maximum load position. A similar algorithm can be used with the DMLHP Motor, but only one end of the cycle can be measured when using the DMLHP. Normally this would be the maximum load position.

Example 1 — Stretching with the Load Cell Clamp

```

comment DR Stretch with Load Cell
comment Set Load Calibration Value on Template
comment Set Jig Position to No-Load Location
comment Set +SP and -SP and Stop Position
comment Sample is 115 mm, stretch 10%
comment DR / Linear Converter 1 mm = 3.183°
comment No-Load=0° Load = 11.5 * 3.183 = -36.6° Stop="+"
comment Tension is Positive, Compression is Negative
comment Record Sample Load and 4-wire Resistance
device DR apply Default
device DAQ apply DAQc-LC-Mux
Load calibration DAQ config DAQc-LC-Mux on Load
comment Get No-Load Voltage
experiment record
comment Get Max Load
move DR speed 100 count 1 Template override Sp+ -36.6 Sp- -36.6 Stop Position -
experiment record
repeat 5 times
do
move DR speed 100 count 3 Template override Sp+ 0 Sp- -36.6 Stop Position -
comment Get Min Load
experiment record
comment Get Max Load
move DR speed 100 count 1 Template override Sp+ -36.6 Sp- -36.6 Stop Position +
experiment record
comment End at No-Load Position
move DR speed 100 count 1 Template override Sp+ 0 Sp- 0 Stop Position -
comment Get Final No-Load Value
experiment record

```

Figure 2. Stretching Algorithm

The Algorithm in Figure 2 is more complicated than a Stretching Algorithm has to be, but several points will be worthy of inclusion in the example. The DR11MR with Linear Converter is being used because it can stop at both ends of a stretching cycle and this allows measurement of Load under both no-load and maximum load conditions. The actual test will be to make an initial measurement at the no-load position, move to the maximum load position to make an initial measurement of maximum load, then run some number of cycles and then stop to measure no-load and then measure maximum load, then continue to run more cycles. When all cycles have been run, the sample will be returned to the no-load position and a final measurement of

load will be made. The sample has a resistance that can be measured, so this will be done also at every measurement point. Because the resistance is low, a 4-wire resistance measurement will be made each time. Numerous comments are given throughout the algorithm to explain the steps that are happening.

In addition, several comments are given before the testing portion of the algorithm begins to explain the test. The first comment is the algorithm name. The second comment, "Set Load Calibration Value on Template", is a reminder that each Load Cell can have a different Sensitivity, and the value for that specific Load Cell needs to be entered on the meter template. In this Application Note, Sensitivity of the Load Cell is defined as the Load Cell's output voltage change when a 1 kg load change is made. Typically the sensitivity will be obtained by hanging the Load Cell, as shown on the left side of Figure 3, measuring the output voltage, then hanging a 1 kg weight on the Load Cell, and then the voltage change will be the Sensitivity. Or, Sensitivity can be obtained by placing the Load Cell on the benchtop, as shown on the right side of Figure 3, measuring the output voltage, then placing a 1 kg weight on the Load Cell, and then the voltage change will be the Sensitivity. With low cost Load Cells, those two values might not be the same. With the Load Cell selected for the Yuasa Test Machines, the two values will be the same.

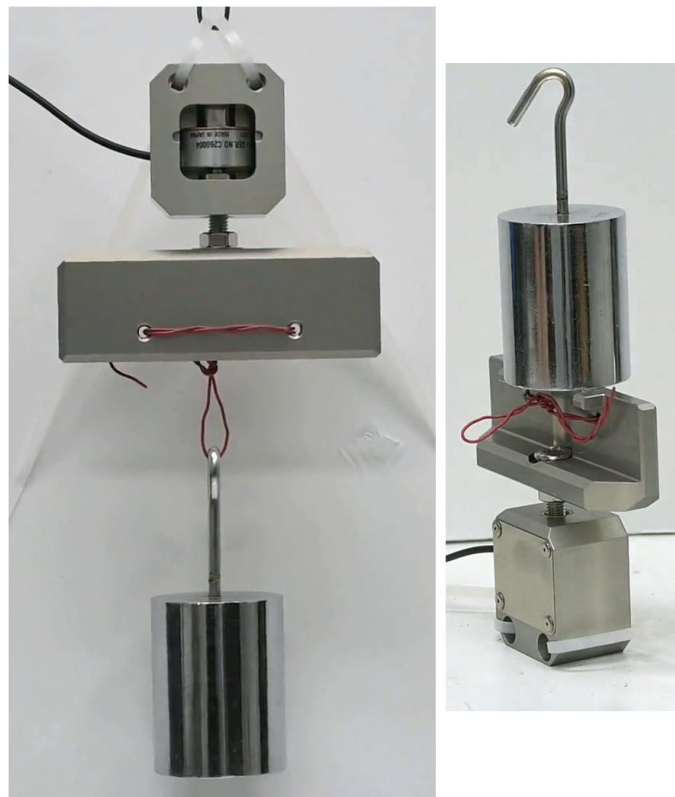


Figure 3. Sensitivity Test by Hanging on Left and on Benchtop on Right

The Load Cell that was used in this test had a sensitivity of $46\mu\text{V}$ per kilogram. This is the value that must be placed in the meter template. That value, "0.000046", is entered in the window on

the right side of the red arrow in Figure 4 for the channel used for measuring the load. That value must be determined prior to the Load Cell being mounted on the stretching Jig.

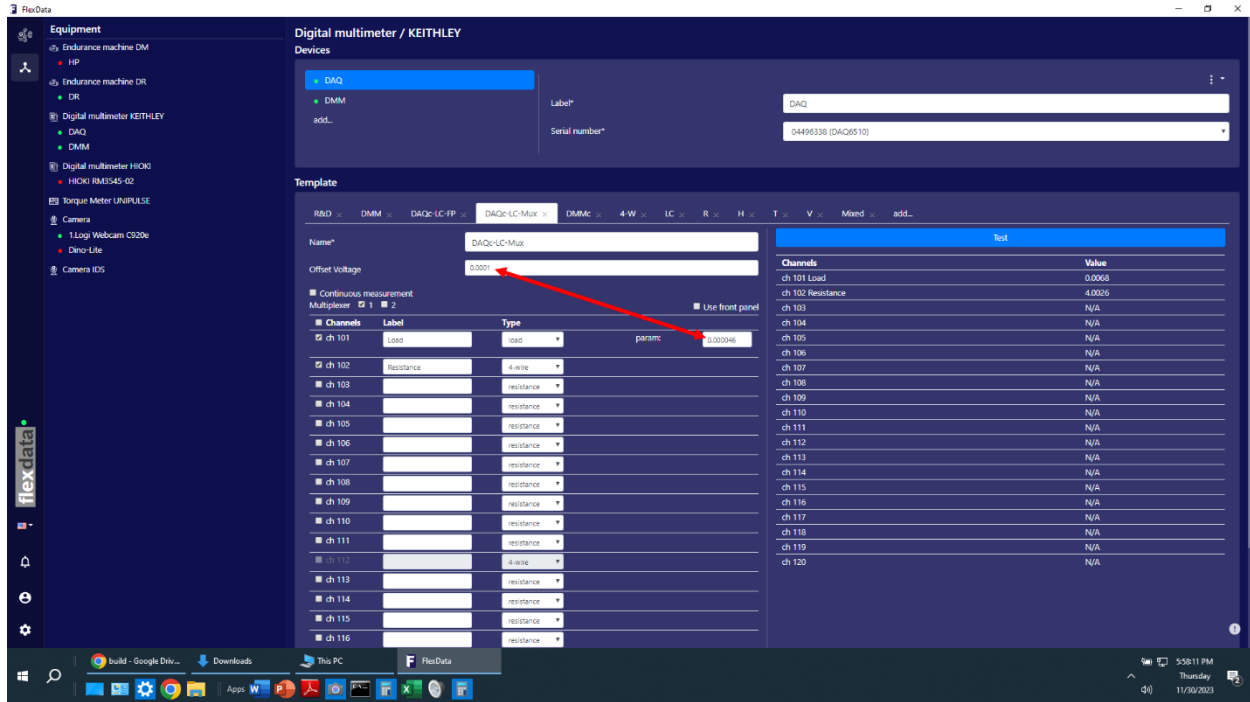


Figure 4. Sensitivity and Offset Settings in Meter Template

The window at the left side end of the red arrow is for entering the offset voltage, the voltage measured when no load is applied. This value must be determined when the sample is mounted on the stretching jig just before beginning the test. This value will not be used during the test, it is only for the convenience of the Operator. To determine this value, the parameter for the Load channel is set to voltage, then the Operator clicks on "Test" in the blue bar on the right side of the template. This will be the offset voltage if no load is being applied to the sample already mounted on the stretching jig. That offset voltage is entered into the window at the left side of the red arrow. Then the Operator must change the parameter back to Load. If then Operator clicks on "Test" again, it will display the Load value. For example, the Operator may wish to Jog the motor slightly to stretch the sample, then click on "Test" and the Operator will know how much Tension or Load is being applied to the sample at that moment. If the Operator did move the jig, then it should be reset to the initial position prior to starting the test.

The next two comments in the algorithm are reminders for the jig setup prior to beginning the test. They are setting the jig to the no load position, and putting the +SP and -SP and Stop Position values into the motor move blocks. The next comment "Sample is 115 mm, stretch 10%" are the setup for this specific test.

The comment "DR / Linear Converter 1 mm = 3.183° " is very specific to the use of the Linear Converter on the DR. The original design of the DR was a rotary-only motor. A kit has been designed to install onto the rotary drive shaft and use a rack and pinion device to allow a linear motion output. The +SP and -SP values must still be in degrees, so a conversion must be made

between degrees and millimeters. That value is 3.183° of rotary motion for every millimeter of linear motion.

The next comment "No-Load= 0° Load= $11.5*3.183=-36.6^\circ$ Stop="+ " " is giving the specific setup values for this specific test.

The next comment "Tension is Positive, Compression is Negative" is a reminder that the convention for tension and compression is that tension is always considered a positive load and compression is always considered a negative load. If test results are not consistent with this convention, some error has been made in the setup data entry.

The final comment before the test is a note that both the load on the sample and the 4-wire resistance of the conductor on the sample will be measured.

The next two blocks are "device" blocks which must be used to specify any and all equipment that will be used in the test. In this example, there are two, the DR motor and the DAQ meter.



Figure 5. Load Calibration Block

Figure 5 shows another block in the algorithm that is worthy of mention, the "Load calibration" block. The purpose of this block, which must be placed in the "device" location of the algorithm, is to cause FD2 to measure the offset voltage of the Load Cell. The sample must be in what is considered the "no load" position at this point in the test. That voltage will be subtracted from the measured Load Cell output voltage during the test to calculate the Load. That voltage difference will be divided by the Sensitivity that was placed on the meter template to determine load in kilogram force (kgf) and then multiplied by 9.8 to convert the load to Newtons. The Load given in the FD2 table will always be expressed in Newtons.

Example 2 — Compression with the Load Cell Clamp

```
comment DR Compress with Load Cell
comment Set Load Calibration Value on Template
comment Set Jig Position to No-Load Location
comment Set +SP and -SP and Stop Position
comment Sample is 67 mm, compress 10%
comment DR / Linear Converter 1 mm = 3.183°
comment No-Load=0° Load = 6.7 * 3.183 = 21.3° Stop="-"
comment Tension is Positive, Compression is Negative
comment Record Sample Load
device DR apply Default
device DAQ apply DAQc-LC-Mux
Load calibration DAQ config DAQc-LC-Mux on Load
comment Get No-Load Voltage
experiment record
comment Get Max Load
move DR speed 100 count 1 Template override Sp+ 21.3 Sp- 21.3 Stop Position +
experiment record
repeat 5 times
do
  move DR speed 100 count 3 Template override Sp+ 21.3 Sp- 0 Stop Position -
  comment Get Min Load
  experiment record
  comment Get Max Load
  move DR speed 100 count 1 Template override Sp+ 21.3 Sp- 21.3 Stop Position +
  experiment record
comment End at No-Load Position
move DR speed 100 count 1 Template override Sp+ 0 Sp- 0 Stop Position -
comment Get Final No-Load Value
experiment record
```

Figure 6. Compression Algorithm

The Compression Algorithm in Figure 6 is very similar to the Stretching Algorithm in Figure 2. Everything stated in the first example will be true in the second example except for the differences noted here.

The first difference is that Compression will be done by moving the jig sliders together to compress instead of moving the sliders apart to stretch. Note that the final move command on both Algorithms is to move the motor to the setting of $+SP = -SP = 0^\circ$. However, the maximum

load on stretching is with a negative angle (-36.6° or -6.7 mm) and the maximum load on compression is a positive angle ($+21.3^\circ$ or 11.5 mm).

The second difference is that Stop Position on stretching was "+" and on Compression the Stop Position is "-". That is to cause the first measuring position in the "repeat" loop to be at no-load.

The third difference is that the data for the sample is different in the two examples because the sample is different in the two examples.

The fourth difference is that only Load is being measured in the second example, the sample does not have any conductor for which the resistance can be measured.

Otherwise, the two Algorithms are the same.

Flexdata Continuous Measurements

Flexdata has been upgraded to offer Continuous Measurements. In the past, measurements controlled by Flexdata were always taken in a static state, while the motor was stopped at some point in the reciprocating cycle. The latest version of Flexdata can now capture about 50 sample points per second if the front panel of the meter is being used. Multiple channels can be tested in the continuous mode using a multiplexer, but the sample rate will decrease when the multiplexer is used for one channel, and it will decrease further as the number of channels is increased.

Example 1 — Using Front Panel on Two Meters — Measure Tension and 2-wire Resistance of Sample being Stretched

Example 2 — Using Front Panel on Two Meters — Measure Tension and 4-wire Resistance of Sample being Stretched

Example 3 — Using Multiplexer on Two Channels — Measure Tension and 2-wire Resistance of Sample being Stretched

Example 4 — Using Multiplexer on Two Channels — Measure Tension and 4-wire Resistance of Sample being Stretched

Test Setup

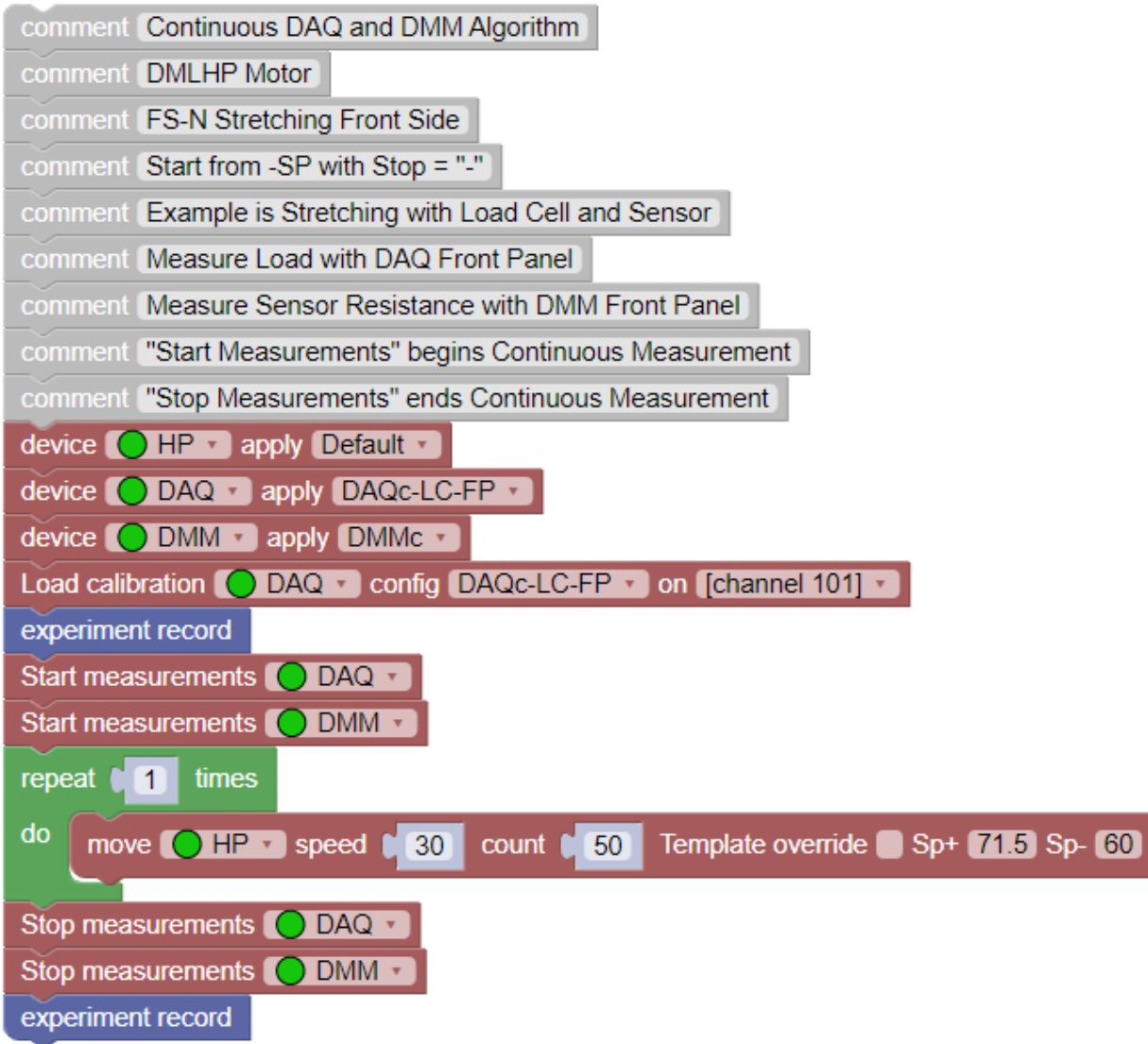


Figure 1. Algorithm for Continuous Measurement Application Note

Figure 1 is the Algorithm that will be used in this Application Note. Two meters will be used, the Keithley DAQ6510 with 7710 Multiplexer, and the Keithley DMM 6500 with no multiplexer. The DMLHP Motor Drive Unit will be used to show just how powerful Continuous Measurements can be. In standard Flexdata operation, the DMLHP can be stopped at only one point in the reciprocation cycle, at +SP or at -SP or halfway between. If making measurements like Load, the value will be changing throughout the reciprocation cycle. In standard Flexdata operation, typically the maximum load position would be selected. However, the profile of the Load and the minimum value of the Load would not be available from the test. With Continuous Measurements the DMLHP can yield the entire profile of the Load. And if a multiplexer or second meter is used, the profile of the resistance of the sample will be available also.

There are several blocks in the algorithm that are set specifically for this experiment. The blocks "Start measurements" and "Stop measurements" are used only for Continuous Measurements.

The two blocks tell the meters when to start measuring and when to stop measuring, there might be occasions where Continuous Measurements are used only in a portion of the test. This pair of blocks must be present for every meter that will be used for Continuous Measurements in the test.

As is the case in any test algorithm, the meters and motors that will be used in the test must be identified in the "device" blocks at the beginning. And the Templates that will be used must be specified. For example, when using the Front Panel, the meter's template must be set to "Front Panel" in the checkbox window. That is a different Template from one that would be used if the multiplexer will be used.

The block "Load calibration" must be used if Load is going to be measured. And the Sensitivity factor must be placed on the Template in the window for the channel that is used for "load".

The tests that will be run in the examples will always run for 50 stretch cycles. Graphs of the measurements will be shown for the entire test. Graphs will be shown also for two cycles in the middle of each test so the sampling rate can be seen.

Example 1 — Using Front Panel on Two Meters — Measure Tension and 2-wire Resistance of Sample being Stretched

The first example will have the highest sampling rates. Two meters will be used. The resistance measurement will be 2-wire which will yield faster sampling than 4-wire, although 4-wire can give more accurate measurements when the resistance is low. That will be seen in the results of the tests.

Figure 2 is presented for an overview. It can be seen that the Load and the Resistance follow the same curve throughout the test. The Load (green curve) varies between about -0.3 Newtons and about 4.1 Newtons. The Resistance (red curve) varies between about 4Ω and about 4.65Ω .

Figure 3 shows two cycles in the middle of the test. The coarseness of the sampling can be seen in this graph. In this specific test, the actual sampling rate is 43.7 samples per second for both the Load and the Resistance. The curves are plotted with a small diamond for the actual sampled value, and with straight lines connecting the samples. In this curve the noise of the Load Cell is visible. The noise is in the order of $1\mu V$.

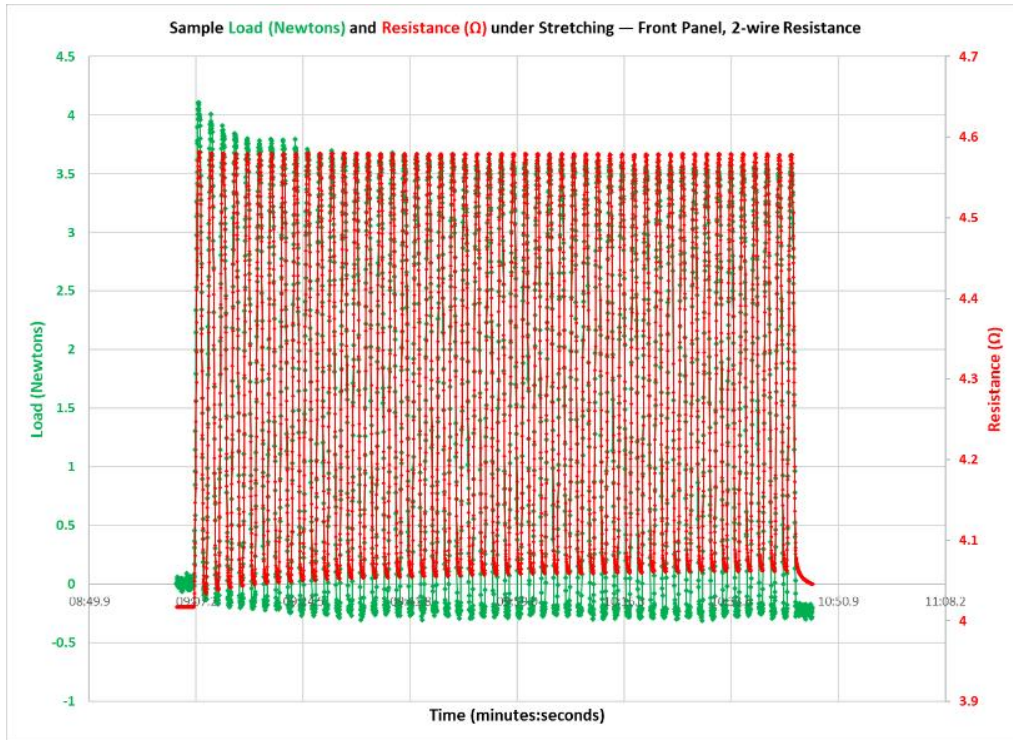


Figure 2. Continuous Measurements, the Full 50 Cycles of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 2-wire Resistance

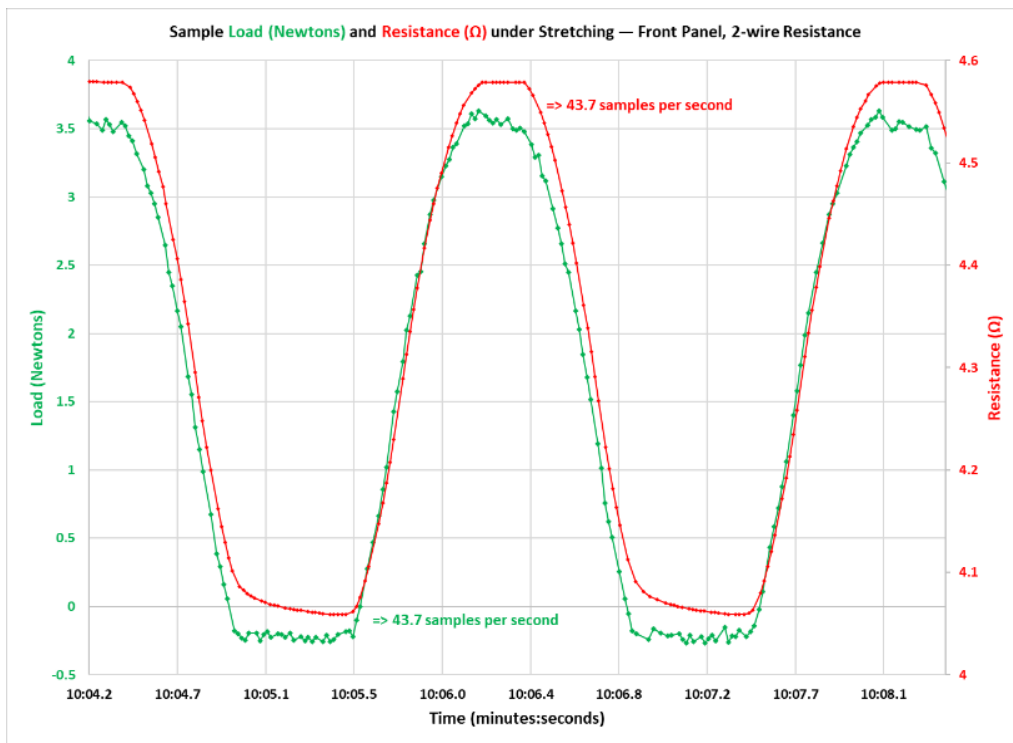


Figure 3. Continuous Measurements, Two Cycles in the Middle of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 2-wire Resistance

Example 2 — Using Front Panel on Two Meters — Measure Tension and 4-wire Resistance of Sample being Stretched

Figure 4 is presented for an overview. It can be seen that the Load and the Resistance follow the same curve throughout the test. The Load (green curve) varies between about 0 Newtons and about 3.9 Newtons. The Resistance (red curve) varies between about 3.9Ω and about 4.45Ω .

Figure 5 shows two cycles in the middle of the test. The coarseness of the sampling can be seen in this graph. In this specific test, the actual sampling rate is 43.7 samples per second for the Load and 13.0 samples per second for the Resistance. The use of 4-wire resistance measurements causes the sample rate to be lower because four connections are used instead of the normal two. The curves are plotted with a small diamond for the actual sampled value, and with straight lines connecting the samples. In this curve the noise of the Load Cell is visible. The noise is in the order of $1\mu V$.

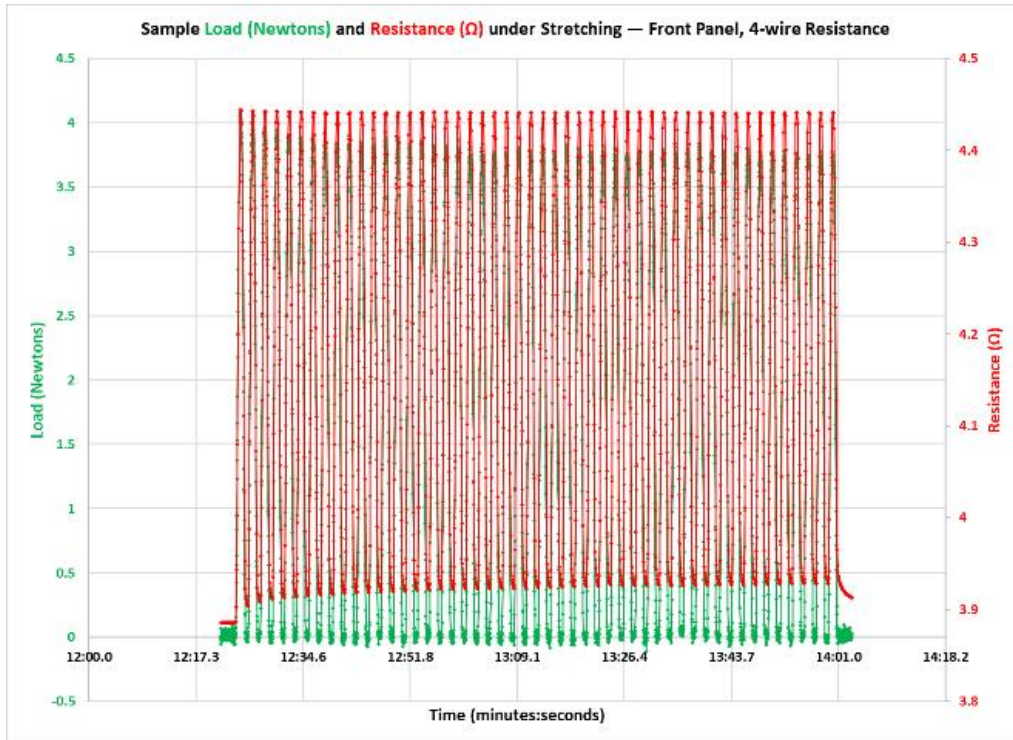


Figure 4. Continuous Measurements, the Full 50 Cycles of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 4-wire Resistance

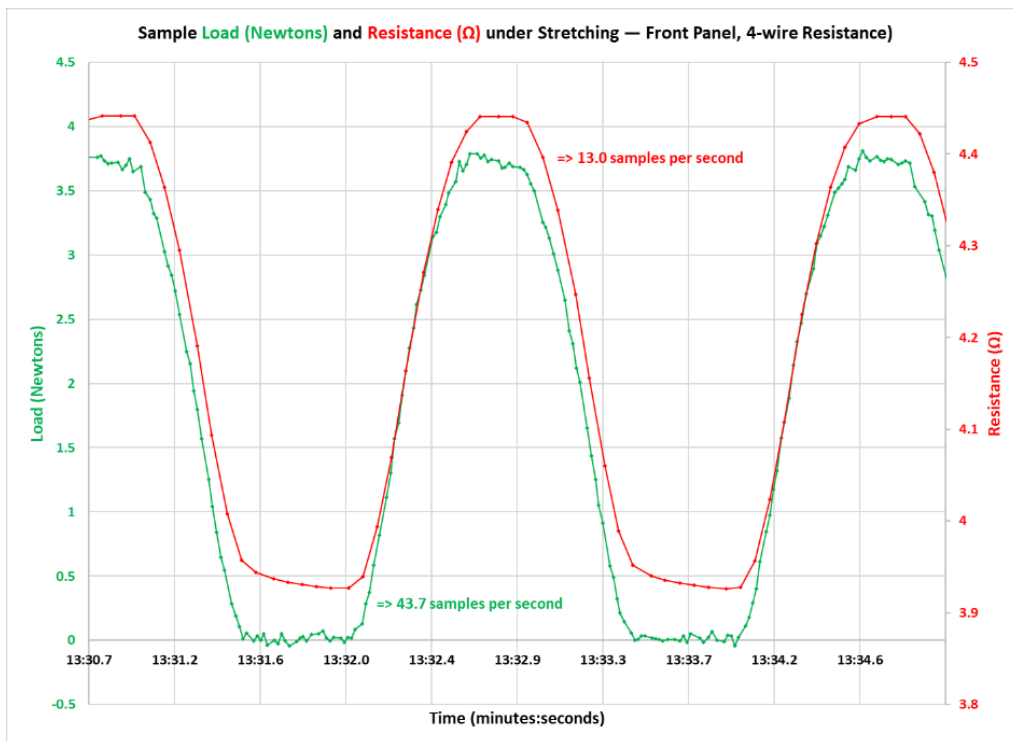


Figure 5. Continuous Measurements, Two Cycles in the Middle of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 4-wire Resistance

Example 3 — Using Multiplexer on Two Channels — Measure Tension and 2-wire Resistance of Sample being Stretched

Figure 6 is presented for an overview. It can be seen that the Load and the Resistance follow the same curve throughout the test. The Load (green curve) varies between about 0 Newtons and about 3.9 Newtons. The Resistance (red curve) varies between about 8.45Ω and about 9Ω .

This is an example where 2-wire resistance will be less accurate. With the Front Panel measurement, and using special probes, the 2-wire resistance and 4-wire resistance measurements were about the same. But when connecting to the multiplexer, normal wires were used, and the resistance of those wires affected the measured value.

Figure 7 shows two cycles in the middle of the test. The coarseness of the sampling can be seen in this graph. In this specific test, the actual sampling rate is 9.2 samples per second for both the Load and the Resistance. The curves are plotted with a small diamond for the actual sampled value, and with straight lines connecting the samples. In this curve the noise of the Load Cell is visible. The noise is in the order of $1\mu V$.

The sampling rate has decreased significantly, it was 43.7 samples per second with two meters, and now is 9.2 samples per second with one meter using two channels on a multiplexer.

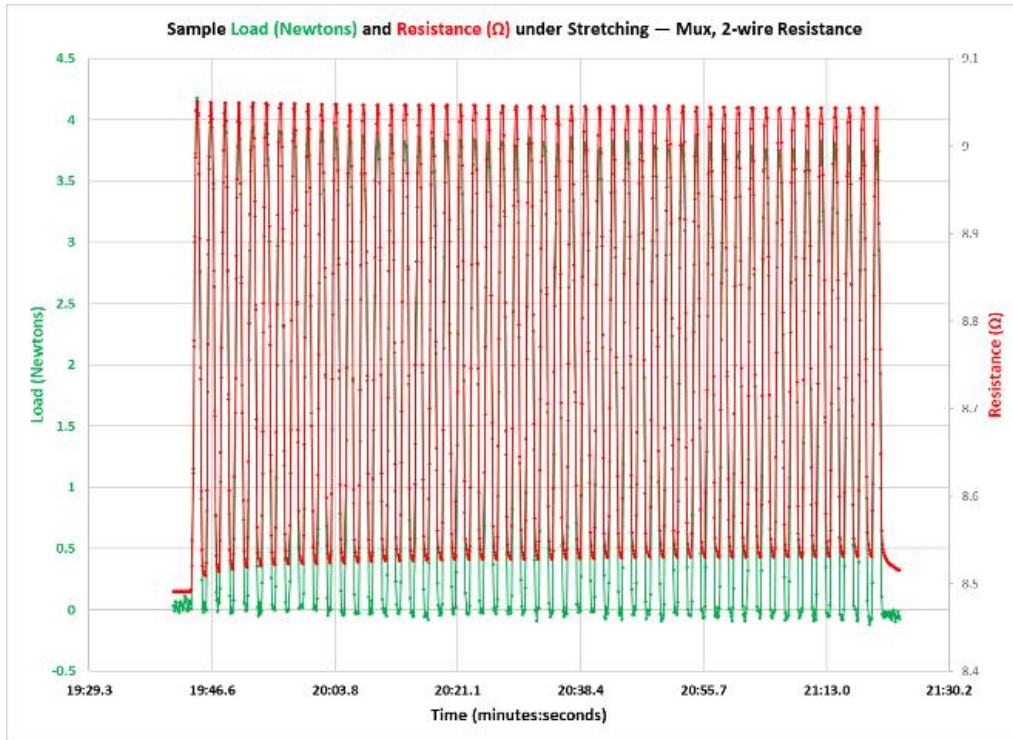


Figure 6. Continuous Measurements, the Full 50 Cycles of the Test are Shown, Using One Meter with a Multiplexer, Measuring Load and 2-wire Resistance

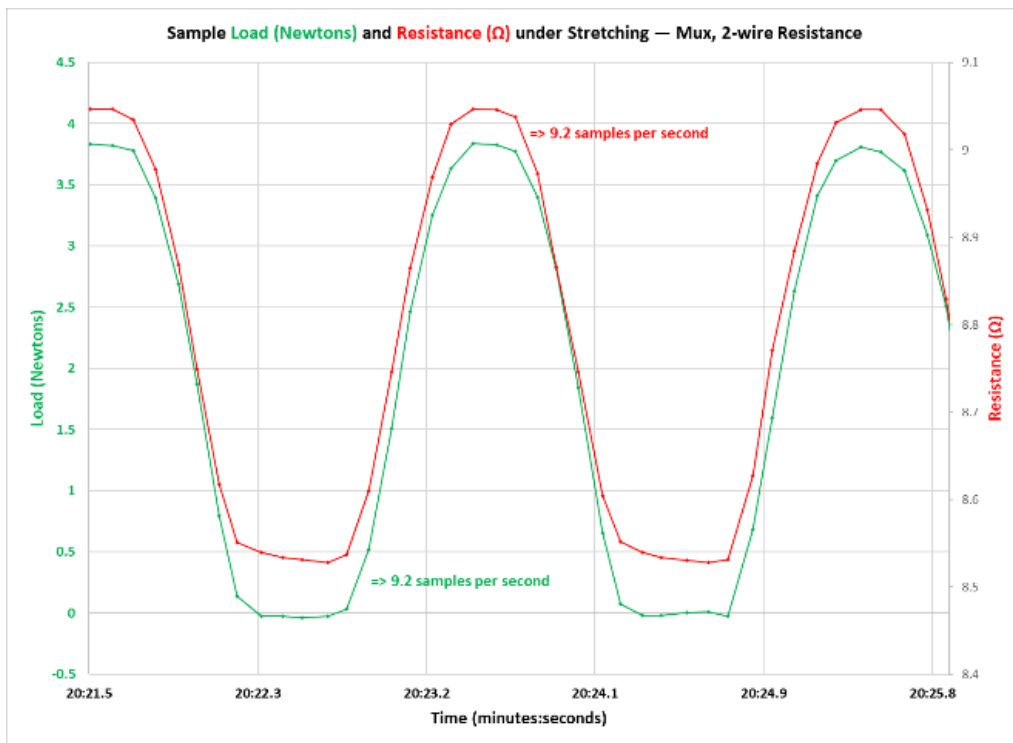


Figure 7. Continuous Measurements, Two Cycles in the Middle of the Test are Shown, Using One Meter with a Multiplexer, Measuring Load and 2-wire Resistance

Example 4 — Using Multiplexer on Two Channels — Measure Tension and 4-wire Resistance of Sample being Stretched

Figure 8 is presented for an overview. It can be seen that the Load and the Resistance follow the same curve throughout the test. The Load (green curve) varies between about 0.2 Newtons and about 4.2 Newtons. The Resistance (red curve) varies between about 3.85Ω and about 4.45Ω .

The measured resistance is about the same as it was for 4-wire measurement using the Front Panel. This was not the case for 2-wire measurement using the multiplexer.

Figure 9 shows two cycles in the middle of the test. The coarseness of the sampling can be seen in this graph. In this specific test, the actual sampling rate is 4.3 samples per second for the Load and for the Resistance. The use of 4-wire resistance measurements causes the sample rate to be lower because four connections are used instead of the normal two. The curves are plotted with a small diamond for the actual sampled value, and with straight lines connecting the samples. In this curve the noise of the Load Cell is visible. The noise is in the order of $1\mu V$.

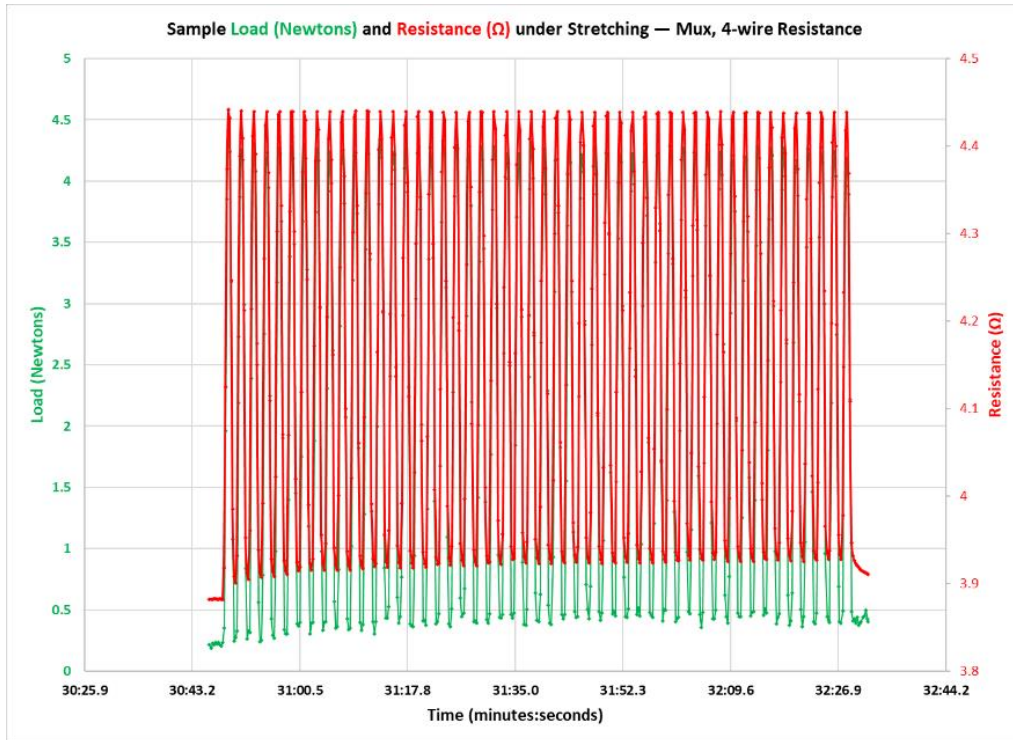


Figure 8. Continuous Measurements, the Full 50 Cycles of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 4-wire Resistance

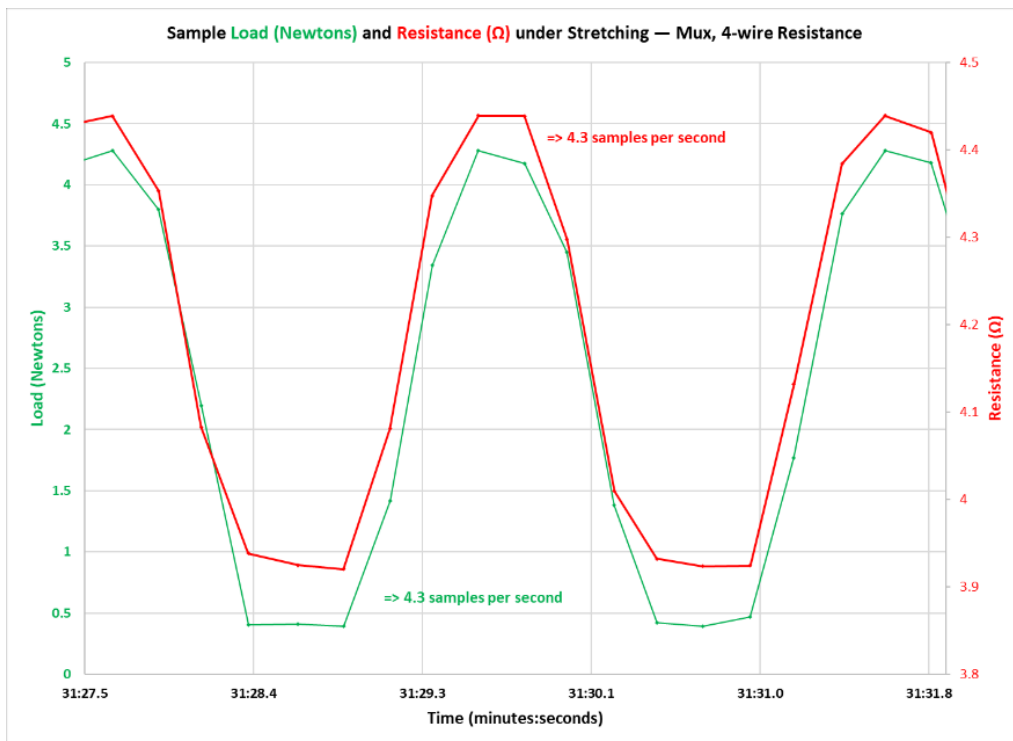


Figure 9. Continuous Measurements, Two Cycles in the Middle of the Test are Shown, Using the Front Panel of Two Meters, Measuring Load and 4-wire Resistance

Summary

This Application Note has illustrated the value of Continuous Measurements. A motor drive unit that normally can measure at only one point in a reciprocation cycle can be used to show the entire profile of a parameter that changes throughout the cycle.

Sample rates up to about 50 samples per second can be achieved using the Front Panel connections. The Continuous Measurement can run using a multiplexer, but the sample rates will fall, and in some cases significantly.

The value of 4-wire measurements was shown using a sample resistance of about 4Ω , which was approximately doubled when using regular clip leads about 3 feet long.

Flexdata SmartStop

Automatically Pause a Test

Flexdata functionality includes SmartStop. SmartStop allows the Flexdata software to use the measured data to change the direction of a test. The test algorithm can set maximum, or minimum values that if exceeded can cause the test to pause. This gives the Operator a chance to examine the sample once a critical threshold has been reached. The Operator can decide to continue the test or to stop the test. The test algorithm can be designed to offer subsequent critical points with different tolerances, or different motor parameter values. Flexdata also can change the direction of a test using Variables, parameters that are derived from calculations based on the standard measured values.

Example 1 — Check Sample Resistance Until It Cracks or Breaks

The resistance of a conductive element on a sample on a Yuasa Test Machine can be measured. If the sample is being stressed, whether stretching or folding or twisting, the resistance of the conductive element may increase. Flexdata gathers data from the meter and checks to see if the resistance exceeds a preset limit. If that happens, Flexdata will stop the Test Machine so the sample can be inspected. If the Operator wishes to proceed, another higher preset limit can be used. For this segment of the test the Test Machine parameters can be changed to increase the stress to hasten further damage to the sample, or decreased to slow down the damage. When the new preset limit is reached, Flexdata will stop the Test Machine again. If desired, a further segment can be added to stress the sample even more.

The resistance could be measured as 2-wire or 4-wire. Other parameters, like voltage, or temperature, or humidity, or tension could be measured in the same manner.

Example 2 — Increase Tension On Sample Until It Exceeds Preset Limit

The Yuasa Stretching Tester ST and Yuasa Multi-Purpose Tester FS-N can be outfitted with a Unipulse Load Cell that can measure Tension on a Sample. Flexdata gathers the data from the Load Cell. The Operator can set the maximum desired tension in Newtons in the testing algorithm. Flexdata can stretch the Sample in steps. The tension can be measured every step. When the tension reaches the specified value, the test will stop automatically. The test data will show the amount of stretching when the specified tension was exceeded.

Example 1 — Check Sample Resistance Until It Cracks or Breaks

An Experiment in Flexdata is designed by using blocks to show desired steps, or commands, or the “Algorithm”. Flexdata is a serial program, each step in the Algorithm is executed by itself. When that step is complete, the next step will begin. An Algorithm is an xml file and can be exported from Flexdata, and can be imported into a Flexdata later, or on a different different computer.

The Algorithm that will be used in this example is too large, or long, for one Figure. It will be broken into four sequential Figures. Figure 1 is the top of the Algorithm.

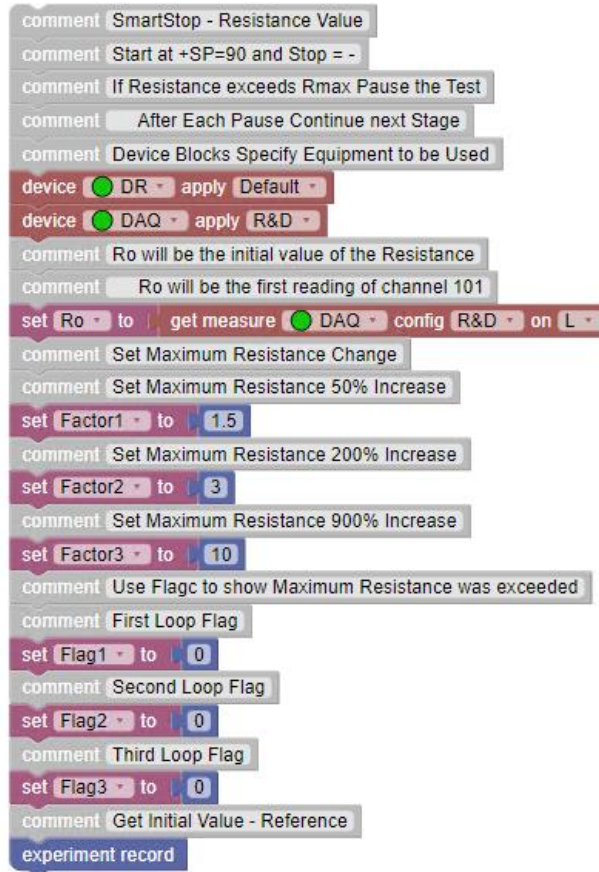


Figure 1. SmartStop Resistance Algorithm, Part 1

The grey blocks at the top are comments to explain the Algorithm. That is followed by two brown blocks which specify the equipment that will be used. The magenta block “Set Ro” will be used to cause Flexdata to get the initial resistance by reading the meter. The three blocks “Factor” will set the amount of resistance increase that will be acceptable. When the resistance exceeds that value, the Flags will cause the program to pause awaiting Operator intervention. The resistance increase in the first stage will be 50%, or a factor of 1.5. In the second stage, the resistance can have a total increase of a factor of 3. In the third stage the resistance can have a total increase of a factor of 10. After those blocks there is an “experiment record” block which will cause the meter to read the resistance value, and that is the initial value. This is followed by three large green blocks, the “repeat” blocks. Figure 2. is the first block, referred to as the Resistance Loop.

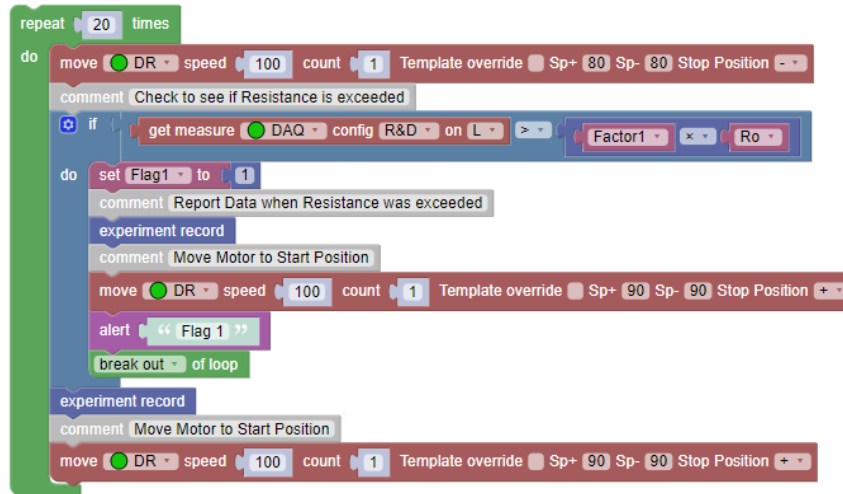


Figure 2. SmartStop Resistance Algorithm, Part 2, "Resistance Loop 1"

There is a blue loop inside the green loop. Ignore the blue loop for the moment. The first block in the green loop, “move DR”, will cause the motor to cycle one time to move from the initial position of 90° which was noted in the second comment at the top of the Algorithm. After the blue block is an “experiment record” which will get the resistance value after the motor made the first move. Then the brown block after that will cause the motor to move back to the initial position of 90°. The next step will be to return to the first "move DR" block in the "repeat" loop. The green repeat loop can run 20 times total.

The blue loop is an “if / do” loop. The first block inside the loop gets the value of the resistance from the meter. Then, in the same block, that value is compared with the Factor times the initial resistance, and if the value from the meter is not greater the instructions immediately following will be ignored and the process will exit the loop. If the value from the meter was greater, the instructions immediately following will be executed. Those instructions are to make a measurement and place it in the record Table, then move the motor back to the start position, and then report an “Alert” which stops the program. The word “Flag” with the flag number will show in the upper right of the Dashboard in Flexdata. The Operator may wish to examine the sample, or may decide to simply continue the test by clicking on “Continue” which is under the Flag Alert, or may decide to terminate the test by clicking on "Stop". If the Operator wishes to continue, the next block will be the "break out of loop" block which will end the Figure 2 portion of the test and go on to Figure 3.

The other two portions of the Algorithm in Figure 3 and Figure 4 are a repeat of Figure 2, but with greater stretching and with a higher tolerance for resistance change.

In Figure 3 the Factor is "Factor2" instead of "Factor1", the Flag is "Flag2" instead of "Flag1", and the stretching is 20° (6.28 mm) instead of the 10° (3.14 mm) of Figure 2.

In Figure 4 the Factor is "Factor3" instead of "Factor1", the Flag is "Flag3" instead of "Flag1", and the stretching is 25° (7.85 mm) instead of the 10° (3.14 mm) of Figure 2..

```

repeat 20 times
do
comment Increase Stretching for more Stress
move DR speed 100 count 1 Template override Sp+ 70 Sp- 70 Stop Position --
comment Check to see if Resistance is exceeded
if get measure DAQ config R&D on L > Factor2 x Ro
do
set Flag2 to 1
comment Report Data when Resistance was exceeded
experiment record
comment Move Motor to Start Position
move DR speed 100 count 1 Template override Sp+ 90 Sp- 90 Stop Position --
alert " Flag 2 "
break out of loop
experiment record
comment Move Motor to Start Position
move DR speed 100 count 1 Template override Sp+ 90 Sp- 90 Stop Position --

```

Figure 3. SmartStop Resistance Algorithm, Part 3, "Resistance Loop 2"

```

repeat 20 times
do
comment Increase Stretching for more Stress
move DR speed 100 count 1 Template override Sp+ 65 Sp- 65 Stop Position --
comment Check to see if Resistance is exceeded
if get measure DAQ config R&D on L > Factor3 x Ro
do
set Flag3 to 1
comment Report Data when Resistance was exceeded
experiment record
comment Move Motor to Start Position
move DR speed 100 count 1 Template override Sp+ 90 Sp- 90 Stop Position --
alert " Flag 3 "
break out of loop
experiment record
comment Move Motor to Start Position
move DR speed 100 count 1 Template override Sp+ 90 Sp- 90 Stop Position --

```

Figure 4. SmartStop Resistance Algorithm, Part 4, "Resistance Loop 3"

The next portion of this Application Note will examine the Dashboard screen in Flexdata at each of the three points where the resistance exceeded the preset value which caused Flexdata to pause the program. Figure 5 is the Flexdata screen at the first Flag, when the resistance exceeded a 50% increase over the initial resistance value.

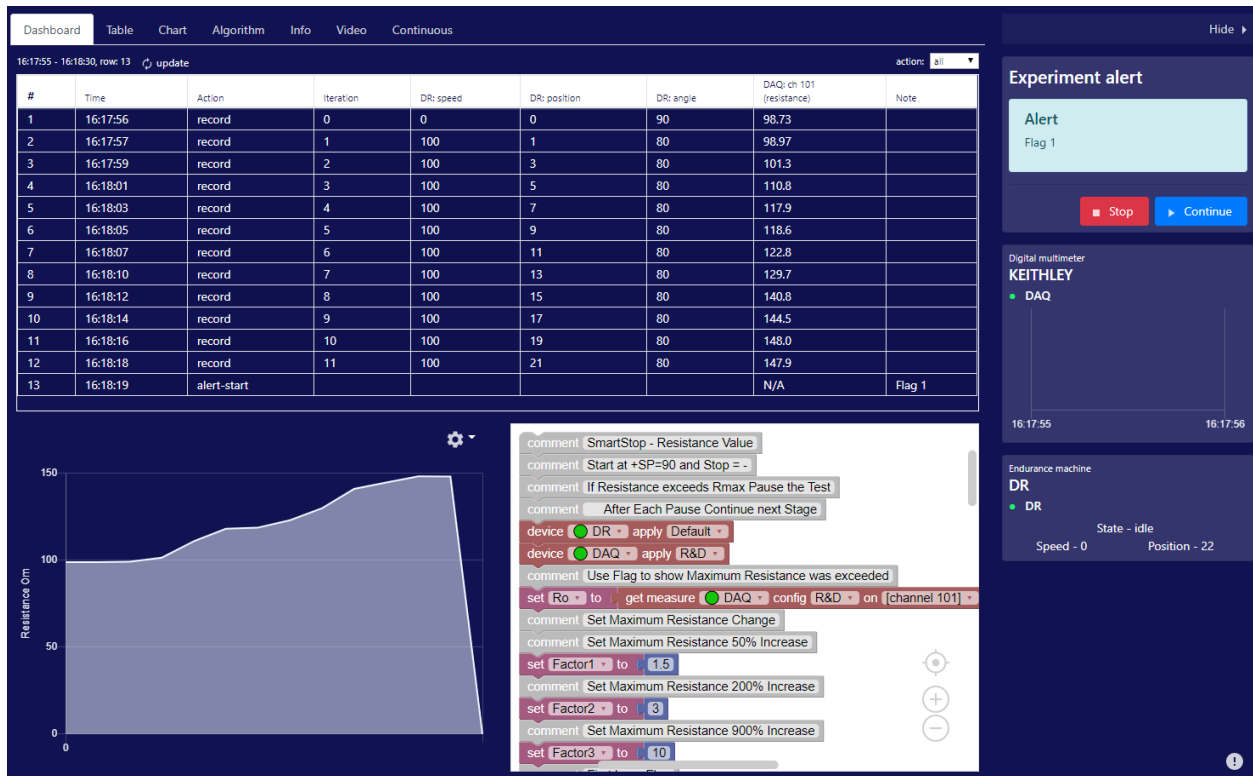


Figure 5. Flexdata Dashboard at First Flag

Line #1 in the Table shows the initial resistance value in the column “DAQ ch 101”, and that value was slightly less than 100Ω. As the test proceeded, the value gradually rose to 147.9Ω in Line #12, which was greater than 1.5 times the initial value. Looking at the graph in the bottom left corner it can be see that the resistance did increase gradually, but then the curve dropped to the axis when the motor stopped.

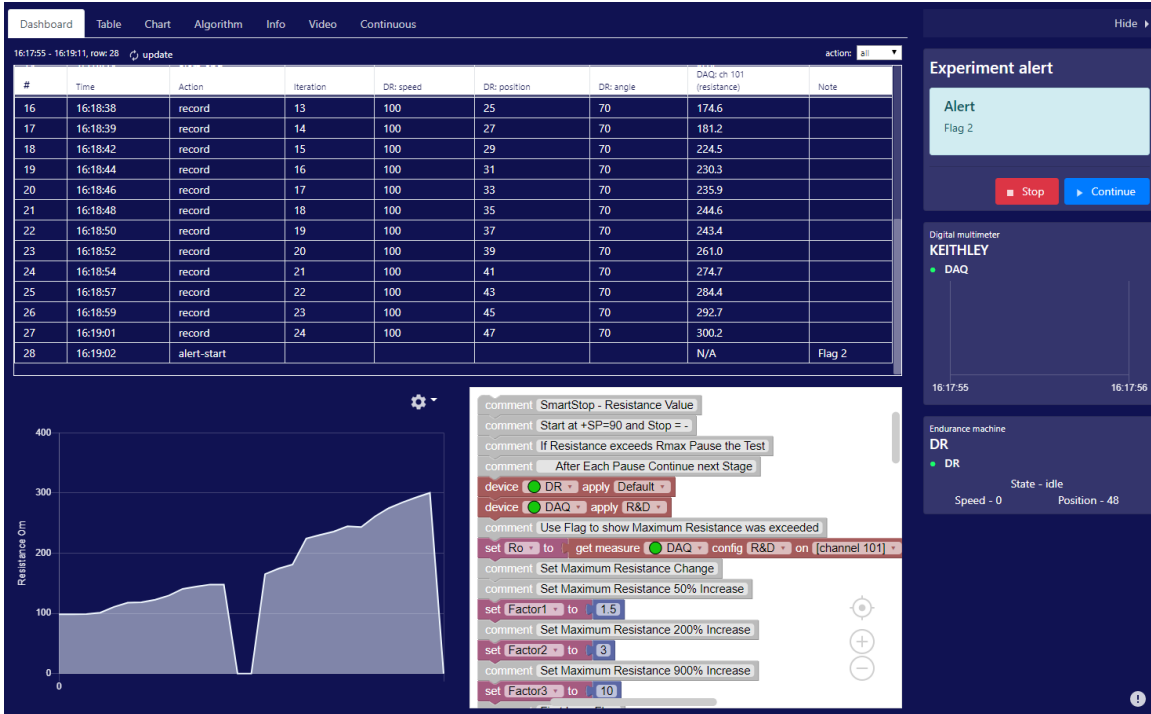


Figure 6. Flexdata Dashboard at Second Flag

Looking at the graph in Figure 6, one can see the pause for the first Flag, and now the pause for the second Flag. The resistance continued to increase and on Line #27 reached 300Ω which resulted in the second Flag. As before, the Operator has the option of examining the sample. And the option of clicking on “Stop” to end the test, or on “Continue” to resume the test.

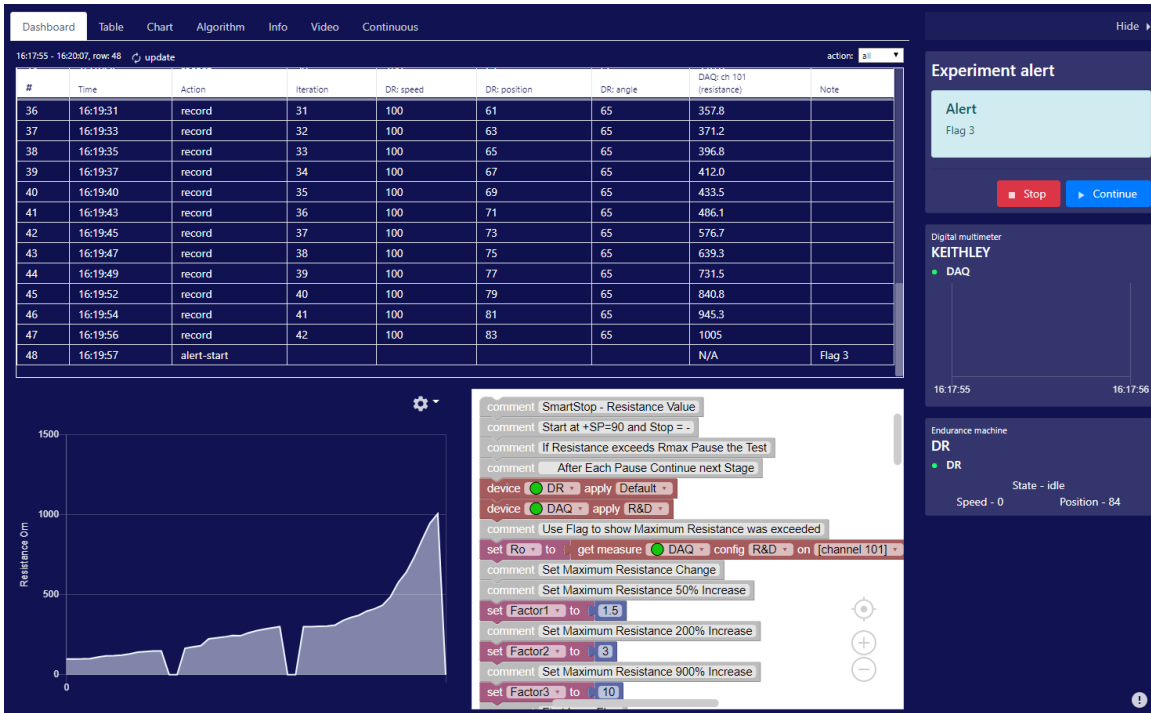


Figure 7. Flexdata Dashboard at Third Flag

Figure 7 shows the Flexdata Dashboard when the third Flag happened. Again the graph drops to the bottom when the test pauses. The resistance measured on Line #48 was just over 1,000Ω, ten times the value at the start of the test.

This example has shown how Flexdata SmartStop can measure and use the test results to change how the test proceeds. If the resistance had never exceeded any of the values for which the Flags were set, the program would have proceeded smoothly and 61 measurements would have been made — the initial measurement plus 20 measurements in each of the three loops. At each Flag the Operator has choices, simply continue, examine the sample and then continue, or stop the test. More complex algorithms can be designed for a more complicated test. Or, this algorithm could be used or made simpler by having fewer repeat loops. There could be more than one sample on the test machine, and the resistance of all the samples measured, and any one of the sample resistances exceeding the preset for that sample would cause the machine to pause.

Example 2 — Increase Tension On Sample Until It Exceeds Preset Limit

Figure 8 shows the top half of the Algorithm that was used to demonstrate SmartStop to determine the maximum stretching of a sample that does not exceed a preset value.

```
comment SmartStop Load for DR
comment Set Maximum Acceptable Load
comment Start at +SP=90 and Stop = -
comment Move DR mm by mm
comment Check Peak Load each Move
comment If Peak Load exceeds Lmax Stop the Test
comment At End Move back to No Load Position
comment If Flag = 1 maximum load was exceeded
comment Peak Load when Flag = 1 was the maximum Load
comment DR Angle when Flag = 1 was the maximum Stretching
device DR apply Default
device DAQ apply R&D
comment Get No-Load "Offset Voltage"
Load calibration DAQ config R&D on L
comment Set a Flag to "1" if Maximum Load exceeded
set Flag to 0
comment Set Peak Load = Maximum Load each Cycle
set Peak Load to 0
variable to report Peak Load
variable to report Flag
comment Set Lmax = Maximum Acceptable Load (Newtons)
set Lmax to 2
comment Get Initial Value - Reference
set Peak Load to get measure DAQ config R&D on L
experiment record
```

Figure 8. Top Half of SmartStop Maximum Load Algorithm

The first several lines in the Algorithm are comments to explain the Algorithm. The first two brown blocks (device) specify the test equipment that will be used in the Experiment. The third brown block (load calibration) is used to measure the Load Cell offset voltage under no-load conditions. The next two magenta blocks (set) define Variables that will be used in the Algorithm. “Flag” is used to signal that the maximum load has been exceeded and that the experiment must be paused. The variable peak level will be the peak load during each Iteration, or Step of the Experiment. The following blue blocks define variables that are to be reported on the Table in Flexdata. The block “experiment record” causes meters to make their measurements, cameras to take snapshots, etc.

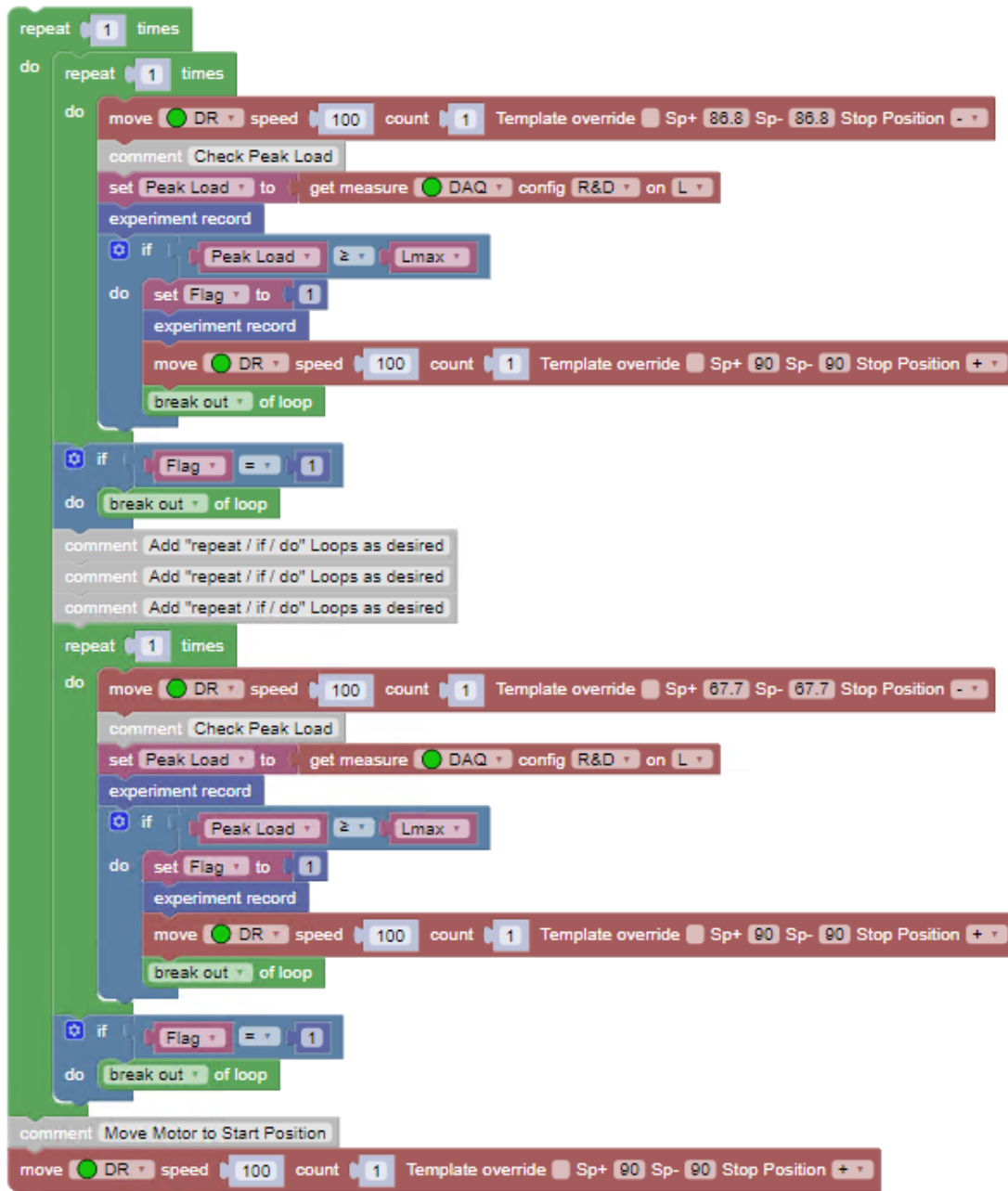


Figure 9. Bottom Half of SmartStop Maximum Load Algorithm

Figure 9 shows the bottom half of the Algorithm. Virtually all of the remainder of the Algorithm is two other repeat loops inside a single repeat loop that runs only the first time. The first of the two internal loops is shown alone in Figure 10.

The portion in Figure 10 could be called a “repeat / if / do” loop. This loop can be used multiple times in the Algorithm. Each of those loops stretches the sample another millimeter, then makes measurements, and then stops the experiment if the Maximum Load was exceeded. If the maximum load was not exceeded, all the rest of the blocks are skipped, then the next “repeat / if / do” loop will begin. If the maximum load was exceeded, the Flag will be set to “1”, final

measurements will be made, the motor will remove stretching on the sample, and then the experiment will end. It is often desirable to remove the load on the sample at the end so the sample does not remain stretched for what could be a lengthy time. At the very end of the Algorithm, the motor moves back to the non stretching position if the maximum load was never exceeded.

```

comment "repeat / if / do" Loop
repeat 1 times
do
  move DR speed 100 count 1 Template override Sp+ 88.8 Sp- 88.8 Stop Position --
  comment Check Peak Load
  set Peak Load to get measure DAQ config R&D on L
  experiment record
  if Peak Load >= Lmax
  do
    set Flag to 1
    experiment record
    move DR speed 100 count 1 Template override Sp+ 90 Sp- 90 Stop Position +
    break out of loop
  if Flag = 1
  do
    break out of loop
  
```

Figure 10. The “repeat / if / do” Loop

Figure 11 shows the results from the experiment. Each row is an “Iteration” in the Experiment. The column “DR angle” shows what the motor position was at the end of each Iteration. The column “Variable: Peak_Load” shows the Peak Load during each Iteration. The column “Variable: Flag” shows the value set for the Flag for each Iteration. Note that all lines up to Iteration #6 are the initial “0” value of the Flag. Iteration #8 shows the Flag value of “1”. That means the previous line result for Load was greater than the preset value of 2.0 Newtons, thus the experiment was stopped. Iteration #5 almost reached 2.0 Newtons, Iteration #6 exceeded the “Maximum Load”. Each row in the Table represents 1 additional millimeter of stretching. Thus the “Iteration” is also the number of millimeters of stretching. To not exceed 2.0 Newtons, the stretching would have to be limited to 5 mm. The sample was 110 mm in length, so that would be 4.5% stretching.

#	Time	Action	Iteration	DR: speed	DR: position	DR: angle	DAQ: ch 101 (load)	Variable: Peak_Load	Variable: Flag	Note
1	17:56:28	record	0	0	326	90.0	0.02500	0.006012086956523661	0	
2	17:56:29	record	1	100	327	86.8	0.3314	0.2898946521739125	0	
3	17:56:30	record	2	100	328	83.6	0.8051	0.7997204782608704	0	
4	17:56:31	record	3	100	329	80.5	1.223	1.1893130869565238	0	
5	17:56:32	record	4	100	330	77.3	1.597	1.5973211739130442	0	
6	17:56:33	record	5	100	331	74.1	1.977	1.9440260000000014	0	
7	17:56:34	record	6	100	332	70.9	2.263	2.3475516521739146	0	
8	17:56:34	record	7	100	332	70.9	2.333	2.3475516521739146	1	

Figure 11. Flexdata Table showing experiment results.

Flexdata Action Replay

Play Back Snapshots as a Movie

The latest version of Flexdata includes a new feature that can be helpful when examining a sample following an endurance test. If the sample at the end of the test has changed from what it was at the beginning of the test, the problem, and when the problem occurred, might be discoverable by examining snapshots taken at the measurement periods during the test. Flexdata Action Replay can play back the snapshots at high speed, like a movie, to make the discovery quicker and easier.

Example 1 — Use DMLHP with One Snapshot per Cycle - Short Test

Example 2 — Use DR11MR with Two Snapshots per Cycle - Long Test

Example 1 — Use DMLHP with One Snapshot per Cycle - Short Test

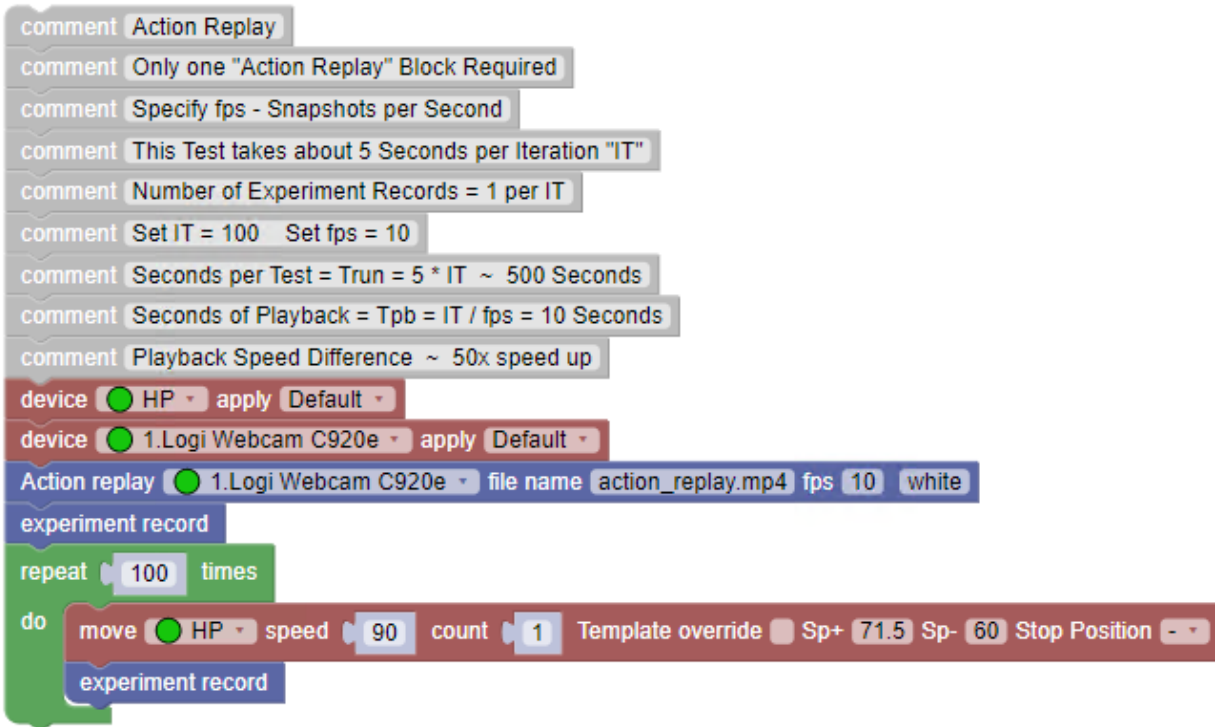


Figure 1. Algorithm for Action Replay using DMLHP

The Action Replay Algorithm in Figure 1 is short:

- Specify the motor with a "device" block
- Specify the camera with a "device" block
- Insert the "Action replay" block
- Insert a "repeat" loop with a motor "move" block and an "experiment record" block

Several comments were placed at the beginning of the algorithm to explain the test and parameters for the test. Examining the Action Replay block in detail in Figure 2 shows that only one parameter is truly required, and that is the fps (snapshots per second) desired for playback. The other windows allow the Operator to select the camera that will be used, the filename of the mp4 playback video (default is "action_replay.mp4"), and the color of the overlay.

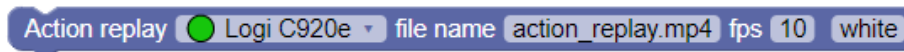


Figure 2. Action Replay Block

Note that the playback speed must be specified before the test is executed. This is required so the FD2 processor can select the snapshots that will be placed in the playback video.

The test took 8:27 or 507 Seconds, the estimate made prior to the test based on a short test was 5 seconds per "iteration" or 500 seconds. The playback speed will be exactly 10 seconds because there were 100 snapshots and playing them at 10 snapshots per second means 10 seconds elapsed time. The Flexdata Action Playback screen is shown in Figure 3. The play button is in the lower

left corner, and it shows also that the video length is 10 seconds. Clicking on the play button will show the 10 second video. The test was stretching a sample, and the snapshots were taken when the sample was at the no-load position. Typically the sample will stretch slightly during a test and will sag. Playing the video shows in high speed how much the sample sags as the stretching cycles progress.

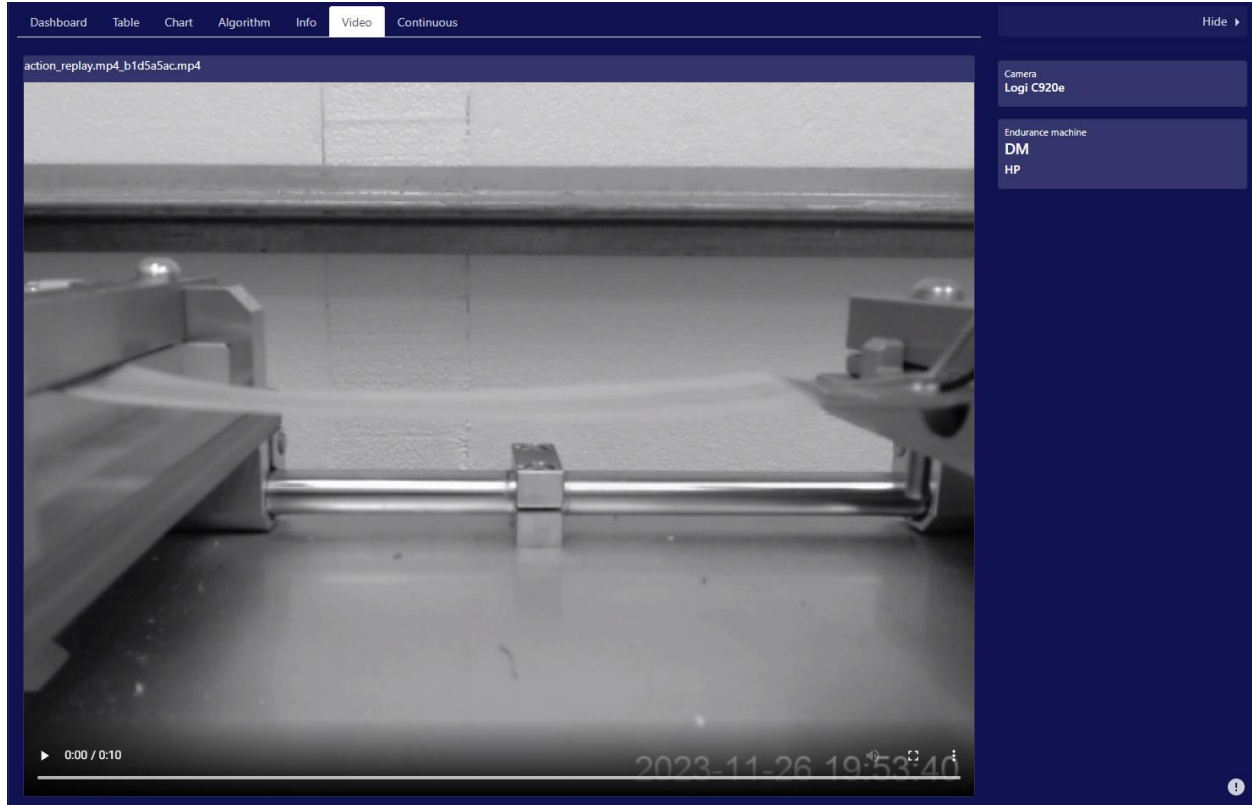


Figure 3. Action Replay Playback Screen

This test has been a short duration test, suitable for a demonstration but not necessarily a valid endurance test unless the sample is fragile and might be expected to be damaged during a short test.

Example 2 — Use DR11MR with Two Snapshots per Cycle - Long Test

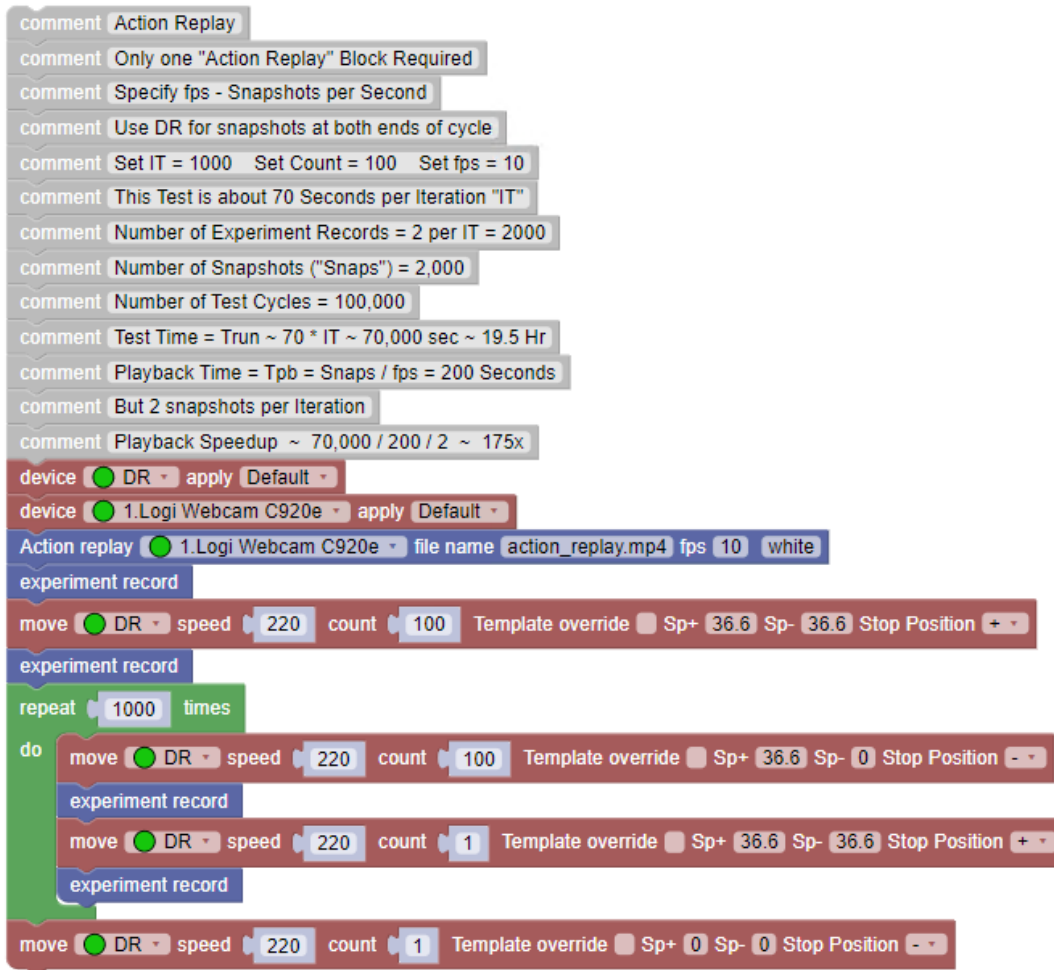


Figure 4. Algorithm for Action Replay using DR11MR

The test in Figure 4 might be a more realistic endurance test with 100,000 cycles stretching a sample. By using the DR11MR Motor Drive Unit, which can stop at both ends of a test cycle, snapshots can be taken when the sample is not stretched and when it is stretched. The algorithm is modified only by adding a second "move" block inside the "repeat" loop. It is only one count to move the Stretching Jig to the other end of the cycle. The "move block prior to the "repeat" loop is to have a snapshot at both ends at the beginning of the test. The "move" block at the end is to move the sample back to the unstretched position.

The action speedup is different in this case because there are two snapshots per "iteration". The estimated time duration is 70 seconds per iteration which is 70,000 seconds with 1,000 iterations. The fps value is the same in this test, "10", but that really means that the speedup is half the calculation from the first example. The playback duration will be 2,000 snapshots divided by 10 fps which means 200 seconds. That would lead one to think the speedup is 70,000 divided by 200, but it is actually half that value, 175x, because there are two snapshots per "iteration". That means that it will be 175 times faster to watch the Video Replay than to watch the 19.5 hour test.

Flexdata ConnectBox

Trigger for External Measurements

Flexdata is an open platform that connects popular mechanical testers, measuring instruments and data visualization tools. Clients may wish to add other accessories or proprietary hardware into the Flexdata platform. The ConnectBox enables engineers to do so; with their own programming to customize their requirements; for simple on/off functionality. The latest version of Flexdata now makes that possible by use of the Flexdata ConnectBox. The device has two sets of terminals, one to tell external equipment to begin their measurements, and the second for the external equipment to signal that the external measurements are complete. This application note will demonstrate how to include that function in a test algorithm.



Figure 1. Flexdata ConnectBox

Because the ConnectBox must be initialized prior to an experiment, the ConnectBox USB cable should be connected to the Computer while FD2 is closed. If the ConnectBox is connected while FD2 is open, it will not operate properly.

Example of a ConnectBox Algorithm

Figure 2 is one example of a Connect Box algorithm.

```
comment ConnectBox Algorithm
comment "Signal" Starts External Measurement
comment "Wait Signal" awaits End of External Measurement
comment Terminal #1 is Output to External
comment Terminal #2 is Output from External
comment Example is +SP & -SP Measurements
comment Start from +SP with Stop = "-"
device DR apply Default
device DAQ apply R&D
comment Measure Initial Condition
repeat 1 times
do
comment ConnectBox Measurement Loop
Signal
experiment record
Wait signal for: 10 message: Push Button
repeat 10 times
do
move DR speed 40 count 5 Template override Sp+ 90 Sp- 70 Stop Position -
comment Measure at -SP
repeat 1 times
do
comment ConnectBox Measurement Loop
Signal
experiment record
Wait signal for: 10 message: Push Button
move DR speed 40 count 1 Template override Sp+ 90 Sp- 90 Stop Position +
comment Measure at +SP
repeat 1 times
do
comment ConnectBox Measurement Loop
Signal
experiment record
Wait signal for: 10 message: Push Button
```

Figure 2. An Example of a ConnectBox Algorithm

The grey blocks at the top of the algorithm are comments explaining the algorithm and giving initial setup details. The “Signal” block is used by Flexdata to signal the beginning of an external measurement. To explain the algorithm, the “ConnectBox Measurement Loop” will be examined in detail.

```

repeat 1 times
do
  comment Start of ConnectBox Measurement Loop
  Signal
  comment Green LED on, Terminal #1 = HIGH
  comment Green LED stays on, Terminal #1 stays HIGH
  comment Run Internal Experiment Record
  experiment record
  Wait signal for: 10 message: Push Button
  comment Waiting for Terminal #2 = LOW
  comment Or Waiting for Push Button
  comment Or Waiting for 10 second Timeout
  comment Whenever Terminal #2 goes LOW
  comment Then Terminal #1 goes LOW, LED goes OFF
  comment End of ConnectBox Measurement Loop

```

Figure 3. ConnectBox Measurement Loop

Because of the “Signal” block, Terminal #1 on the ConnectBox will go to a HIGH voltage. This signal will be shown by the green LED turning ON and staying ON. This appearance of the LED could be used by the Operator to manually begin the external measurements, or the HIGH voltage could be used by software written by the Operator to start the external measurements. At this point Flexdata will begin its own internal measurements. The “Wait Signal” block will cause Flexdata to wait until Terminal #2 goes LOW. That signal must stay LOW until Flexdata has concluded that the entire measurement loop, including Flexdata’s own internal measurements, are complete. That LOW could come about by the Operator, after completing external measurements, manually pressing and holding the Stop Button until the green LED goes OFF. Or it could come about by the Operator’s software generating a LOW that stays LOW until Terminal #1 goes LOW. Or it could come about after a Timeout, in this case 10 Seconds. This Timeout is present because Flexdata will simply wait and wait until Terminal #2 goes LOW. The Timeout must be set for a longer time than is expected to complete the external measurements. The characteristic of the signal presented on Terminal #2 must be that it must stay LOW until Terminal #1 goes LOW which Flexdata uses to signal that the ConnectBox Measurement Loop has ended. Then the ConnectBox Measurement Loop is complete.

The ConnectBox Measurement Loop is used three times in this example of a test algorithm. It is used to establish initial conditions at the beginning of the test. Then it is used two times inside the repeat loop that is the actual operating program. The first time is to make measurements at -SP and the second time is to make measurements at the other end of the cycle +SP.

Of course more complicated algorithms can be designed, and the ConnectBox Measurement Loop would be used every time measurement data is needed.

Robert Hopkins
 Bayflex Solutions
 20 December 2023