

Liverpool John Moores University

Title: LabVIEW
Status: Definitive
Code: **7136ENG** (120347)
Version Start Date: 01-08-2018

Owning School/Faculty: Maritime and Mechanical Engineering
Teaching School/Faculty: Maritime and Mechanical Engineering

Team	Leader
Christian Matthews	Y

Academic Level: FHEQ7
Credit Value: 10
Total Delivered Hours: 24
Total Learning Hours: 100
Private Study: 76

Delivery Options

Course typically offered: Runs Twice - S1 & S2

Component	Contact Hours
Practical	8
Tutorial	12

Grading Basis: 50 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Portfolio	Portfolio	Continuous Assessment	30	
Exam	Exam	Practical Examination	70	4

Aims

To enable students with a background in using a traditional high-level programming language, but no experience using National Instruments (NI) LabVIEW, to create virtual Instruments (VI's) and interface them with compatible hardware for data acquisition (DAQ) and control applications.

Learning Outcomes

After completing the module the student should be able to:

- 1 Construct a range of LabVIEW VI's which demonstrate the typical functionality which would be encountered in industrial and research applications.
- 2 Configure and communicate with DAQ hardware to collect and store data from a physical system
- 3 Interpret a set of written functional requirements and create an original LabVIEW VI which implements that functionality
- 4 Document the construction and functionality of an original LabVIEW VI created by themselves

Learning Outcomes of Assessments

The assessment item list is assessed via the learning outcomes listed:

Portfolio	1		
Exam	2	3	4

Outline Syllabus

The list below provides an indicative list of topics which may be covered in this module:

The LabVIEW User Interface – understanding the LabVIEW user interface to enable students to navigate their way around the software, and to find help using the built in documentation system.

Virtual Instruments – understanding the architecture of a LabVIEW VI in terms of front panel and block diagram views. Introduce commonly used tools in the control and function palettes.

Dataflow – understanding the dataflow model of execution in LabVIEW

Data Types – String, Boolean and Numeric (Integer and Floating Point) data-types.

Arrays & Clusters – handling 1D and 2D arrays as well as multiplexed collections of signals in the form of a cluster.

Loops – Implementing standard programmatic loops such as 'While' and 'For' in LabVIEW

Structures – Case structures and sequences to control the execution of graphical programs based upon events and conditions.

Timing – Using shift-registers, wait and tick counts to add simple timing functions to VI's. The use of timed loops for more advanced timing applications.

File I/O – High-level and low-level file handling for access and storage of data stored in both standard formats (.csv, .txt etc) and proprietary formats (.tdms).

Debugging and Error Handling – approaches to debugging programs and using built-in error handling to control the behaviour of a program in the event of undesirable or unexpected behaviour.

Documentation – using tip-strips and text fields to provide informative comments regarding the construction and functionality of a VI.

DAQ Hardware – Configuration and communication with real and simulated DAQmx hardware.

Learning Activities

Tutorials, practical exercises

Notes

Topics will be delivered weekly, with students assessed at regular intervals throughout the course to encourage engagement with the curriculum and tutorial materials. A single 4 hour practical exam will be completed at the end of the module, with students working in an invigilated laboratory environment to complete a summative practical task based upon information provided in advance.

Participants will be provided with a student version of LabVIEW to allow them to complete a series of self-paced tutorials, in conjunction with a series of instructor led sessions.

Students will have access to NI hardware such as myDAQ, ELVIS II or other DAQmx compatible hardware to gain experience of simple data acquisition and control applications.