Project Topics and Ideas

Computer and Information Sciences,
Undergraduate (CISC)

Summer 2019

Instrumentation Design and Implementation

Douglas B. Rumbaugh

Mina Gabriel,
CISC Experiential Learning Coordinator
Harrisburg University
326 Market St,
Harrisburg, PA 17101
(717) 265-3727
MGabriel@HarrisburgU.edu
http://harrisburgu.edu/

Title:
Instrumentation Design and Implementation

Author:
Douglas B. Rumbaugh - dbrumbaugh@harrisburgu.edu

Difficulty:
Hard

Specialization:
Microcontroller/Electronics Programming

If other, please specify:

Most Appropriate Course:
Project II

Brief Description:
The student will design and construct a measurement instrument for a selected application. This system will run on micro-controllers or SBCs (such as an Arduino or Raspberry Pi), and connect to electronic sensors selected for the measurement to be made. The specific device to be designed can be tailored to student interest, however possible examples might include a refractive index detector, position sensor, etc.

Number of students needed:
1

Outcomes and Deliverable:
Documentation of the research that went into the project, the design of the instrument, a functional instrument, a presentation featuring demonstration of the instrument.

Skills Required:
Programming (C or C++), basic electronics

Available Resources:
This one will require the purchase of components, depending upon the specific instrument being designed and built. Usually the required components are fairly cheap, unless the student wants to do something particularly esoteric or complex.

Program Goal:
CISC 1.1: Mathematical Analysis, CISC 1.2: Sound Reasoning, CISC 1.3: Develop Solution, CISC 1.4: Deploy Solution CISC 2.1: Hardware Platform, CISC 2.2: Software Platform CISC 4.1: Written Communication, CISC 4.2: Oral Communications

Student Learning Outcomes:
1a: The student should be able to analyze a problem in a manner that facilitates the design of its solution., 1b: The student should be able to apply relevant principles of computing during their analysis of a problem., 1c: The student should be able to apply relevant principles of related, non-computing disciplines during their analysis of a problem., 2a: Student is able to create a formal software design based on a given set of requirements., 2b.:Student is able to develop a software solution from a formal design specification., 2c: Student is able to evaluate a software solution to determine its compliance with the specification., 6a: Student will be able to produce computer-based solutions by applying applicable computer science theory and software development fundamentals