ROCHESTER INSTITUTE OF TECHNOLOGY
COURSE OUTLINE FORM

COLLEGE OF SCIENCE

Chester F. Carlson Center for Imaging Science

NEW COURSE: COS-IMGS-180 Introduction to Computing and Control

1.0 Course Designations and Approvals

Required course approvals:	Approval request date:	Approval granted date:
Academic Unit Curriculum Committee	5/7/2013	6/4/2013
College Curriculum Committee	8/26/2013	10/15/13

Optional designations:	Is designation desired?		*Approval request date:	**Approval granted date:
General Education:	Yes	No √		
Writing Intensive:	Yes	No √		
Honors	Yes	No √		

2.0 Course information:

Course title:	Introduction to Computing and Control
Credit hours:	3
Prerequisite(s):	Matriculation as Imaging Science or Motion Picture Science or permission of instructor
Co-requisite(s):	none
Course proposed by:	Carl Salvaggio
Effective date:	8/26/2013

	Contact hours	Maximum students/section
Classroom	75 minutes / twice per week	30
Lab		
Studio		
Other (specify)		

2.a Course Conversion Designation*** (Please check which applies to this course).

*For more information on Course Conversion Designations please see page four.

	Semester Equivalent (SE) Please indicate which quarter course it is equivalent to:
	Semester Replacement (SR) Please indicate the quarter course(s) this course is replacing:
√	New

2.b Semester(s) offered (check)

Fall √	Spring	Summer	Other

All courses must be offered at least once every 2 years. If course will be offered on a bi-annual basis, please indicate here:

2.c Student Requirements

Students required to take this course: (by program and year, as appropriate) Imaging Science / First year Motion Picture Science / First year
Students who might elect to take the course: Any student in any major that is interested in an introductory course in computing in the Linux operating system, programming, and simple hardware control using single-board computers may elect to take this course.

In the sections that follow, please use sub-numbering as appropriate (e.g. 3.1, 3.2, etc.)

3.0 Goals of the course (including rationale for the course, when appropriate):

3.1 To develop programming skills necessary to solve scientific problems 3.2 To develop programming skills required to control simple electronics; optical detectors, LED sources, servo motors, etc. 3.3 To provide an understanding of simple electronics and electronics assembly techniques 3.4 To provide an understanding of the low-cost, single-board computers available and the use of these devices to create instrumentation useful in scientific experiments 3.5 To develop a proficiency in the setup and administration of these Linux-based single-board computers

4.0 Course description (as it will appear in the RIT Catalog, including pre- and co-requisites, and quarters offered). Please use the following format:

COS-IMGS-180 Introduction to Computing and Control This hands-on course is an introduction to computer programming, simple electronics, and the control of electronic devices using commercially available, single-board computers (*e.g.* Raspberry Pi). Emphasis will be placed on utilizing the analog and digital input/output ports available on these single-board computers to control and acquire data from electronic devices like optical detectors, LED sources, and servo-motors. The use of open-source software libraries to assist in the control and real-time acquisition of image data from peripheral imaging devices and cameras will be covered in detail. The student will be introduced to object-oriented programming using Python. Fundamentals of flow control, object types and creation, input/output, and problem-solving approaches such as the use of randomness, divide-and-conquer, Monte Carlo, and search will be examined in detail and applied to scientific, mathematical, and imaging-specific related problems. (Prerequisites: Matriculation as Imaging Science or Motion Picture Science or permission of instructor) **Class 3, Lab 0, Credit 3 (F)**

5.0 Possible resources (texts, references, computer packages, etc.)

5.1 Lee, K.D., Python Programming Fundamentals, Springer-Verlag, London
5.2 Langtangen, H.P., A Primer on Scientific Programming with Python, Springer-Verlag, New York
5.3 Guttag, J., Introduction to Computer Science and Programming, MIT Open Courseware, http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00sc-introduction-to-computer-science-and-programming-spring-2011/index.htm

6.0 Topics (outline):

- 6.1 Single-board computer architecture
  - 6.1.1 Power
  - 6.1.2 I/O
  - 6.1.3 Ports
- 6.2 Linux administration
  - 6.2.1 Installation and maintenance
  - 6.2.2 Fundamental Unix commands
  - 6.2.3 Organization / hierarchy
  - 6.2.4 Networking
  - 6.2.5 Devices
  - 6.2.6 X11
- 6.3 Shell scripting
  - 6.3.1 Editors
  - 6.3.2 bash
  - 6.3.3 Fundamental scripting commands
  - 6.3.4 init scripts
  - 6.3.5 cron
- 6.4 Package and file management
  - 6.4.1 apt-get
  - 6.4.2 rsync, ftp, sftp
  - 6.4.3 backup
- 6.5 Python programming
  - 6.5.1 idle development environment
  - 6.5.2 Basic objects
  - 6.5.3 Flow control
  - 6.5.4 Classes and objects
  - 6.5.5 Module creation and management
- 6.6 Scientific-problem solving
  - 6.6.1 Randomness
  - 6.6.2 Divide-and-conquer
  - 6.6.3 Monte Carlo
  - 6.6.4 Graphs
- 6.7 Basic electronics
  - 6.7.1 Current
  - 6.7.2 Resistance
  - 6.7.3 Capacitance

6.7.4 Switches/Relays
6.7.5 Amplifiers
6.8 Electronics assembly
6.8.1 Soldering
6.8.2 Wire wrapping
6.8.3 Crimping / mechanical connectors
6.8.4 Testing
6.8.4.1 Multi-meters
6.8.4.2 Oscilloscope basics
6.8.4.3 Simple logic analyzer
6.9 Electronics control and acquisition
6.9.1 GPIO
6.9.1.1 Digital
6.9.1.2 Analog
6.9.2 I2C
6.9.3 Devices
6.9.3.1 Simple displays
6.9.3.2 Optical detectors
6.9.3.3 LED sources
6.9.3.4 Servo motors
6.10 Visualization
6.10.1 pylab
6.10.2 matplotlib
6.10.3 PIL
6.11 Open-source libraries
6.11.1 numpy/scipy
6.11.2 OpenCV
6.11.2.1 Real-time image acquisition
6.11.2.2 Video

7.0 Intended course learning outcomes and associated assessment methods of those outcomes (please include as many Course Learning Outcomes as appropriate, one outcome and assessment method per row).

Course Learning Outcome	Assessment Method
7.1 Use the Python programming language to solve scientific problems; general and specific to imaging	Projects
7.2 Use the Python programming language to analyze and visualize scientific data; plotting, visualization, video	Projects
7.3 Use the Python programming language to control and acquire data from simple electronic devices; optical detectors, LED sources, servo motors, board-level and USB cameras	Projects
7.4 Develop skills necessary to design and implement simple electronics devices; breadboard circuit design, soldering,	Projects and practical exams

and circuit debugging	

8.0 Program outcomes and/or goals supported by this course

8.1 Develop the programming and visualization skills necessary to solve problems in subsequent courses within the department including the freshman imaging project, optics, color science, radiometry, image processing, detectors, and all imaging-related mathematics 8.2 Develop proficiency in a general-purpose programming language that is currently very popular in the imaging industry (remote sensing, astronomy, medical imaging, computer vision) 8.3 Develop problem-decomposition skills needed to solve complex problems 8.4 Develop proficiency in designing a device to create a solution to a scientific measurement problem; control, acquisition, and analysis 8.5 Develop skills necessary to formulate, analyze, and solve practical problems in imaging science

9.0

	General Education Learning Outcome Supported by the Course, if appropriate	Assessment Method
Communication		
	Express themselves effectively in common college-level written forms using standard American English	
	Revise and improve written and visual content	
	Express themselves effectively in presentations, either in spoken standard American English or sign language (American Sign Language or English-based Signing)	
	Comprehend information accessed through reading and discussion	
Intellectual Inquiry		
	Review, assess, and draw conclusions about hypotheses and theories	
	Analyze arguments, in relation to their premises, assumptions, contexts, and conclusions	
	Construct logical and reasonable arguments that include anticipation of counterarguments	
	Use relevant evidence gathered through accepted scholarly methods and properly acknowledge sources of information	
Ethical, Social and Global Awareness		
	Analyze similarities and differences in human experiences and consequent perspectives	
	Examine connections among the world's populations	
	Identify contemporary ethical questions and relevant stakeholder positions	
Scientific, Mathematical and Technological Literacy		
	Explain basic principles and concepts of one of the natural sciences	
	Apply methods of scientific inquiry and problem solving to contemporary issues	
	Comprehend and evaluate mathematical and statistical information	
	Perform college-level mathematical operations on quantitative data	
	Describe the potential and the limitations of technology	
	Use appropriate technology to achieve desired outcomes	
Creativity, Innovation and Artistic Literacy		
	Demonstrate creative/innovative approaches to course-based assignments or projects	
	Interpret and evaluate artistic expression considering the cultural context in which it was created	

10.0 Other relevant information (such as special classroom, studio, or lab needs, special scheduling, media requirements, etc.)

10.1 The class should be scheduled twice per week, since there is a significant preparation time on the students' part for each session, as they will need to setup their individual single-board computers.

10.2 The class will need to be scheduled in a classroom that provides individual, hardwired network links at each student's position.

10.3 The class will need to be offered in a classroom with a high-resolution projector.

10.4 The class will require external access to electronic assembly kits including simple tools, soldering station, and digital multi-meter.

10.5 Each student will be required to purchase a kit containing the single-board computer, power supply, cables, breadboard and breadboard accessories, board-level camera, servo-motors, LEDs, ADC board and amplifiers, as well as other miscellaneous electronic components.