Project-Based-Learning: Outcomes, Descriptors and Design

Peter D. Hiscocks
Electrical and Computer Engineering,
Ryerson University
Toronto, Ontario
phiscock@ee.ryerson.ca

Abstract
The paper contains three sections on project based learning. First, we provide a rationale and a high-level view of projects and their organization. Second, we present some examples from Ryerson University and show how these project descriptors apply. Finally, we show a new system of electronic instrumentation that supports independent and project-based learning.

1 What is a Project?

Project: an activity where the participants have some degree of choice in the outcome. The result is complete and functional, that is, it has a beginning, middle and end. Usually, it spans multiple lab periods and requires work outside scheduled lab periods.

Since there are choices in implementation, design is inherently a component of a project. A project is inherently different from an analysis or exercise, in which the solution has a predictable form.

Projects span a wide variety of possibilities: design and build, identify a system, do a forensic analysis, evaluate a product or assess some environmental situation.

1.1 The Role Project-Based Learning in Engineering

Projects have a central role in engineering so engineering educators naturally turn to project-based-learning (PBL) as an educational tool. Engineers implicitly understand that PBL is useful and many projects are introduced into the curriculum without a detailed analysis of their pedagogical value. However, it is useful to make explicit why we do project-based-learning and how it may be implemented.

First, project-based-learning actively involves students in the learning process, and results in increased motivation, satisfaction and confidence. Motivation is critical: a popular project will cause students to put in a huge effort to succeed and complete the task, and this enhances learning.

From [1]: compare the statements

Learn this because we know you will need it later.

vs

Learn this because you can see you need it now.

Students are more motivated to learn when they have an immediate application for the knowledge, a situation which applies in to project-based learning.

Second, compared to lecture assignments and lab exercises, project-based learning teaches a different set of skills:

- Project Management: How do we ensure that this project gets done in an orderly manner?
- Time Management: When do we have to have each item completed in order to meet the final deadline?
- Organization: What are the tasks that need doing, and in what order do they get done?
- Teamwork: Who does what? How do we cooperate?
- Research: How did other people approach this problem? Where can we get advice?
- Procurement: Where do we find project material and resources? How can we adapt existing resources to our project?

\(^\text{1}\)Emphasis added.
• Debugging: How do we troubleshoot these real-world problems? What tools should we know about that can help?

Furthermore, projects lead several kinds of meta-knowledge about engineering:

• Projects must be planned and managed,
• Projects take longer than you expect, even allowing for that effect
• System complexity is a non-linear function of system size
• Anything that works correctly the first time should be regarded with intense suspicion.

Battle-hardened veterans of project management have their own sardonic rules, encapsulated in such sayings as ‘Systems only fail when being demonstrated to someone important,’ and ‘The Law of Maximum Aggravation applies.’ Participating in projects gives these rules resonance in the mind of a student engineer and prepares them for project management in the engineering workplace.

1.2 The Design of an Educational Project

There are many ways of structuring a student engineering project. In this section, we’ll look at some of the alternatives. As always, the learning objectives of the project should be carefully considered and the parameters of the project then mapped to those objectives.

1.3 Guidance

We naturally first learn to swim in shallow end of the pool. For beginners, the deep end is (or should be) frightening and dangerous. In the same way, project beginners need the equivalent of floatation devices to keep them from drowning in project management. This is accomplished by a certain degree of guidance in the project which is provided by the supervising staff.

Early projects can be subject to significant guidance, that is, limited and artificial in comparison with full-blown engineering projects. In effect they are guided designs which are provided with most of the required information and some direction toward a solution to the project.

However, even a project that is strongly guided can increase motivation and confidence and introduce some of the management concepts of full-blown projects.

For example, second-year Mech students can be required to design and build a lever mechanism with a required output trajectory. Instructors provide the technical background and an example. Students to fill in some limited elements of the design, acquire parts, construct the circuit, troubleshoot errors and document the result.

Although the outcome is not in doubt, students obtain much the same sense of accomplishment as senior students completing a much more complex design.

1.4 Structured and Free-Form

Some industrial engineering projects are defined by a list of specifications driven by marketing information and the legal requirements, such as the placement of headlights on a vehicle or the emission of interference from a computer. Other projects are more free-form, in which the design is driven by what is possible and what engineers define as the coolness factor.

In the same way, student engineering projects may be tightly specified (we use the term structured) or more open-ended (free-form). An example from electronics:

<table>
<thead>
<tr>
<th>Structured</th>
<th>Free-Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design a flashlight using 2 light emitting diodes. The flashlight must:</td>
<td>Design a flashlight using light emitting diode(s). Marks are awarded for:</td>
</tr>
<tr>
<td>Use a standard 9V battery for power</td>
<td>Compactness</td>
</tr>
<tr>
<td>Include intensity control</td>
<td>Battery life</td>
</tr>
<tr>
<td>Occupy no more that 10 cubic cm</td>
<td>Light output</td>
</tr>
<tr>
<td>Operate more than one hour</td>
<td>Design aesthetics</td>
</tr>
</tbody>
</table>

The Structured form of the project assignment requires strong engineering skills and a disciplined approach. The Free-Form form of the project assignment allows a wide latitude in the design encourages big-picture creative thinking. Both methods are equally valid – they address different skills.
1.5 Size of the Project Team

Should projects be done by individual students, or should students be grouped into teams?

A team size of one is simple to evaluate: there is no place to hide. Furthermore, there are no issues of team composition, dynamics and evaluation to deal with.

On the other hand, larger teams force cooperative behaviour, important preparation for the real-world engineering workplace. A larger team can take on a more extended project, and from the standpoint of the instructor, supervision and marking may be simplified. However, effective student teamwork does require some up-front organization [9], [10]. The major objection to group projects - parasitic behaviour - can be dealt with by techniques [8] to evaluate team members on their individual contributions to the project.

1.6 Provision of Materials

Assuming that the project is about constructing some physical system (such as an audio amplifier or model cargo airplane), students will need to obtain certain material. There are three alternatives:

1. Provide a completed hardware unit: This might be appropriate where the project is centred around some application of the hardware, rather than its construction. For example, students may be given a completely assembled robot. The project requires programming the robot to execute some specific task (eg, fire-fighting), and the learning objectives are entirely centered on this software development.

2. Provide a kit of parts: Students assemble the kit, thereby obtaining some experience in construction in addition to the other project requirements. In competitive project, a kit puts everyone on the same footing. Sometimes certain critical parts (motors) are provided to deliberately restrain the scope of the project.

3. Require students to chase the parts: Students learn how to track down parts and deal with suppliers, an important real-world skill.

2 Project Examples

In this section, we show some descriptors and images of projects from Electrical Engineering at Ryerson University.

Light Emitting Diode Project

**Audience:** Theatre technology and high school students. **Objective:** Learn to build simple electronic project. **Concept:** Build light emitting diode 'device'. **Guidance:** High, extensive written manual. **Structure:** Free-form, do something interesting, examples provided. **Team size:** Individual. **Hardware:** Given parts list, required to shop.

![LED Flower](image1)

**Figure 1:** LED Flower

Extremely Simple Processor

**Audience:** Intro to Digital Electronics, EE students. **Objective:** Learn how a microcomputer works at the hardware level. **Concept:** Write low and medium level code to make functional computer. **Guidance:** High, extensive written material. **Structure:** Constrained, must meet specifications. **Team size:** Two. **Hardware:** Provided.

![Extremely Simple Processor](image2)

**Figure 2:** Extremely Simple Processor
**eebot: Microprocessor Robot**

**Audience:** Introduction to Microprocessor Programming, EE students. **Objective:** Learn microprocessor concepts and programming. **Concept:** Program a microprocessor-guided mobile robot to execute various tasks including line following and solving a maze. **Guidance:** Medium, pointers to information provided but students must design and construct their own complete program. **Structure:** Constrained, specific goals. **Team size:** Individual. **Hardware:** Provided.

![Figure 3: eebot](image-url)

**Weather Station**

**Audience:** Fourth year EE students. **Objective:** Learn analog design for complete microprocessor system. **Concept:** Build functional weather station with three instruments. **Guidance:** Medium: students must do some hardware design and complete software. **Structure:** Structured, specifications are given and must be met. **Team size:** One. **Hardware:** Parts, including microprocessor kit, purchased by students.

![Figure 5: Weather Station](image-url)

**mechbot: Analog Guided Robot**

**Audience:** Intro to Electronic Circuits, Mech students. **Objective:** Learn basic analog circuits. **Concept:** Create circuits to control robot, ultimately for sumo wrestling competition. **Guidance:** High, most circuits described in lab manual. **Structure:** Constrained at first, then open form. **Team size:** Two. **Hardware:** Partial kit provided.

![Figure 4: Mechbot](image-url)

**eelab Instrumentation**

**Audience:** Fourth year EE students. **Objective:** Learn instrumentation systems. **Concept:** Program modular electronic instrument. **Guidance:** Medium, some documentation provided. **Structure:** Open-ended, choose and implement a system (from examples). **Team size:** One. **Hardware:** Hardware is purchased and assembled by students from kit.

![Figure 6: eelab](image-url)
**Thesis Project**

**Audience:** Fourth year EE students. **Objective:** Major (capstone) project of EE Courses. **Concept:** Original project with significant design. **Guidance:** Low, students must do all research and design. **Structure:** Specifications are established by agreement. **Team size:** Two. **Hardware:** Purchased by students.²

![Thesis Project](image)

**Figure 7: Thesis Project**

### 3 Xtck: Equipment for Project-Based Learning in Electrical Engineering

![Xtck Prototype Hardware](image)

**Figure 8: Xtck Prototype Hardware**

In Electrical Engineering, project-based learning requires access to certain fundamental items of instrumentation: power supplies, signal generator, voltmeter and oscilloscope. Traditionally, the university provides this equipment but restricts access to specific locations and times. For students, purchasing their own electronic equipment is not an option: it’s too expensive.

The Xtck project is based on one salient idea: if students had access to their own personal electronic equipment then significant project-based learning - and the execution of conventional lab laboratory exercises - could take place outside scheduled university lab sessions. This would enhance student learning, allow more sophisticated projects, and reduce pressure on university resources.

We have developed a system based on open software that provides these same instruments to students. The system uses the hardware shown in figure 8 and open-source software that runs on a personal computer. For the cost of about 2 textbooks, it is feasible for a student to own an electronics lab.

The prototype of figure 8 (shown out of its case) includes (from top to bottom):

- Triple output power supply
- Volt-ohmmeter
- Waveform generator
- Dual channel 20MS/sec oscilloscope

These are the instruments used in basic electricity and electronics laboratories.

#### 3.1 The Xtck Architecture

A block diagram of the prototype system is shown in figure 9.

The Xtck instrument hardware attaches to a PC host via the serial port³. The instrument modules are daisy-chained via an extension of the serial bus and powered from an AC power adaptor. The host PC operates a graphical user interface for each of the various instruments.

An example of the software – the oscilloscope GUI (Graphical User Interface) – is shown in figure 10. There are no mechanical switches or controls - all operation of the instrument takes place on the video display.

In addition to the control function, the host PC provides the usual printing, storage and analysis facilities for data.

#### 3.2 The Host Software

The PC host software is written in the Tcl/Tk language. The Tcl/Tk language is not as well known as

³A USB port can be used with an inexpensive serial-usb adaptor, a device that is readily available in the commercial market.

²Photo credit, Luis Fernandez, Ryerson University.
Java or Visual Basic, but has a number of attractive features for this type of application:

- It is open source, maintained and extended by a large user community and freely available for any use [11].

- It is multi-platform and runs under Linux, Unix, Windows and Apple operating systems. Engineering students at Ryerson University often do software development on a Windows machine at home and then demonstrate the same software on a Linux machine at Ryerson. The GUI changes to suit the look and feel of its host machine.

- Tcl is a scripting language, and so no compilation or linking are required to run a program. Programs are pure text and are interpreted by the host.

- The Tk part of Tcl/Tk is a tool kit for the construction of a graphical user interface. As the result of some clever design decisions, Tk is powerful and easy to learn. A graphical user interface can be constructed in a matter of minutes. An event handler is intrinsic in the language, so mouse and keyboard events are trivial to program and manage.

- Access to the serial port is built into the language. No drivers are required.

- Sockets are available to support the inter-process communication of multiple GUI message sources. This simplifies the design of the software. As well, sockets can support connections to other networked sources. These could be computers in the same laboratory, or via the internet to some remote location. This has the potential for supporting remote access to the hardware, or the duplication of GUI clients at different locations so that other people can see what is being measured.

- The Tcl language supports multiple interpreters, each with its own variable space. This is useful when the system includes multiple units of the same type. For example, if the system consists of 3 voltmeter modules, each voltmeter GUI must maintain its own set of variables. Ideally,
the code for each voltmeter GUI should be identical, which requires that the variables of each GUI must have a private variable space. This happens automatically when each voltmeter runs under its own interpreter and communicates with the hardware via the socket mechanism.

The Tcl instrument software will be available as open-source software [12].

3.3 The Hardware Bus Structure

To simplify matters, the PC is bus master at all times. Instrument traffic is managed in each instrument module by a local microprocessor. Each instrument is opto-isolated from the host PC in order to prevent a student fault from propagating back to the host computer. The system runs reliably at 115kBaud. The most intensive bus traffic is required by the oscilloscope, which can update waveforms at rate of about 4 times per second.

Commands from the host and messages from the instruments are structured as ASCII strings. ASCII strings greatly simplify debugging at a small cost in efficiency compared to binary messages. A simple terminal emulator on the host can be used to send ASCII commands and observe ASCII-encoded messages. Any program that can talk to the serial port can control an instrument module. For example, Matlab was used to control the instrument hardware during the development process.

3.4 Configurable Instrumentation

It is possible to configure the hardware as new instruments for specific measurement applications. This is accomplished by creating a virtual instrument and graphical user interface on the host computer, without modification of the hardware.

For example:

- The waveform generator and oscilloscope are programmed to perform frequency response plot of a circuit network.
- The programmable power supply and voltmeter form a simple curve tracer.
- The power supply is used to sweep a heatsink through a range of powers while the voltmeter uses a thermistor to measure temperature. The GUI calculates and displays thermal resistance vs power dissipation.

- The system contains two function generators for a communications exercise. One generator provides the information signal and the second generator provides the modulation signal. The oscilloscope software performs a Fourier Transform and displays the spectrum of the modulated signal.

In this manner, the system can be used for projects in instrumentation or perform specialized functions in a larger project.

3.5 Applications in Education

Any real technological innovation has applications and social implications beyond its obvious first use. The first and primary use, as indicated in the introduction, is to support project-based learning. However, there are other interesting possibilities for an affordable, modular electronic instrumentation system:

- Staged Introduction Because this instrumentation is modular and affordable, students could stage their acquisition of a complete system. The power supply and voltmeter support an introductory circuits course. The waveform generator and oscilloscope complete a functional electronics laboratory. The total cost could be distributed over a period of time.

- Presentation Because this instrumentation uses a PC host, it is possible to use a digital projector to demonstrate operation of the instruments in a classroom or laboratory. Commercial instruments generally do not have this facility.

- Networking Socket communication in the host software supports various forms of distance education via the Internet.

- New Exercises and Demonstrations Traditional lab equipment has limited functionality. For example, the standard university lab waveform generator is capable of sine, square and triangle waveforms. If an arbitrary waveform generator is available, then a number of interesting exercises are possible. For example, the generator is set up to produce a cardiogram waveform plus 60Hz noise. The oscilloscope is set up in spectrum-analysing mode to show the spectrum of this signal. The student then designs an active filter to bandpass filter the desired signal.

- Instrumentation Project As part of a course in instrumentation, students could add features to the
existing software or, at a more advanced level, add new hardware and software modules to the system.

- **Open Software Enhancements** The instrument graphical user interface code for this project will be in the public domain. In the tradition of Open Software, we expect that students will contribute and benefit from contributions that enhance and modify the instruments. For example, students have already contributed a waveform editor for the generator and a waveform math package for the oscilloscope.

4 Conclusion

Project-based-learning is an effective method of learning in the engineering curriculum. It encourages involvement of the students in the learning process and conveys important information on project management.

Projects can have a variety of forms and structures. In the process of organizing an engineering project, some explicit consideration should be given to the alternatives and how they support the learning objectives.

It is now feasible to support project-based learning by providing students with their own set of electronic instruments. The combination of low-cost electronic components, open software and powerful personal computers makes it possible to provide a modular, versatile system at very modest cost.

Providing students with electronic instrumentation holds out the promise of de-structuring lab access and improving learning. It also creates a number of interesting possibilities at the lab curriculum level and in the organization of university laboratories.

Acknowledgements

Jim Koch of Ryerson’s Department of Electrical and Computer Engineering was a major contributor to the implementation of the projects shown in this paper. Devin Ostrom of Mechanical Engineering provide many of the electronics ideas for the mechbot. Jason Naughton and Ryerson EE technical staff assisted in a variety of ways. James Gaston, graduate student in EE at Ryerson, is a major contributor to the design of the eelab and the Xtick system.

References


