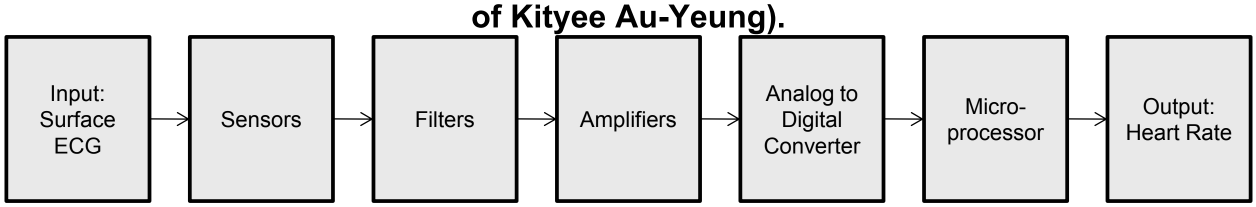
Appendix 4.5.3
Special Considerations for the Development of Electrical Device Prototypes

An increasing number of medical devices involve components that require electronics and signal processing, which generally is related to the discipline of electrical engineering. Examples of devices that utilize electronics include pacemakers, oxygen saturation sensors, fiber optic endoscopes, and blood analysis machines. While some of the discussion that follows may be difficult for someone not trained as an electrical engineer to appreciate, the goal is to highlight that concepts requiring expertise in electrical engineering can be approached in much the same way as any other concept. Whether or not innovators personally have electrical engineering experience, they should understand what needs to be done in the prototyping phase and whom to recruit to assist with this work.

If a concept being considered has one or more functional blocks that involve electronics design, it is important to expand on the design requirements to specify what the electronics components need to do within the context of the solution. For example, as part of a device to treat slow heart rhythms, there may be a functional block focused on sensing the heart rate from the surface electrocardiogram (ECG) signal. With this functional block in mind, the innovator can create an electronics block diagram that outlines the high-level input-output relationships and flow of information within this functional block. (Using the term “block diagram”—a standard convention in electrical engineering—to describe the components that make up a “functional block of the concept” may be confusing, but it emphasizes how the overall approach is similar at various levels of the prototyping and development process.) For the ECG example, a block diagram may look as follows:

Figure 4.5.3-1 – An example of an electrical engineering block diagram (courtesy of Kityee Au-Yeung).

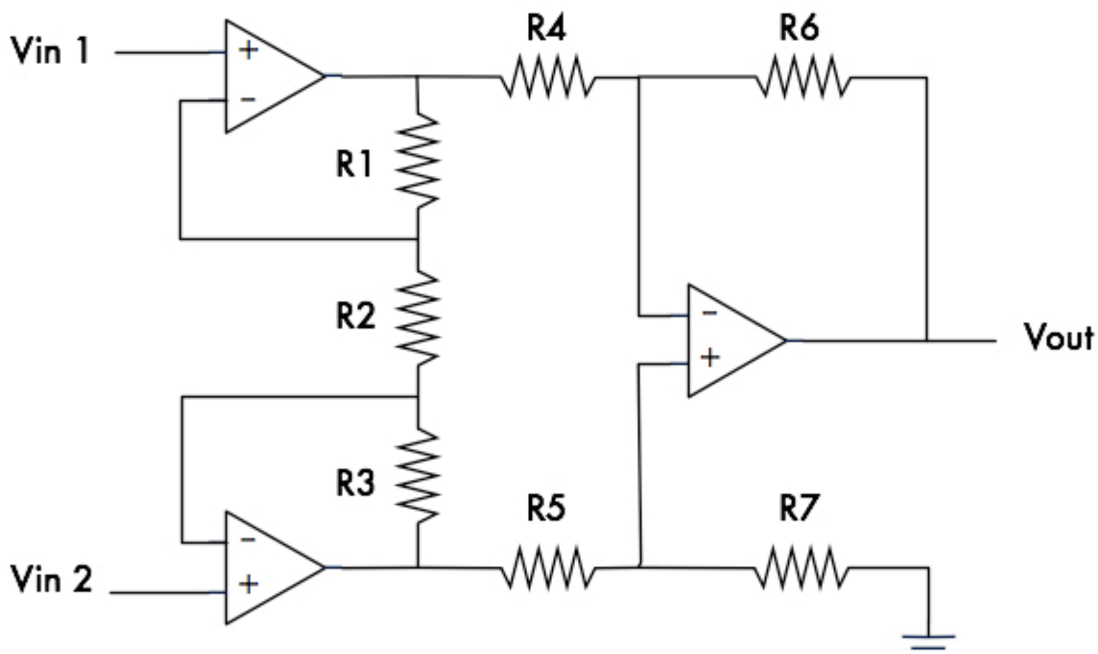


By creating this block diagram, the innovator can understand the various electrical functions and use them to create more specific design requirements. For instance, the innovator may define requirements such as: (1) the need to filter or amplify the sensed signal; (2) a need to convert the signal to a digital form; (3) the need to include a microprocessor to perform a logic or decision step about the incoming signal; or (4) the type of output that is required. Additionally, how the sequence of blocks is laid out can be important to creating a good design. While there are generally accepted conventions for the elements and sequence of the components of a block diagram, engineers may produce diagrams that vary in terms of the sequence and level of detail of the listed blocks.

Without knowledge of electrical engineering or a background in signal processing, it may be difficult to come up with a simple block diagram. However, through some basic research, the innovator should be able to break the high-level functional blocks of the concept (e.g., sensor to determine heart rate) into the high-level electrical components of an electronics block diagram (e.g., amplifier, analog-to-digital converter). Electronics design textbooks and published patents (particularly expired ones) can be of some help in constructing a new block diagram or modifying an existing one. These resources contain many useful and commonly used examples of electronics designs that can be leveraged across various solution concepts with minor fine-tuning.

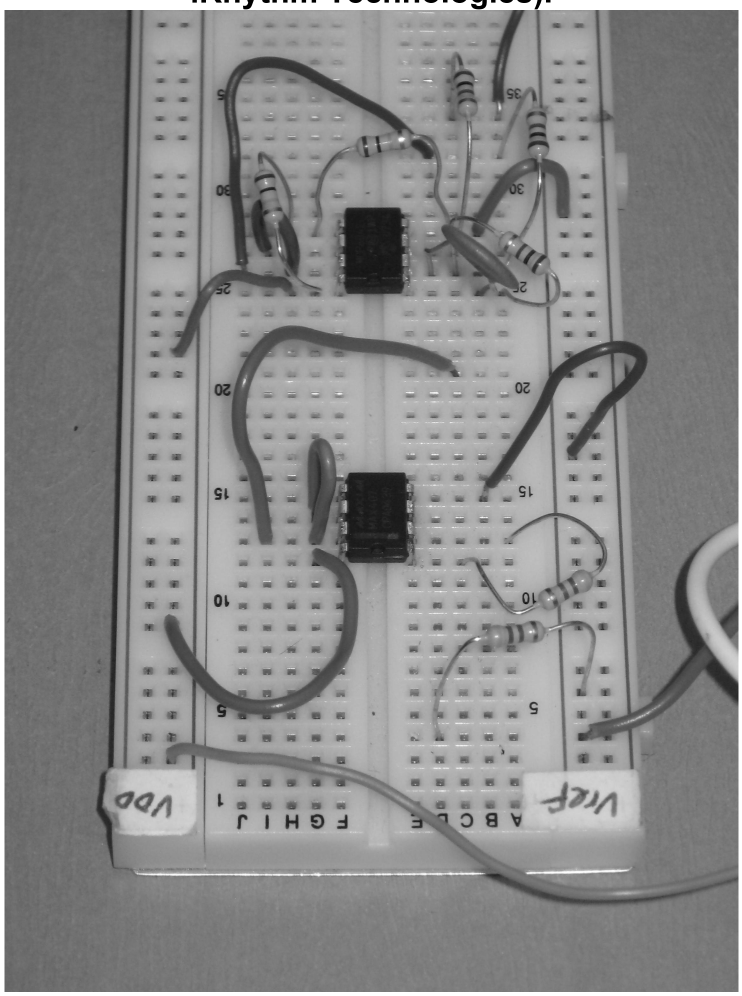
Once a block diagram has been created or modified for the solution concept, an innovator should determine how to prototype each component of the diagram. For the blocks that do not perform logic steps (e.g., making decisions), the function can be simulated using software programs or by building a hardware version. For the hardware version, a drawing of a circuit diagram showing the required hardware should be made first. At a basic level, a circuit is the design implementation of a block diagram. For example, in the block diagram shown as Figure 4.5.3-1, for the amplifier block, a potential circuit diagram showing components such as resistors (R) could be drawn as shown in Figure 4.4.3-2.

Figure 4.5.3-2 – An example of an electrical engineering circuit diagram (courtesy of Kityee Au-Yeung).



Again, without knowledge of electrical engineering, it would be challenging for an innovator to create this kind of output alone. However, many of these blocks are commonly used, so a survey of an electronics design textbook could make it easier to find or develop a diagram such as this. Furthermore many of these blocks are available in off-the-shelf modules often referred to as breakout boards or shields. These modules are typically designed to easily interface with other off-the-shelf hardware and often come with detailed tutorials. The help of an electrical engineer will also be invaluable. Using these resources, the innovator could then obtain the various components outlined in the circuit diagram and build the circuit. This could involve purchasing various breakout boards, resistors, capacitors, etc. and a **breadboard**—a reusable solderless device usually used to build a temporary prototype of an electronic circuit and/or experiment with circuit designs. A general example of a breadboard and its components would look like that shown in Figure 4.5.3-3.

Figure 4.5.3-3 – An example of an electrical engineering breadboard (courtesy of iRhythm Technologies).



Electronic designs generally fall into three categories: analog, digital, and mixed (i.e., the use of both analog and digital domains) designs. A microprocessor is often useful in digital designs. In the example shown above, a microprocessor is used to take the digitized data from the analog-to-digital converter and determine what the heart rate is. To develop the hardware that can perform this type of operation, an innovator first needs to pick a hardware platform with which to implement the design. Examples of hardware platforms include digital signal processors (**DSP**), application specific integrated circuits (**ASIC**), field-programmable gate array (**FPGA**), and microcontroller or microprocessors (for example, Arduino, 8051, ARM). An innovator would choose a particular platform based on his/her familiarity with it, its accompanying software tools, and its specific pros and cons. In general, there are tradeoffs between cost, speed, and power consumption. The latter can be a crucial factor in portable applications where battery life is critical. Typically, the choice of a hardware platform is made based on its functionality, cost, and ease of use.

These hardware platforms are often accompanied by software tools, which allow an engineer to simulate and program the hardware with embedded software (also referred to as firmware) in order to realize a desired functionality. This results in a chip that can then be incorporated onto a circuit board (e.g., breadboard) with the other components to create the entire circuit represented by the block diagram. One popular choice for rapid prototyping is the Arduino platform. The Arduino platform is a combination of well-documented open source hardware and a simple software language similar to C. Its low cost and ease of use has made it a frequent choice for mocking up electrical prototypes that can be optimized later.

Once all the pieces representing the components outlined in the block diagram have been realized, they can be assembled on a breadboard or custom circuit board. This board may contain some of the blocks already prototyped, as described above, or may be populated with the entire design at once. When possible, it is desirable to test each block individually before interfacing it with the rest of the system. Basic lab tools such as oscilloscopes and power supplies are often useful during the testing stages.

Simulating hardware using software tools is another way to test for correctness. Software simulations allow the designer to test, refine, and prove functionality rapidly. Generally speaking, software simulations should be used at the beginning of the design phase in order to validate functionality. In the end, however, there is no substitute for testing the actual hardware.

Finally, with the prototyped electronics in hand, the innovator can gather the required input and display the desired output. For the input, there are usually straightforward ways to sample the signal, which can be directly fed into the prototype. In the ECG example, electrodes could be attached directly to a person and the signal could be fed into the breadboard itself. For other concepts in which the sensor or input signal itself is novel, this may be its own functional block, but the idea here is that it is usually feasible to directly input a signal into the breadboard. Additionally, it is often useful to interface a computer to the electronics prototype. Programs such as LabView can be used to create an interface for displaying the output from the circuit. Again, in the ECG example, it would be used to display the heart rate determined by the microprocessor. The system might look something like that shown in Figure 4.5.3-4. Alternatively data-loggers can store the circuit output to a file, which can be opened on a computer (e.g., to plot the data in Excel).

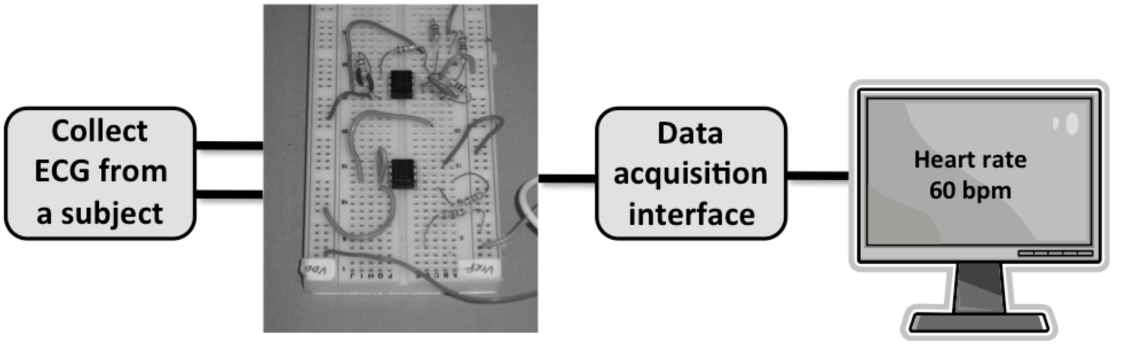


Figure 4.5.3-4 – An example of how these steps come together (courtesy of Kityee Au-Yeung).

In general, it is important to create a working prototype early in the process so that the innovator can observe the solution's interface with the biological signal. As a rule of thumb, when combining predictable systems (such as circuits and sensors) with dynamic systems (such as human physiology) problems may arise that can only be identified by observing their interaction in a working prototype.

When first getting started with prototyping the electronics components of a concept, it makes sense to use as many existing, off-the-shelf components and solutions as possible in order to minimize the development time required to get to the stage of testing with animals or humans. Today, more than ever, there are myriad resources for off-the-shelf, ready-to-use hardware modules. Platforms such as Arduino can be useful in realizing a functionally correct design. In general, the cost, size, and power consumption of electronic designs scale well with manufacturing volumes. To that end, the innovator should prioritize testing functionality with the understanding that cost, size, and battery life can be optimized later. The importance of testing an electronics solution in a biological setting early in the process cannot be emphasized enough in order to ensure that the final electronics design meets all the needs criteria and critical design requirements. The effort and cost associated with creating a custom solution can be expensive and time-consuming, so it is crucial to verify at an early stage of development that the effort will be worthwhile.

Additional Resources:

- Paul Horowitz and Winfield Hill, *The Art of Electronics, Second Edition* (Cambridge University Press, 1989) – An authoritative text and reference on electronic circuit design.
- Darren Ashby, *Electrical Engineering 101, Third Edition* (Newnes, 2011).

- Michael Margolis, *The Arduino Cookbook, Second Edition* (O’Reilly Media, 2011) – A guidebooks for experimenting with the Arduino microcontroller and programming environment.
- www.sparkfun.com - An online marketplace for electronics designed specifically for prototyping, which also includes tutorials, references, a blog, and other resources.
- Electronics.stackexchange.com – A question and answer site for electronics and electrical engineering professionals, students, and enthusiasts.

