9

ICT 9 Activity Sheet
Quarter 4 | Weeks 3-4

Microcontroller Programming

MARICAR R. PORNEL, *Oton NHS*
ZALDY M. TONDO, *Division Science Coordinator*
WRITERS

Introductory Message

Welcome to ICT 9!

The **Learning Activity Sheet** is self-directed instructional materials aimed to guide the learners in accomplishing activities at their own pace and time using the contextualized resources in the community. This will also assist the learners in acquiring the lifelong learning skills, knowledge and attitudes for productivity and employment.

For learning facilitator:

The **ICT 9 Activity Sheet** will help you facilitate the leaching-learning activities specified in each Most Essential Learning Competency (MELC) with minimal or no face-to-face encounter between you and learner. This will be made available to the learners with the references/links to ease the independent learning.

For the learner:

The **ICT 9 Activity Sheet** is developed to help you continue learning even if you are not in school. This learning material provides you with meaningful and engaging activities for independent learning. Being an active learner, carefully read and understand the instructions then perform the activities and answer the assessments. This will be returned to your facilitator on the agreed schedule.

Name of Learner:________________________ **Grade and Section:**______________
School:___________________________________ **Date:** ____________________________

ICT-9 ACTIVITY SHEET
Microcontroller Programming

Learning Competency:

Explain how microcontroller programming works.

Support Competencies:

1. Explain the selection of a microcontroller.
2. Discuss the programming process using Arduino.

Background information for the learners

In our previous lesson, we have discussed what is a microcontroller and its applications. As we progress with our lesson in microcontroller, you will know the basics on how programming is done in a microcontroller. To help you further understand this lesson, you will be guided by a video about Arduino programming.

Activity Proper
Activity 1.

Selecting a Microcontroller

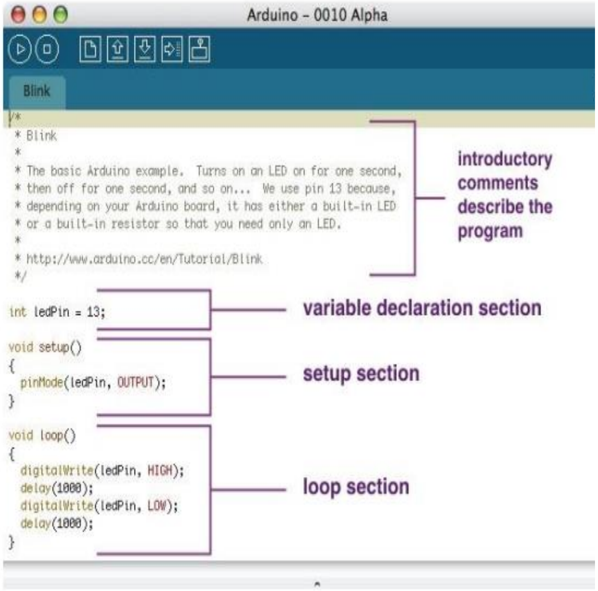
The features and functionality of a microprocessors are unique to each brand or model. Before coding a microcontroller for your project, you must select a model that meets all the requirements of your design. Common features people look for in a microcontroller include I/O pins, clock frequency and storage memory, however a countless number of other specifications exist. Writing code is pointless if the hardware can't interact with your circuit the way you need it to.

For beginners, Arduino is a brand of microcontroller commonly used amongst hobbyists and professionals alike. Its software is open source, meaning anyone can contribute to the growing pool of recourses available to its users.

Programming

Microcontrollers are typically programmed in higher-level languages such as C++ or Java. One of the essential tools needed to program a microcontroller is an integrated development environment (IDE). This software is usually developed by the creators of the microcontroller, and contains useful tools to help you program your microcontroller. Common tools found in IDE's include, code editors, compilers, and debuggers. Depending on the application of the microcontrollers, additional features may be added as well.

Once a suitable IDE is obtained, you can begin writing code. For explanatory purposes, this guide will show an example of the Arduino IDE in use. Below is an example of a simple Arduino program that makes an LED blink on and off at a frequency of 1Hz. The code is split into 4 different sections as follows:


Arduino - 0010 Alpha
Blink
introductory comments describe the program
variable declaration section
setup section
loop section

```
/*
 * Blink
 *
 * The basic Arduino example.  Turns on an LED on for one second,
 * then off for one second, and so on...  We use pin 13 because,
 * depending on your Arduino board, it has either a built-in LED
 * or a built-in resistor so that you need only an LED.
 *
 * http://www.arduino.cc/en/Tutorial/Blink
 */

int ledPin = 13;

void setup()
{
  pinMode(ledPin, OUTPUT);
}

void loop()
{
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

Comments

The program started with a comment section containing a general description of what the code/program does. While this section has no effect on the programs functionality, it's always a good to document it for future reference. User instructions, company and copyright information are also commonly placed here as well.

Variable Declaration

The second section is the Variable decoration. These variables are global, and can be called in any sections that follow. It is also common to create variables to describe each pins function, and set them equal to the pin number on the board to make coding more intuitive.

Void Setup

Digital pins on microcontrollers are commonly used as inputs or outputs, but very rarely can they be both. In this section, the user defines which pins are inputs or outputs, as well as any other parameters that must be initialized. While the method of doing so varies for different microcontrollers, almost all of them require a similar step to configure the microcontrollers internal circuitry to fit the needs of your design.

Void Loop

This section is where the function of your microcontroller is written. Any actions that require reading or writing values from pins, or computing the values of different variables is done here.

Compiling and Uploading

This step is almost always handled by the IDE. Once your code is written, it must be uploaded to the microcontrollers. Most have USB interfaces, but some smaller microcontrollers require a special hardware to be programmed. While we typically program microcontrollers in higher level languages, the microcontroller itself runs on assembly. To translate code to a format usable by a microcontroller, a compiler must be used.

A compiler is a software tool that takes higher level code and optimizes it for assembly. Assembly provides specific instructions to the microcontroller on what register operations to perform to match the operation of the original code. Once the assembly code is created, it can be uploaded to the microcontroller for testing.

Debugging

Debugging is the process of identifying and removing errors from computer hardware or software

Most IDE's contain debuggers that are able to run the code step by step at the users control while keeping track of variable and register values so the user can learn exactly what point the program behaves differently than as intended. This is a form of white box testing, and provides much more information that can be used to deduce the cause of a faulty program.

Answer the questions below:

1. Why is the importance of selecting a microcontroller?

2. How compling and uploading is done in a microcontroller programming?

Activity 2.

In this activity, you will watch a video about how microcontroller works.

Link: **https://www.youtube.com/watch?v=nL34zDTPkcs**

After watching the video, answer the questions below.

1. What is Arduino?
2. Explain how Arduino works.

Activity 3.

Multiple choice. Select the letter of the correct answer.

1. What is the section of a program that gives the general description of what the code can do?
   a. Variable declaration
   b. Comments
   c. Void loop
   d. Compiling
2. The following are the sections of a program, except
   a. comments c. debugging
   b. void Loop d. void setup
3. Microcontrollers are programmed using in a low-level language.
   a. True b. False

4. Microcontrollers work in this type of environment.
   a. IDE c. integrated circuit
   b. Arduino d. CPU
5. What are the common features you should in selecting a microcontroller?
   a. I/O pins c. Memory
   b. clock frequency d. All of the above
6. Arduino is a microcontroller?
   a. True b. False
7. Which is NOT a common tool found in IDE?
   a. code editors c. compilers
   b. debuggers d. uploading
8. What do you need to translate code to a format usable by a microcontroller?
   a. functions c. IDE
   b. debugger d. compiler
9. A microcontroller may work without a program running on it.
   a. True b. False
10. A step in programming process that identify and corrects program code.
    a. Uploading c. debugging
    b. Compiling d. optimizing

Reflection.

Complete the statements below.

I understand____

____

I don’t understand ____

____

I need more information about ____

____

Links and/or Other References

https://www.egr.msu.edu/classes/ece480/capstone/spring15/group13/assets/app_note_john_foxworth.docx.pdf
https://www.google.com/search?q=debugging+meaning
https://www.youtube.com/watch?v=nL34zDTPkcs
https://www.slideshare.net/EmmanuelObot4/introduction-to-arduino-hardware-and-programming-83637660