Learning game development with Unity3D engine and Arduino microcontroller

G Chursin, M Semenov

National Research Tomsk Polytechnic University, Lenin Avenue, 30, Tomsk, Russia, 634050

E-mail: `sme@tpu.ru`

Abstract. In this paper, we propose a new approach to the development and learning of children and students. The main idea is to combine a full-fledged video game and real microcontrollers. To complete the quests in the game, it is necessary to collect electronic circuits with microcontroller and various sensors. The Unity3D game engine is selected to develop the virtual game. Arduino microcontroller is used to create real electronic circuits. For testing, several puzzles with two-dimensional graphics were developed. The results showed that some sensors work with a delay of about half a second. For the rest, Unity3D correctly receives and processes data from Arduino microcontroller.

1. Introduction

The video game industry is rapidly developing nowadays. There are over 2.2 billion gamers around the world and billions of dollars in profit [1]. In 2018, the global video games marked was valued at around USD 134.9 billion and there is an estimation that video games could be a USD 300 billion industry by 2025.

From the time of video games creation at 1970s, they have been hugely used for military and corporate training purposes. By 2008, in the study [2] have found that an estimated 70% of major domestic employees used games for purposes of training. At moment video games can be considered as a huge part of popular culture, as an art form, as entertainment, as a learning tool and even as a sport. Modern games engines [3] Game Maker, JMonkey, Marmalade, Ogre3D, Shive, Sio2, Turbulenz, Unreal Engine 4 and Unity3D actively used in non-game contexts too [4,5]. A detailed description of these different applications may be found in [6,7].

Learning through entertainment is becoming a powerful alternative to the traditional practice of school education. Project-based learning is a modern approach to learning. According to the research [8], this allows to improve the academic performance of students. But this approach remains an educational project, which can not attract children who like computer games. In addition, according to study [9], project-based learning has such problems as the lack of motivation of students to self-education, compulsory group work, unequal opportunities in different schools, and the replacement of traditional teaching with projects. Without motivation and interest, it is not possible to qualitatively absorb useful information.

Many modern children spend a great deal of time playing games and other entertainment. For them, the virtual world is more interesting than learning. There are attempts to create educational games. But most of them do not take hold, as it is a simple replacement of a real teacher with a virtual teacher.

Based on the above mentioned contributions, however, there are two problems with the existing research. At first, the current studies did not quantify the impact among the different approaches to learning, which does not make it possible to argue the need for a complete change of classical learning to a different learning option. At second, there is no study to research the relation between educational games and practical knowledge. This indicates the low efficiency of regular educational games.

Motivated by these issues, this paper works from two aspects. First, we offer a different approach to development and learning. Combining the virtual and real world into one will allow a student to solve the problem of choosing between learning and entertainment. Playing a game, a student will be able to use real microcontrollers, sensors, boards, and electronic components. Along the game player understands how they work together. This approach will not discourage the student from using complex electronic components. And the desire to pass the game will motivate a student to understand base concepts and learn new one automatically. Second, we suggest to use a game as the education techonogy for providing an interactive learning environment for students and generaing an individual learning trajectory of the student.

The study's aim is to create a virtual game that needs to be played with a real Arduino microcontroller and plug-in sensors.

2. System Design Approach

At the moment the most popular game engines are Unreal Engine 4 [6] and Unity3D [7]. These video game developer platforms can be used to implement virtual and augmented reality experiences. The engines is written mostly in C++ and C# programming languages. We used the Unity3D game engine, it is the most suitable for virtual game development. The main features of cross-platform Unity3D game engine are support C# programming language for the development of two- and three-dimensional (2D, 3D) graphics games, and flexible licensing policy. C# to be developed by Microsoft Corporation. Therefore, using this programming language, it is easy to interact with Microsoft Windows. This is an important feature of this study.

An Arduino microcontroller is selected as the hardware platform. Arduino is suitable for starting in the study of microcontrollers. Thanks to it is possible to quickly learn to understand the basic ideas and principles of working with hardware. In addition, this platform has many development's libraries, its own compiler and tools for firmware microcontroller. Also Its main features are low cost and easy in setup and usage.

In figure 1 shown an example game workflow. At the system prototyping stage, we concentrated on the Unity3D game engine and an Arduino microcontroller interaction, and we do not paid attention to a story-line that should connect the different game levels. Because of this, a single level is elementary puzzle that is solved with simple schemes. Before the starting level, the player is told the ultimate goal, for example, melt the ice, find distance to wall, or moving through a maze. The player decides which of the proposed sensors to use according to the scheme to achievement the level goal. Arduino IDE is commonly used as the primary development environment for the Ardunio platform. This environment allows us to compile and load code to the board. But in order not to interrupt the game, it is necessary to flash Arduino microcontroller during the game. For this purpose we put sketches in advance. For this purpose we put Arduino sketchers in advance. We use the *arduino-cli* tool to compile and download the sketchs. It allows you to interact with Arduino without using third-party programs, using only the command line interface.

The interaction scheme Unity3D and Arduino microcontroller is presented in figure 2. We used circuit with temperature sensor DS18B20, ultrasonic distance sensor HC-SR04, RC522

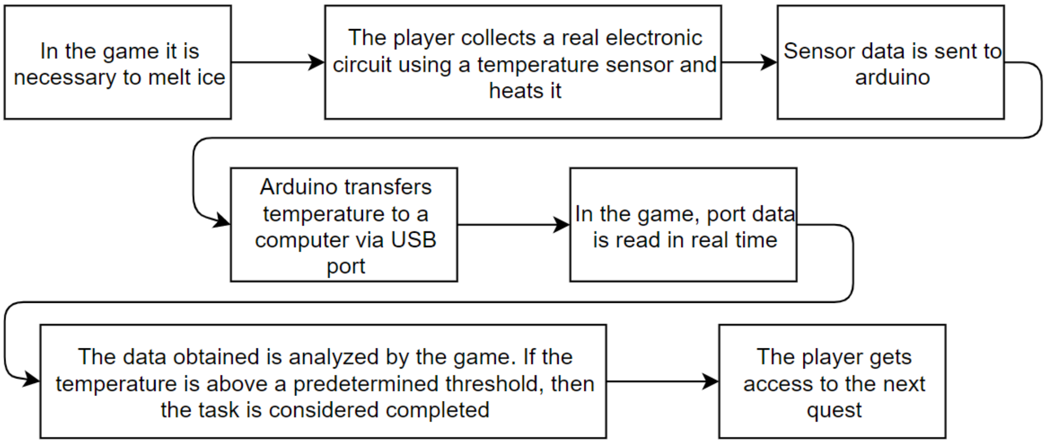
In the game it is necessary to melt ice
The player collects a real electronic circuit using a temperature sensor and heats it
Sensor data is sent to arduino
Arduino transfers temperature to a computer via USB port
In the game, port data is read in real time
The data obtained is analyzed by the game. If the temperature is above a predetermined threshold, then the task is considered completed
The player gets access to the next quest

Figure 1. Example game workflow: melt the ice.

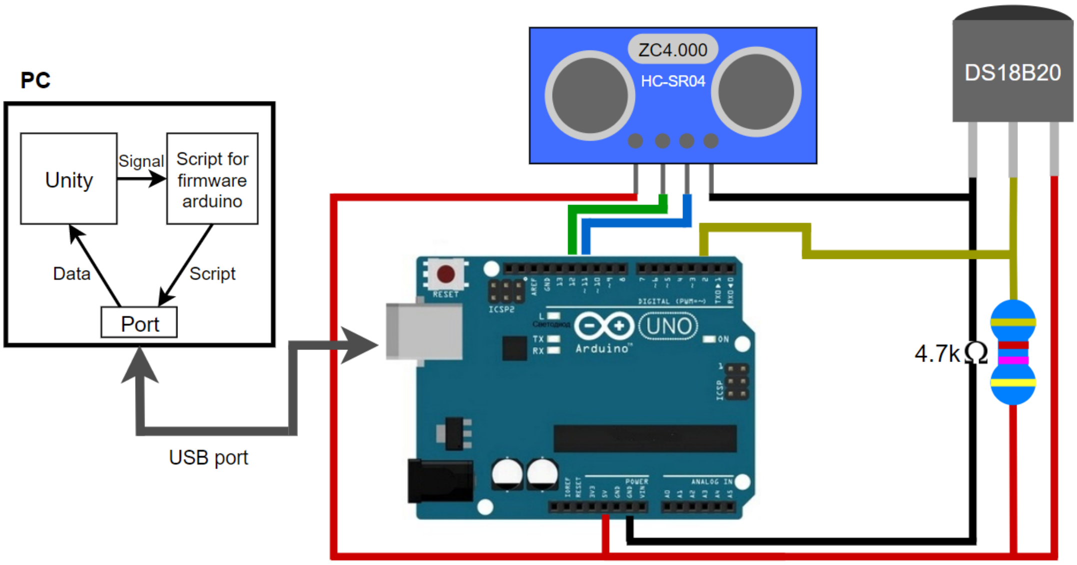

Figure 2. Unity3D and Arduino microcontroller interaction scheme. Description in the text.

RFID module, and two-axis analog joystick. A 4.7 kΩ resistor required for the sensor DS18B20. By colored lines we denoted physical connections between used sensors and Arduino microcontroller: red – 5 V power supply, black – ground, others – digital pin.

Unity3D script reads a serial signal from USB port and analyzes information based on the task. When changing a puzzle, Unity3D submits calls script to write new firmware to Arduino microcontroller. For firmware, we create a child process. This allows us to run command string scripts and obtain a standard output. Based on the output, we can analyze the compilation and firmware results and report problems to the player.

The main problem with using Arduino in games is the delay in reading data from some sensors. For example, between data from a temperature sensor, the delay can reach 1 second. Because of this problem, it is impossible to use Arduino in dynamic scenes where fast decision making and performing several actions at the same time is required. Problems can also include corrupted data arriving at a computer. This may be due to poor connections between devices or freezes of iron. Otherwise, the proposed game works adequately. There is practically no additional load on the central processor, which is important for low-end computers.

3. Results and Discussion

During the work, a number simple puzzle with 2D graphics were created. At the testing stage, it is possible to draw conclusions about the operability of the proposed game model, and Unity3D game engine adequately interacts with Arduino micro-controller. Weaknesses include sensor error and the speed of information processing by third-party Arduino libraries. In our tests, compilation and firmware take an average of $5 - 7$ seconds. This time may vary depending on the power of the machine used. According to our observations, the proposed scheme can be used to solve tasks and puzzles that are not limited by time.

To get feedback on the game idea, three adult independent players were asked to play it. Due to the lack of understanding of the principle of connecting modules to Arduino micro-controller as well as the absence of explicit clues in the game itself (for example, how to connect certain sensors with a port), circuits of connecting sensors to Arduino micro-controller were offered to each level.

According to the test reports, such a game model would be interesting to children as well as students, because it gives a new experience in computer games usage. It is worth noting main remark from players it is necessary to move from 2D to 3D version of the game.

4. Conclusion

In this work we demonstrate the idea implementation of a game aimed at the development and learning of children and students. The main idea is to combine a full-fledged video game and real microcontrollers. In the first stage, the main task of the game is to a player develop an understanding of where to use certain sensors and microcontrollers. A player without specific knowledge can create his/her first schemes and implement them in the gameplay. As a result, the player gets real knowledge of how sensors and microcontrollers work, sees their possible advantages and disadvantages of realized scheme. This contributes to the development of interest in science and technology, engage to in an interdisciplinary and applied learning approach.

We chose C# as the core programming language. The widespread use of this language for game development allows us to easily interact with the Micrisoft Windows and associate the game with various third-party programs, scripts and file formats.

The further research of our longitudinal study can be continued in the following directions. At first, we are planning to move from 2D to 3D game. This transition will increase interest in the game, reach a large audience and expand the possible use of Arduino sensors. Secondly, we are considering porting the game to mobile devices. For this it is necessary replace the Arduino with another microcontoller, for examplem it can be esp8266 that has built-in Wi-Fi and bluetooth. Such features will simplify the connection of the controller wirelessly. Finally, for educational purposes, it is planned to introduce a self-programming mechanism. This will allow to develop a player theoretical knowledge as well as practical skills.

References

[1] Risi S and Togelius J 2019 *arXiv e-prints* arXiv:1911.13071
[2] Wolf M J P 2017 *Quaderns* 119–128
[3] Christopoulou E and Xinogalos S 2017 *International Journal of Serious Games* **4** 21–36
[4] Resch J, Ocelewski I, Ehrentraut J and Barnett-Cowan M 2018 *arXiv e-prints* arXiv:1901.00729
[5] Matallaoui A, Herzig P and Zarnekow R 2015 *48th Hawaii International Conference on System Sciences* (Kauai, HI, USA)
[6] Unreal 4 game engine, Epic Games Incorporated, USA `https://www.unrealengine.com/`
[7] Unity3D game engine, Unity Technologies, USA `https://unity.com/`
[8] Kazemi F and Ghoraishi M 2012 *Social and Behavioral Sciences* **46** 3852–3856
[9] Aksela M and Haatainen O 2019 *Integrated Education for the Real World* 9–16