# **Abstract**

Contained within this report is the design and implementation process of the robotic arm control system. This system aims to provide a more intuitive, wireless and real-time control system by allowing the user to control a simulated robotic arm with the motion of their arm. This has been accomplished by allowing hardware to be attached to the arm. Internal measurement units were used to provide data which was then processed and then converted to degrees using Euler angles to be used in the simulation. Testing has shown that despite the control system being operational, with a robust communication network, is inadequate to serve as a control system for a robotic arm due to the use of Euler angles to calculate the orientation of each part of the arm which only provided accurate orientation estimates within and defined range of motion.

# **Introduction**

The aim of the project is to design and implement a robotic arm control system which would enable the user to project own’s motion onto a simulation. This system could later be integrated with a real robotic arm which would be controlled by the system. Using such system would provide an alternative, more intuitive solution to controlling robotic arms compared to the more common method of using a joystick. An alternative use for this kind of system would be to capture motion of the arm to be used in virtual reality game industry which is becoming increasingly popular with each passing year [1].

The project consists of physical hardware and software which is used to control the hardware such as a microcontroller or the FPGA board. Other hardware includes batteries, Bluetooth modules, USB to UART bridges and internal measurement units. Two different programming languages were used: C to program the simulation and C++ to program the microcontroller. Additionally, a hardware description language, VHDL, was used to program the FPGA.

Because majority of the hardware has to be attached to the arm in order to retrieve orientation data of the arm, a series of attachment platforms must be designed in order to allow the user to wear the hardware. The user must be able to move their arm freely without being restricted by connection cables and therefore the system must be able to transmit the data wirelessly to the computer reliably, in real-time, without corrupting the data in the process. For this reason, a communication network needs to be established to maintain the flow of data between different components in the system. The orientation data extracted must be processed and fused through the use of complementary in order to eliminate the effect of gyroscopic drift. The data in the x, y and z axes needs to be retrieved from the internal measurement unit in order to allow the system to estimate the orientation of the arm in three dimensions.

In order to confirm that the robotic arm control system functions in the intended manner, the simulation will be created to resemble the structure of a human’s arm. This means that the simulated arm will have to consist of three parts: the upper arm, forearm and the hand.

Aims

* Build a communication network that allows data to be transmitted around the hardware without corrupting the data in the process. This system should communicate with the simulation wirelessly.
* Enable the system to measure orientation of the arm in all three axes.
* Test the data from the internal measurement units and improve accuracy and reliability of the readings and perform additional tests on the implemented methods to verify their functionality.
* Design the sensing system such that it can be attached to the arm and build an attachment platform for all the components.
* The system should work in real-time. This means that the latency between the time between the acquisition of data and the receiving data in the simulation should be small enough to not notice this delay.

[1] <https://www.statista.com/statistics/528779/virtual-reality-market-size-worldwide/>