CS4099 DOER: Rust implementation of the sACN protocol

Paul Lancaster

September 17, 2019

Description

The project aims to expand an existing implementation [2] of the streaming architecture for control networks (sACN) protocol [1] in rust so that it fulfils the scope of the protocol as defined in [3].

Background

DMX512

DMX512 is an protocol used in the entertainment industry for the control of lighting, effects and other devices. It works by daisy chaining devices together into distinct physical chains (called universes) and is a one way protocol. This means that the devices in the line cannot communicate their presence back to the controller so the controller must know about the devices ahead of time and their addresses so it can broadcast packets down the line which the devices then receive and use. The DMX packets are a fixed size and contain five hundred and twelve 8-byte channel (+ a start code) which allows them to control up to 512 different devices on a singular line. A device may support the use of multiple channels to control different functionalities so for example a light with RGB colour mixing may use 3 channels to allow control of the Red, Green and Blue individually. Since there are only 512 channels available on a single universe this quickly imposes a limitation to the number of devices that can be connected together, especially as modern lighting fixtures commonly use upwards of 30 channels each for a moving light with usage of many more not uncommon. The solution to this was previously to simply have more physical lines (universes) and in this way allow more devices to be controlled simultaneously. This comes with a number of problems however as each new physical line means a new cable coming directly from the control desk.

DMX512 Problems

As the control desk is often far from the devices themselves (at the back of the venue whereas the lights/devices are above the stage) it means that many cables need to be run which can be expensive and time consuming.

The length of the cable runs can cause signal interference / degradation and DMX as a 1 way protocol does not have any error correction (bad frames if detected are thrown out).

The protocol only allowing 512 channels per physical line means that a device cannot have more channels than this. This is particularly a problem recently with the advent of complex fixtures which may have many LED's with individual colour control.

The sACN protocol is used to transport DMX512 packets over UDP/IP. DMX512 packets are themselves used to control things within the events industry remotely - primarily lighting equipment.

DMX512 as a protocol doesn't allow non-broadcast communication (every device on the line recieves it) and only allows up to 512 channels to be controlled simultaneously as the size of a DMX packet is

Bibliography

[1] ANSI E1.17 - 2015 Entertainment Technology—Architecture for Control Networks

[2] https://github.com/lschmierer/sacn

[3] ANSI E1.31 — 2018 Entertainment Technology Lightweight streaming protocol for transport of DMX512 using ACN

[4] https://www.element14.com/community/groups/open-source-hardware/blog/2017/08/24/dmx-explained-dmx512-and-rs-485-protocol-detail-for-lighting-applications (17/09/2019)

It generally consists of about two pages of text and must include these four sections: Description The title and a short description of the project aims, context and background. It should explain the big picture of what you would like to achieve, why it is important, and how you intend to go about doing it (e.g. by using some kind of technology or developing a new algorithm, or following a particular methodology, etc.)

Objectives This is a list of clearly defined, measurable goals you intend to achieve by the end of your project. This could include any software artefacts you intend to submit in the end, results of an evaluation (for surveys or research algorithms), etc. Your performance will be measured against these objectives. Typically, you will list about 3-5 primary objectives which are necessary for a project to be deemed successful, and further 3 or so secondary objectives which allow a successful project to be extended in an interesting direction. Occasionally, tertiary objectives may also be listed, but these are comparatively rare. Ethics Here you should discuss any ethical considerations pertaining to your project. Start with the selfassessment form from the Student Handbook (Ethics section). If you can answer "No" to all questions on the self-assessment form, this section of the DOER document will be brief and state that there are no ethical considerations. If you are planning to work with people (especially children), animals, sensitive private data, or if there are other considerations, you should discuss them here, and explain how you went about obtaining necessary approval (any Ethics applications). The self-assessment form and any other relevant documents (if applicable) should be scanned and uploaded to the "Ethics" slot on MMS. 1 2017/18 Resources This is a list of any special resources your project will need: hardware, software, licenses, access to infrastructure (e.g. compute servers), drones, etc. Think ahead, but be realistic – the School will not be able to fulfill all requests. Most projects can be completed using standard school equipment, in which case this section will contain only a short statement confirming this. You and your supervisor will have to agree on everything in the DOER document. Typically, the process looks like this: 1. Schedule a meeting with the supervisor to flesh out the description, objectives, and any needed resources, 2. Write this all up in a word processor, following the structure presented above, 3. Make sure both you and the supervisor agree about the contents (via email or in person), 4. Submit the DOER to MMS.