

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 12

**POVEZIVANJE VIRTUALNOG I STVARNOG  
OSVJETLJENJA KORIŠTENJEM UNREAL SUSTAVA**

Bernard Bačani

Zagreb, lipanj 2021.

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 12

**POVEZIVANJE VIRTUALNOG I STVARNOG  
OSVJETLJENJA KORIŠTENJEM UNREAL SUSTAVA**

Bernard Bačani

Zagreb, lipanj 2021.

## **ZAVRŠNI ZADATAK br. 12**

Pristupnik: **Bernard Bačani (0036515322)**

Studij: Elektrotehnika i informacijska tehnologija i Računarstvo

Modul: Računarstvo

Mentor: doc. dr. sc. Mirko Sužnjević

Zadatak: **Povezivanje virtualnog i stvarnog osvjetljenja korištenjem Unreal sustava**

### Opis zadatka:

Tržište video igara raste velikom brzinom posljednjih godina te se na tržištu pojavio i veliki broj sustava za razvoj igara. Unreal sustav za izradu igara je jedan od trenutno najpopularnijih. Osim za igre ovakvi sustavi se koriste za razvoj aplikacija različitih svrha među kojima je i virtualna produkcija. Koristeći virtualnu produkciju može se olakšati produciranje video uradaka. Primjeri primjene su prikaz virtualnog pozadinskog pejzaža koristeći ekrane umjesto zelenih panela koji su kasnije zamijenjeni digitalnom slikom. Vaš zadatak je proučiti podršku u Unreal sustavu za virtualnu produkciju. Dodatno, potrebno je proučiti protokole za upravljanje osvjetljenjem. Na temelju proučenog materijala potrebno je implementirati upravljanje osvjetljenjem koje će u virtualnoj sceni uspostaviti isto osvjetljenje kao što je i u stvarnosti. Potrebno je dobiveni sustav detaljno opisati i testirati te integrirati s već postojećim sustavom za detekciju položaja kamere koristeći sustav za virtualnu stvarnost. Svu potrebnu literaturu i uvjete za rad osigurat će Vam Zavod za telekomunikacije.

Rok za predaju rada: 11. lipnja 2021.

*Zahvaljujem se mentoru doc. dr. sc. Mirku Sužnjeviću na vodstvu i savjetima, te  
ustupanju svih materijala potrebnih za izradu ovog rada.  
Zahvaljujem i Tiboru Jakovecu na pomoći oko snimanja te što je išao sa mnom u menzu  
i svaki put kasnio.*

# SADRŽAJ

<b>1. Uvod</b>	<b>1</b>
<b>2. Virtualna produkcija</b>	<b>2</b>
2.1. Slučajevi upotrebe . . . . .	3
2.2. Opis prototipa . . . . .	5
<b>3. Upravljanje osvjetljenjem</b>	<b>6</b>
3.1. DMX . . . . .	6
3.1.1. Kontroleri . . . . .	7
3.1.2. Rasvjetni uređaji . . . . .	7
3.1.3. Komunikacija signala . . . . .	7
3.2. Mrežni protokoli . . . . .	9
3.2.1. Art-Net . . . . .	9
3.2.2. sACN . . . . .	11
3.3. GDTF . . . . .	11
<b>4. Model rješenja</b>	<b>12</b>
4.1. Simulacija osvjetljenja . . . . .	12
4.2. Integracija s prototipom . . . . .	12
<b>5. Korištene tehnologije</b>	<b>14</b>
<b>6. Simulacija osvjetljenja</b>	<b>16</b>
6.1. Postavljanje simulatora . . . . .	16
6.2. Povezivanje s Unreal Engineom . . . . .	16
6.3. Simulacija . . . . .	21
6.3.1. Upravljanje simulatorom . . . . .	21
6.3.2. Korištenje sekvencera . . . . .	23

<b>7. Integracija s prototipom</b>	<b>24</b>
<b>8. Rezultati</b>	<b>27</b>
<b>9. Zaključak</b>	<b>29</b>
<b>Literatura</b>	<b>30</b>

# 1. Uvod

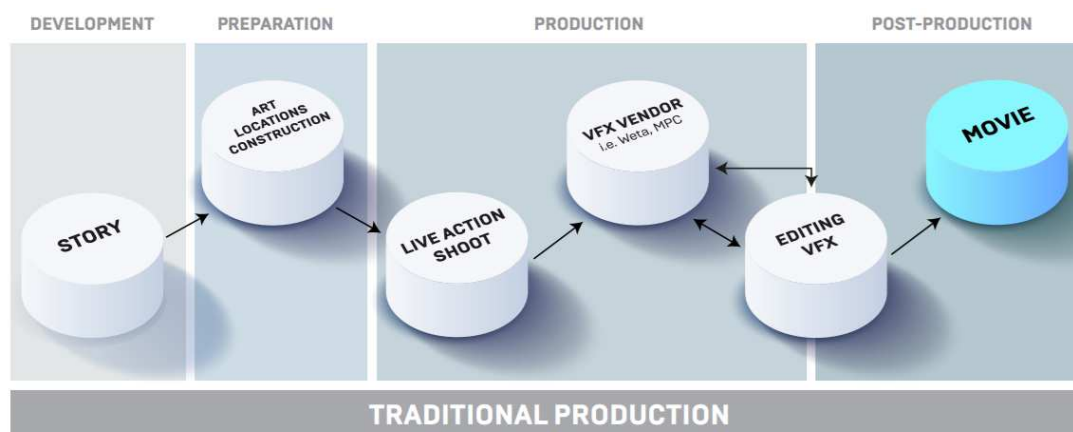
Ogromna popularnost industrije digitalnih igara dovela je do razvoja velikog broja sustava za razvoj igara poput Unreal Enginea. Ovakvi sustavi imaju podršku za upravljanjem virtualnom scenom i objektima u njoj što se pokazalo korisnim u filmskoj industriji. Tako pomoću virtualne produkcije umjesto korištenja zelenih panela moguće je prikazati pozadinski pejzaž i vizualne efekte u stvarnom vremenu, projiciranjem pozadine scene iza glumaca i objekata seta na velikim LED ekranima.

Zadatak ovog rada je proučiti protokole za upravljanje osvjetljenjem i podršku za virtualnu produkciju u Unreal Engineu te zatim implementirati sustav za upravljanje osvjetljenjem koji povezuje virtualno i stvarno osvjetljenje. Za upravljanje osvjetljenjem korišten je simulator upravljačke konzole za osvjetljenje koji šalje signale do Unreal Enginea. Na kraju, sustav se integrira sa sustavom za detekciju položaja kamere.

Rad je strukturiran na sljedeći način. Nakon uvoda, u drugom poglavlju opisana je virtualna produkcija s njezinim slučajevima upotrebe i sklopovlje koje će se koristiti za njezino snimanje. U trećem poglavlju objašnjen je DMX protokol i mrežni protokoli za prijenos DMX podataka podržani u Unreal Engineu te GDTF protokol. Potom u četvrtom i petom poglavlju opisan je model rješenja i korištene tehnologije. Šesto i sedmo poglavlje opisuje izradu simulacije osvjetljenja te povezivanje simulacije s prototipom za virtualnu produkciju. U osmom poglavlju analizirani su rezultati simulacije, a deveto poglavlje je zaključak, nakon čega slijede literatura i sažeci rada na hrvatskom i engleskom jeziku.

## 2. Virtualna produkcija

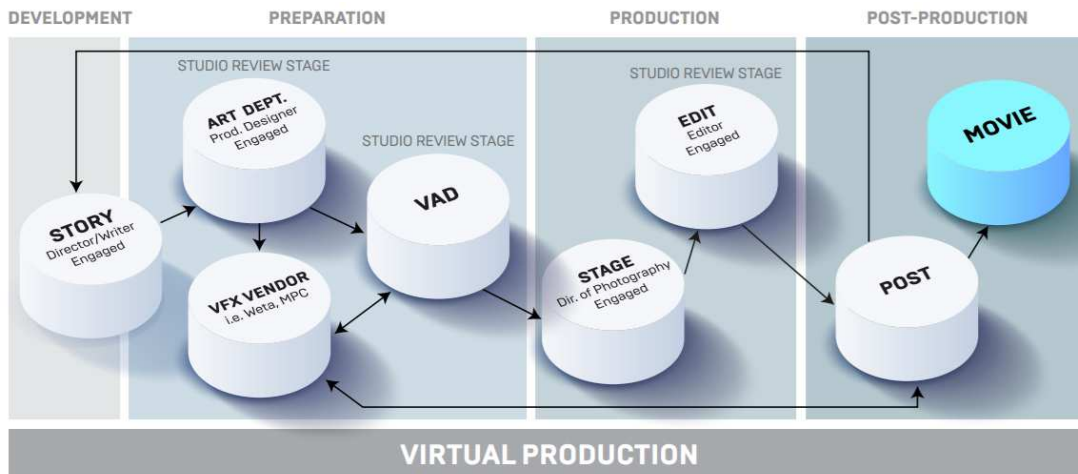
Filmska produkcija je izuzetno kompleksan proces koji je tipično linearan te se sastoji od razvoja, preprodukcije, produkcije i postprodukcije (slika 2.1). Problem tradicionalne produkcije je izoliranost sudionika u pojedinoj fazi i neprilagodljivost promjenama na samom setu. Konačna verzija scene ne zna se do samog kraja produkcije, što uvelike otežava cijeli proces snimanja.



**Slika 2.1:** Tradicionalna produkcija [13]

Virtualna produkcija je metoda slična agilnom razvoju, gdje se korištenjem pogonskog sklopa igara (engl. *game engine*) prati kretanja kamere u stvarnom vremenu te se pozadinski sadržaj prikazuje na LED ekranu. Za razliku od tradicionalne, virtualna produkcija uvodi fleksibilnost u snimanje, omogućujući promjenu priče ili bilo kojeg dijela seta uz prisutnost cijelog tima, čime se štedi na vremenu i resursima (slika 2.2).





Slika 2.2: Virtualna produkcija [13]

## 2.1. Slučajevi upotrebe

Vizualizacija je vjerojatno najčešći oblik upotrebe virtualne produkcije, a posebice predvizualizacija, koja u novoj paradigmi pruža prvi uvid u ideje i sredstva koja će se koristiti kroz produkciju [14]. Poznata uporaba ove tehnike je u seriji *The Mandalorian* (2019.-) gdje je virtualna produkcija bila korištena u snimanju više od pola cijele prve sezone (slika 2.3).

Osim u filmovima i serijama može se koristiti u predvizualizaciji raznih javnih događaja poput koncerata. Primjerice, tvrtka Moment Factory je u 2021. godini objavila projekt *DMX Previs*, digitalni svjetlosni šou (engl. *light show*) napravljen koristeći DMX dodatak (engl. *plugin*) u Unreal Engineu (slika 2.4) koji je sličan primjeru iz stvarnog života te ga se može i upravljati upravljačkom konzolom za osvjetljenje.

Također, virtualna produkcija koristi se za hvatanje pokreta (engl. *motion capture*) - praćenja pokreta objekata ili glumaca za animaciju digitalnih modela i kod vizualnih efekata u kameri (engl. *in-camera visual effects*, skraćeno *in-camera VFX*).



**Slika 2.3:** Set u seriji *The Mandalorian* [10]

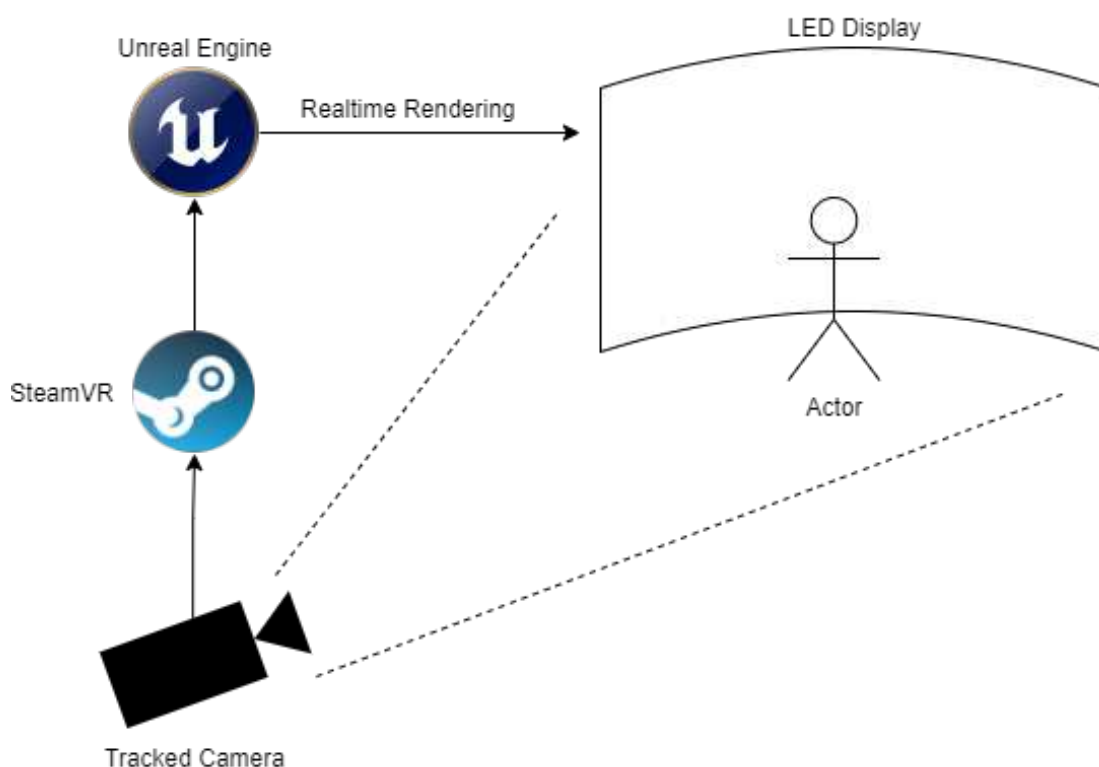


**Slika 2.4:** Svjetlosni šou *DMX Previs* [16]

## 2.2. Opis prototipa

Prototip za snimanje virtualne produkcije se sastoji od HTC Vive sustava s dvije bazne stanice i HTC Vive Trackera nalijepljenog na kameru te LED ekrana računala. Kretanjem kamere HTC Vive Tracker detektira pokrete te se oni šalju iz stvarnog svijeta u virtualni svijet napravljen u Unreal Engineu, što rezultira promjenom virtualne kamere i scene prikazane na LED ekranu.

Bazne stanice su postavljene na zamišljenu dijagonalu prostora gdje se obavlja snimanje te su priključeni na izvor struje, a HTC Vive Tracker je bežično spojen putem USB adaptera (HTC Vive Tracker Dongle) priključenim na računalo kako bi se dobila sloboda kretanja kamere. Navedeno sklopovlje je povezano s računalom pomoću SteamVR aplikacije [17].



**Slika 2.5:** Skica prototipa virtualne produkcije

## 3. Upravljanje osvjetljenjem

Za upravljanje osvjetljenjem u industriji se koristi konzola za upravljanje rasvjetom, elektronički uređaj koji komunicira sa svjetlima, prigušivačima i ostalim specijalnim efektima putem protokola upravljanja. Osim komuniciranja direktno, neke konzole mogu komunicirati s uređajima putem mreže pomoću mrežnih protokola.

### 3.1. DMX

Digital Multiplex (skraćeno DMX) je standardni digitalni komunikacijski protokol koji se uobičajeno koristi za upravljanje svjetlima i svjetlosnim efektima dajući im naredbe poput promjene boje, pozicije i intenziteta. Izvorno namijenjen kao standardna metoda za kontrolu prigušivača svjetlosti, brzo je proširio svoju namjenu na kontrolu ostalih uređaja, poput vatrometa, lasera, mikrokontrolera, inteligentna svjetla, uređaja za maglu itd.

DMX možemo gledati kao paket digitalnih informacija koji se šalju s izvora na odredište pri čemu svaki paket sadrži polje od 512 bajtova čije su vrijednosti između 0 i 255 (slika 3.1).



**Slika 3.1:** DMX paket [5]

### 3.1.1. Kontroleri

DMX kontroleri (engl. *controllers*) djeluju kao izvor DMX paketa te šalju podatke do jednog ili više povezanih uređaja. Postoje dva oblika DMX kontrolera, USB/mrežno sučelje koje pretvara USB signale ili IP pakete u DMX te standardna DMX konzola koja omogućuje ručno aktiviranje slanja DMX paketa. Konzole također mogu imati podršku za komunikaciju preko mreže.

### 3.1.2. Rasvjetni uređaji

DMX rasvjetni uređaji (engl. *fixtures*) su uređaji koji primaju DMX pakete te na temelju primljenih podataka obavljaju određenu naredbu. Naredbe mogu biti razne, obično uključivanje/isključivanje, pojačavanje/smanjivanje intenziteta, promjena boje te rotacija za neki kut.

Svaki rasvjetni uređaj ima niz atributa koji su unaprijed definirani na hardverskoj razini te su organizirani u grupe koje zovemo odama (engl. *odes*). Često uređaji imaju više načina rada (engl. *mode*) kako bi korisnici mogli odabrati samo one funkcije koje im zaista trebaju te time pojednostavili upravljanje. Npr. ako neki uređaj ima ugrađenih 8 funkcija, a način rada mu se postavi na 4 kanalni, tada će samo prve 4 funkcije biti dostupne.

Svemir (engl. *universe*) u DMX-u predstavlja niz spojenih rasvjetnih uređaja koje čitaju iste podatke (pakete). Svaki svemir sadrži 512 adresa te se na jedan svemir može spojiti najviše 512 rasvjetnih uređaja (uz pretpostavku da su svi uređaji 1 kanalni). Dakle, rasvjetni uređaj će zauzimati onoliko adresa u svemiru koliko atributa ima u trenutnom načinu rada (4 kanalni način rada = 4 adrese).

### 3.1.3. Komunikacija signala

Svaki kontroler je zadužen za jedan ili više svemira te u svakom svemiru se nalazi nekoliko rasvjetnih uređaja. Kada kontroler šalje DMX paket, locira svemir kojemu ga treba poslati te šalje isti paket svakom rasvjetnom uređaju u tom svemiru. Paket se šalje slijedno, prvi uređaj pročita podatke u paketu te šalje paket dalje sljedećem. Kako bi uređaj pravilno primio podatke koji se odnose na njega, treba slušati odgovarajuće podatke u paketu, zbog čega je uvedeno adresiranje.

Adresiranje (engl. *fixture patching*) svakom rasvjetnom uređaju dodjeljuje specifičnu početnu adresu u svemiru, čime rasvjetni uređaj tada zauzima raspon adresa od početne adrese do početne adrese plus koliko je atributa u trenutnom načinu rada (slika 3.2).

Universe 1			
<u>Fixture</u>	<u># of Attributes</u>	<u>Starting Address</u>	<u>Address Range</u>
1	7	1	1-7
2	8	8	8-15
3	4	16	16-19
4	10	20	20-29
5	5	30	30-34
6	11	35	35-45

**Slika 3.2:** Adresiranje [5]

Prema primjeru na slici, kada se pošalje DMX paket, rasvjetni uređaj 4 će od 512 adresa u paketu slušati samo one na adresama 20-29, a ostale će zanemariti.

Većinu vremena atributu će biti dovoljan opseg unosa od jednog bajta (0-255), ali ako želimo veću kontrolu te jedan bajt nije dovoljan, atribut može zauzeti više adresa u svemiru. Primjerice, atribut koji zauzima dvije adrese može poprimiti vrijednosti u rasponu 0-65,536. Ostali primjeri:

1. 8-bitni atribut - Min: 0, Max: 255 - zauzima 1 adresu
2. 16-bitni atribut - Min: 0, Max: 65,536 - zauzima 2 adrese
3. 24-bitni atribut - Min: 0, Max: 16,777,215 - zauzima 3 adrese
4. 32-bitni atribut - Min: 0, Max: 4,294,967,296 - zauzima 4 adrese

## 3.2. Mrežni protokoli

Porastom kompleksnosti i broja uređaja korištenih u predstavama pojavila se potreba za lakšim i bržim adresiranjem i upravljanjem rasvjetnim uređajima. Napretkom komunikacijskih tehnologija razvijeni su ethernet protokoli koji omogućuju prijenos više DMX svemira putem Cat5 kabla. Unreal Engine podržava dva ethernet protokola, Art-Net i sACN.

### 3.2.1. Art-Net

Art-Net je ethernet protokol temeljen na TCP/IP protokolnom složaju te se koristi za prijenos velikog broja DMX paketa putem UDP protokola. Najnovija revizija je Art-Net 4 i podržava prijenos 32,768 svemira (teoretski, stvarni broj ovisi o fizičkom sloju mreže i metodi prijenosa) i preko 1000 DMX priključaka (engl. *ports*), gdje je adresa priključka svakog svemira kodirana kao 15-bitni broj (slika 3.3).

Bit 15	Bits 14-8	Bits 7-4	Bits 3-0
0	Net	Sub-Net	Universe
Port-Address			

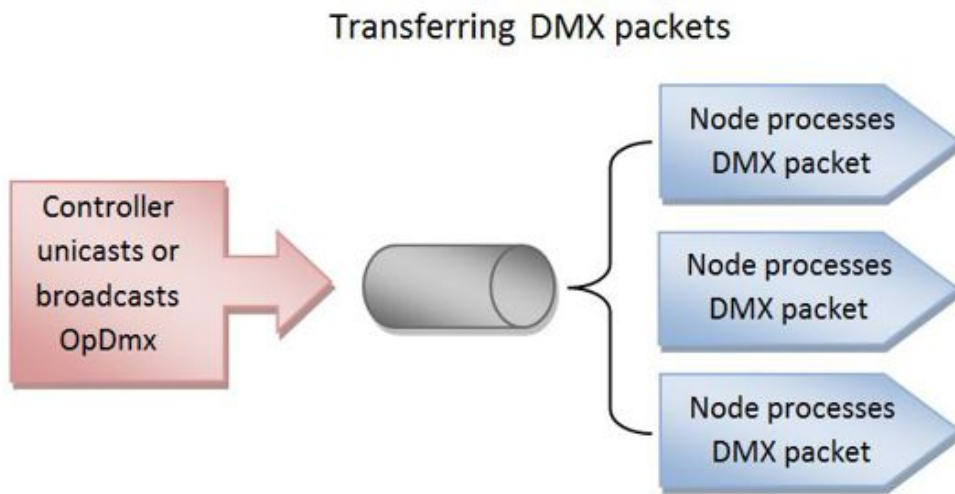
Slika 3.3: Adresa priključka svakog DMX svemira [1]

Podmreža (engl. *Sub-Net*) je grupa od 16 uzastopnih svemira, a mreža (engl. *Net*) je grupa od 16 uzastopnih podmreža ili 256 uzastopnih svemira.

Kod prvog spajanja mreže, kontroleri ne znaju broj uređaja u mreži ni njihove IP adrese te se za njihovo otkrivanje koristi *broadcast* adresa - kontroler pošalje *ArtPoll* paket na IP adresu 2.255.255.255 na UDP priključku 0x1936. *ArtPoll* paket šalje samo kontroler, a odgovor *ArtPollReply* šalju i kontroleri i uređaji. Maksimalno vrijeme čekanja između slanja *ArtPoll* paketa i primanja *ArtPollReply* paketa je 3 sekunde. Kontroler koji je poslao *ArtPoll* paket bi također trebao odgovoriti na svoju poruku sa *ArtPollReply*, čime se osigurava da bilo koji drugi kontroleri koji slušaju na mreži detektiraju sve uređaje bez potrebe da svaki spojeni kontroler šalje *ArtPoll* paket. Svi kontroleri moraju poslati *ArtPoll* svakih 2.5 do 3 sekunde, čime se lako otkriva prekid veze.



Za prijenos DMX podataka koristi se *unicast* ili *broadcast* adresa te se šalje *ArtDmx* podatkovni paket (slika 3.4).



**Slika 3.4:** Prijenos DMX paketa [20]

Iako se za slanje *ArtDmx* paketa može koristiti *broadcast* adresa, u velikom sustavu dolazi do ogromnog opterećenja obrade podataka te se koristi *unicast* adresa. Kako bi se mogla koristiti *unicast* adresa, kontroleri trebaju znati koji uređaji slušaju na određenom priključniku, što se ostvaruje preplatom (engl. *subscription*). Pretplata se odvija na sljedeći način: [21]

1. Kontroler na *broadcast* adresu šalje *ArtPoll* svakih 2.5 do 3 sekunde.
2. Kontroler analizira sve *ArtPollReply* pakete i sastavlja popis do 8 mogućih priključnih adresa i pridružene IP adrese za svaki uređaj koji odgovori (pretplatnici).
3. Kontroler koristi ovaj popis za izračunavanje IP adrese koju bi trebao koristiti za slanje *ArtDmx* paketa na *unicast* adresu.

*ArtSync* paket koristi se za sinkronizaciju slanja *ArtDmx* paketa. Sinkronizacija se odvija tako da kontroler na *unicast* adrese svemira šalje *ArtDmx* pakete, a zatim na *broadcast* adresu pošalje jedan *ArtSync* paket koji signalizira da je zadnji *ArtDmx* paket stigao na odredište.

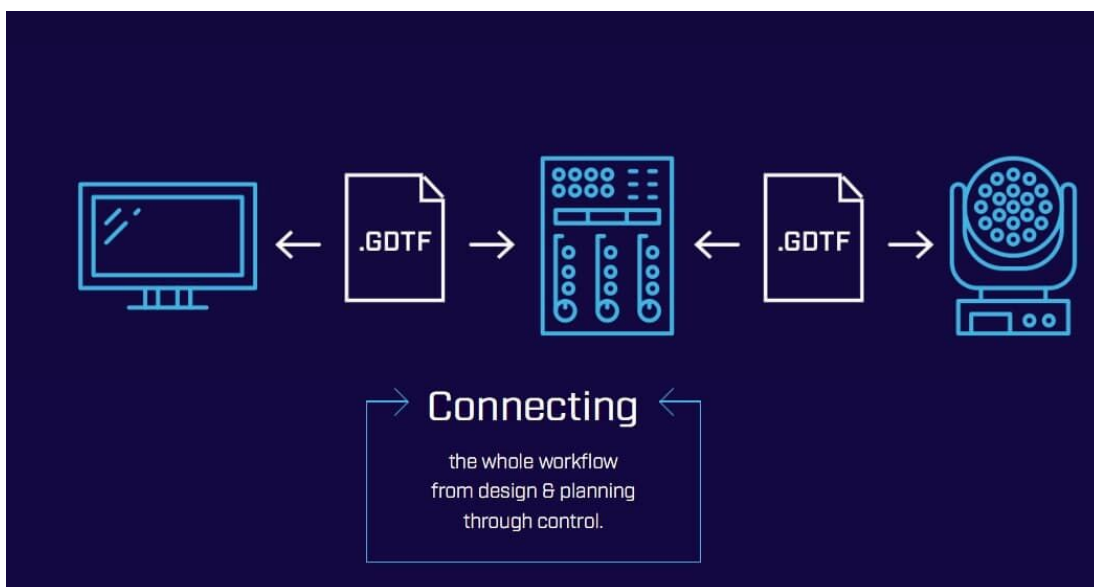


### 3.2.2. sACN

Arhitektura strujanja za upravljačke mreže (engl. *Streaming Architecture for Control Networks*, skraćeno sACN) je standardni protokol razvijen za efikasni prijenos DMX svemira preko mreže. Konceptualno je veoma usporediv s Art-Netom te se sastoji od podatkovnog paketa (*E1.31 Data Packet*) za prijenos DMX podataka, paketa otkrivanja (*E1.31 Universe Discovery Packet*) za praćenje prometa i sinkronizacijskog paketa (*E1.31 Synchronization Packet*). Za razliku od Art-Neta, podržava prijenos do 63,999 svemira i *multicast* adrese, čime se mogu prenositi podaci bez prethodnog znanja topologije mreže [9].

### 3.3. GDTF

*General Device Type Format* (skraćeno GDTF) protokol je otvoreni standard razvijen kako bi se olakšala primjena novih rasvjetnih uređaja (slika 3.5). GDTF datoteka je otvorenog formata te se sastoji od XML datoteke koja sadrži informacije o pojedinom uređaju poput geometrije, atributa, načina rada i pretpostavljenih vrijednosti [4]. Baza podataka svih dostupnih GDTF datoteka je besplatna [11].



Slika 3.5: GDTF [11]

## 4. Model rješenja

Upravljačka konzola za osvjetljenje je uređaj kojim se upravlja svjetlosnim i drugim uređajima putem upravljačkog protokola. Zbog povećane količine uređaja na pozornici, osim komuniciranja direktno ili preko USB-a, neke konzole također mogu komunicirati putem mreže kako bi se olakšala kontrola nad složenim sustavima. Za ovaj rad korišten je simulator konzole koji šalje podatke preko ethernet protokola.

Svjetlosni efekti korišteni u simulaciji su nalik na one u javnim događajima, poput koncerata gdje se koriste primjerice reflektori, laseri, dimni efekti i drugo. Zatim, simulacija se integrira s postojećim sustavom za detekciju položaja kamere u kojem se pomoću stvarne kamere upravlja virtualnom kamerom u stvarnom vremenu.

### 4.1. Simulacija osvjetljenja

Simulacija je zamišljena kao predvizualizacija mini pozornice u kazalištu. Pozornica sadrži nekoliko reflektora, statičnih svjetala i lasera. Svjetlosnim uređajima se upravlja simulatorom konzole te se svjetla pale i gase, mijenja im se pozicija, pojačava ili smanjuje intenzitet i slično. Promatra se slanje naredbi korištenjem konzole u stvarnom vremenu te unaprijed programiran skup naredbi koji se izvodi u sekvenceru. Simulacija je uspješna ako je dobiven rezultat sličan korištenju prave konzole te je utjecaj kašnjenja minimalan.

### 4.2. Integracija s prototipom

Povezivanje s prototipom ostvaruje se korištenjem stvarne kamere, sustava za virtualnu stvarnost i LED zaslona. Kako se simulacija izvodi, kretanjem stvarne kamere mapiraju se pokreti u virtualnu kameru te se time krećemo po pozornici. Ispred LED zaslona stavljaju se objekti u stvarnom životu te se snima smješten objekt

u virtualnoj sceni. Integracija je uspješna ako se dobije dojam da se objekt stvarno nalazi na pozornici, a simulirana svjetlost nalikuje stvarnoj svjetlosti.

## 5. Korištene tehnologije

**Unreal Engine** [22] je pogonski sklop igre (engl. *game engine*) prvenstveno namijenjen za razvoj digitalnih igara za PC. Može se koristiti za izradu dvodimenzionalnih i trodimenzionalnih igara, aplikacija za virtualnu i proširenu stvarnost te za razvoj virtualnih setova korištenih u filmskoj industriji. Skripte se pišu programiranjem u jeziku C++ ili korištenjem vizualnog sustava skriptiranja *Blueprints*. Za izradu i testiranje simulacije korišten je Unreal Engine 4.26.2.

**MagicQ Software** [15] je programsko sučelje za vizualizaciju i kontrolu osvjetljenja do 256 svemira (64 svemira za PC verziju). Omogućuje slanje DMX paketa preko mreže do aplikacija na računalu korištenjem Art-Net, Pathport, sACN i drugih protokola. Korisničko sučelje i funkcionalnosti su jednake kao i na fizičkoj konzoli te je moguć međusobni prijenos podataka preko mreže ili USB-a. Za izradu i testiranje simulacije korišten je MagicQ v1.8.9.0.

**SteamVR** [19] je alat za pokretanje VR sadržaja na sustavima za virtualnu stvarnost poput HTC Vive, Oculus Rift i drugi. Koristi *Lighthouse* sustav, laserski sustav za praćenje položaja i orijentacije korisnika u stvarnom vremenu te *Chaperone* sustav za postavljanje i pregled granica prostora u sobi kako bi se spriječilo kretanje izvan dosega senzora te zaletavanje u zidove i druge predmete. Za testiranje simulacije korišten je SteamVR 1.16.10.

**HTC Vive** [12] je sustav za virtualnu stvarnost koji omogućuje korisniku kretanje u 3D prostoru te interakciju s okolinom korištenjem ručnih kontrolera. Osim *headseta*, kontrolera i baznih stanica može se koristiti dodatak HTC Vive Tracker za praćenje pozicije predmeta na koji je pričvršćen preko *Lighthouse* sustava. Za testiranje simulacije korišten je HTC Vive Tracker, HTC Vive Tracker Dongle, HTC Vive Tracker Dongle Cradle i dvije bazne stanice.

**DMX Template** [7] je demo projekt u Unreal Engineu s već omogućenim svim DMX dodacima te sadrži nacрте (engl. *blueprints*) za DMX rasvjetne uređaje. Nacrti sadrže komponente koje su odgovorne za primanje DMX paketa. Projekt također sadrži tri mape s unaprijed snimljenim sekvencama.

## 6. Simulacija osvjetljenja

### 6.1. Postavljanje simulatora

Slanje DMX paketa preko mreže ostvareno je pomoću *ChamSys MagicQ* simulatora. Stvoren je novi šou u načinu programiranja *Normal (live)* te kako bi se paketi mogli poslati lokalno na aplikacije na računalu, potrebno je otići u *Setup* te u izborniku *Network* postaviti opciju *Send to Applications on this PC* na *Yes*. Dodatno, u *Setup* u izborniku *View DMX I/O* potrebno je omogućiti svemire kojima želimo upravljati postavljanjem opcije *Status* na *Enabled*. Za potrebe ovog rada omogućeni su svemiri 1 i 2. Ovdje se također mogu mijenjati ostale opcije poput imena svemira, protokola za prijenos paketa te način slanja paketa. Unreal Engine trenutno podržava samo Art-Net i sACN protokole te je ostavljena zadana vrijednost, odnosno Art-Net.

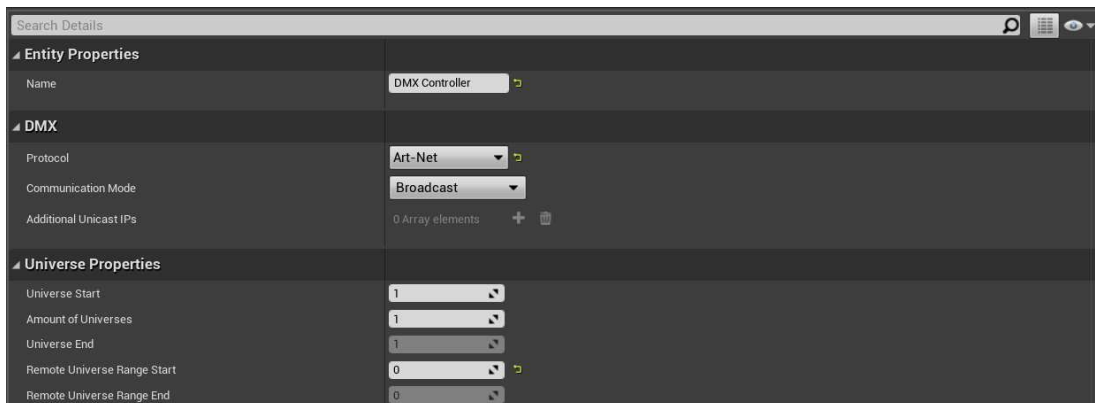
Rasvjetne uređaje povezujemo u sustav u izborniku *Patch*. Uređaj se dodaje odabirom *Choose Head* čime se može izabrati uređaj iz postojeće kolekcije uređaja koji dolaze instalirani sa simulatorom ili se može napraviti vlastiti odabirom *Edit Head > New Head*. Za testno povezivanje odabran je uređaj *Clay Paky Sharpy* u standardnom načinu rada (16-kanalni). Nakon što je odabran uređaj, klikne se na *Patch It* te se upiše koliko uređaja će se spojiti. Pojavljuje se iskočni prozor s opcijom umetanja uređaja u *MagicQ* vizualizator, što se odbija jer se simulator koristi samo za slanje DMX paketa do Unreal sustava [3].

### 6.2. Povezivanje s Unreal Engineom

Napravljen je novi prazan (engl. *Blank*) projekt te su uključeni dodaci (engl. *plugins*) za DMX podršku, *DMX Engine*, *DMX Fixtures*, *DMX Pixel Mapping* i *DMX Protocol*. Dodatak *DMX Fixtures* je trebao uvesti osnovne sheme, ali pošto nije uspio ručno je uvezena datoteka *BP\_MovingHead.uasset* kopiranjem s lokacije

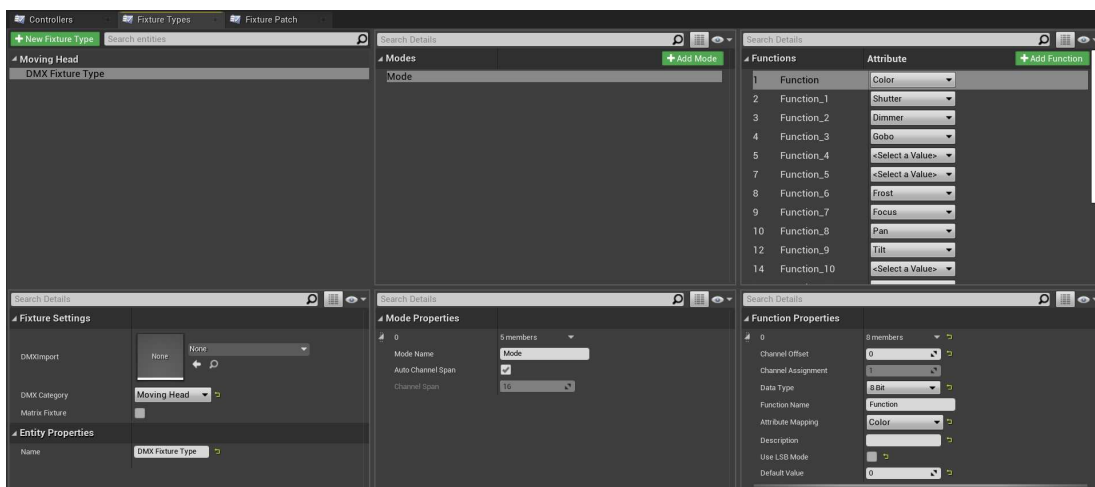
C:\ProgramFiles\EpicGames\UE\_4.26\Engine\Plugins\  
VirtualProduction\DMX\DMXFixtures\Content\LightFixtures  
u *Content* direktorij našeg projekta.

Nakon toga, potrebno je kreirati *DMX Library* odabirom *Add/Import > DMX > DMX Library*. U izborniku *Controllers* klikom na gumb *New Controller* doda se novi DMX Controller te mu se uređuju svojstva (slika 6.1). *Remote Universe Range Start* postavljen na 0 jer *Output Universe* u *MagicQ* simulatoru počinje s indeksom 0 [18].



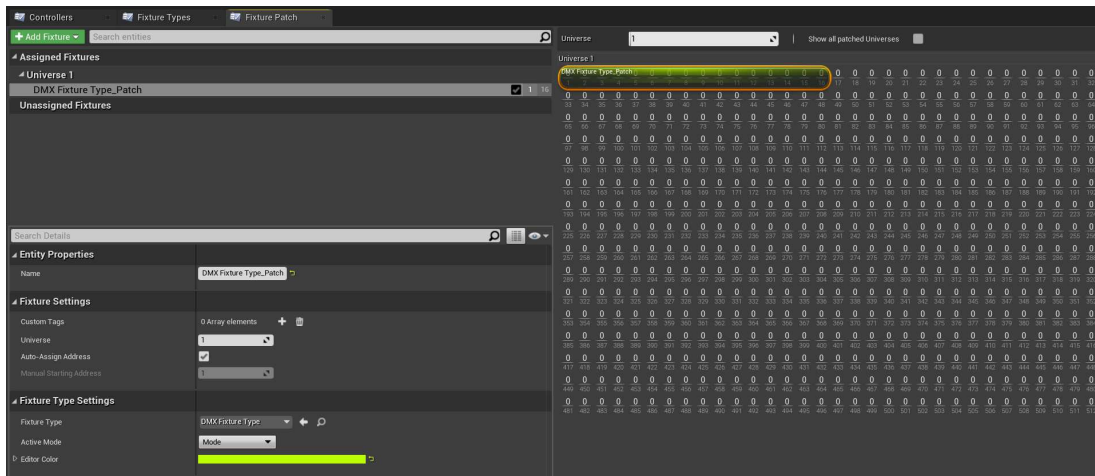
Slika 6.1: Postavke kontrolera

Zatim u izborniku *Fixture Types* klikom na gumb *New Fixture Type* doda se nova vrsta rasvjetnog uređaja. Postavke uređaja mogu se postaviti uvozom GDTF datoteke ili ručno. U *Modes* i *Mode Properties* pregledniku postavlja se način rada, a u *Functions* i *Function Properties* pregledniku postavljaju se funkcije uređaja (slika 6.2). Ako neka funkcija zauzima dva ili više kanala, *Data Type* se postavi na odgovarajući broj bitova (jedan kanal zauzima jedan bajt).



Slika 6.2: Postavke uređaja

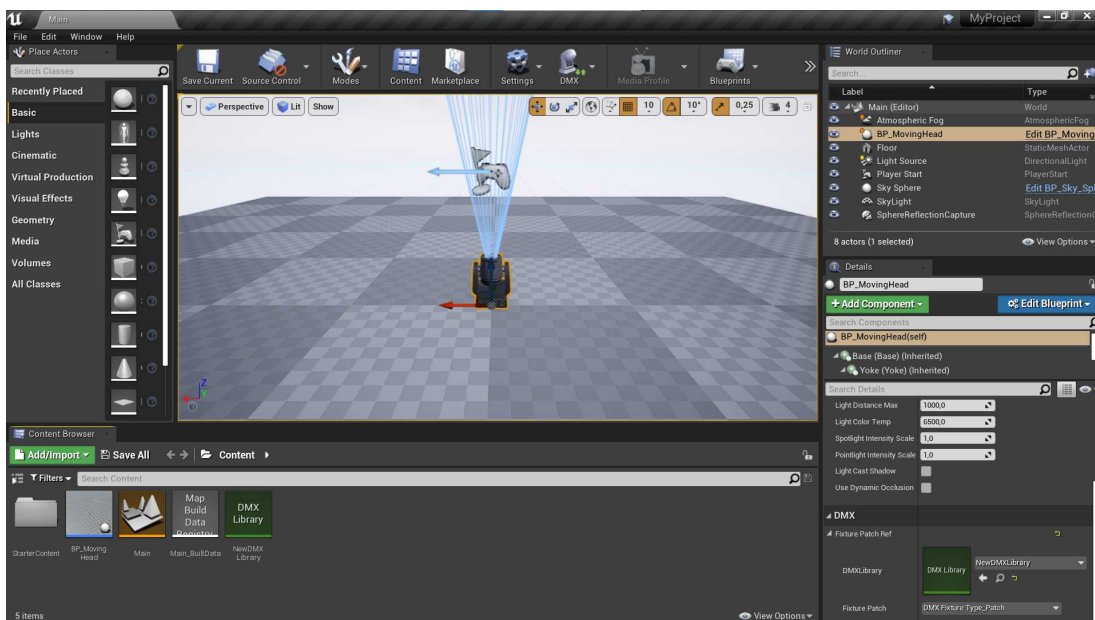
Na kraju, potrebno je obaviti adresiranje u izborniku *Fixture Patch*. Klikom na gumb *Add Fixture* odabere se novokreirani uređaj te mu se postave svemir i adrese (slika 6.3).



Slika 6.3: Adresiranje uređaja

Pregled svih dolaznih DMX paketa za pojedini svemir može se vidjeti u *DMX Channel Monitoru* ili u *DMX Activity Monitoru* [6].

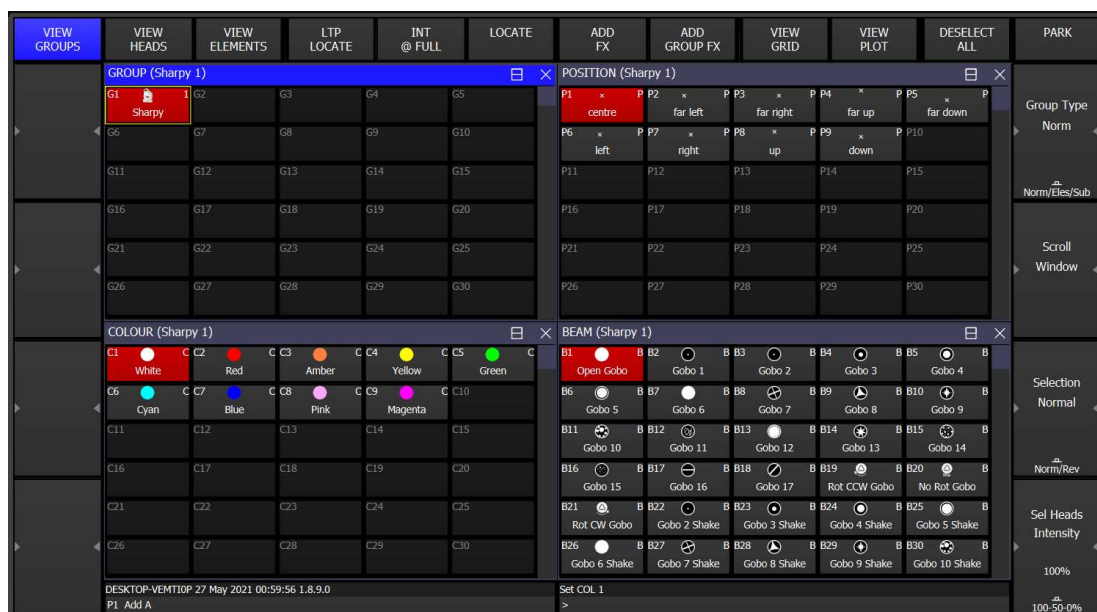
Za testiranje je ubačen *BP\_MovingHead* u scenu te mu je unutar *Details* izbornika postavljen *DMX Library* i *Fixture Patch* (slika 6.4).



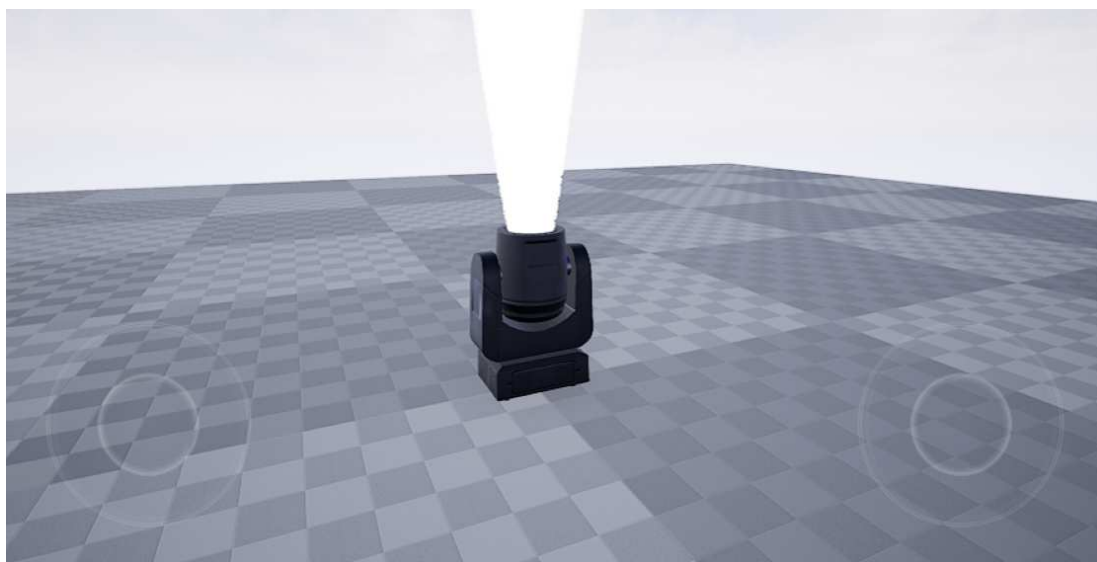
Slika 6.4: Scena i DMX postavke pokretne glave

Nakon pokretanja scene, korištenjem simulatora šalju se DMX paketi do Unreal Enginea što uzrokuje upravljanje pokretnom glavom (slike 6.5 do 6.8).





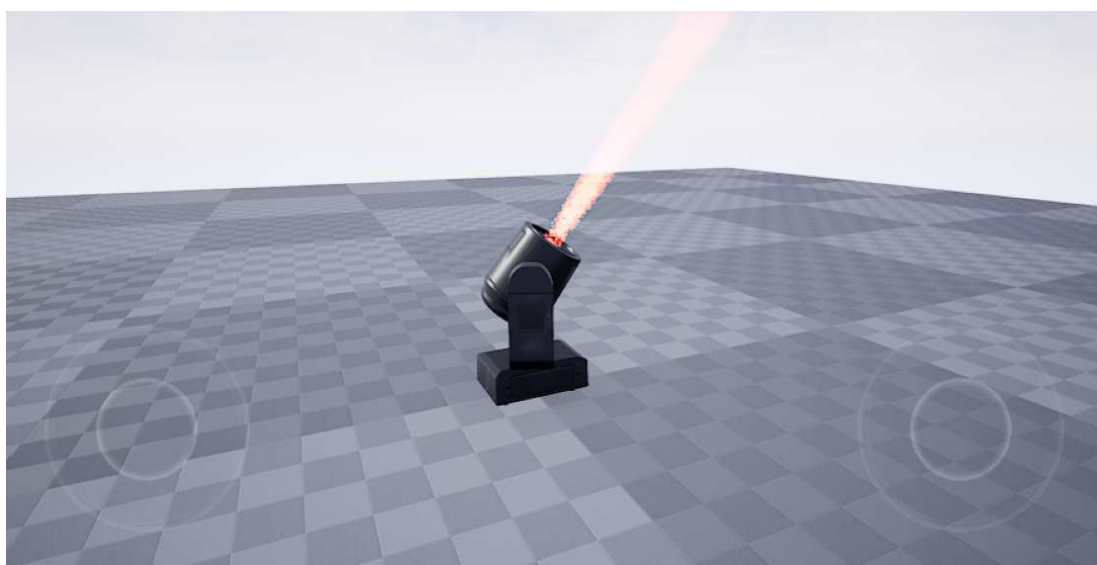
Slika 6.5: Početno stanje simulatora



Slika 6.6: Početno stanje pokretne glave



**Slika 6.7:** Promjena pozicije, boje i zrake



**Slika 6.8:** Promjena stanja pokretne glave

## 6.3. Simulacija

Za simulaciju je korišten *DMX Template* koji već sadrži podešen cijeli *DMX Library* i posložene rasvjetne uređaje u sceni. Potrebno je samo promijeniti *Remote Universe Range Start* s 1 u 0 kako bi se postigla kompatibilnost sa *MagicQ* simulatorom.

### 6.3.1. Upravljanje simulatorom

U *MagicQ* simulatoru za svaku vrstu rasvjetnog tijela u sceni stvori se nova glava (engl. *Head*) te se dodijele jednake funkcije i početne vrijednosti. Nakon kreiranja, uređaje adresiramo na iste adrese kao i u *DMX Libraryju*. Na kraju dobivamo grupe uređaja kojima možemo upravljati slanjem DMX paketa (slika 6.9).

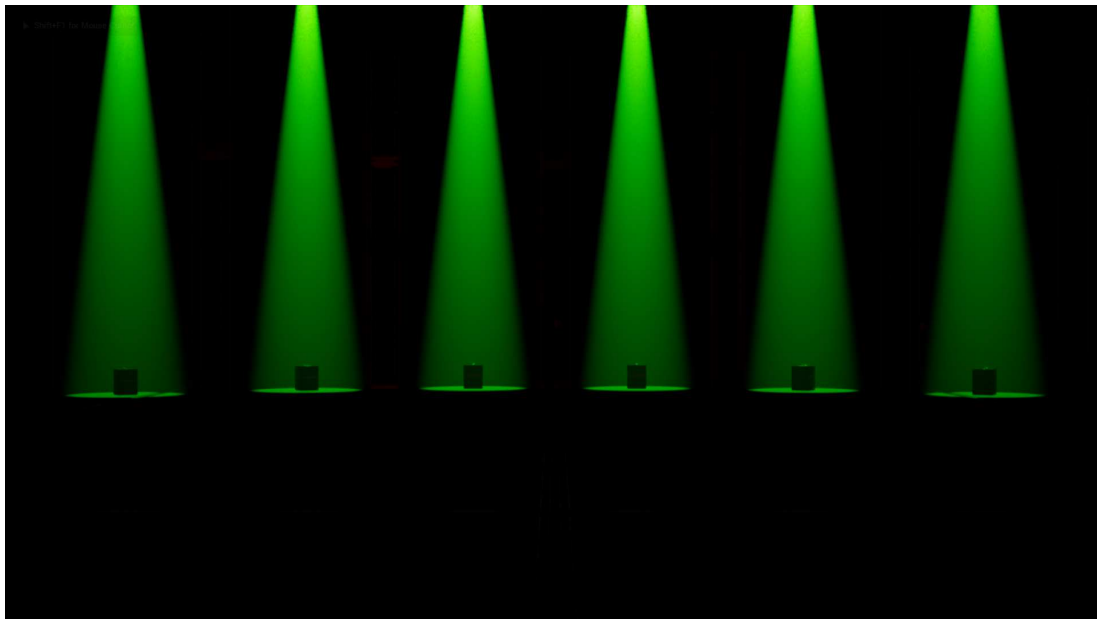


G1 Heads 8	G2 Mirrors 2	G3 Static 9	G4 Strobe 6	G5 LEDWash 2
G6	G7	G8	G9	G10
G11	G12 Pyro 2	G13 Laser 2	G14 Fireworks 3	G15
G16	G17	G18	G19	G20

Slika 6.9: Grupe uređaja

Primjerice, odabirom grupe *Heads* svim pokretnim glavama na svoj prvi kanal (kanal za boju) pošalje se vrijednost 45, na treći kanal (kanal za intenzitet) pošalje se vrijednost 255 te na sedmi i deveti kanal pošalje se vrijednost 128 (kanali za nagib i pomicanje) što rezultira upravljanjem pokretnim glavama u Unreal Engineu kao na slici 6.10.

Drugi primjer, odabirom grupe *Pyro* te slanjem vrijednosti 128 rezultira paljenjem pirotehničkih efekata (slika 6.11). Na ovaj način se može upravljati cijelom setom u stvarnom vremenu.



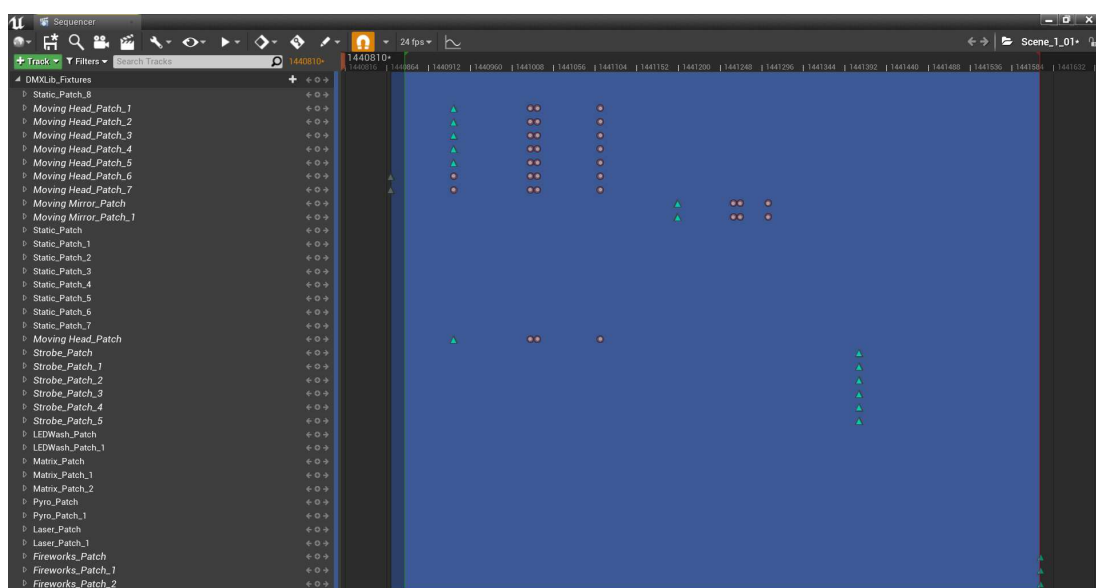
**Slika 6.10:** Upravljanje pokretnim glavama



**Slika 6.11:** Paljenje pirotehnike

### 6.3.2. Korištenje sekvencera

Poslani DMX paketi mogu se snimiti, uređivati i reproducirati pomoću sekvencera (engl. *sequencer*). Za snimanje dolaznih DMX podataka koristi se alat *Take Recorder* te se kao izvor odabere *DMX Library* i njegovi uređaji želimo snimati. Nakon toga, pokreće se scena i pritisne *Record* za snimanje te se na kraju snimka spremi na `/Game/Cinematics/Takes/[RecordDate]/`. Kako bi se omogućilo uređivanje sekvence, potrebno ju je otključati klikom na lokot (*Unlock the animation so that it is editable*) u gornjem desnom kutu sekvencera [8].



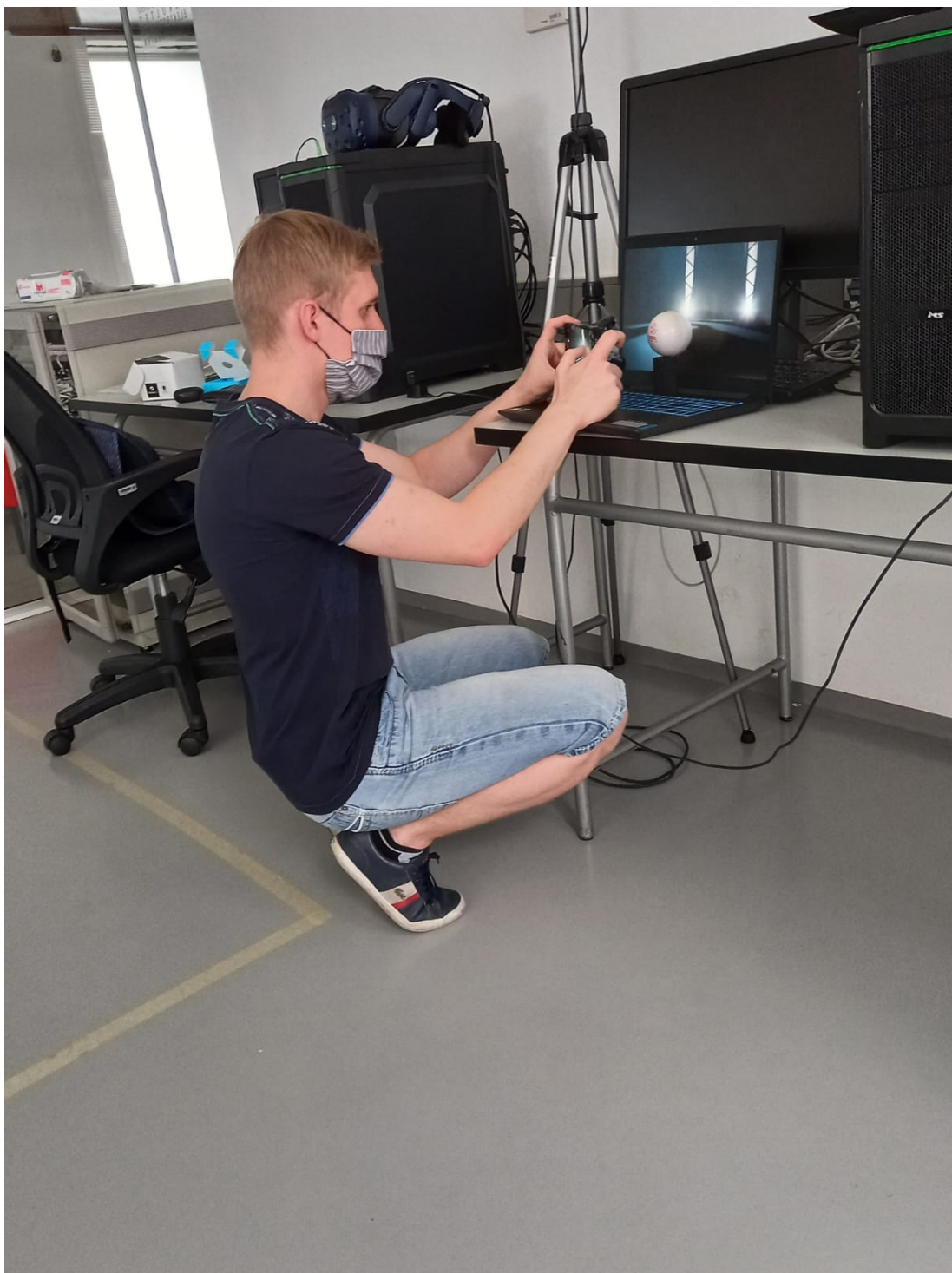
Slika 6.12: Sekvencer

## 7. Integracija s prototipom

Povezivanje prototipa s Unreal sustavom ostvareno je korištenjem SteamVR aplikacije unutar Steam okruženja i SteamVR dodatka u Unreal Engineu. SteamVR aplikaciju potrebno je prilagoditi za snimanje bez headseta (engl. *head-mounted display*, skraćeno HMD) omogućavanjem i aktivacijom SteamVR *null driver*a. Nakon toga, tracker se bežično povezuje s računalom preko USB adaptera koji je priključen na korišteno računalo kako bi se dobila sloboda kretanja kod snimanja [17].

HTC Vive Tracker je potrebno upariti unutar SteamVR aplikacije čime ako je uspješno statusna lampica na trackeru postane zelena. Zatim, u Unreal Engineu je stvoren novi pijun (engl. *Default Pawn*) te mu je kao dijete dodan *Cine Camera Actor* koji u sebi sadrži komponente *SceneComponent* i *CameraComponent*. Nad virtualnom kamerom se stvara nacrt (engl. *blueprint*) koji prati pokrete HTC Vive Trackera te prenosi informacije o njegovom položaju. Također, potrebno je urediti *Level Blueprint* kako bi novostvorena kamera postala glavna kamera. Na kraju, tracker se pričvršćuje na mobitel tako da je statusna lampica na njemu okrenuta prema gore te je usmjeren prema LED ekranu [17].

Ispred LED ekrana postavljaju se objekti iz stvarnog života te se može pokrenuti snimanje pritiskom na *Play* u Unreal Engineu, čime se pokreće snimljena sekvenca *LS FixtureDemo* dostupna u projektu *DMX Template*. Može se pokrenuti bilo koja druga sekvenca ili samo scena bez sekvence pa se upravljanje scenom može ostvariti korištenjem *MagicQ* simulatora konzole. Kretanjem u prostoru detektiraju se pokreti stvarne kamere što se prenosi do virtualne kamere te se dobiva efekt virtualne produkcije (slika 7.1). Kretanje je ograničeno veličinom LED ekrana na kojem je prikazan virtualni svijet i veličinom prostora u kojem se snima, odnosno prostora koji pokrivaju dostupne bazne stanice.

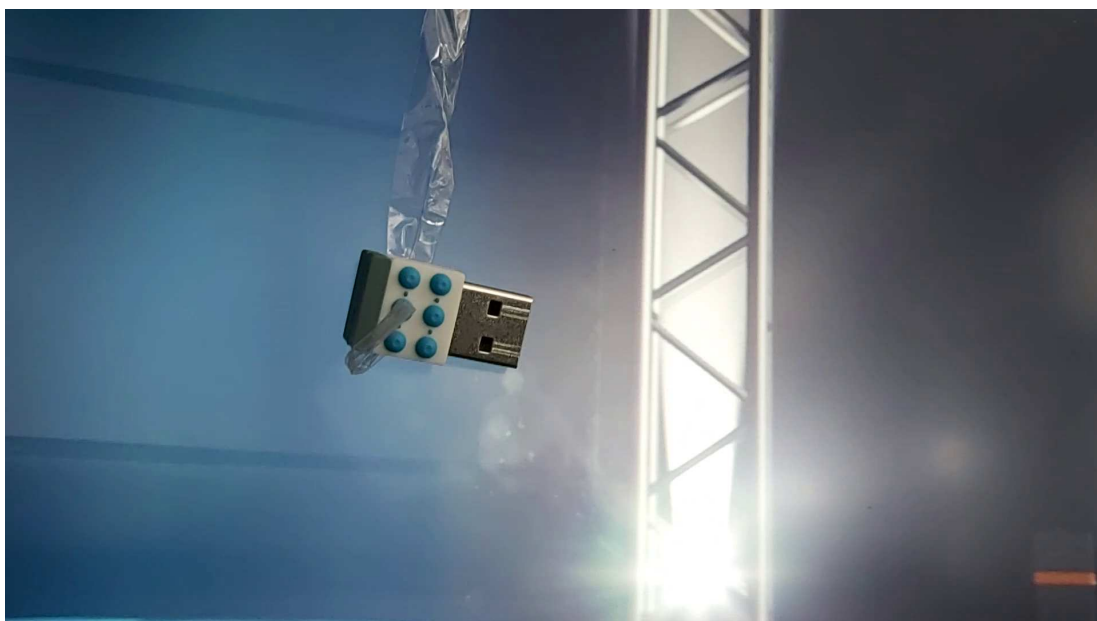


**Slika 7.1:** Snimanje videa pomoću metode virtualne produkcije





**Slika 7.2:** Snimka statičnog predmeta



**Slika 7.3:** Snimka dinamičnog predmeta



## 8. Rezultati

Podrška za virtualnu produkciju i DMX je u trenutku pisanja ovog rada u eksperimentalnoj fazi te se pojavljuju tehničke poteškoće. Pri postavljanju postavki simulatora ako se radi na Windows računalo potrebno je pokrenuti MagicQ kao administrator jer inače promjene neće biti spremljene zbog ograničenja ovlasti nad datotekama. Nadalje, iako je Chamsys bio prvi veliki proizvođač konzola s podrškom za uvoz GDTF datoteka [2], kod svakog pokušaja se pojavila pogreška *Could not read GDTF file*. Ručno stvaranje rasvjetnih uređaja u MagicQ simulatoru nije u potpunosti kompatibilno s Unreal Engineom te se neke funkcije mogu različito zvati ili uopće ne postojati.

Slanje DMX paketa se uspješno provodi, a upravljanje scenom je jednostavno i vrlo prilagodljivo s mogućnošću delikatnog upravljanja jednog uređaja ili grupe uređaja. Virtualna svjetla se ponašaju identično kao i stvarna svjetla te im se mogu mijenjati boja, intenzitet, svjetlina, položaj i orijentacija jednako kao i odgovarajući rasvjetni uređaji u stvarnom svijetu. Sekvencer se pokazao kao izvrstan alat za snimanje, uređivanje i reproduciranje DMX paketa te se pomoću njega može dizajnirati cijeli svjetlosni šou za predvizualizaciju. Trenutno DMX snimanje radi samo tijekom izvođenja [8].

Unreal Engine osim pogreške u nemogućnosti uvoza nacрта i ostalih objekata iz DMX dodataka nije predstavljao problema za korištenje. Uspješno uvozi GDTF datoteke, ali trenutno je podržan uvoz samo XML podataka s informacijama o načinima rada, funkcijama i pretpostavljenim vrijednostima, ali planiran je uvoz tekstura i modela uređaja [4]. Povezivanje sa SteamVR aplikacijom nije bilo pouzdano te je HTC Vive Tracker prepoznat samo na jednom od dva USB porta. Također, kako bi se sinkronizirala stvarna kamera s trackerom i virtualna kamera potrebno je prvo pokrenuti SteamVR, a tek nakon toga projekt u Unreal Engineu.

Kod snimanja videa prostor je potrebno zamračiti kako bi se spriječio odsjaj vanjskog svjetla na LED ekranu. Pri kretanju potrebno je paziti da kamera ne izađe iz okvira ekrana jer inače se gubi iluzija prisutnosti stvarnog objekta u virtualnom prostoru. Virtualna produkcija koristi puno računalnih resursa te u slučaju pokretanja na laptopu nakon 10 minuta dolazi do pregrijavanja zbog čega je potrebno pričekati par minuta prije ponovnog snimanja.

## 9. Zaključak

U ovom radu proučena je podrška za virtualnu produkciju kao i podrška za upravljanje osvjetljenjem DMX protokolom u Unreal Engineu. Za testiranje je korištena predvizualizacija svjetlosnog u čijoj sceni su sadržani različiti rasvjetni uređaji. Upravljanje osvjetljenjem ostvareno MagicQ simulatorom konzole koji šalje DMX pakete preko Art-Net mrežnog protokola do Unreal Enginea.

Povezivanje simulatora konzole i scene se trebalo raditi ručno zbog nemogućnosti čitanja GDTF datoteka u simulatoru što je inače odlično rješenje za ubrzavanje povezivanja te rješava problem kompatibilnosti za razliku od ručnog unosa postavki uređaja. Također, u budućnosti će Unreal Engine imati podršku za uvoz tekstura i modela uređaja iz GDTF datoteke što je veoma poželjna funkcionalnost.

Integracija s prototipom trenutno ima tehničkih poteškoća kod povezivanja sa SteamVR aplikacijom, ali nakon postavljanja snimanje se odvija bez problema uz ograničeno kretanje zbog veličine ekrana i prostora koji pokrivaju bazne stanice VR sustava te se virtualna svjetlost u predvizualizaciji ponaša kao stvarna svjetlost.

# LITERATURA

- [1] Artistic Licence, *Specification for the Art-Net 4 Ethernet Communication Protocol*, v1.4dd2, 01.02.2021.,  
URL: <https://artisticlicence.com/WebSiteMaster/User%20Guides/art-net.pdf>,  
pristupljeno 10.06.2021.
- [2] Baroulette, S., *ChamSys MagicQ Software Now Supports Import Of GDTF Personality Files*, 02.12.2019.,  
URL: <https://chamsyslighting.com/blogs/news/chamsys-magicq-software-now-supports-import-of-gdtf-personality-files>, pristupljeno 10.06.2021.
- [3] ChamSys, *MagicQ User Manual*, v1.9.1.0, 08.04.2021.,  
URL:  
<https://secure.chamsys.co.uk/help/documentation/magicq>,  
pristupljeno 10.06.2021.
- [4] DMX GDTF Support,  
URL: <https://docs.unrealengine.com/en-US/WorkingWithMedia/DMX/GDTFSupport>,  
pristupljeno 10.06.2021.
- [5] DMX Overview,  
URL: <https://docs.unrealengine.com/en-US/WorkingWithMedia/DMX/Overview>,  
pristupljeno 10.06.2021.
- [6] DMX Quickstart,  
URL: <https://docs.unrealengine.com/en-US/WorkingWithMedia/DMX/Quickstart>,  
pristupljeno 10.06.2021.

- [7] DMX Template Overview,  
URL: <https://docs.unrealengine.com/4.26/en-US/WorkingWithMedia/DMX/TemplateOverview>,  
pristupljeno 10.06.2021.
- [8] DMX Tracks,  
URL: <https://docs.unrealengine.com/4.26/en-US/WorkingWithMedia/DMX/Tracks>,  
pristupljeno 10.06.2021.
- [9] Entertainment Services and Technology Association - ESTA, *ANSI E1.31 - 2018, Entertainment Technology Lightweight streaming protocol for transport of DMX512 using ACN*, 07.11.2018.,  
URL: [https://tsp.esta.org/tsp/documents/docs/ANSI\\_E1-31-2018.pdf](https://tsp.esta.org/tsp/documents/docs/ANSI_E1-31-2018.pdf),  
pristupljeno 10.06.2021.
- [10] Farris, J., *Forging new paths for filmmakers on "The Mandalorian"*, 20.02.2020.,  
URL: <https://www.unrealengine.com/en-US/blog/forging-new-paths-for-filmmakers-on-the-mandalorian>, pristupljeno 10.06.2021.
- [11] GDTF Share,  
URL: <https://gdtf-share.com>,  
pristupljeno 10.06.2021.
- [12] HTC Vive,  
URL: <https://www.vive.com>,  
pristupljeno 10.06.2021.
- [13] Kadner, N., *The Virtual Production Field Guide v1.2*, 2019.,  
URL: <https://cdn2.unrealengine.com/Unreal+Engine%2Fvpfieldguide%2FVP-Field-Guide-V1.2.02-5d28ccec9909ff626e42c619bcbe8ed2bf83138d.pdf>, pristupljeno 10.06.2021.
- [14] Kadner, N., *The Virtual Production Field Guide Volume 2, v1.0*, 2021.,  
URL: <https://cdn2.unrealengine.com/Virtual+Production+Field+Guide+Volume+2+v1.0-5b06b62cbc5f.pdf>,  
pristupljeno 10.06.2021.

- [15] MagicQ Software,  
URL: <https://chamsyslighting.com/products/magicq>,  
pristupljeno 10.06.2021.
- [16] Moment Factory, *DMX Previs Sample*, 22.02.2021.,  
URL: <https://unrealengine.com/marketplace/en-US/learn/dmx-previs-sample>,  
pristupljeno 10.06.2021.
- [17] Moslavac, M., Brzica, L., Šimić, Z., *Virtualna produkcija u Unreal Engine 4 razvojnom okruženju, v2.0*, 20.01.2021.,  
URL:  
<https://gitlab.com/mirtamoslavac/projekt-r-fer-2020>,  
pristupljeno 10.06.2021.
- [18] Nierras, J., *Using Chamsys MagicQ to control DMX Lighting in Unreal*,  
13.02.2021.,  
URL: <https://www.youtube.com/watch?v=xxliSWwjL6Q>,  
pristupljeno 10.06.2021.
- [19] SteamVR,  
URL: <https://store.steampowered.com/app/250820/SteamVR>,  
pristupljeno 10.06.2021.
- [20] Streaming Packets,  
URL:  
<https://art-net.org.uk/how-it-works/streaming-packets>,  
pristupljeno 10.06.2021.
- [21] Subscription,  
URL: <https://art-net.org.uk/how-it-works/streaming-packets/subscription>,  
pristupljeno 10.06.2021.
- [22] Unreal Engine,  
URL: <https://www.unrealengine.com>,  
pristupljeno 10.06.2021.

## **POVEZIVANJE VIRTUALNOG I STVARNOG OSVJETLJENJA KORIŠTENJEM UNREAL SUSTAVA**

### **Sažetak**

U ovom radu opisana je metoda virtualne produkcije te su dani njeni slučajevi upotrebe. Proučeni su DMX protokol za upravljanje osvjetljenjem, ethernet protokoli za prijenos DMX podataka preko mreže, Art-Net i sACN i GDTF protokol za postavljanje svojstava uređaja u virtualnoj sceni.

Razvijena je simulacija osvjetljenja u Unreal Engineu kojom se upravlja slanjem DMX paketa korištenjem MagicQ simulatora konzole. Dolazni paketi su također bili snimani kao sekvence te su se mogli uređivati i reproducirati pomoću sekvencera. Za testiranje je korišten DMX Template projekt.

Simulacija se povezala s prototipom za praćenje kamere korištenjem HTC Vive sustava za virtualnu stvarnost i SteamVR aplikacije. Za snimanje videa HTC Vive Tracker je bio pričvršćen na mobitel te su se snimali objekti iz stvarnog života postavljeni ispred LED ekrana.

**Ključne riječi:** virtualna produkcija, DMX, simulacija osvjetljenja, Unreal Engine, praćenje kamere, virtualna stvarnost

# **CONNECTING VIRTUAL AND REAL LIGHTING USING THE UNREAL SYSTEM**

## **Abstract**

This thesis describes the method of virtual production and its use cases. The DMX protocol for lighting control, ethernet protocols for transmitting DMX data over a network, Art-Net and sACN, and the GDTF protocol for setting device properties in a virtual scene were studied.

A lighting simulation was developed in Unreal Engine that controls the sending of DMX packets using the MagicQ console simulator. Incoming packets were also recorded as sequences that could be edited and played back using a sequencer. The DMX Template project was used for testing.

The simulation was connected to a prototype camera tracking system using the HTC Vive virtual reality system and SteamVR app. For video recording, the HTC Vive Tracker was attached to a mobile phone, and real-life objects were placed in front of the LED screen.

**Keywords:** Virtual Production, DMX, lightning simulation, Unreal Engine, camera tracking, Virtual Reality