

OLEDを用いた3Dバーサイタの試作

Web 版資料(2024.03 v6)

目次

- ・研究概要
- ・3Dバーサライタの概要
- ・本年度の改善点
- ・OLEDについて
 - 発光原理
 - リフレッシュレート測定
- ・ハードウェアの概要
- ・ソフトウェアの概要
- ・投影実験の様子
- ・同期機構の追加
- ・科学イベントへの出展
- ・まとめ

研究概要

現在主流の
三次元技術



(1)

Virtual Reality



(2)

Augment Reality

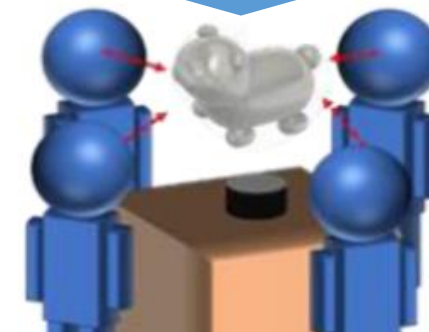


(3)

Mixed Reality

ゴーグルやスマホなどのデバイスを不必要にしたい

新しく提案する
三次元技術

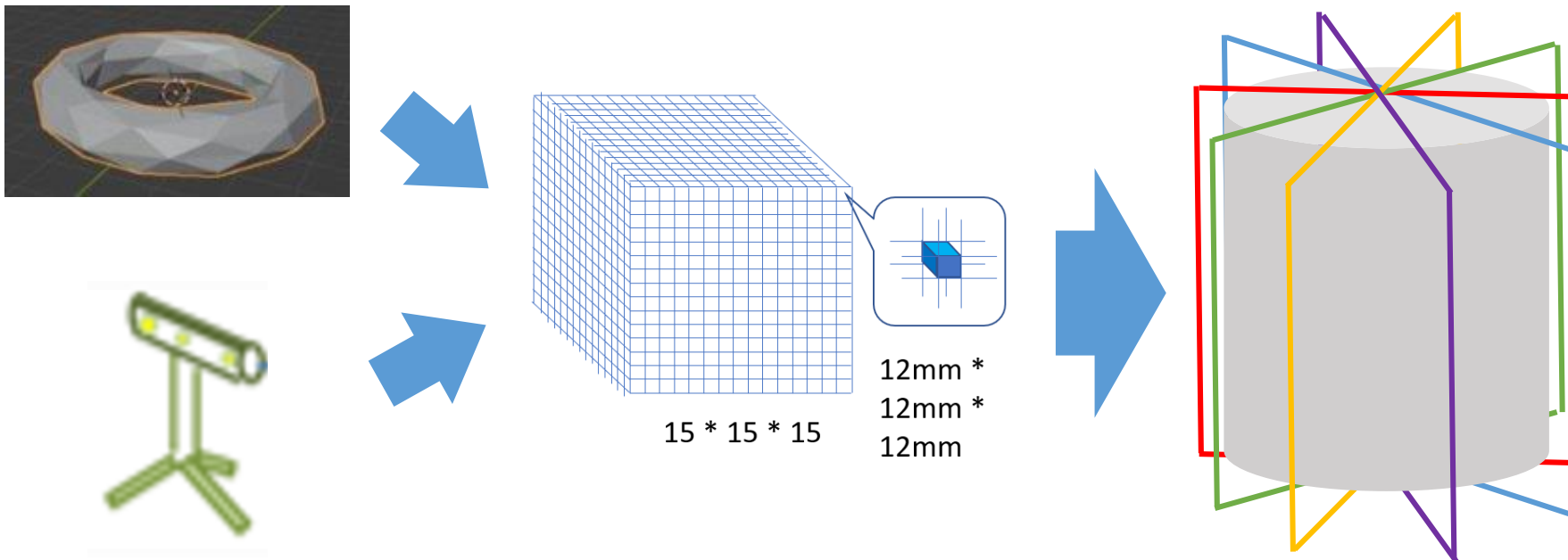


3Dバーサライタ

(1) GoodsPress 両手で操れ！スマホでお手軽VR体験ができる「ポッツニューVR」 <https://www.goodspress.jp/news/53214/>
(2) Pokemon GO Japan 3分でわかる！「ポケモンGO」の基本的な遊び方 <https://youtu.be/2rhjcX-cqic>
(3) 清水教弘 オンラインイベント「医療・ヘルスケア業界DX Day～最先端のデジタル技術活用で実現する医療・ヘルスケア変革～」開催によせて <https://www.microsoft.com/ja-jp/industry/blog/health/2022/04/01/medical-healthcare-industry-dx-day/>

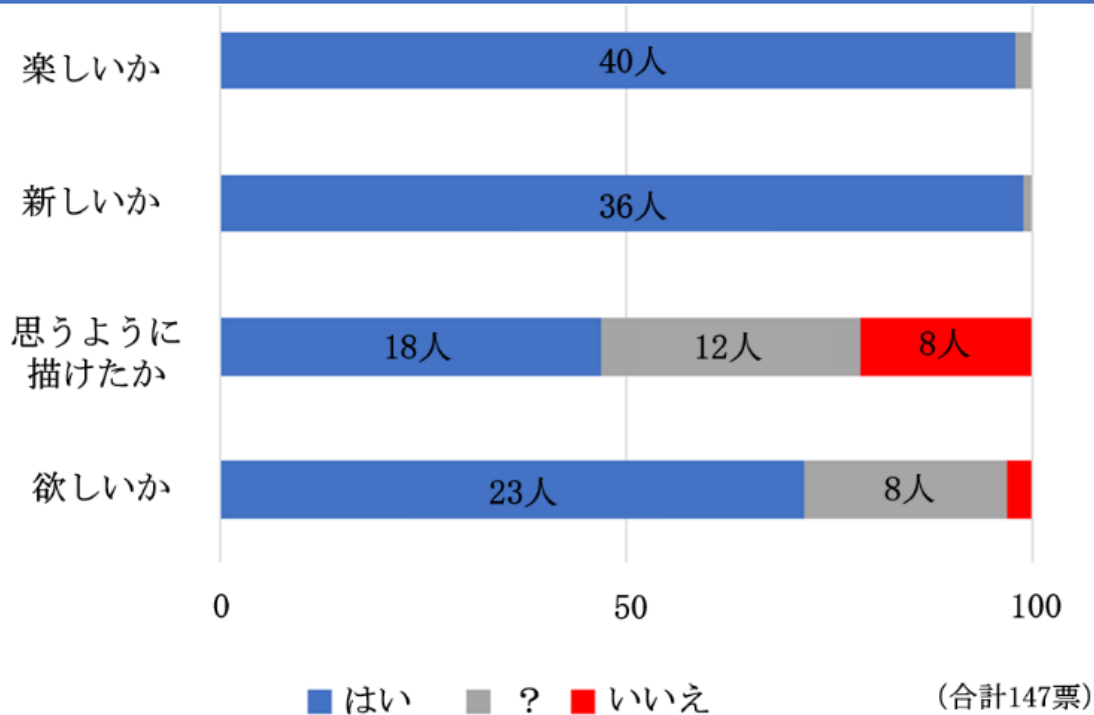
3Dバーサライタの概要

- 発光しているLEDマトリクスを高速回転させ、目に残像を残すことで立体物が投影されて見える装置
- 大人数がデバイスを介することなく視認可能



実際の投影物

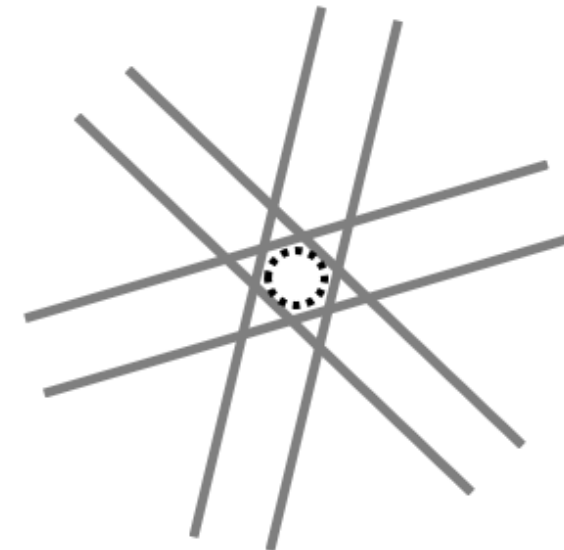
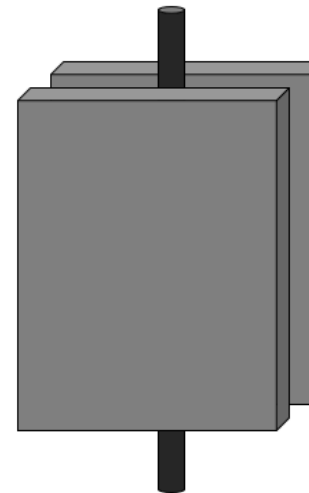
先行研究にて製作された3Dバーサイタの改善点



- ・動いていることは分かるが、手だとはわからない
- ・リアルタイムな映像の出力が楽しい
- ・さらに大きなパネルを使ってほしい
- ・立体の中心が黒くて見えない 今年度の改善点

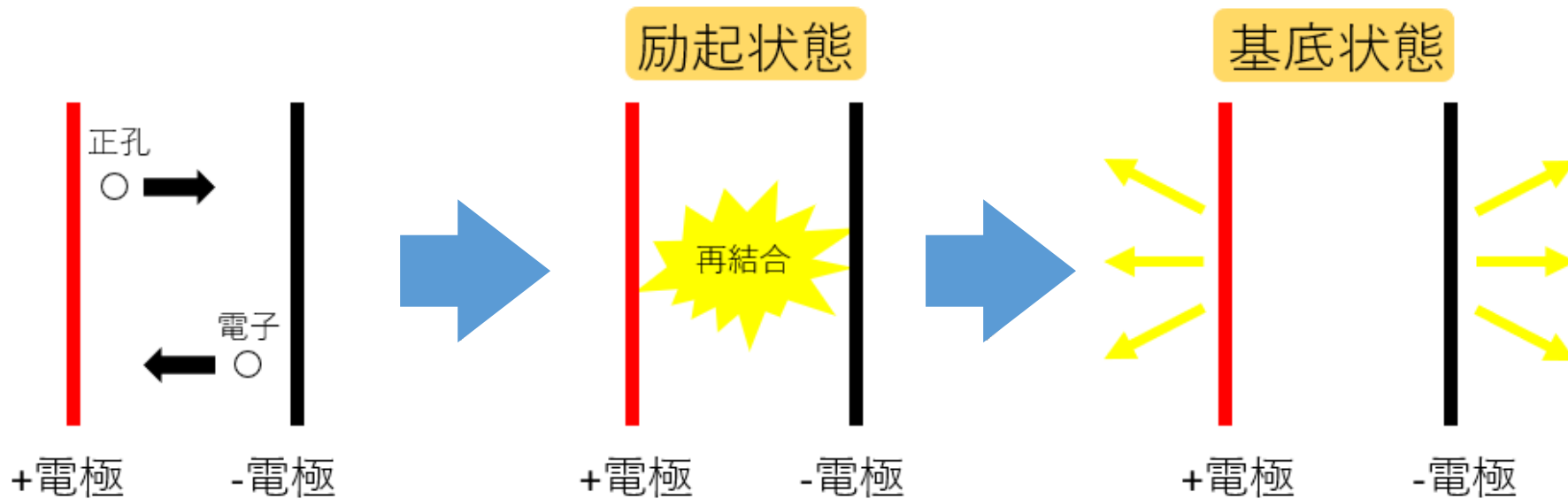


Maker Faire Kyoto(4月29、30日)
けいはんなオープンイノベーションセンターにて



回転時に上から見た図

OLEDの発光原理

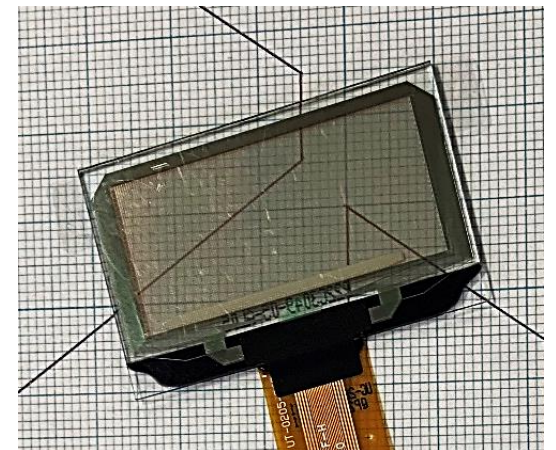


液晶に比べ、表示速度が速いと言われている

OLEDに電圧をかけると、+、-電極それぞれから正孔と電子が注入

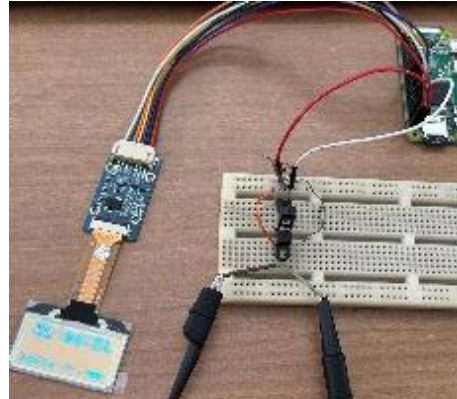
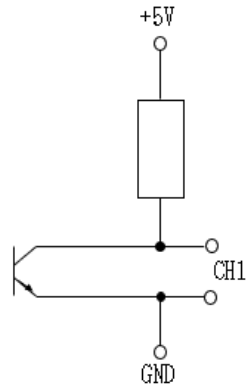
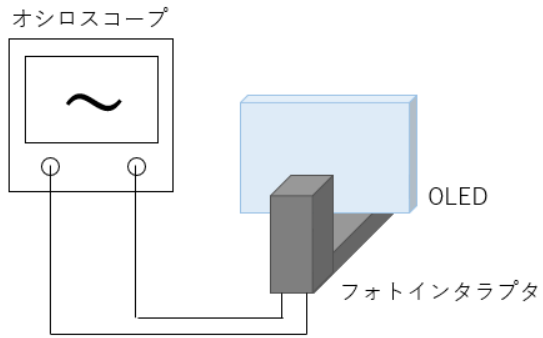
有機物でできた発光層で再結合し、有機物は励起状態へ

元の安定状態へ戻る際、過剰なエネルギーが光として放出



本研究で使用するOLED

リフレッシュレート測定



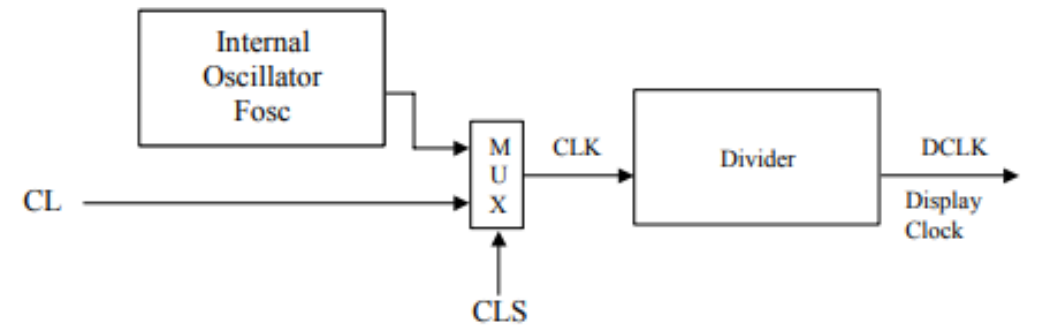
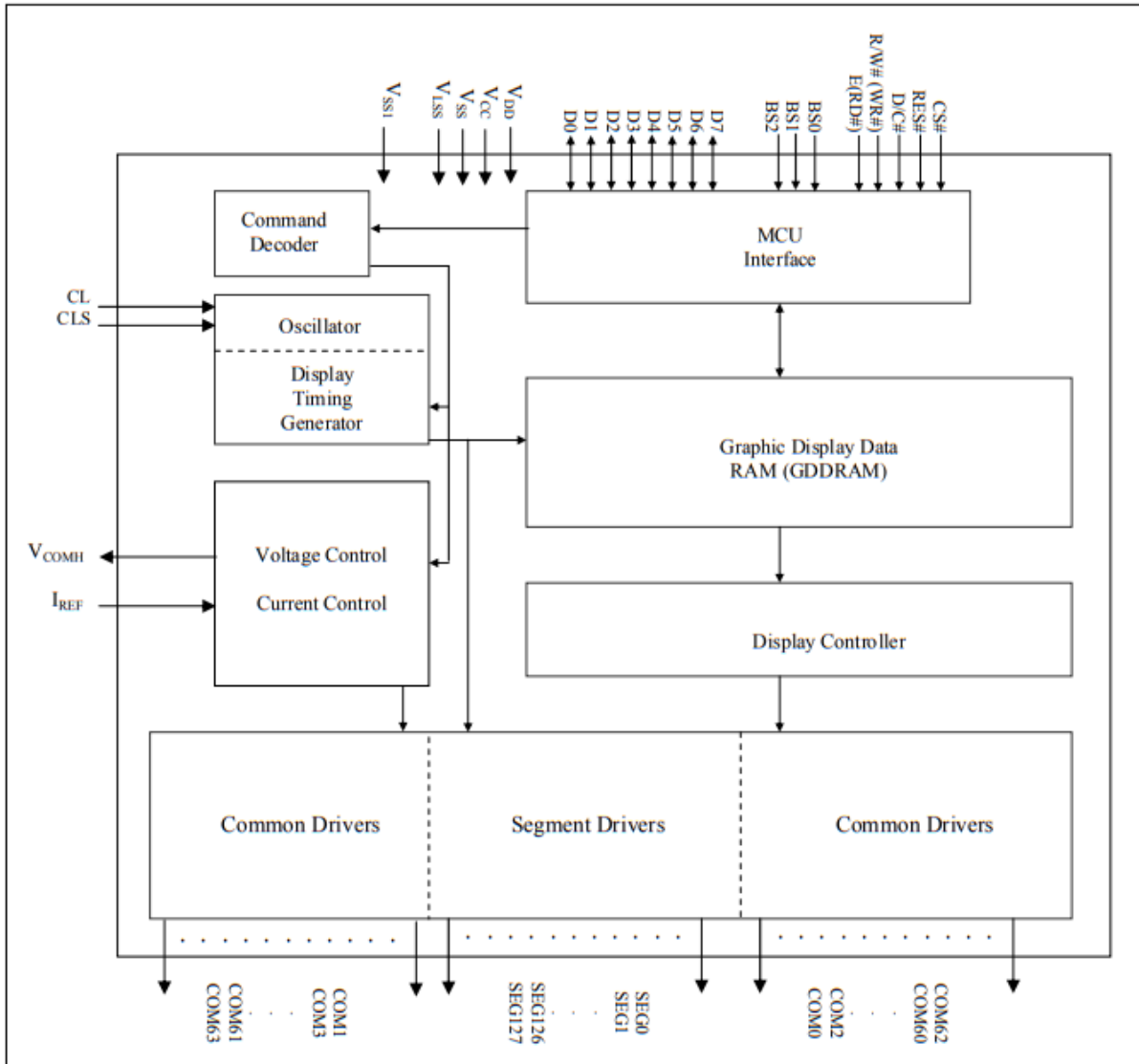
・110Hzであることを確認 ➡ コントローラの設定を変更後、140Hzへ高速化したことを確認

・モータを5回転/sで回転するため、立体物は22分割できると推定

補足資料 リフレッシュレートの高速化



補足資料



書き換えの速度(DCLK)を初期設定プログラムにて変更

ハードウェア製作

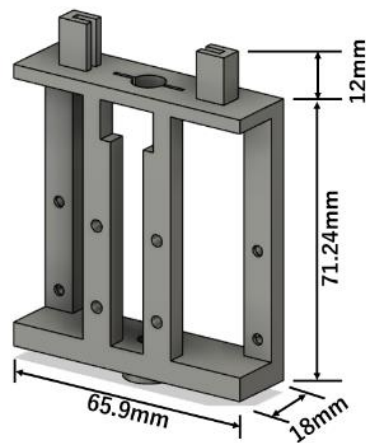


回転部分

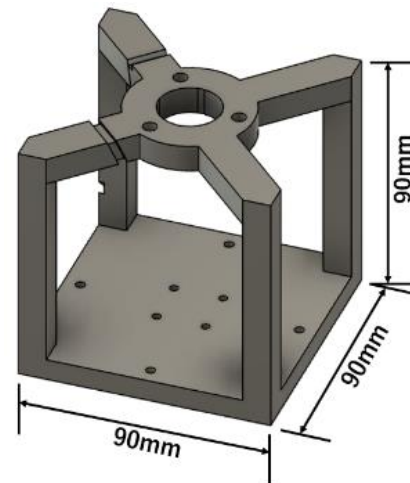
土台部分

- ・Raspberry Piごと回転させる構造
- ・スリッピングの使用により、配線が絡まない

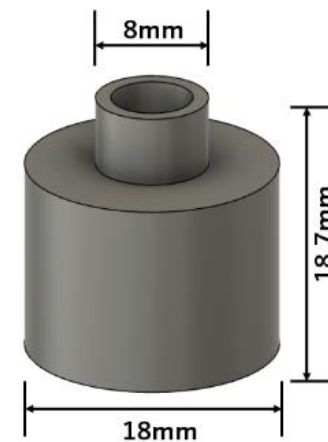
3Dプリンタで製作した部品



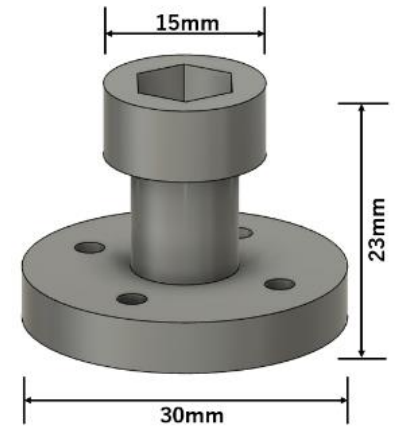
基盤を取り付ける部品



土台と回転部の接続部品



モータの動力を伝える部品

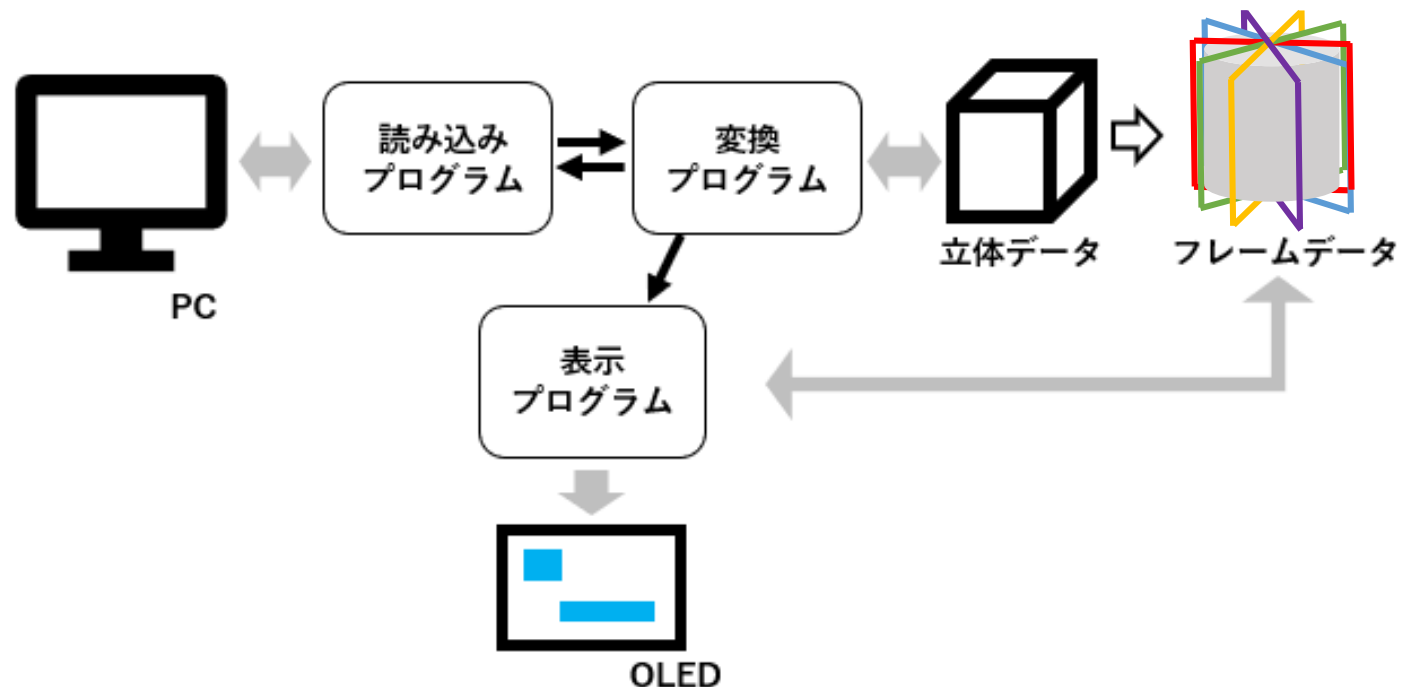


プログラムの流れ

- 座標の読み込みプログラムと、データの変換プログラムを別々に用意
- 2×2 を1マスとして計算する



高速な画面表示に対応



1×1での表示
(128×64)



2×2での表示
(64×32)

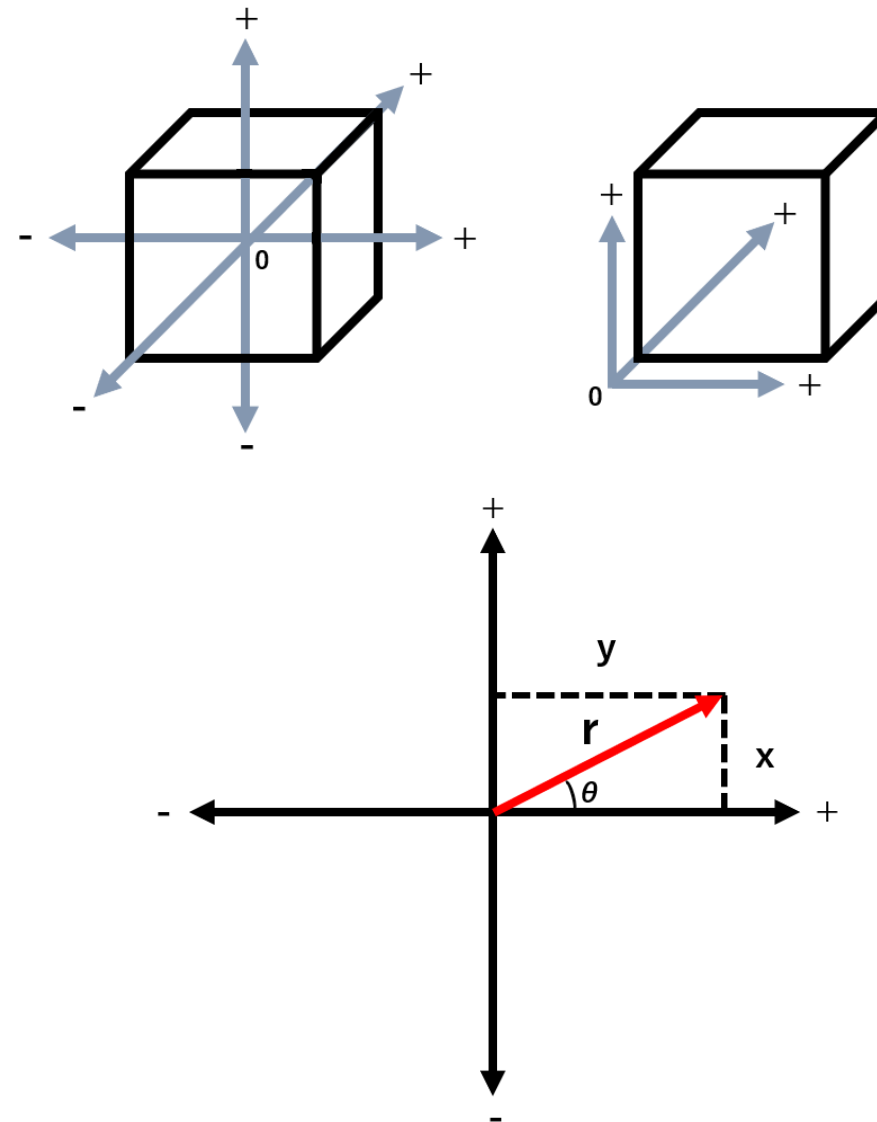
補足資料 3Dデータの展開

```
int henkan(void)
{
    double c, q;
    f=0;

    for(c=0; c<11; c++){
        for(r=0; r<32; r++){
            for(k=-15; k<17; k++){
                q=(c/22)*6.28;//度からラジアン
                i=cos(q)*r;//データ上のx
                j=sin(q)*r;//データ上のy
                i1=i+32;
                j1=j+32;
                k1=k+16;

                if(efg[i1][j1][k1]==1){
                    x=r+32;
                    y=16-(k);
                    abc[x][y][f]=1;}
                else{
                    x=r+32;
                    y=16-(k);
                    abc[x][y][f]=0;}
            }
        }
        f=f+1;
    }
}
```

```
for(f=0; f<11; f++){
    for(y=0; y<32; y++){
        for(x=0; x<64; x++){
            if(abc[x][y][f]==1){
                t=63-x;
                e=f+11;
                abc[t][y][e]=1;}
            else{
                t=63-x;
                e=f+11;
                abc[t][y][e]=0;}
        }
    }
}
```



補足資料 UDP受信例

```
int sock;
struct sockaddr_in addr;

char buf[2048];

sock = socket(AF_INET, SOCK_DGRAM, 0);

addr.sin_family = AF_INET;
addr.sin_port = htons(12345);
addr.sin_addr.s_addr = INADDR_ANY;

bind(sock, (struct sockaddr *)&addr, sizeof(addr));
while(1){
    memset(buf, 0, sizeof(buf));
    recv(sock, buf, sizeof(buf), 0);

    i1=((buf[1]-'0')*10+(buf[2]-'0'))-10;
    j1=((buf[3]-'0')*10+(buf[4]-'0'))-10;
    k1=((buf[5]-'0')*10+(buf[6]-'0'))-10;
    printf("i1=%d\n",i1);
    printf("j1=%d\n",j1);
    printf("k1=%d\n",k1);
    printf("%s\n", buf);
```

補足資料 3Dデータ表示例

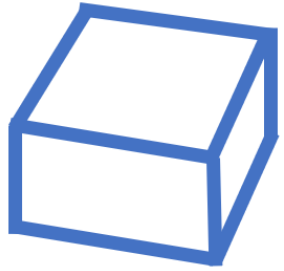
```
struct timespec t;
int t0=0;
int t11, t3;

while(1) {
    henkan();

    for(f=0; f<22; f++){
        for(y=0; y<32; y++){
            for(x=0; x<64; x++){
                if(abc[x][y][f]==1)
                    Paint_DrawPoint(x*2, y*2, WHITE, DOT_PIXEL_2X2, DOT_STYLE_DFT);
            }
        }
    }

    // Show image on page1
    OLED_1in51_Display(BlackImage);
    while(1){
        clock_gettime(CLOCK_REALTIME, &t);
        t11=t.tv_nsec/8000000;
        t3=t11%22;
        if(t3 != t0)
        {
            t0=t3;
            break;
        }
        t0=t3;
    }
    Paint_Clear(BLACK);
    OLED_1in51_Clear();
}
return 0;
```


投影実験



表示したい立体物



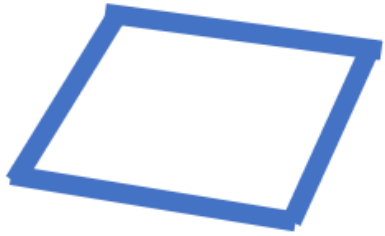
カメラで撮影した立体物



動画:

<https://doilab.net/web/doilab/lab2012/3dv/oled/cyoku4f.mp4> 10

投影実験



表示したい立体物



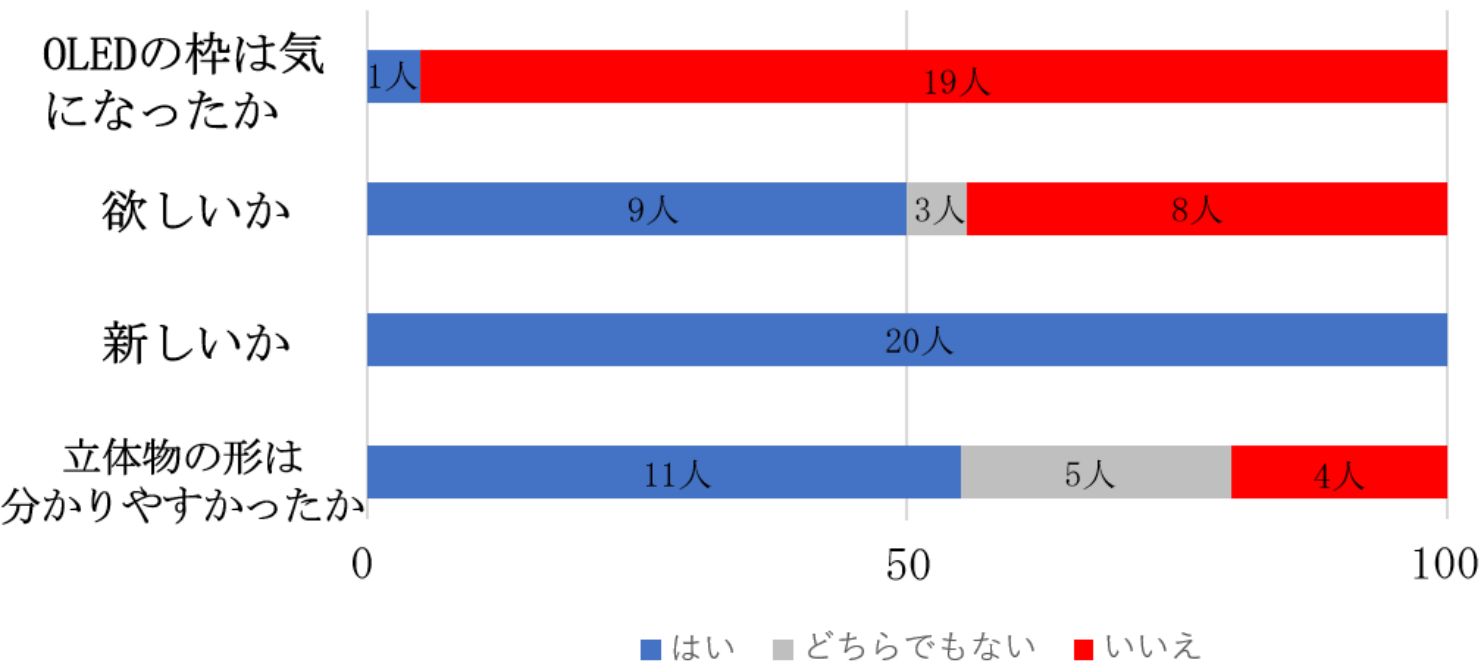
カメラで撮影した立体物



動画:

<https://doilab.net/web/doilab/lab2012/3dv/oled/jyojyo4f.mp4> 11

アンケート結果



- ・輝度を上げてほしい
- ・顔を映し出してみしてほしい
- ・角度によって見えやすさが異なる
- ・回転を速くしてほしい
- ・回転時の音が気になる

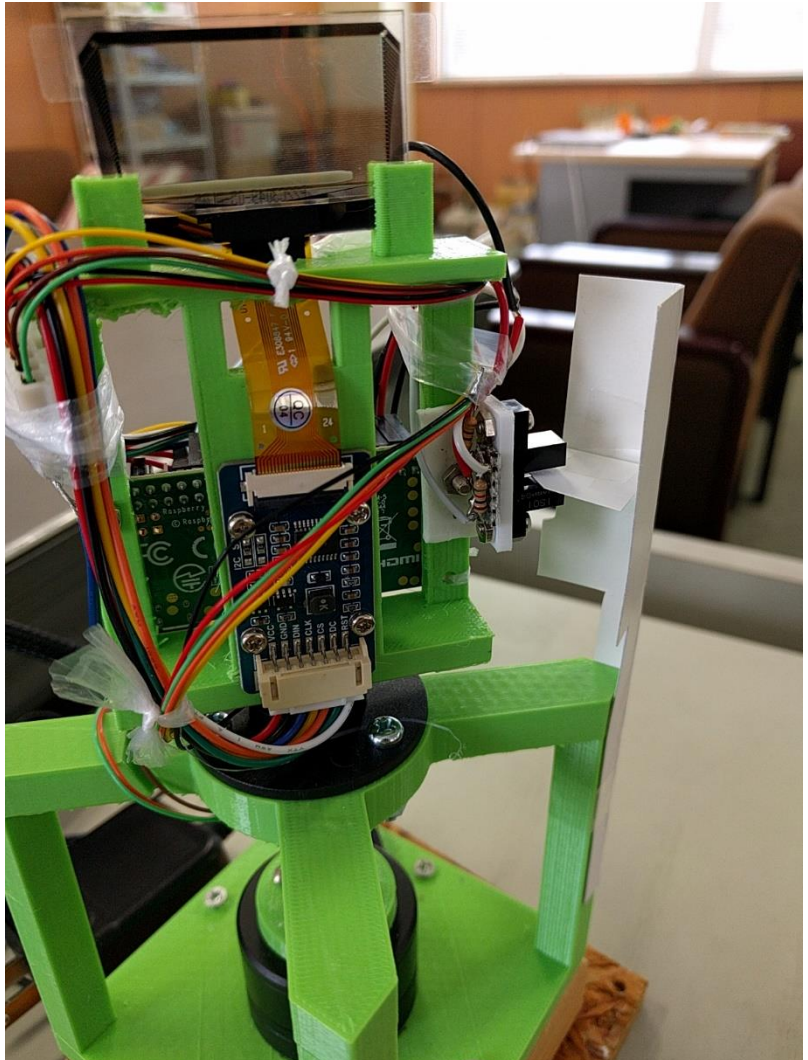
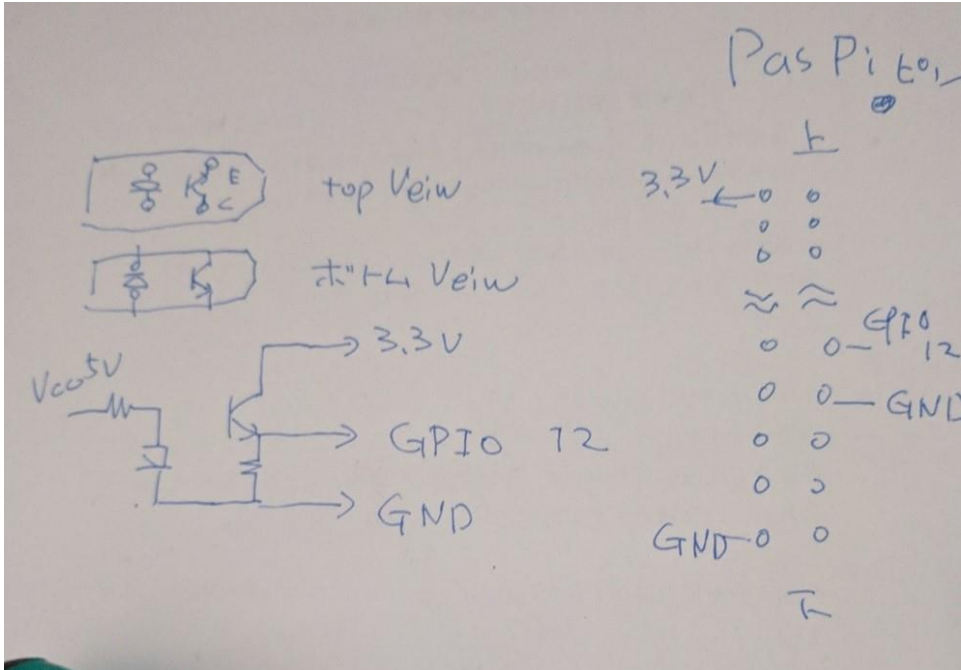
(20人に実施)



- ・新しい技術だという印象は持ったが、立体物の描画の精度が不十分
- ・点灯部分を囲う枠は、視認のし易さには影響しない

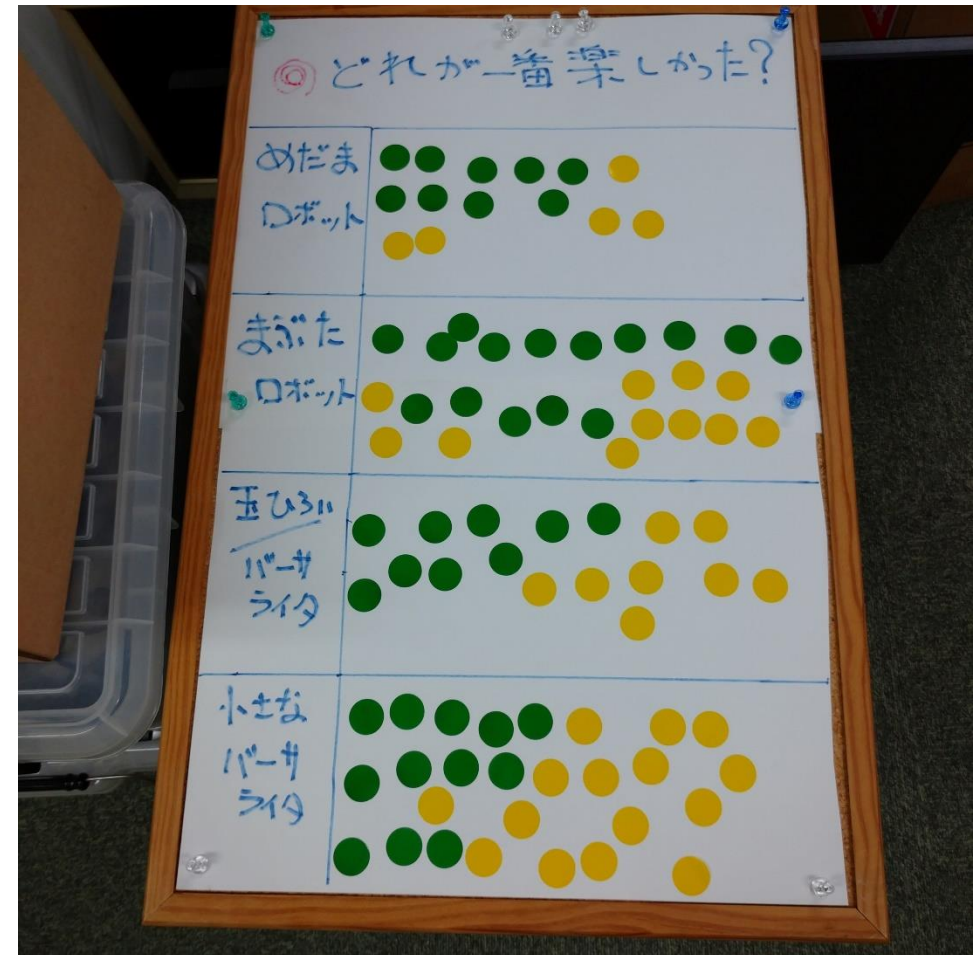
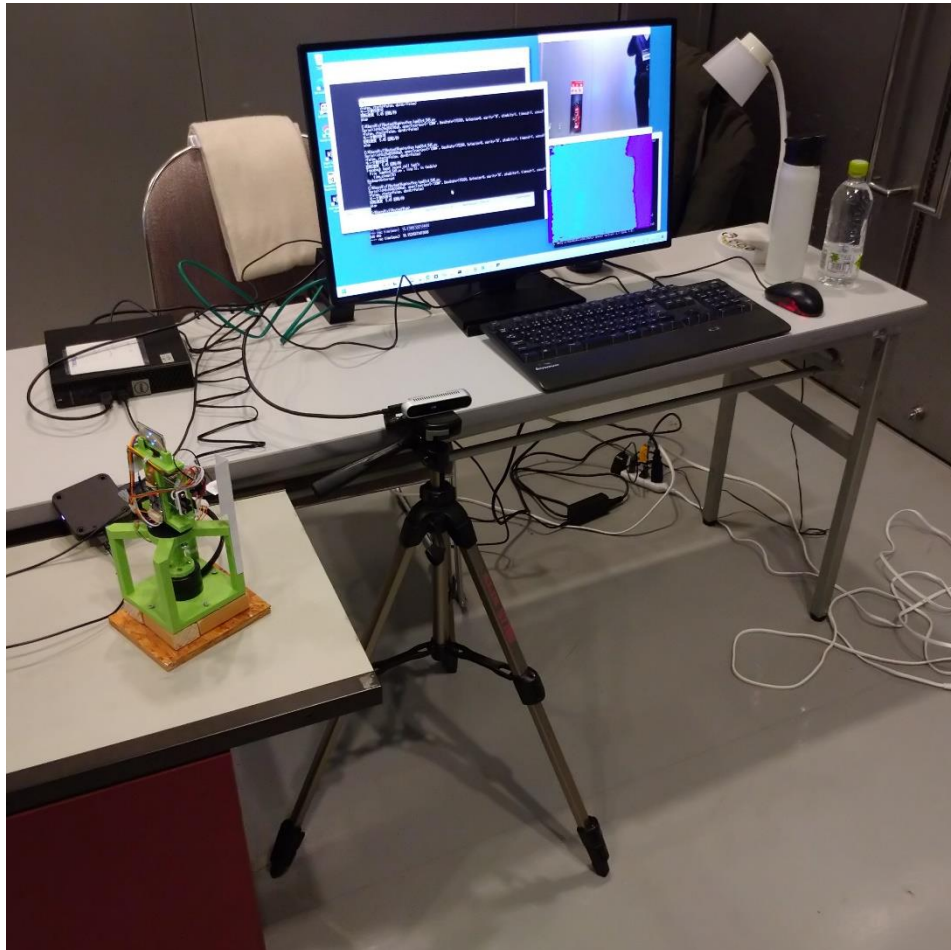
同期機構の追加

同期機構を追加した。フォトインタラプタを利用し、1回転毎にフレーム更新のタイミングを修正する。



科学イベントへの出展

近隣の科学イベントの「けいはんな科学体験フェスティバル」出展した。アンケートを見ると、同時に展示したガジェットの中で高評価となった。



今年度の達成項目

- ・黒い中心部分をなくした立体物の投影
- ・昨年度より細かいボクセルサイズでの投影

今後の改善点

- ・さらに大きなOLEDでの製作
- ・1×1サイズでのさらに細かな立体物表示