

この資料について

この資料は、2025年6月28日に開催された3高専合同学校説明会「宇部高専制御情報工学科体験授業」で用いたスライドを公開したものです。興味を持たれた方は、ぜひその他の宇部高専広報イベントにご参加ください。

- 受験生応援サイト
- 第1回オープンキャンパス → 7月1日（火）申し込み締め切りです！

参考資料（体験実習で用いたツールなど）

今回の体験実習ではTinkercadという仮想環境を使用してArduinoマイコンのシミュレーションを行いました。この環境は一年生のグループワークでラーメンタイマーの設計開発ツールとして試行利用しています。

- 8x8 LED Matrix

リンク先の8x8 LED Matrix回路に対し、`loop()`関数を体験実習用に変更して使用しました。

- 仮想実習環境（Tinkercad）

リンク先から「ニックネームで参加」をクリックし、「ニックネームを入力」部分に1～30までの任意の数字を入力いただくことでログインできます。（体験用に共有していますので、変更を加えてオリジナルを保存するなどの際は、新規にアカウントを取得されてください。）

- Arduino Reference Manual

今回の体験実習には仮想環境を用いしましたが、一年生の実習では実際のArduinoマイコンを用いたプログラミング実習を行います。今回使用したのと同じ関数（命令文）を用います。リンク先は公式マニュアルです。

ミニ電光掲示板プログラミング体験

2025年6月28日

宇部高専 制御情報工学科

教員：江原 史郎・田辺 誠

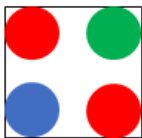
学生（4年生）

じつは → コンピュータとプログラムが活躍！（8x8のLEDで体験）



7.2m

23.4m



3cm

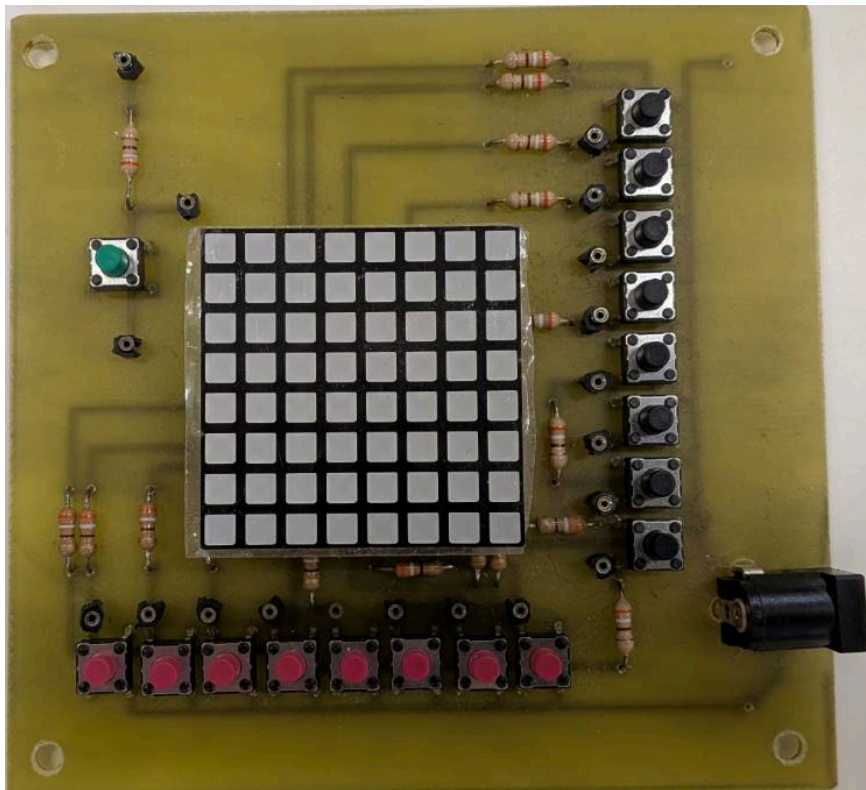


縦240 × 横780 = 187200個！

写真は(株)広島東洋カープ提供

技術情報はパナソニックシステムネットワークス(株)提供

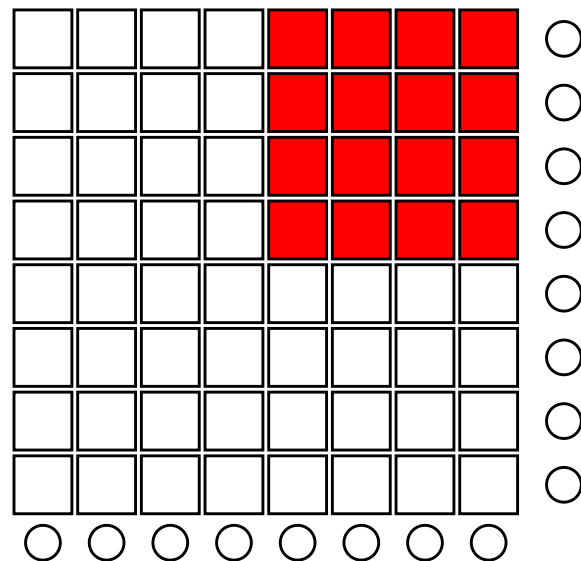
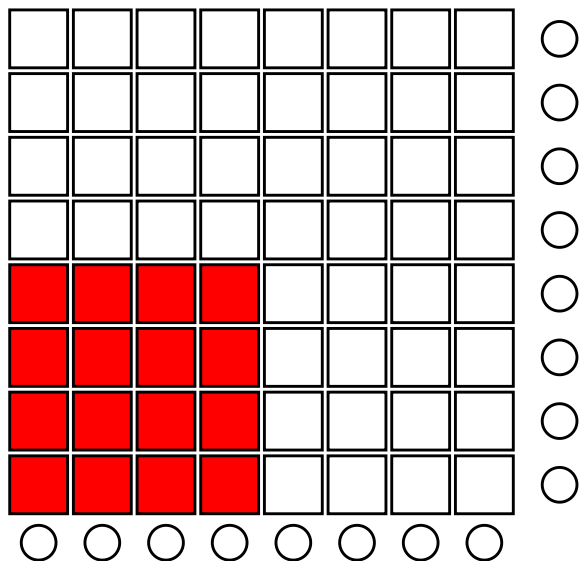
8x8のLED (Matrix LEDとよびます) のしくみ



1. 赤ボタンと黒ボタンを押す。
2. 赤がプラス (5V)
黒がマイナス (0V)
に接続され、
交点のLEDが点灯する。

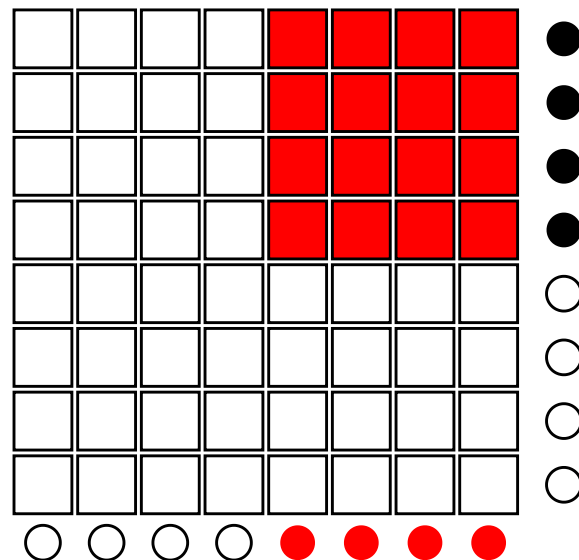
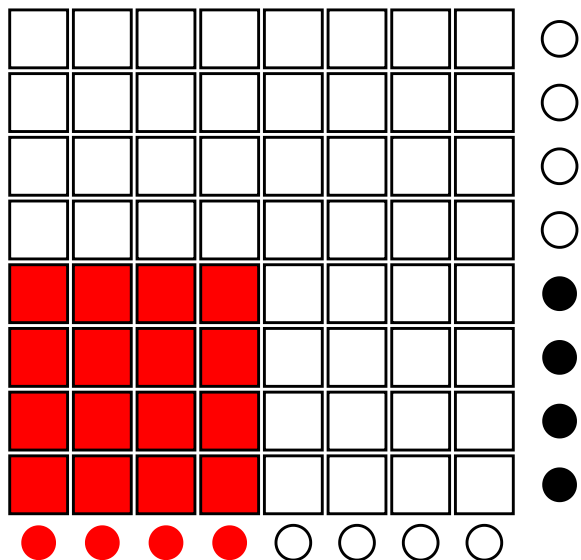
やってみよう！

演習1: 左下の4x4のLEDを光らせよう！ 演習2: 右上の4x4のLEDを光らせよう！

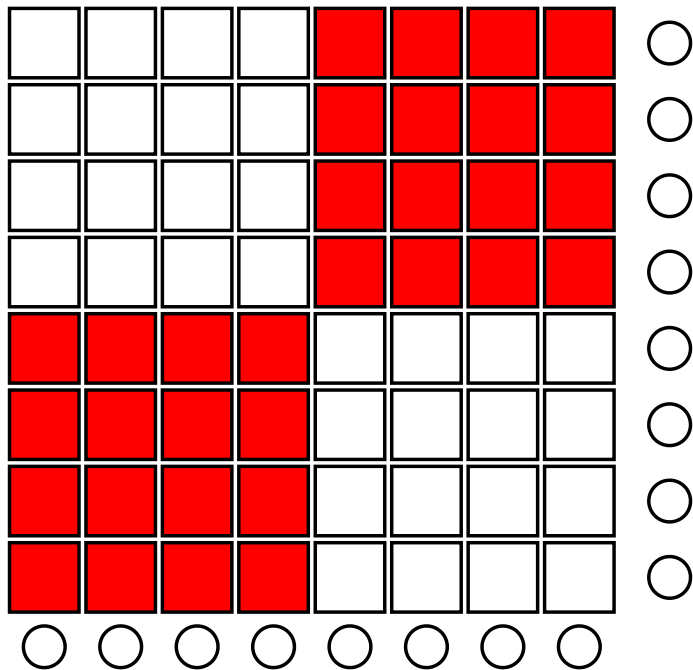


やってみよう！

解答1: 左下の4x4のLEDを光らせよう！ 解答2: 右上の4x4のLEDを光らせよう！



演習3: 左下と右上を光らせよう！



やってみよう！

解答3:

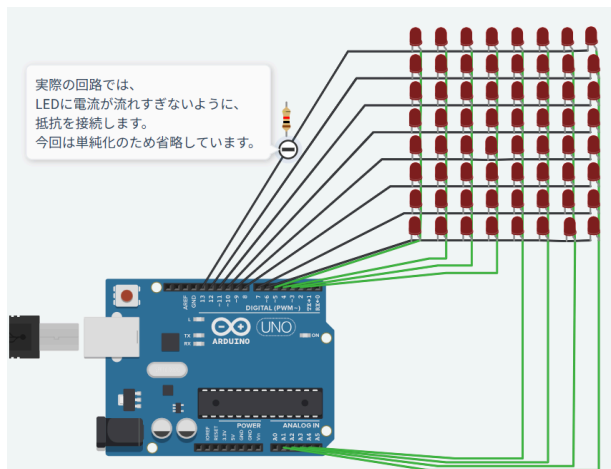
むちゃ手が疲れる！



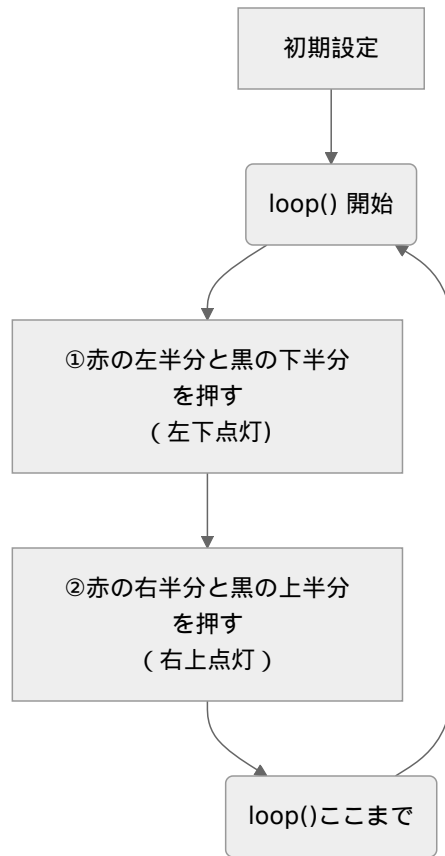
コンピュータに

まかせてしまえ

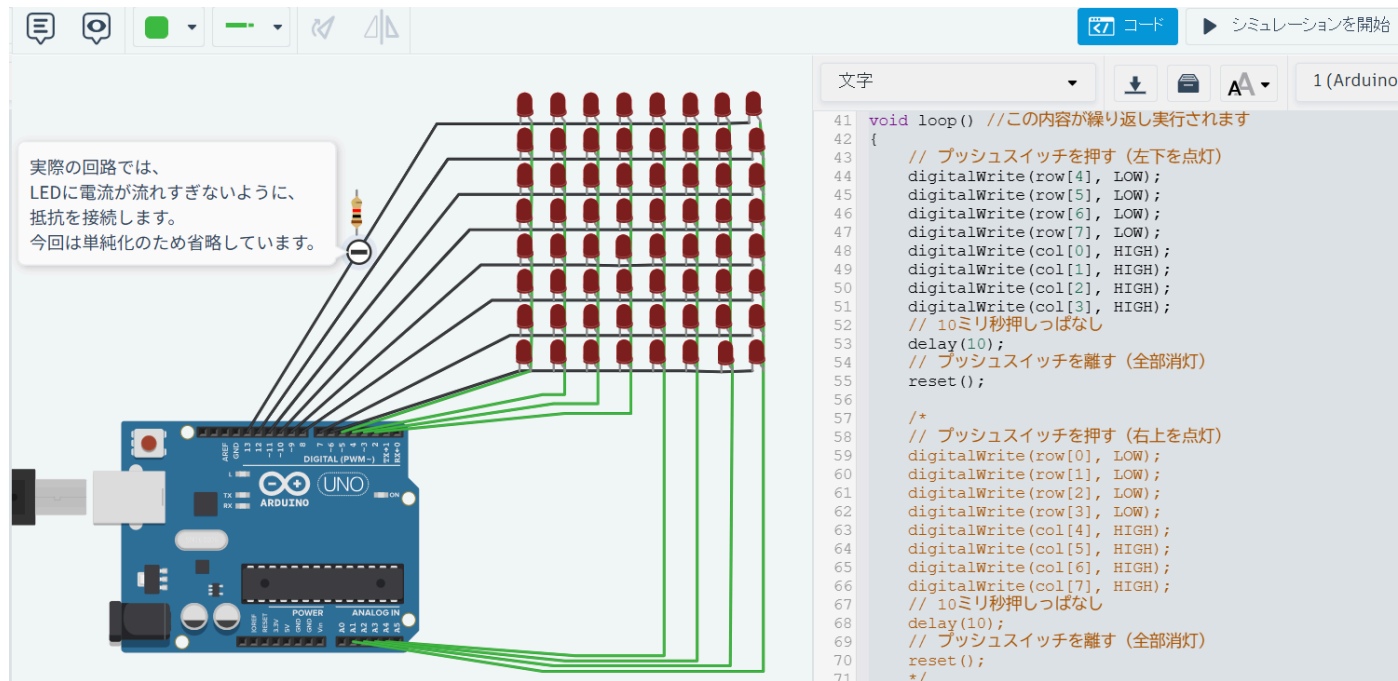
ダイナミック点灯



1. スwitchの代わりにコンピュータを回路に接続する。
2. 「**すごい勢いで①②を繰り返す**」プログラムを作成する。
 - ① 「赤の左半分だけ押す」「黒の下半分だけ押す」
 - ② 「赤の右半分だけ押す」「黒の上半分だけ押す」



演習4: シミュレーションしよう！



The screenshot shows the Arduino IDE interface. On the left, a circuit diagram features an Arduino Uno board connected to a 5x8 Matrix LED. A callout box explains that in a real circuit, a resistor would be used to limit current, but it is omitted here for simplification. On the right, the code editor displays a C++ program that controls the LEDs in a sequence. The code includes comments in Japanese and uses `digitalWrite` to set pin states, `delay` for timing, and `reset` to clear the state.

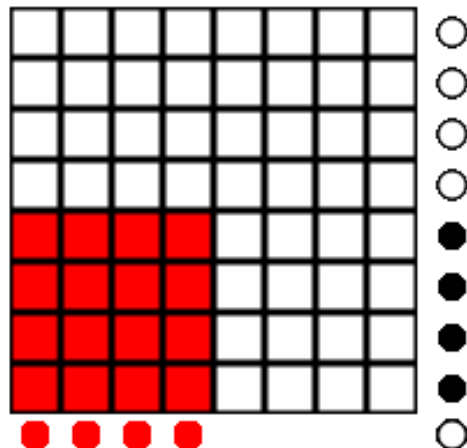
実際の回路では、LEDに電流が流れすぎないように、抵抗を接続します。今回は単純化のため省略しています。

```
41 void loop() //この内容が繰り返し実行されます
42 {
43     // ブッシュスイッチを押す (左下を点灯)
44     digitalWrite(row[4], LOW);
45     digitalWrite(row[5], LOW);
46     digitalWrite(row[6], LOW);
47     digitalWrite(row[7], LOW);
48     digitalWrite(col[0], HIGH);
49     digitalWrite(col[1], HIGH);
50     digitalWrite(col[2], HIGH);
51     digitalWrite(col[3], HIGH);
52     // 10ミリ秒押しっぱなし
53     delay(10);
54     // ブッシュスイッチを離す (全部消灯)
55     reset();
56
57     /*
58     // ブッシュスイッチを押す (右上を点灯)
59     digitalWrite(row[0], LOW);
60     digitalWrite(row[1], LOW);
61     digitalWrite(row[2], LOW);
62     digitalWrite(row[3], LOW);
63     digitalWrite(col[4], HIGH);
64     digitalWrite(col[5], HIGH);
65     digitalWrite(col[6], HIGH);
66     digitalWrite(col[7], HIGH);
67     // 10ミリ秒押しっぱなし
68     delay(10);
69     // ブッシュスイッチを離す (全部消灯)
70     reset();
71     */
72 }
```

1. ▶シミュレーション開始 をクリックして開始する。
2. Matrix LEDの変化を見る。
3. プログラム内の `loop()` を確認する。ただし、57行目～71行目 (`/*` と `*/` の間) は実行されない
4. シミュレーションを停止 をクリックして停止する。

演習5: ダイナミック点灯しよう！

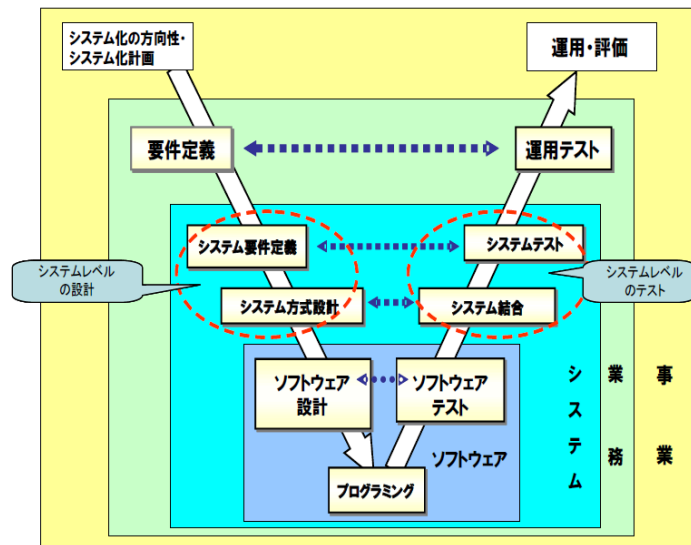
```
41 void loop() //この内容が繰り返し実行されます
42 {
43     // プッシュスイッチを押す (左下を点灯)
44     digitalWrite(row[4], LOW);
45     digitalWrite(row[5], LOW);
46     digitalWrite(row[6], LOW);
47     digitalWrite(row[7], LOW);
48     digitalWrite(col[0], HIGH);
49     digitalWrite(col[1], HIGH);
50     digitalWrite(col[2], HIGH);
51     digitalWrite(col[3], HIGH);
52     // 10ミリ秒押しっぱなし
53     delay(10);
54     // プッシュスイッチを離す (全部消灯)
55     reset();
56
57     // プッシュスイッチを押す (右上を点灯)
58     digitalWrite(row[0], LOW);
59     digitalWrite(row[1], LOW);
60     digitalWrite(row[2], LOW);
61     digitalWrite(row[3], LOW);
62     digitalWrite(col[4], HIGH);
63     digitalWrite(col[5], HIGH);
64     digitalWrite(col[6], HIGH);
65     digitalWrite(col[7], HIGH);
66     // 10ミリ秒押しっぱなし
67     delay(10);
68     // プッシュスイッチを離す (全部消灯)
69     reset();
70
71
72 }
```



まとめ：わかったこと

1. コンピュータを使って、**機械の制御**を安全かつ正確に行うことができる。
2. コンピュータへの指示を**プログラム**で行う。
3. 制御情報工学科では「**機械の制御ができる情報技術者**」を育成する！
4. 設計開発のV字プロセス（設計・実装・検証）を教えます。

プログラミング（実装）



```
void motor_fwd(void)
{
    static motorcp pmc=NULL;
    if (pmc==NULL) pmc=mc();
    if (pmc->cnt!=0x18) { // 同じ場合は実行しない。
        pmc->mflag = (pmc->mflag&~3)|1;
        pmc->cnt = 0x18 ;
        PADR=pmc->cnt;
    }
    return;
}
```


設計

テスト（検証）

