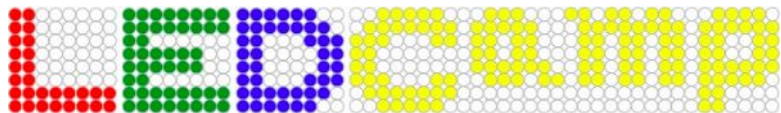
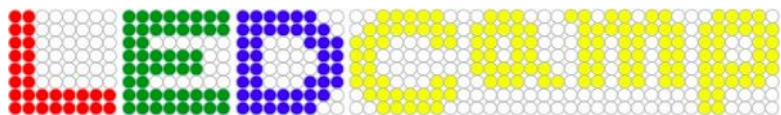


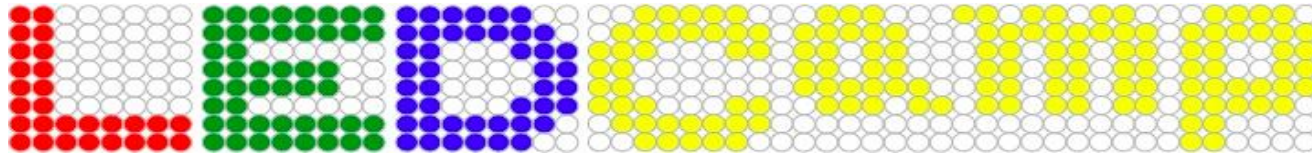
# LED-Camp9 実施報告書



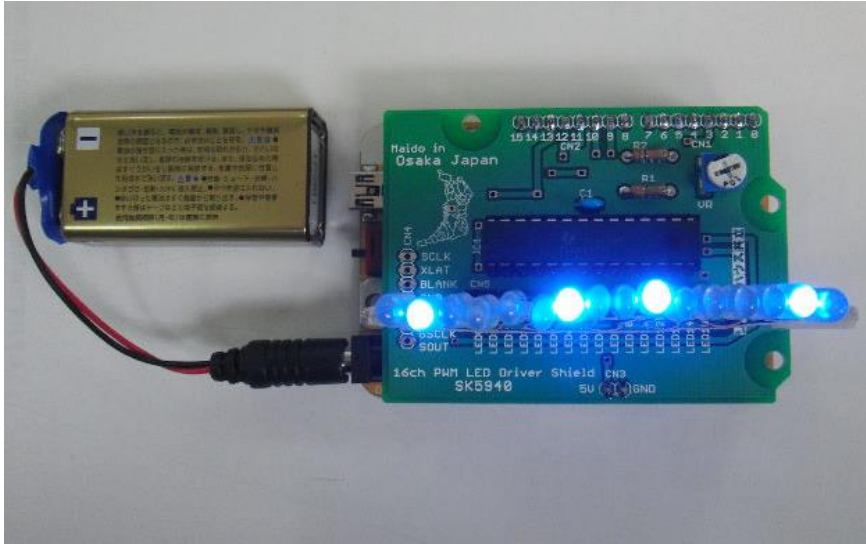
# LED-Camp9開催概要

- ❖ 目的：次世代の組込み技術者の育成
- ❖ 参加対象：組込みシステム分野の若手・初学者
  - 社会人：若手技術者の方
  - 大学生：学部生～修士学生相当
  - 先進的な開発技術に興味のある方
  - 組込み技術者のネットワークを形成したい方
  - チーム開発やプロジェクトマネジメントに興味のある方
- ❖ 実施形態：連続3日間の短期集中形式
  - オンライン開催
  - 2021年8月30日(月)～9月1日(水)





とは？



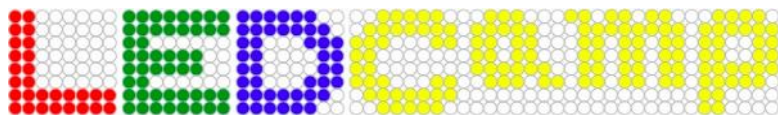
```
hello.c (~) - VIM4
hello.c
1 #include <stdio.h>
2
3 int main (int argc, char *argv[]) {
4     printf ("Hello, World!\n");
5     return 0;
6 }
hello.c 4,26-29 全て
```

LED点滅制御処理

組込み版 “Hello World!”



Learning Embedded software Development Camp



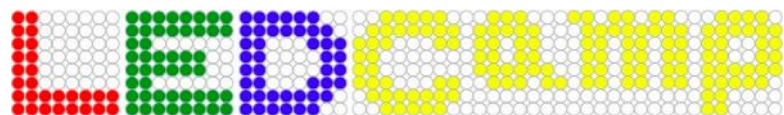
# LED-Camp9の教育目標

## ❖ 組込みシステムの開発技術を習得する

- UMLを用いたモデル駆動開発
- アジャイル開発手法（スクラムフレームワーク）

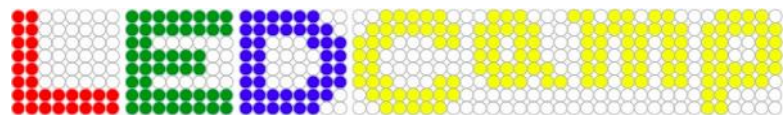
## ❖ チーム開発を円滑にするコミュニケーション技術を習得する

- 目標に向けた合意形成
- メンバーの考えを引き出す話し方・聴き方
- チーム状況を適切に把握するフォローアップ

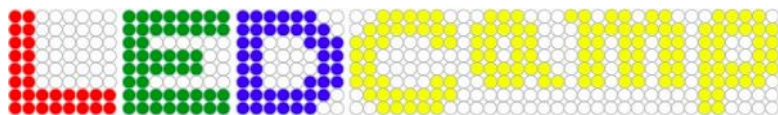
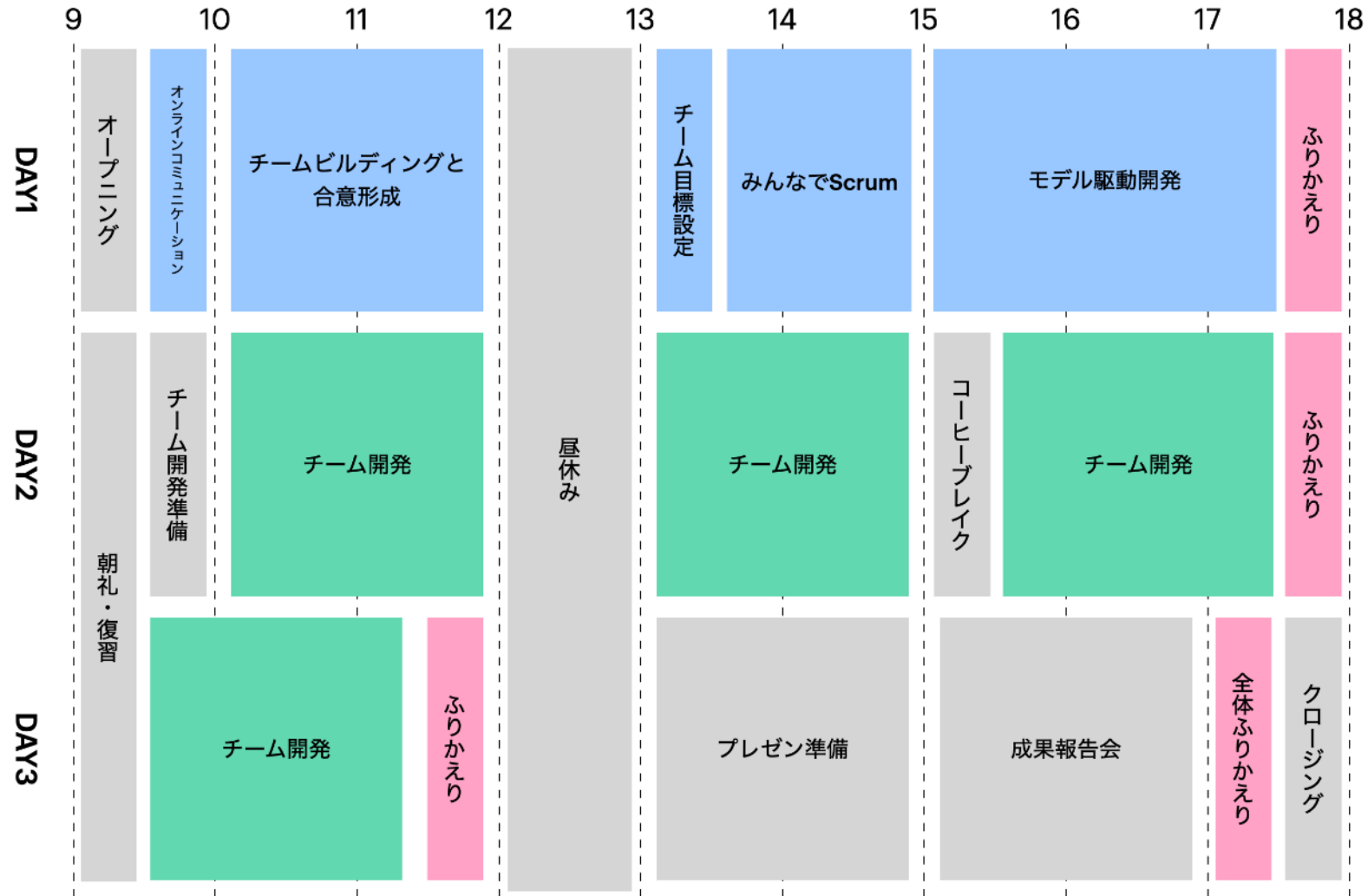


# カリキュラムの狙い

- ❖ モデル駆動開発とアジャイル開発を，体験しながら学ぶ
  - 大規模なソフトウェアを短期間で開発するために注目されている2つの開発手法について学ぶ
  - 開発実習で実践し，深い理解を得る
- ❖ オンラインでの適切なコミュニケーション技術を学ぶ
  - チーム開発実習を通じ，メンバーが同じ目標に向かって進むために何が必要か・何が肝心かを学ぶ
  - 従来とは異なる，オンラインツールを用いたコミュニケーションがどのようなものかを体験し，理解する



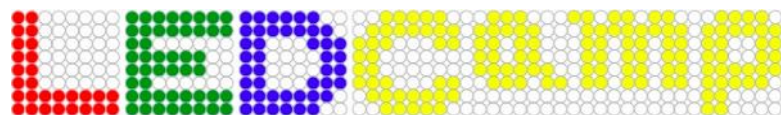
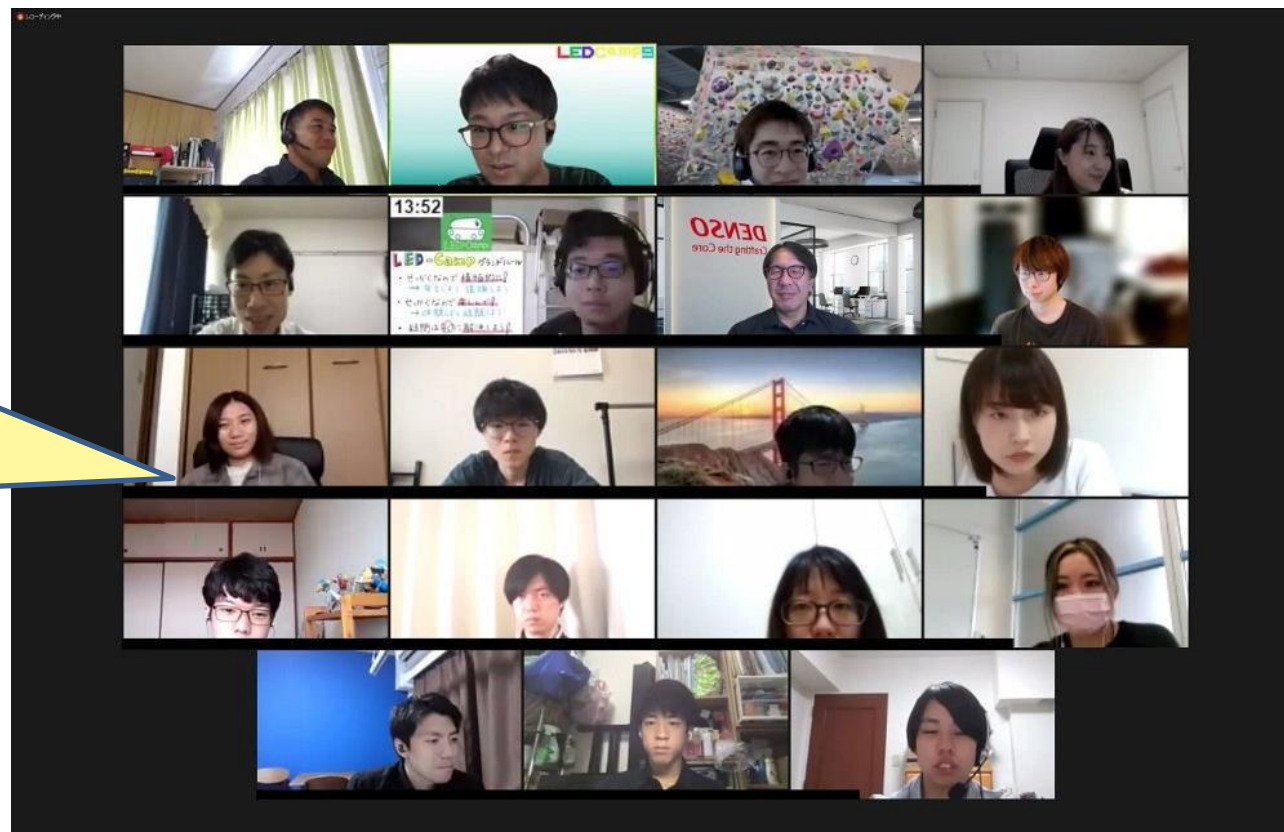
# Camp9タイムテーブル



# ①オンラインコミュニケーション

- ❖ オンラインコミュニケーションの特徴を体験的に学ぶ
- ❖ 後の開発実習に生かすための、円滑なコミュニケーションの手法について考える

いいやり方・悪いやり方に沿った  
オンラインコミュニケーションの  
体験と考察



## ②チームビルディングと合意形成

- ❖ 参加者同士で議論してチーム分けを決めることによりパフォーマンスの良いチームの作り方について考える

チーム分けのやり方を  
アイデア出し&議論

チーム決め

チームわけ

やりたいこと、目的の軸でわける

きたさん  
モデル駆動  
開発の体験

チームA

チームB

チームC

あだ名

なかちゃん  
: チーム開  
発

まーぼ  
チー  
ム開発

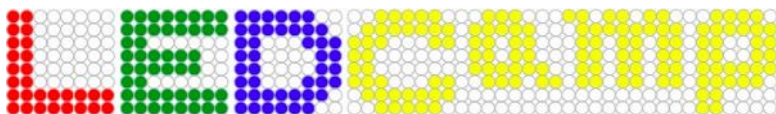
みさきち:  
チーム開発

あいさん:  
アジャイル  
開発手法の  
実践

まき  
コミュニケーション(チ  
ームビルディング)

組込みの勉  
強, チーム  
開発のむさ

るきあ  
チー





# ③みんなでScrum！！

- ❖ アジャイルソフトウェア開発フレームワークであるScrumについて，基礎を学ぶ
- ❖ 演習内でScrumフレームワークを用いることにより，理解を深めるとともにチーム開発実習の基礎を作る

授業風景

みんなでScrum!!!!

### スプリントバックログの作成

1. プロダクトバックログから本スプリントで行う項目を選択し，リスト化→スプリントバックログ
2. 各項目を見積り可能なタスクに分割

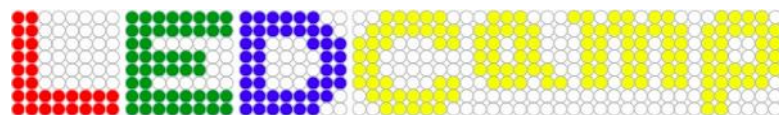
スプリントプランニング

タスク

プロダクトバックログ      スプリントバックログ

プロダクトバックログはすでに優先順位によってソート済みなので上から順に取り出すのみでよい。

LEDcamp      Kazushi Yamashina



## ④モデル駆動開発

- ❖ モデル駆動開発（Model Driven Development）の基礎を学ぶ
- ❖ 開発実習に必要なツールを用いたMDD演習を行い、理解を深めるとともに実習での開発手順に慣れる

モデル駆動開発とは？

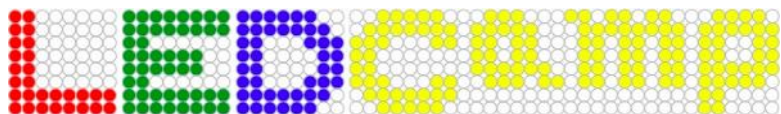
Model Driven Development  
= 開発の抽象度を上げる手段の一つ  
開発コストが減る

```
1 #include <string>
2 #include <vector>
3 #include <map>
4 #include <algorithm>
5 #include <string.h>
6 #include <math.h>
7 #include <stdio.h>
8 #include <stdlib.h>
9 #include <unistd.h>
10 #include <fcntl.h>
11 #include <sys/types.h>
12 #include <sys/stat.h>
13 #include <sys/time.h>
14 #include <sys/mman.h>
15 #include <sys/ioctl.h>
16 #include <linux/usb/lmi.h>
17 #include <linux/usb/lmi.h>
18 #include <linux/usb/lmi.h>
19 #include <linux/usb/lmi.h>
20 #include <linux/usb/lmi.h>
21 #include <linux/usb/lmi.h>
22 #include <linux/usb/lmi.h>
23 #include <linux/usb/lmi.h>
24 #include <linux/usb/lmi.h>
25 #include <linux/usb/lmi.h>
26 #include <linux/usb/lmi.h>
27 #include <linux/usb/lmi.h>
28 #include <linux/usb/lmi.h>
29 #include <linux/usb/lmi.h>
30 #include <linux/usb/lmi.h>
31 #include <linux/usb/lmi.h>
32 #include <linux/usb/lmi.h>
33 #include <linux/usb/lmi.h>
34 #include <linux/usb/lmi.h>
35 #include <linux/usb/lmi.h>
36 #include <linux/usb/lmi.h>
37 #include <linux/usb/lmi.h>
38 #include <linux/usb/lmi.h>
39 #include <linux/usb/lmi.h>
40 #include <linux/usb/lmi.h>
41 #include <linux/usb/lmi.h>
42 #include <linux/usb/lmi.h>
43 #include <linux/usb/lmi.h>
44 #include <linux/usb/lmi.h>
45 #include <linux/usb/lmi.h>
46 #include <linux/usb/lmi.h>
47 #include <linux/usb/lmi.h>
48 #include <linux/usb/lmi.h>
49 #include <linux/usb/lmi.h>
50 #include <linux/usb/lmi.h>
51 #include <linux/usb/lmi.h>
52 #include <linux/usb/lmi.h>
53 #include <linux/usb/lmi.h>
54 #include <linux/usb/lmi.h>
55 #include <linux/usb/lmi.h>
56 #include <linux/usb/lmi.h>
57 #include <linux/usb/lmi.h>
58 #include <linux/usb/lmi.h>
59 #include <linux/usb/lmi.h>
60 #include <linux/usb/lmi.h>
61 #include <linux/usb/lmi.h>
62 #include <linux/usb/lmi.h>
63 #include <linux/usb/lmi.h>
64 #include <linux/usb/lmi.h>
65 #include <linux/usb/lmi.h>
66 #include <linux/usb/lmi.h>
67 #include <linux/usb/lmi.h>
68 #include <linux/usb/lmi.h>
69 #include <linux/usb/lmi.h>
70 #include <linux/usb/lmi.h>
71 #include <linux/usb/lmi.h>
72 #include <linux/usb/lmi.h>
73 #include <linux/usb/lmi.h>
74 #include <linux/usb/lmi.h>
75 #include <linux/usb/lmi.h>
76 #include <linux/usb/lmi.h>
77 #include <linux/usb/lmi.h>
78 #include <linux/usb/lmi.h>
79 #include <linux/usb/lmi.h>
80 #include <linux/usb/lmi.h>
81 #include <linux/usb/lmi.h>
82 #include <linux/usb/lmi.h>
83 #include <linux/usb/lmi.h>
84 #include <linux/usb/lmi.h>
85 #include <linux/usb/lmi.h>
86 #include <linux/usb/lmi.h>
87 #include <linux/usb/lmi.h>
88 #include <linux/usb/lmi.h>
89 #include <linux/usb/lmi.h>
90 #include <linux/usb/lmi.h>
91 #include <linux/usb/lmi.h>
92 #include <linux/usb/lmi.h>
93 #include <linux/usb/lmi.h>
94 #include <linux/usb/lmi.h>
95 #include <linux/usb/lmi.h>
96 #include <linux/usb/lmi.h>
97 #include <linux/usb/lmi.h>
98 #include <linux/usb/lmi.h>
99 #include <linux/usb/lmi.h>
100 #include <linux/usb/lmi.h>
```

扇風機を表現してみよう

- 仕様
- 出力は2段階の設定が可能
- ON/OFF/低/高と書かれたボタンがある
- ON/OFFボタンで運転開始/停止を制御する
- ONを押すと出力高で運転する
- 運転中に低/高ボタンを押すことで出力を制御する
- ボタンが押されると「ピッ」と音がる

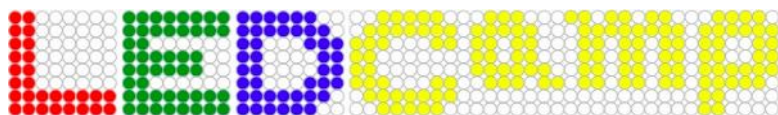
演習  
(扇風機の振る舞いを  
ステートマシン図で表現)



## ⑤チーム開発実習（説明1）

- ❖ 座学・演習で学んだ知識を実践し，さらに理解を深める
- ❖ チームで議論しながら開発することにより，組込み開発およびオンラインでのチーム開発について，気づきを獲得する

競技ルールを基に  
作戦を議論中



## ⑤チーム開発実習（説明2）

- ❖ 1スプリント2時間 × 4 = 8時間の開発期間
- ❖ スプリント毎にチームで振り返りを実施・意見を全体で共有
- ❖ 成果物として、制御プログラムのモデルを提出

開発であった  
K（よかったこと）  
P（問題だったこと）  
T（次からやってみることをふりかえりで全体共有

シミュレータ上で  
振舞を確認

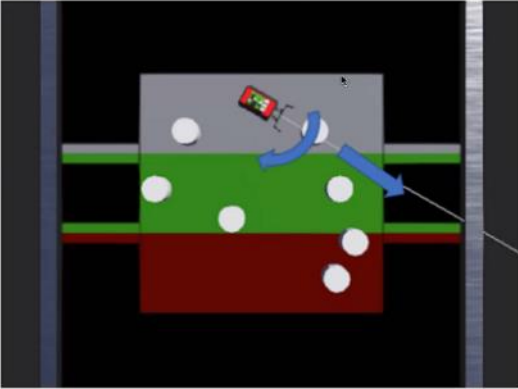
Problem  
時間にルーズだったこと  
方向性を決めるのに悩んで、実  
タスクの工数見積もりが甘かった  
一人の作業に集中してしまい、周りのフォローができなかった  
話し合いの期、他人の意図が理解できなかったり、自分の意図が伝わらなかったりした。  
全体の進捗を気にするばかりになってしまい、自分の稼働が止まらなかった。  
単純な質問でのアカウントが多かった。

実行錯誤しつつ  
観測を考えると  
わかってきた  
さ  
りまず手を  
動かして実  
装ができた  
スプリント中に  
目標に沿って、  
動くものが  
できた

# ⑥成果報告会（プレゼンテーション）

- ❖ 競技会に向けた戦略や，実習中で得た学び等を整理する
- ❖ 成果をプレゼンすることにより，伝え方の工夫を学ぶ．また，各チームが得られた情報を他チームと共有し，学びを深める．

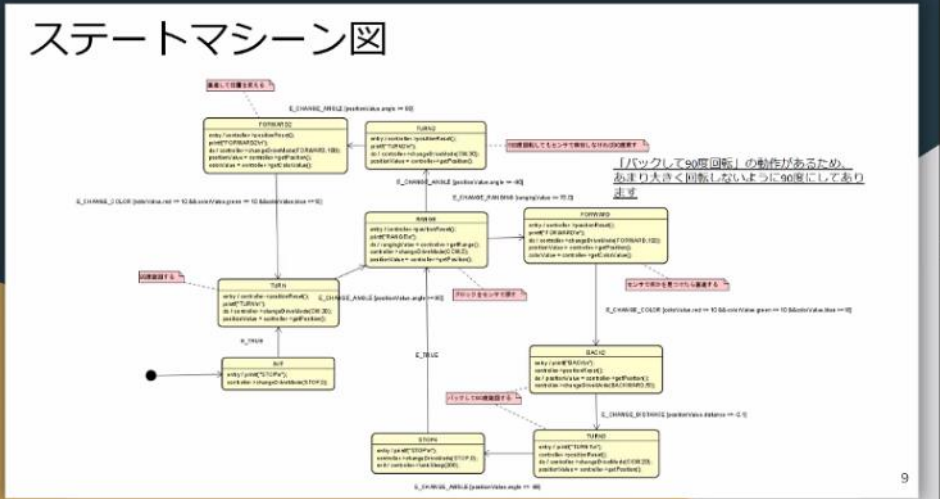
コンセプト  
タンクの振る舞い(攻略方法)



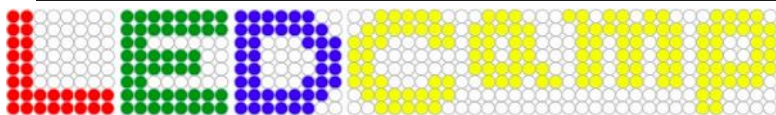
1. 白の初期位置(START位置)に移動する
2. **ブロックを見つける**
3. **ブロックをフィールド外に押し出してSTART位置に戻る**
4. 無得点エリアから高得点エリアにブロックを押しだす
5. 無得点エリアから高得点エリアにブロックを押しだす（未実装）

このスライドには、ゲームのコンセプトと攻略方法が示されています。左側には、タンクが赤いブロックを押し出す様子を示すイラストがあります。右側には、5つのステップで構成された攻略手順がリストアップされています。また、このスライドはオンラインプレゼンテーションの一部であり、右側には参加者のビデオサムネイルが縦に並んでいます。

状態マシン図

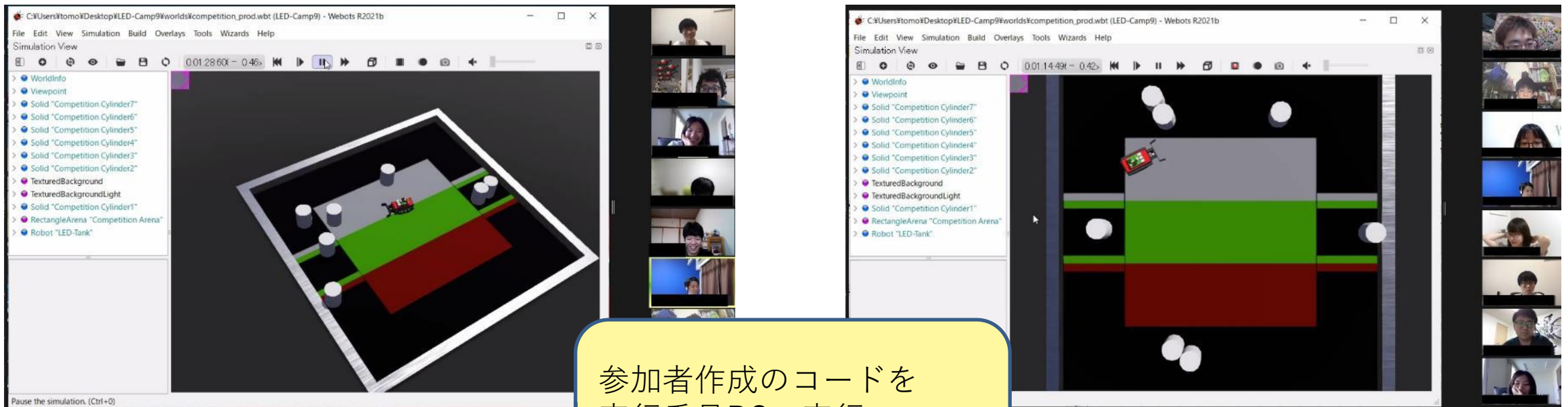


このスライドには、ゲームの動作を制御するための状態マシン図が示されています。図には、START、SEARCH、MOVE、BACKなどの状態があり、各状態間の遷移と実行されるアクションが詳細に記述されています。また、いくつかの注釈が追加されています。例えば、「バックして90度回転」の動作があるため、あまり大きく回転しないように90度にしてあります。このスライドもオンラインプレゼンテーションの一部であり、右側には参加者のビデオサムネイルが縦に並んでいます。

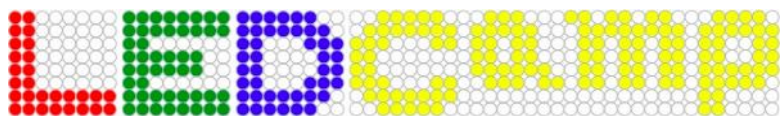


## ⑥成果報告会（競技会）

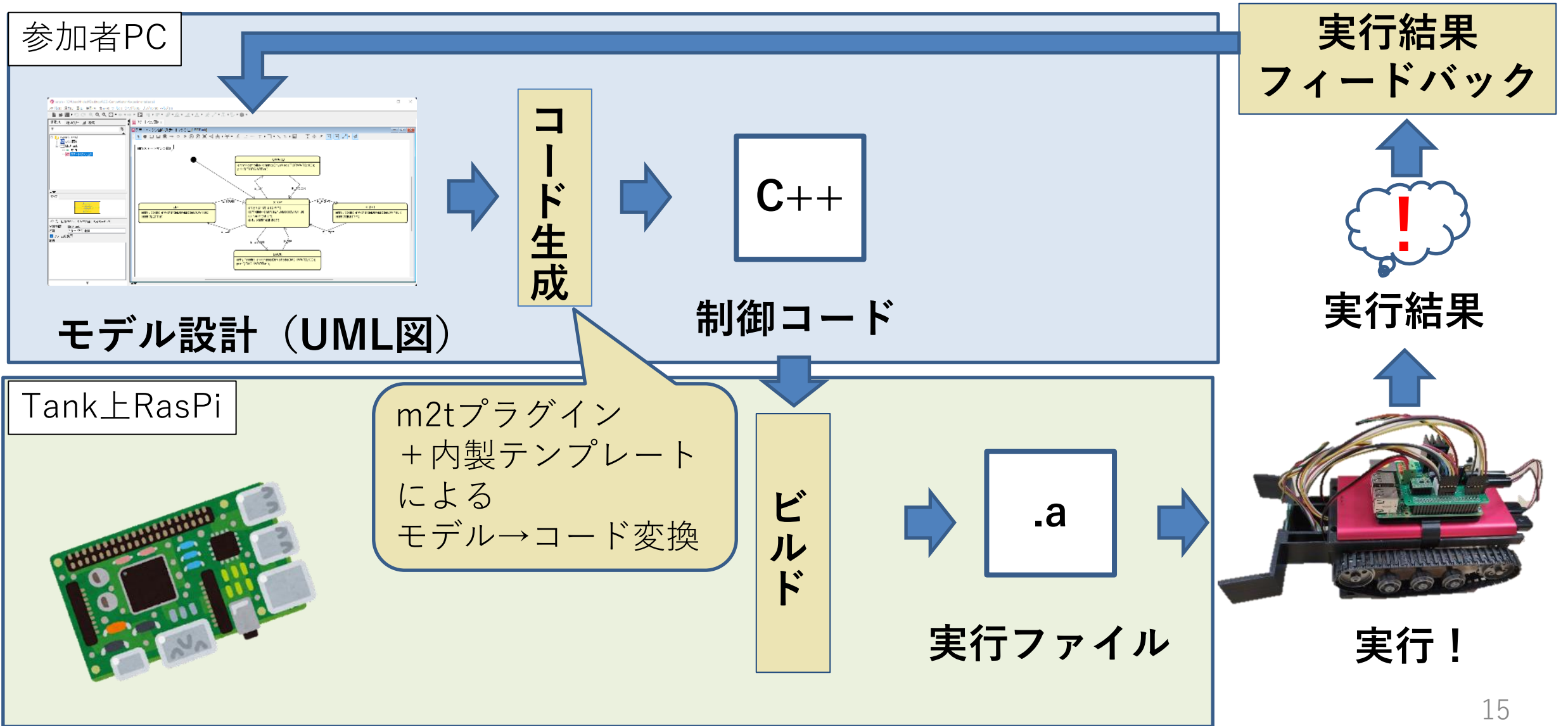
- ❖ チーム開発実習の成果物を実行する
- ❖ f競技会での実行結果を共有し議論することで、組込みなどのシステム開発についての学びを獲得する



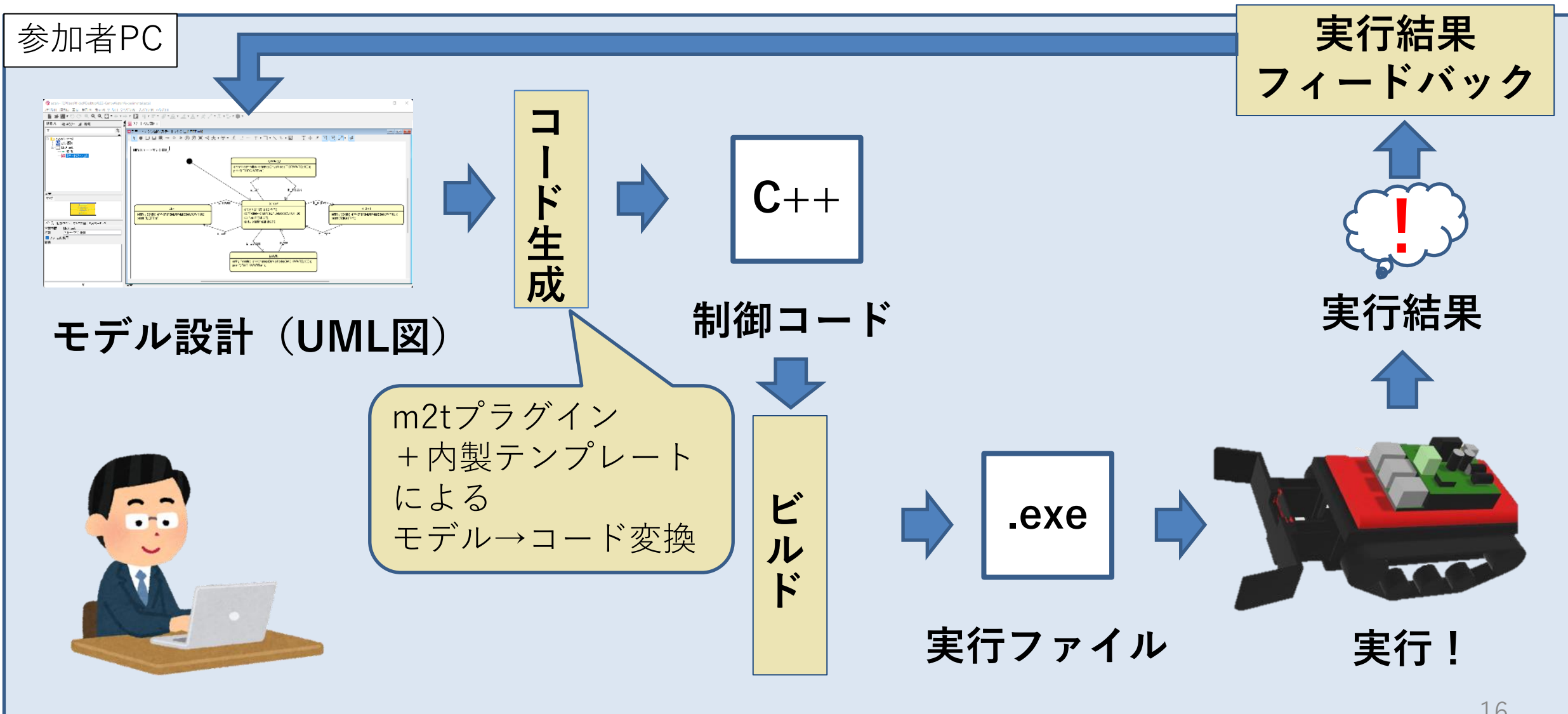
参加者作成のコードを  
実行委員PCで実行  
全員で動きを見守ってます



# 過去(Camp7)の実習での開発フロー



# Camp7（オンサイト開催）実習での開発フロー





# 過去(Camp7)の実習でのLED-Tank構成

## 測距センサ

VL53L0X ToF型  
レーザーセンサー

## カラーセンサ

BH1745(底面)

## ラインセンサ

内製フォトセンサ基板

## 内製キャタピラ台車

タミヤのキャタピラキット  
+内製 (3Dプリンタ出力) 部品  
+モバイルバッテリー

## 内製拡張基板

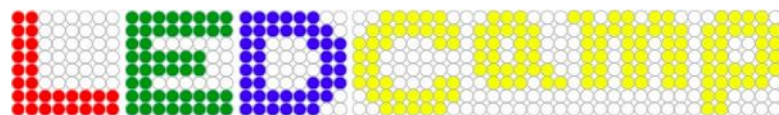
モータドライバ DRV8835  
各センサ接続

## 制御コンピュータ

Raspberry Pi Model 3 B  
Raspbian OS

## 内製エンコーダ基板

DCモータ  
+ロータリーエンコーダー  
(LBT-131フォトインタラプタ使用)



# Camp9（オンライン開催）実習でのLED-Tank構成

**測距センサ**

DistanceSensor  
(Laser)

**カラーセンサ**

Camera

**ラインセンサ**

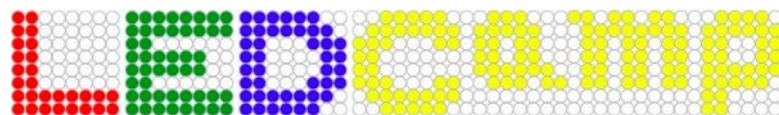
DistanceSensor(IR)

**内製3Dモデル**

Fusion360で作成  
3Dプリント用データを一部使用

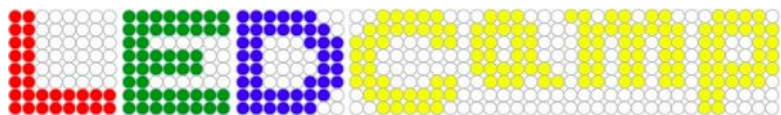
**モータ回転角センサ**

Motor  
+PositionSensor



# Camp9での開発環境

- ❖ ロボットシミュレータ
  - Webots (R021b)
- ❖ モデリングツール
  - astah\* Professional (参加者には半年間ライセンス配布)
  - astah\* m2t プラグイン

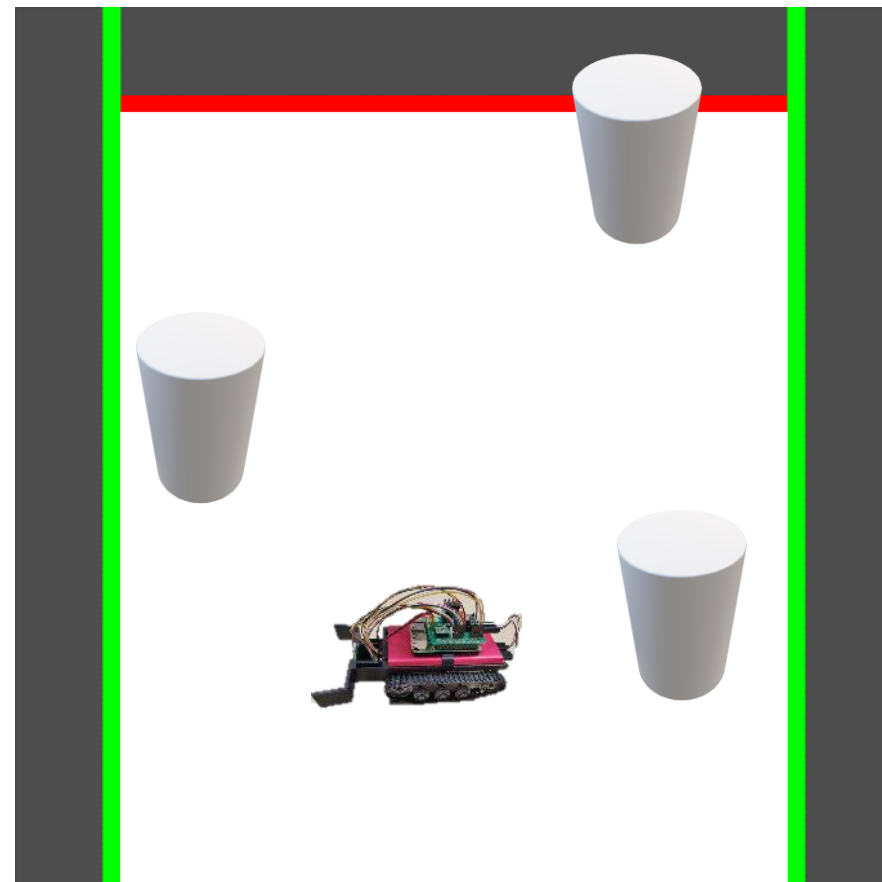


# 競技会

**制限時間（2分）以内に円柱を得点エリアに  
押し出し，より多くの得点を獲得する**

## ルール詳細

- 開発したプログラムでLED-Tankを制御し，  
コースに配置されたブロック（円柱）を押し出す  
（ただし，ブロック配置および開始地点は非公開）
- 得点エリア（黒色領域）は，隣接する色により  
得られる得点が異なる  
（緑：3点 赤：1点 白：0点）



# 競技結果

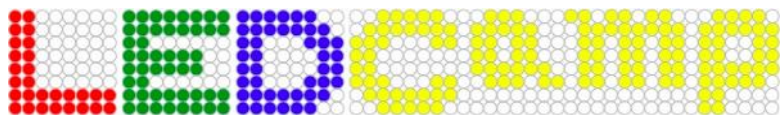
❖ どのチームも高い得点を獲得！

サンシャイン木へん 10点(50秒)

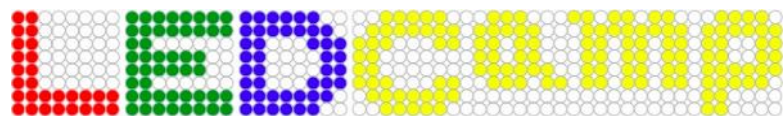
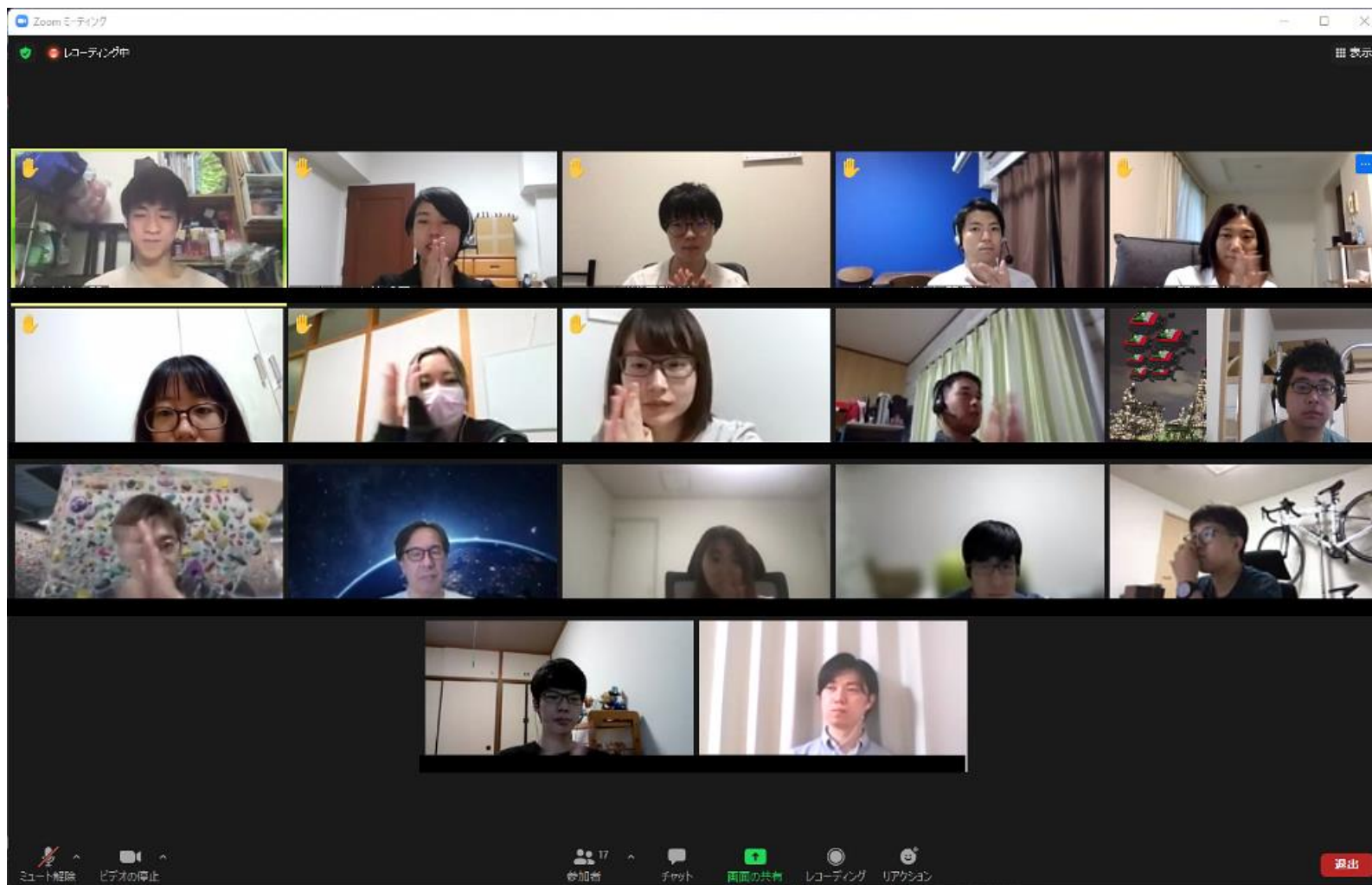
3人で頑張るマン 10点(1分26秒)

BEST OF FUN 8点

入賞：実行委員 7点



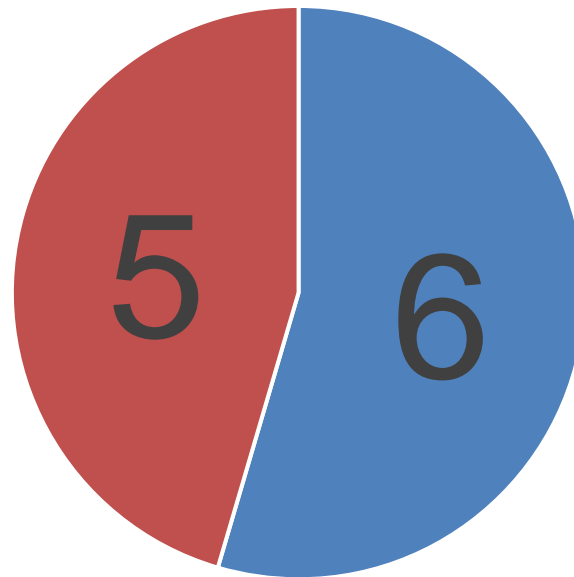
# 集合写真



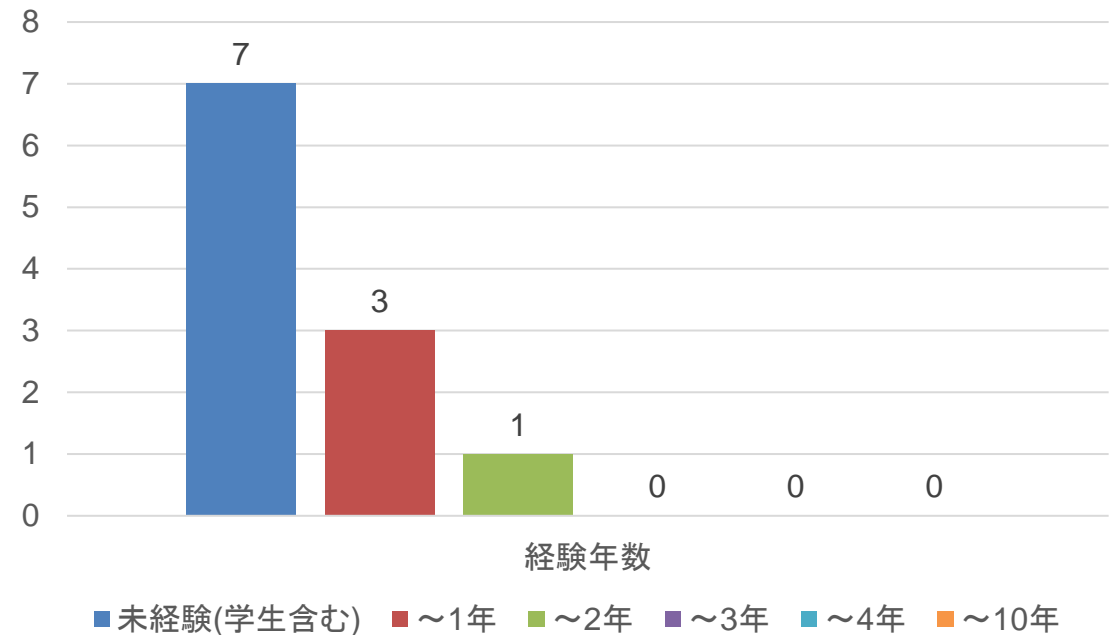
# 参加者の分布

★ 参加者総数：11名

参加費  
学生： ¥3,000  
社会人： ¥14,000

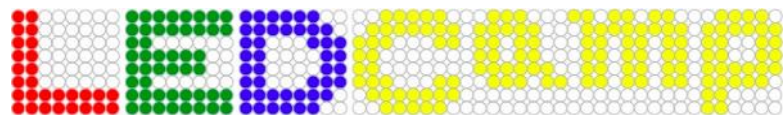


■ 学生 ■ 社会人



# 参加者の声（1）

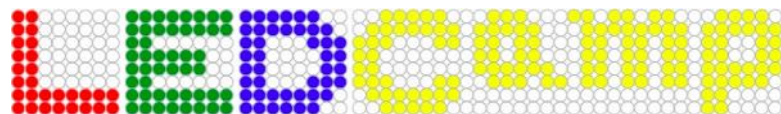
- ❖ スクラム開発，MDD等について今後さらに学ぶための大きなきっかけとなりました。ありがとうございました。
- ❖ あまり長くはなかったですが、ほかのチームの方とも交流できたのはよかったです
- ❖ オンラインでのコミュニケーションや、チーム開発には最初とても不安がりましたが、楽しく実習を進めることができました。貴重な経験ができる機会を用意していただき、ありがとうございました。





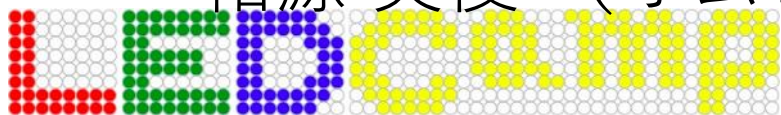
## 参加者の声（2）

- ❖ 最初は、オンラインでチーム開発ということもあり、不安が大きかったが、最終的に、チームのサポートもあり、気軽に相談することができて、開発に楽しむことができた。クラス図やスタートマシン図は、学校の実験で作成したことはあるが、図を使って組込みをしたことがなかったので、動くものを操作することができて、楽しかった。
- ❖ 見知らぬ人どうしてもとても楽しむことができました。大変貴重な経験ができたとも思います。ありがとうございました！
- ❖ もっと開発を作りこむ時間が欲しかったな、と思うぐらい熱中できました。



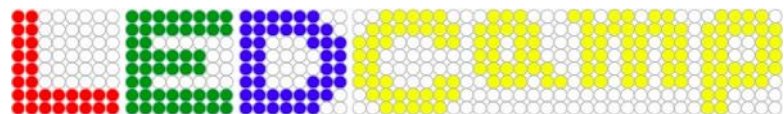
# 主催：LED-Camp実行委員会

- ・ 岩永 知裕 （チェンジビジョン）
- ・ 大栄 豊 （オーバス（デンソー））
- ・ 桐畑 鷹輔 （島津エス・ディー）
- ・ 中野 和香子 （日立製作所）
- ・ 浜名 将輝 （ゆめみ）
- ・ 山科 和史 （日立製作所）
- ・ 山本 健太 （デンソークリエイト）
- ・ 祐源 英俊 （オムロン）



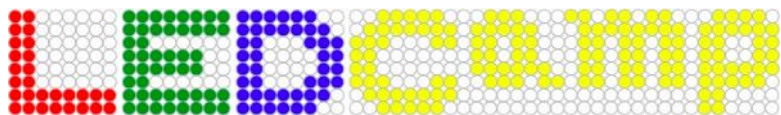
# 共催

- ❖ 一般社団法人 情報処理学会 組込みシステム研究会(SIGEMB)
- ❖ NPO法人 TOPPERSプロジェクト
- ❖ NPO法人 組込みソフトウェア管理者・技術者育成研究会 (SESSAME)
- ❖ 組込みシステム開発技術研究会(CEST)



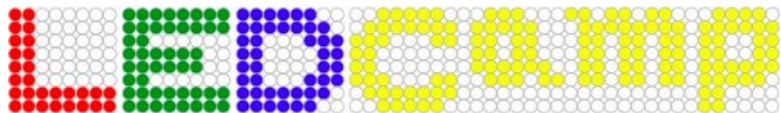
# 共催

- ❖ [一般社団法人 情報処理学会 システムとLSIの設計技術研究会\(SLDM\)](#)
- ❖ [システム開発文書品質研究会 \(ASDoQ\)](#)
- ❖ [ソフトウェア技術者協会 \(SEA\)](#)
- ❖ [車載組込みシステムフォーラム\(ASIF\)](#)
- ❖ [一般社団法人 組込みシステム技術協会\(JASA\)](#)
- ❖ [一般社団法人 電子情報通信学会](#)
- ❖ [NPO法人 ソフトウェアテスト技術振興協会\(ASTER\)](#)
- ❖ [一般財団法人 日本科学技術連盟](#)
- ❖ [派生開発推進協議会\(AFFORDD\)](#)
- ❖ [NPO法人 軽量Rubyフォーラム](#)



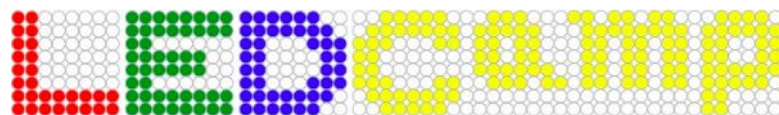
# 協力コミュニティ

- ❖ [NervesJP](#)
- ❖ [IoT ALGYAN \(あるじゃん\)](#)



# 協賛企業（口数順・申込順）

- ❖ [株式会社アルファプロジェクト](#)
- ❖ [STマイクロエレクトロニクス株式会社](#)
- ❖ [アイシン・ソフトウェア株式会社](#)
- ❖ [株式会社ティアフォー](#)
- ❖ [株式会社永和システムマネジメント](#)
- ❖ [エプソンアヴァシス株式会社](#)
- ❖ [APTJ株式会社](#)
- ❖ [京都マイクロコンピュータ株式会社](#)
- ❖ [株式会社チェンジビジョン](#)
- ❖ [東海ソフト株式会社](#)



EOF

**EOF**

