

# **FPGA**ボードと**Bluetooth**通信を行う **Android**アプリで操作する自動車の制作

## 最終発表

**122931B**

木村 駿

# 計画時のシステム構成

アプリ操作



信号を送信



Bluetooth

モーター制御により動作

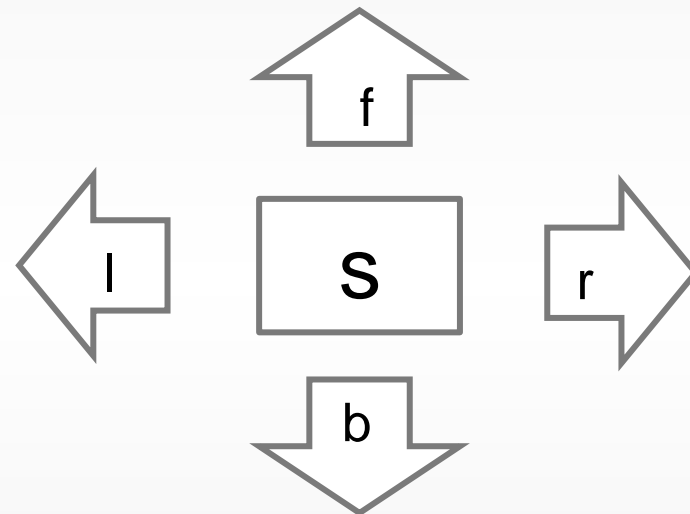


# 使用機器

- 自動車部分
  - ビュートロバー × 1
  - PmodBT2 (Bluetoothデバイス) × 1
  - PmodHB3 (モーターデバイス) × 2
  - BASYS2 (FPGAボード) × 1
- Android端末
  - SH-04E(Androidバージョン 4.4.2)

# アプリの仕様(通信仕様・計画)

- 5つのボタンを持つ。
- 各ボタンの仕様は以下の通り
  - ↑ : 文字 f (0110 0110)
  - ← : 文字 l (0110 1100)
  - → : 文字 r (0111 0010)
  - ↓ : 文字 b (0110 0010)
  - STOP : 文字 s (0111 0011)
- ボタンを押し続ける間、100msごとに信号を送信



# FPGAの仕様(機能仕様・計画)

- 信号を200ms間受信しなければ停止
- 各信号を受信したとき、表の動作をするよう、モーターモジュールに信号を送信する。

受信文字	左モーター		右モーター	
	DIR	PWM	DIR	PWM
f	+	1	+	1
r	+	1	+	0
l	+	0	+	1
b	-	1	-	1
s	x	0	x	0

- PWM制御に関する部分をモジュールに分割する。

# 開発結果(アプリ)

- 出来たもの
  - FPGAボード・Arduinoとの接続の確立
  - 仕様通りの信号の送信
- 出来なかったもの
  - 100msごとの信号の送信  
→ ボタンをクリックしたときのみ信号を送信

# 開発結果(**FPGA**)

目標到達度:70%

- 出来たもの
  - アプリとの接続の確立
  - 仕様通りのモーター制御
- 出来なかったもの
  - 200msごとの信号の受信確認
  - PWM制御を行うモジュールの分割  
→分割したところ、PWM制御が行えなくなった。値の受け渡し方がよくなかったか？

# 開発計画・実際の進歩(中間発表まで)

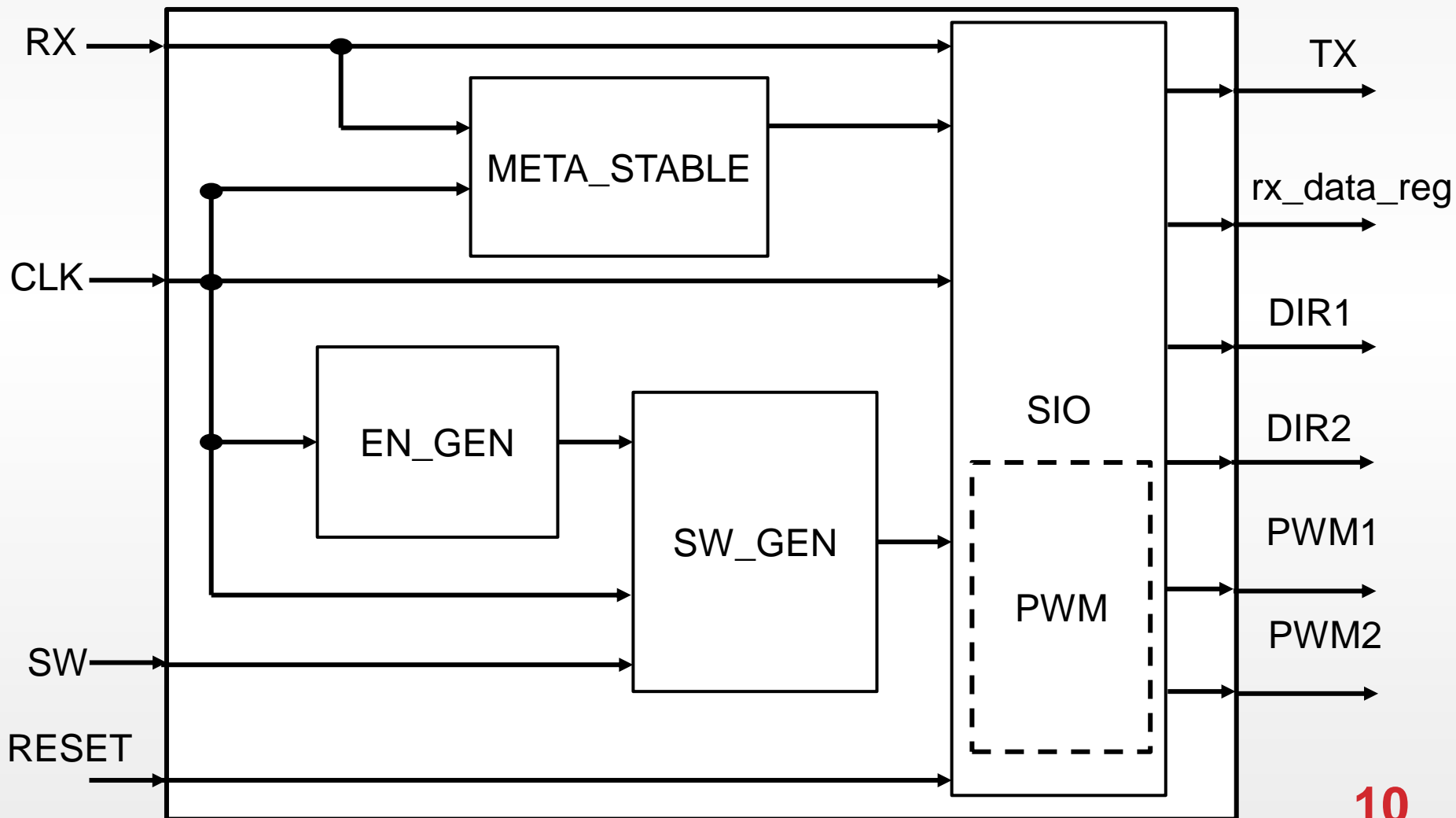
回数	日付	計画	実際の進歩
第1回	10月2日	オリエンテーション	オリエンテーション
第2回	10月16日	テーマ決定	テーマ決定
第3回	10月23日	開発計画の作成 アプリ開発環境のガイダンス	開発計画の作成 アプリ開発環境のガイダンス
第4回	10月30日	計画発表	計画発表
第5回	11月6日	Bluetooth通信・ Androidアプリ開発・ FPGAボード設計の基礎学習	Bluetooth通信 Androidアプリ開発 FPGAボード設計の基礎学習
第6回	11月13日	Androidアプリの制作	Androidアプリ開発の基礎学習 Androidアプリの制作(ボタン配置)
第7回	11月20日	Androidアプリの制作	FPGAボードの設計 (シリアル通信の受信を行う回路)
第8回	12月4日	Androidアプリの制作 FPGAボードの設計	FPGAボードの設計 (シリアル通信の送受信を行う回路)
第9回	12月11日	FPGAボードの設計	実機を用いた動作確認(TeraTermを用いた、 シリアル通信の接続テスト) FPGAボードの設計(PWM制御を行う回路)
第10回	12月18日	実機を用いた動作確認	実機を用いた動作確認 (モーターの制御テスト・TeraTermを用いた、 シリアル通信の信号を元にした モーターの制御テスト)
第11回	12月20日	中間発表	中間発表



# 開発計画・実際の進歩(中間発表以降)

回数	日付	計画	実際の進歩
第12回	12月25日 →1月8日	プログラムの修正 実機を用いた動作確認	通信・機能仕様の決定 PWM制御を行うモジュールの開発
第13回	1月8日 →1月15日	同上	PWM制御を行うモジュールの開発 実機を用いた動作確認 (モーターの制御テスト)
第14回	1月15日 →1月22日	同上	実機を用いた動作テスト (Androidアプリによるモーター制御)
第15回	1月22日 →1月24日	最終発表	最終発表

# FPGAのモジュール構成



# FPGAのモジュール構成

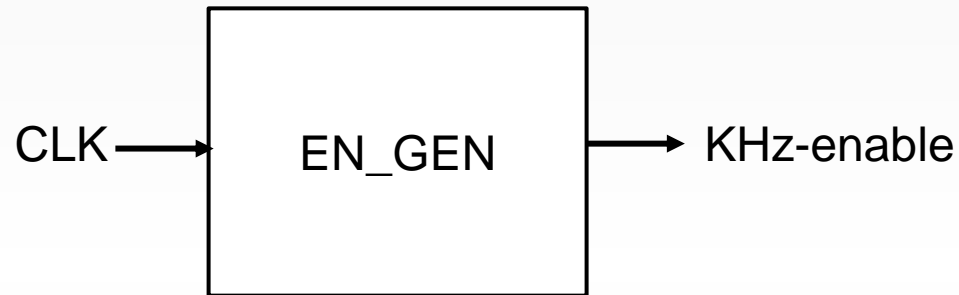
## META\_STABLE



- メタ・ステ이블(0と1の間の中間的な電圧が入力になることで出力が不安定になること)を回避する回路。
- RXをクロック1回分遅らせ、RX\_METAとして出力することでメタ・ステ이블を回避。

# FPGAのモジュール構成

## EN\_GEN



- イネーブルを生成する回路。
- 出力されたKHz-enableは後述のSW\_GENの入力として使用する。

# FPGAのモジュール構成

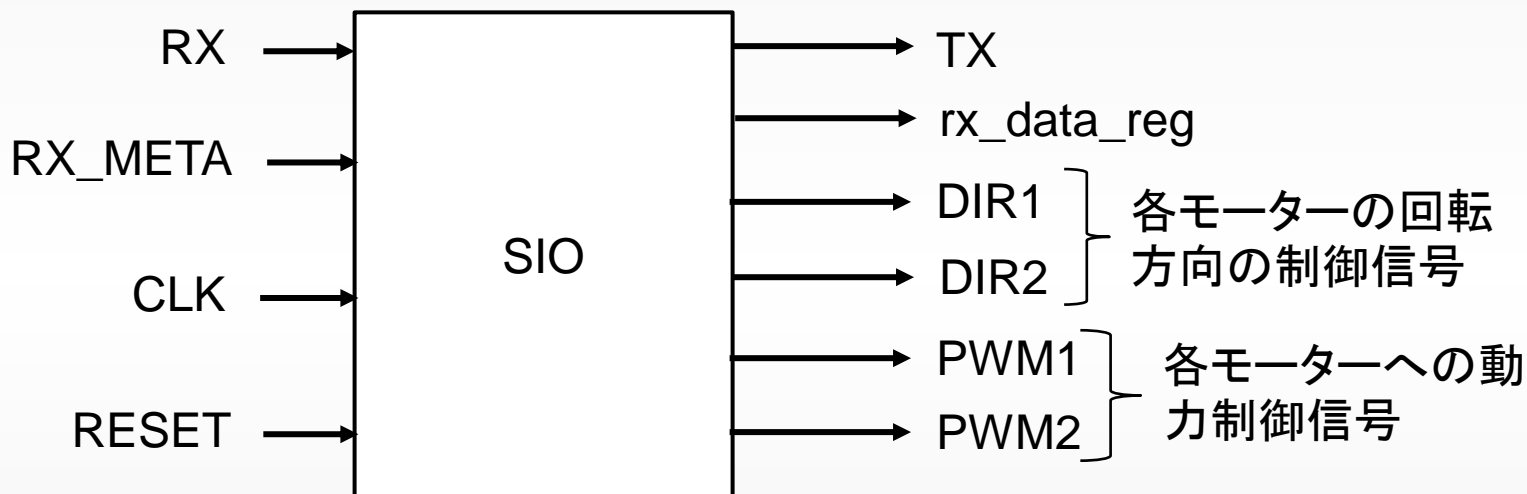
## SW\_GEN



- チャタリングの防止と立ち上がりの検出をする回路。
- 出力のsw\_one\_shotはトップモジュールのSIOで、受信データを送信開始するトリガとして使われる。

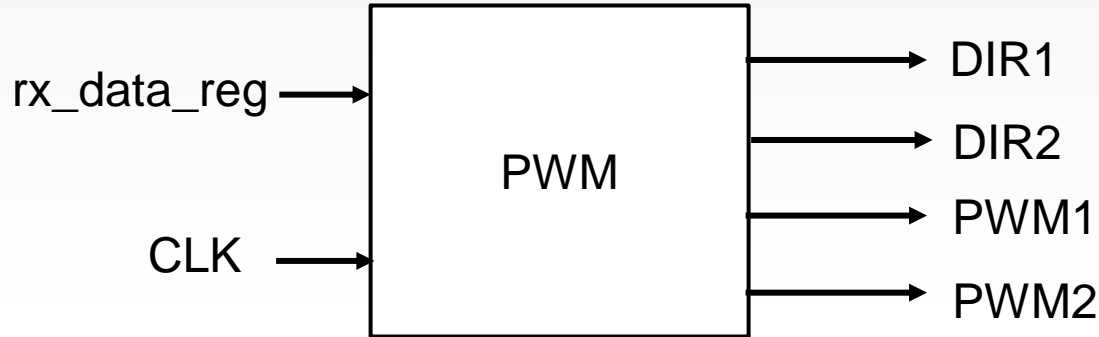
# FPGAのモジュール構成

## SIO



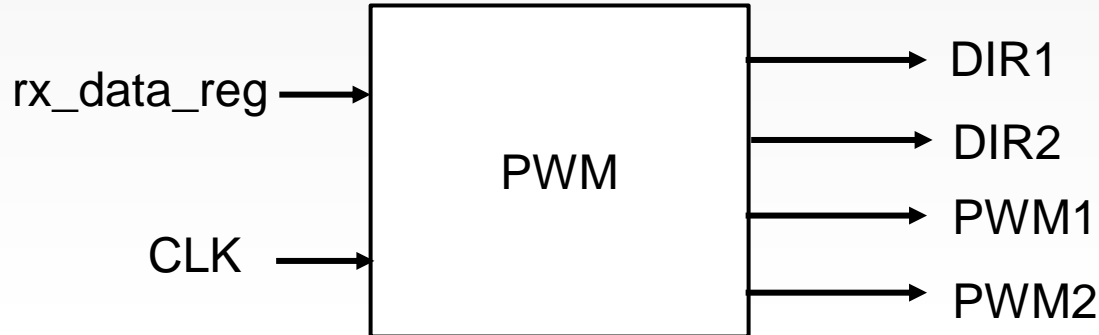
- シリアル通信での信号の送受信およびPWM制御によるモーター制御を行う回路であり、トップモジュール。
- TXはRXの値をそのままBluetoothモジュールへ送信する。
- rx\_data\_regはレジスタでもあり、RXの値を格納する。また、RXの値8ビットをLEDに表示するのにも使われる。
- DIRとPWMはモータードライバへ送信されモーターを制御する。

# PWM制御部



- 仕様決定時ではデューティ比60%で動作するよう設計  
→実機テストでタイヤが回転しなかったため、70%に変更
- 制御変数としてmodeを設定。デューティ比に基づき、5000ns中3500ns間modeに1を設定。それ以外は0。
- modeが1のとき、アプリから送信された信号の判定を行い、受信信号によりPWM1,2を送信。
- Modeが0のとき、's'を受信した時と同じ信号を送信。

# PWM制御部



- 下記の表に対応させてDIRとPWMを送信する。コードの組みやすさから 's' の時のDIRを1と設定。

受信文字	左モーター		右モーター	
	DIR	PWM	DIR	PWM
f	1	1	1	1
r	1	1	1	0
l	1	0	1	1
b	0	1	0	1
s	1	0	1	0



# 本講義を受けてのまとめ

- 当初の計画通りに進まず、1から計画・設計しモノを作り上げる大変さが実感できた。また、実機テストで思うように動くたびに達成感が味わえた。
- 中間発表時の見通しでは前後2方向に動けば、と低い目標設定だったが最終的には4方向＋停止が操作できるしっかりとしたものができたと思う。