

通信システム実験 第2週 実験レポート

信州大学工学部
電子情報システム工学科

実験日: 2023/04/27

実験場所: W2 棟 601 教室

気温: °C

湿度: %

実験者: 21T2166D 渡辺 大樹

共同実験者: 21T2164H 六川 沙絢

21T2167B 渡邊 大翔

21T2804J 伊藤 星斗

1 目的と概要

本実験では Arduino を用いてマイコンの制御、また前回用いたソフトである IFTTT を用いて近年話題となっている IoT(Internet of Things) についての理解を用いる。

1.1 事前学習

本実験の事前学習として以下 Arduino の標準ライブラリ、もしくは実験にて使用する温湿度センサー (DHT11) を用いるために用意されているライブラリの DHT.h にある 7 つの主な関数について調べた。上記 4 つの関数は Arduino 公式リファレンスの有志の日本語訳サイト [1] を参考になっている。DHT ライブラリの関数は公式ドキュメントが見つからなかったため、配布元の GitHub[2] にあったテストコードのコメントを参考に作成している。以下がその関数と調べた結果になる。

- `Serial.print()`

この関数は引数にきた文字、数字をシリアルポートに ASCII テキストとして出力する。浮動小数点はデフォルトだと小数点以下第 2 位までが送信される。これは第 2 引数で変更することができる。

- `digitalRead(pin)`

この関数は引数にデジタルピンの番号を取り、引数に指定したピンから HIGH もしくは LOW の値を読み取り、返す。

- `digitalWrite(pin, value)`

この関数は第 1 引数にデジタルピンの番号、第 2 引数に HIGH もしくは LOW を取る。この引数に、OUTPUT に設定されているピンを指定したときはピンの電圧が value に沿って HIGH(3.3V) もしくは LOW(0V) になる。逆に INPUT に設定されたピンを入力したときは、ピン内部のプルアップ抵抗の有効無効を切り替えることができる。

- `pinMode(pin, mode)`

この関数は第 1 引数で指定したピンを第 2 引数で入力に使うのか出力に使うのかを設定する。第 2 引数の Mode には INPUT もしくは OUTPUT、また INPUT PULLUP が入る。Mode を INPUT PULLUP にすることで内部のプルアップ抵抗を有効にすることが可能である。

- `dht.begin()`

この関数は DHT センサーを初期化するための関数である。

- `dht.readTemperature()`

この関数は DHT センサーから温度を読み取る関数である。戻り値として温度が出力される。引数に何もセットしなかった場合、Celsius 温度を、True を入力すると Fahrenheit 温度を出力する。

- `dht.readHumidity()`

この関数は DHT センサーから湿度を読み取る関数である。戻り値として湿度が出力される。

2 実験内容

本実験では Arduino を用いて二種の実験を行う。

実験 1 では赤外線モジュールを用いて赤外線通信を受信する実験を行う。赤外線モジュールと Arduino と接続し、eAlps 記載のコードをコンパイル、書き出して実行する。実行したのちに実際に赤外線通信を行っているリモコンをモジュールに向けて操作することで赤外線を受信する。

実験 2 では Arduino と IFTTT を用いて、温湿度センサーから Arduino で取得した温度を Web 上で確認する実験を行う。温湿度センサーには DHT11 を用いて Arduino と接続し、eAlps 記載のコードをコンパイル、書き出して実行する。コードでは Arduino を 601 教室のネットワークに接続し、Arduino から IFTTT に Web リクエストを行って、URL パラメータを用いて温湿度センサーから得た温度を IFTTT に送信している。

3 結果

以下に実験の結果を記す。

3.1 実験 1

実験 1 では以下図 1 のような回路を実装し、実験した。

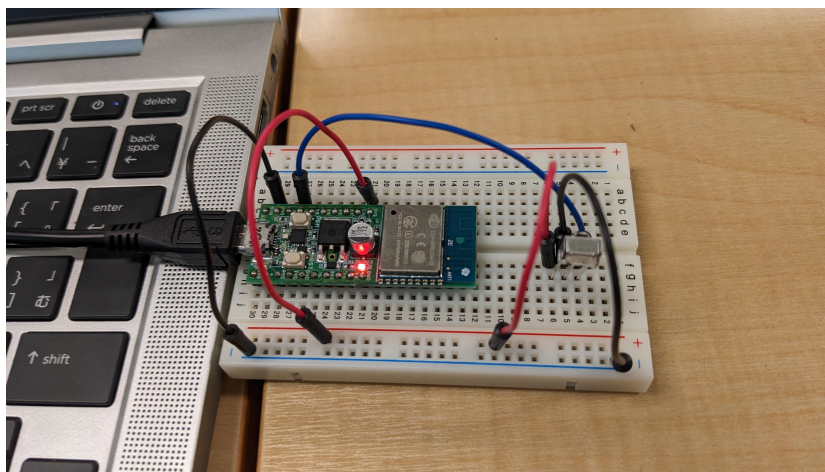


図 1 実験 1 で実装した回路

またこの図 1 の回路にリモコンで赤外線を当てたところ、以下図 2 のような出力が得られた。

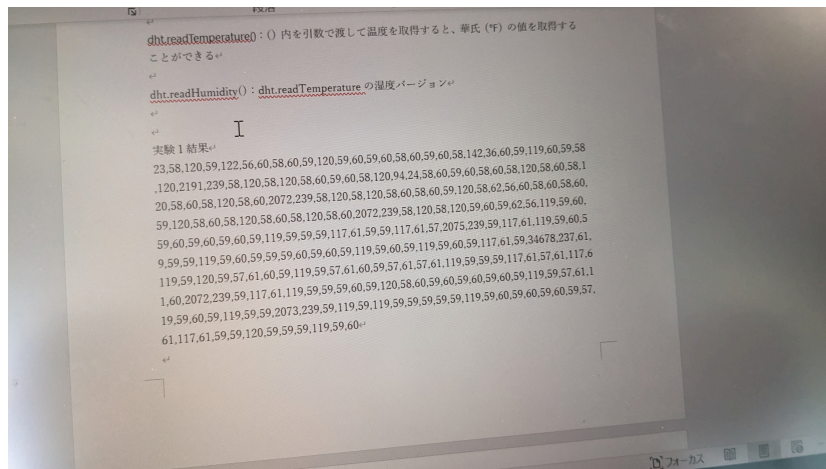


図 2 実験 1 での出力

3.2 実験 2

実験 2 では以下図 3 のような回路を実装し、実験した。

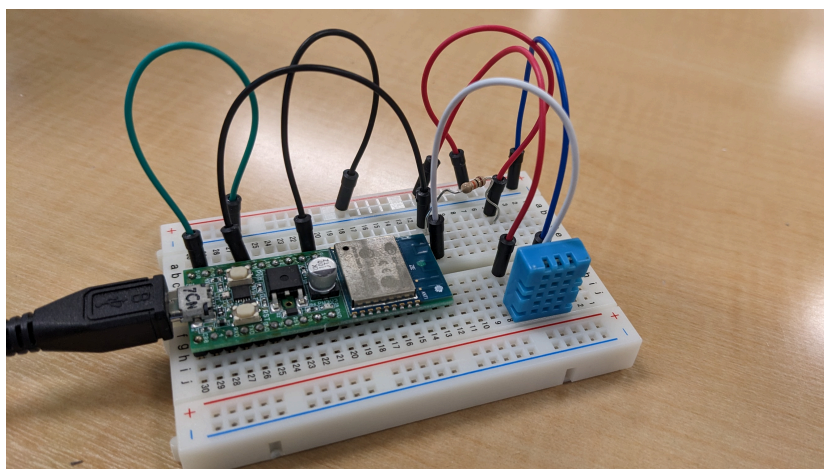


図 3 実験 2 で実装した回路

またこの図 3 の回路を動作させ、IFTTT にアクセスしたところ以下図 4 のような結果を得られた。

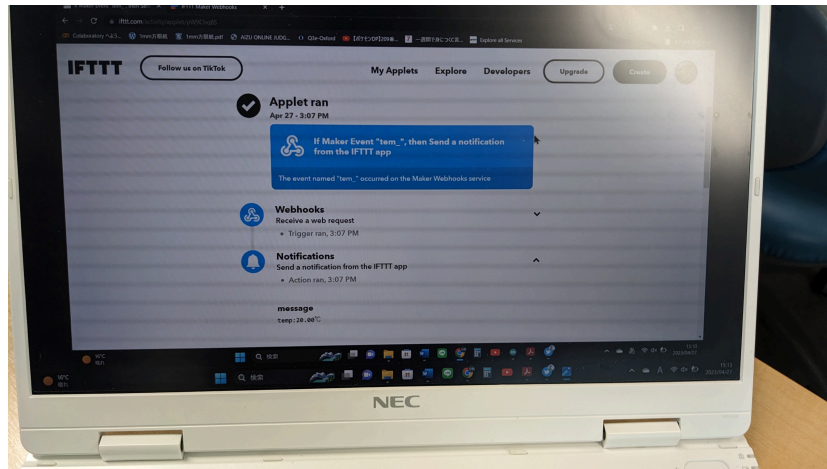


図 4 実験 2 での IFTTT の出力

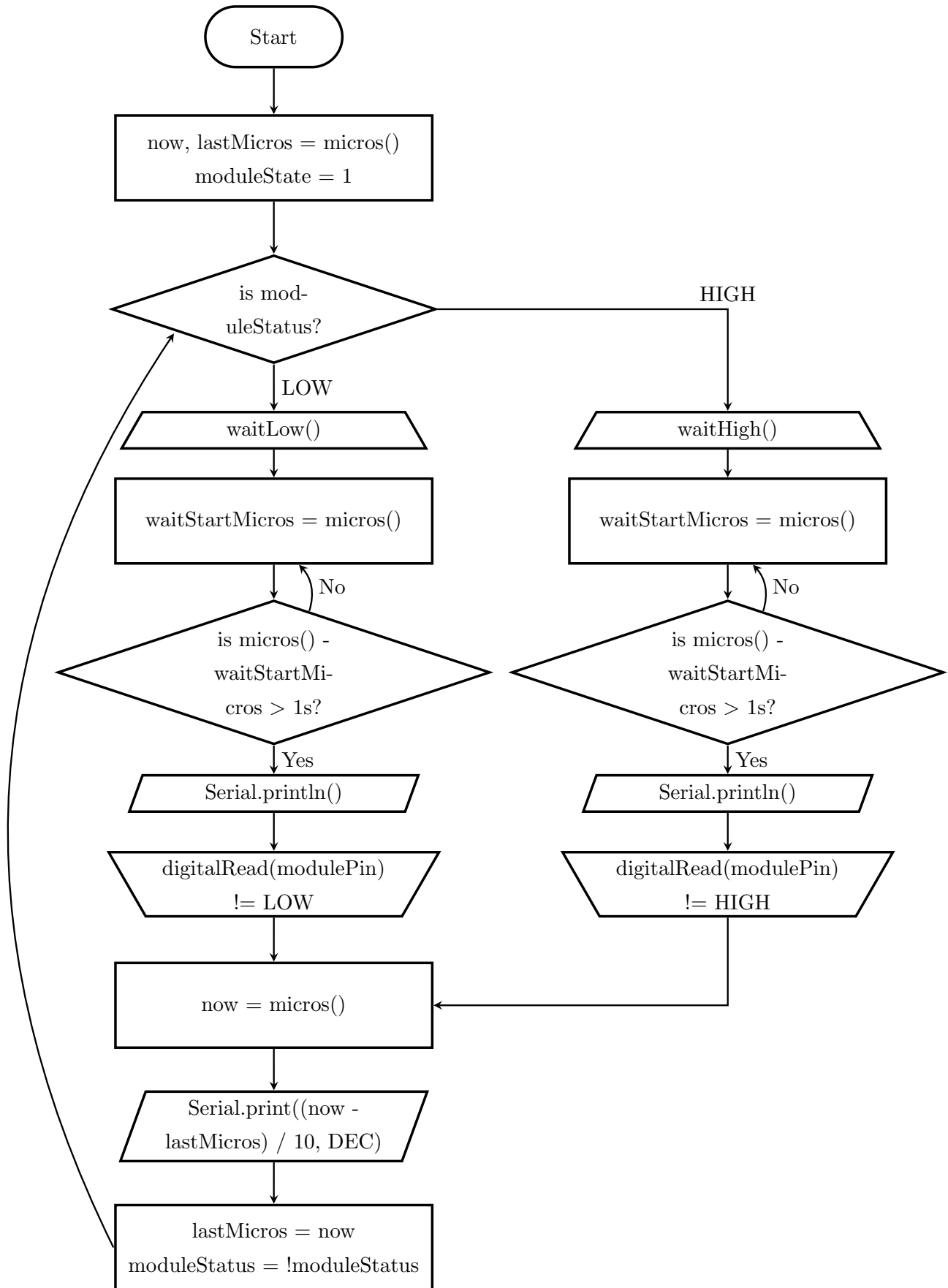
4 考察

以下にこの実験での考察を示す。

4.1 実験 1

実験 1 の結果、特に図 2 の結果から考察していく。図 2 ではリモコンからの赤外線の状態が変わらなかった時間が出力されており、このデータだけでは分析しづらいもののある程度周期的に変化しているのが分かる。特に 59、119 近傍の数字がよく出力されていることを考えると実験で用いたリモコンは $590\mu s$ ごとに振幅の変わる信号を用いているのではと推測した (つまり振幅変調?)。

また資料に記載されていたプログラム全文を載せることはできないため、以下に実験 1 で用いたコードのフローチャートを作成した。このフローチャートは \LaTeX の TikZ を用いて作成しているが、不慣れのため loop の処理は台形での表示としている。

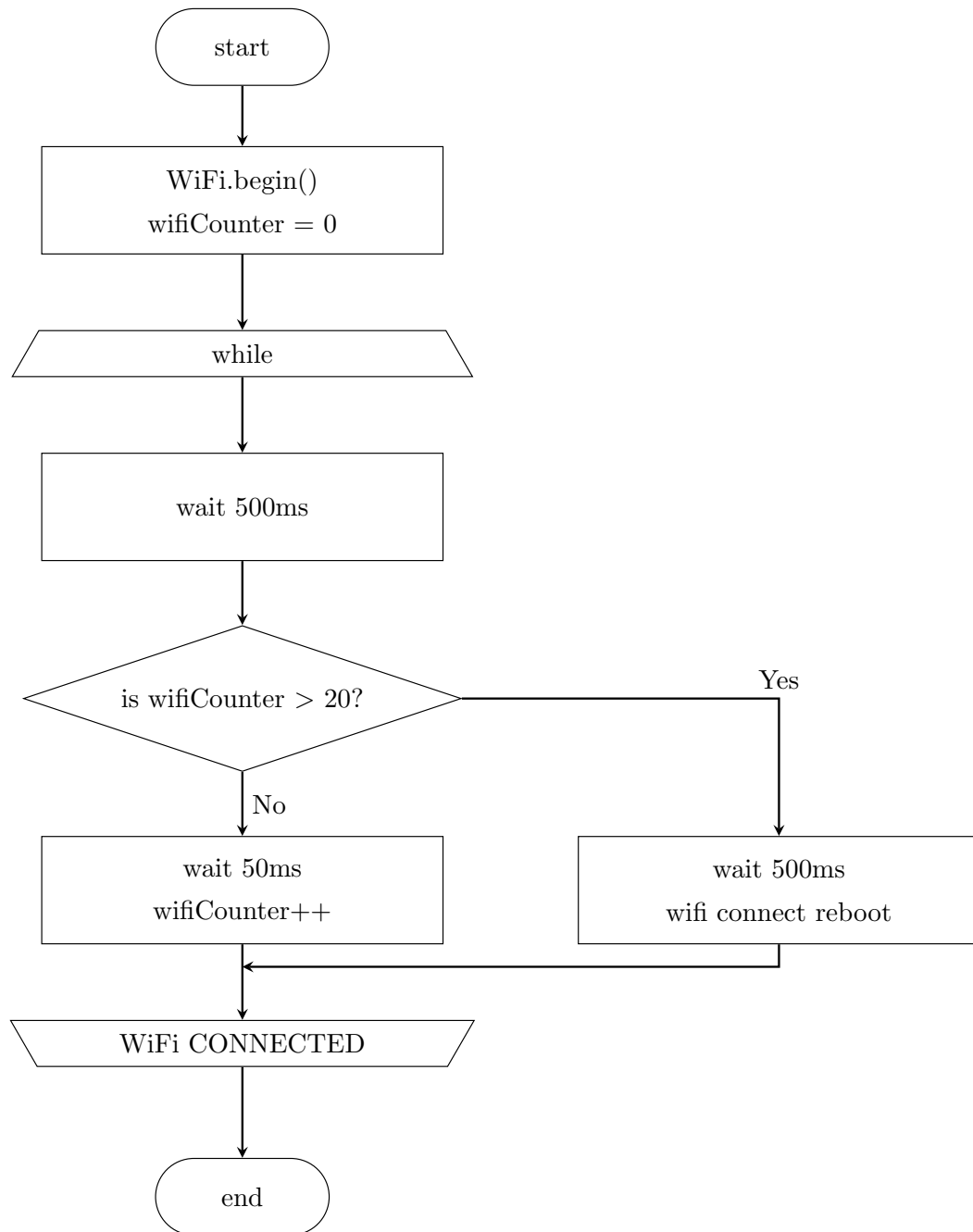


4.2 実験 2

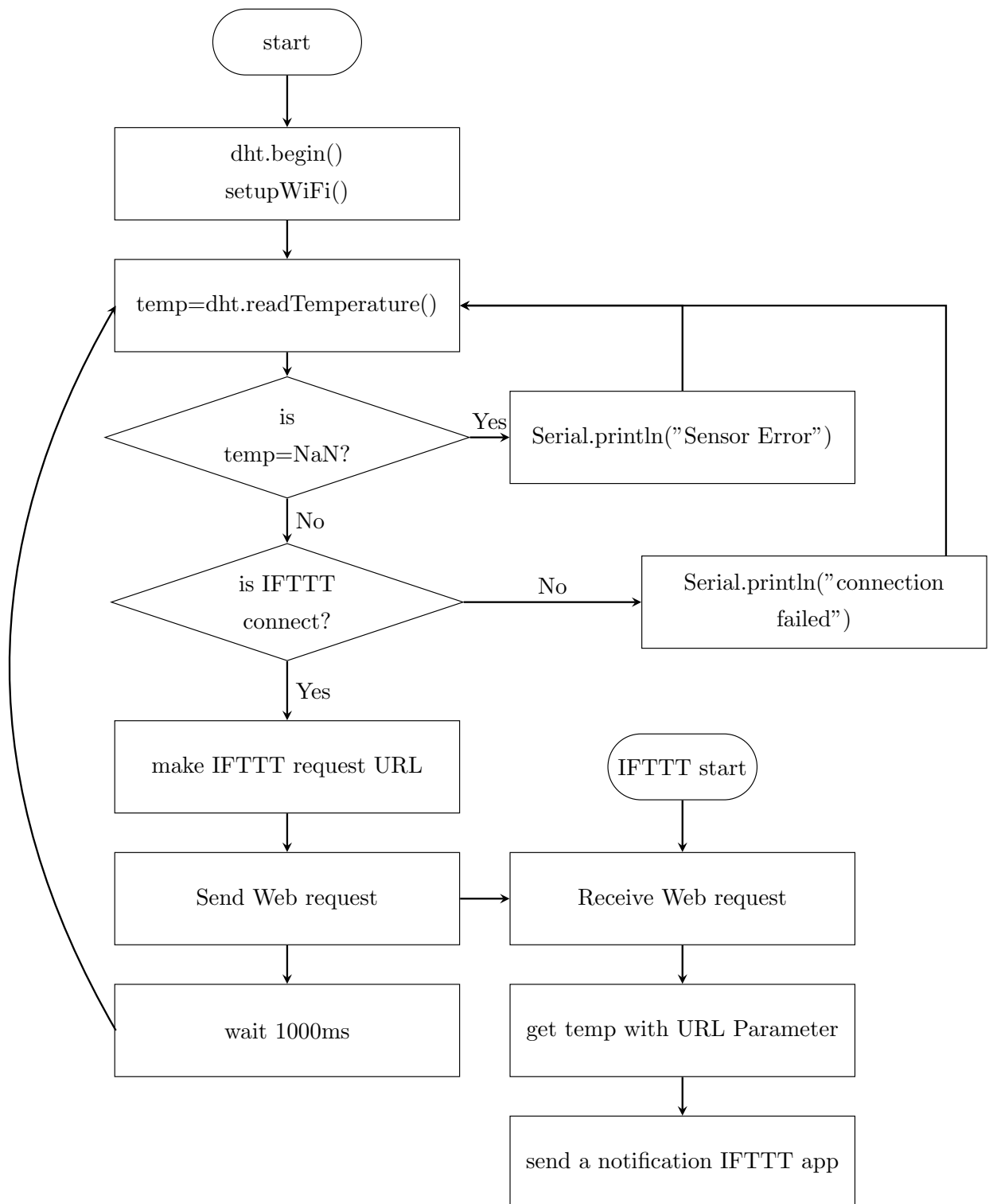
実験 2 の結果より室温が 20.00 °Cであると計測された。正確な他の機器での計測は行えていないが実際にこの日の室温は 20 °C前後であったため、おおよそ正しいといえる。

またこのプログラムでもコードをフローチャートにしている。複雑な関数がいくつか用意されていたため `setupWiFi()` のみ別表示としている。

Arduino (WiFi setup)



Arduino (Web request) and IFTTT



参考文献

- [1] Arduino 公式リファレンス 日本語訳ページ https://garretlab.web.fc2.com/arduino_reference/
- [2] GitHub DHT-sensor-library DHTtester.ino <https://github.com/adafruit/DHT-sensor-library/blob/master/examples/DHTtester/DHTtester.ino>