Mobile Robotics

Week 2: Multitasking & Wi-Fi

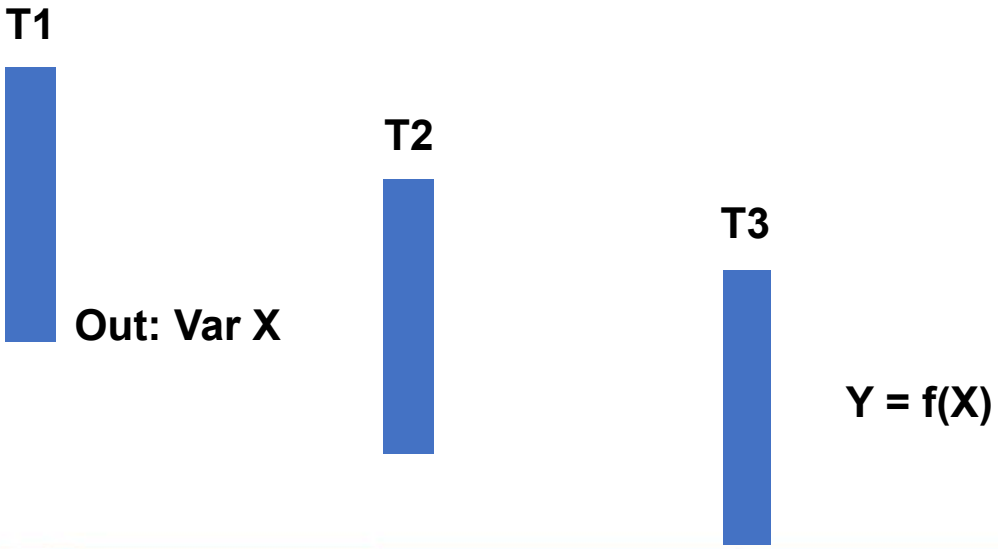
EEN1025 Semester 2, 2025

Jennifer Bruton

DCU Ollscoil Chathair Bhaile Átha Cliath Dublin City University

Multitasking

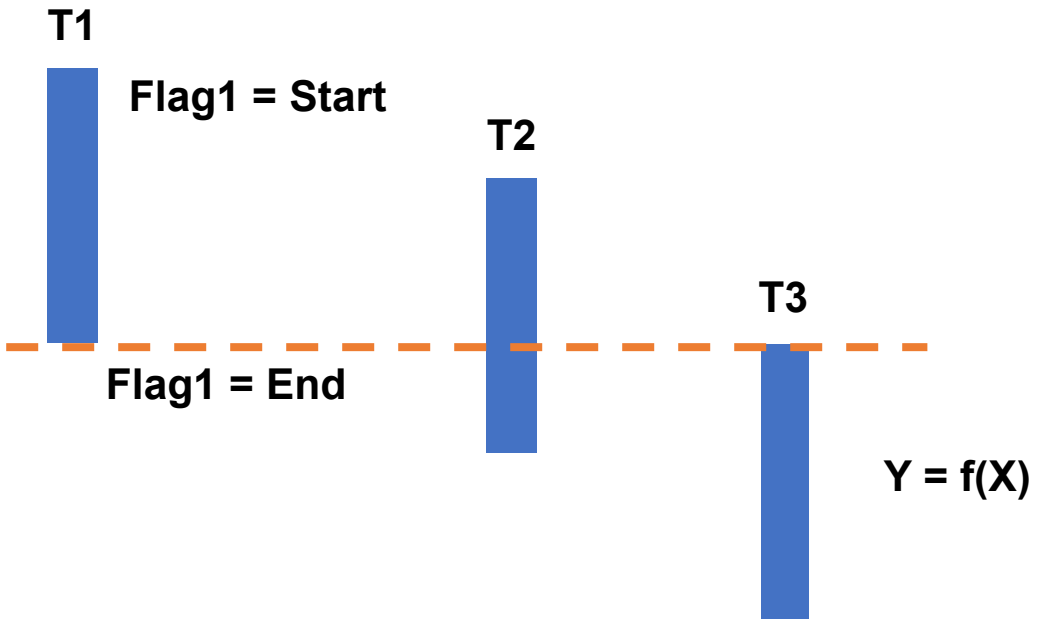
- In your code: you can define multiple loops and run these “at the same time”.
- Remember my baking analogy? – Check out example from the Arduino Forum
- But…how do you synchronize your tasks?



- Task 1 produces x
- Task 3 uses x
- How do you know when to start T3?

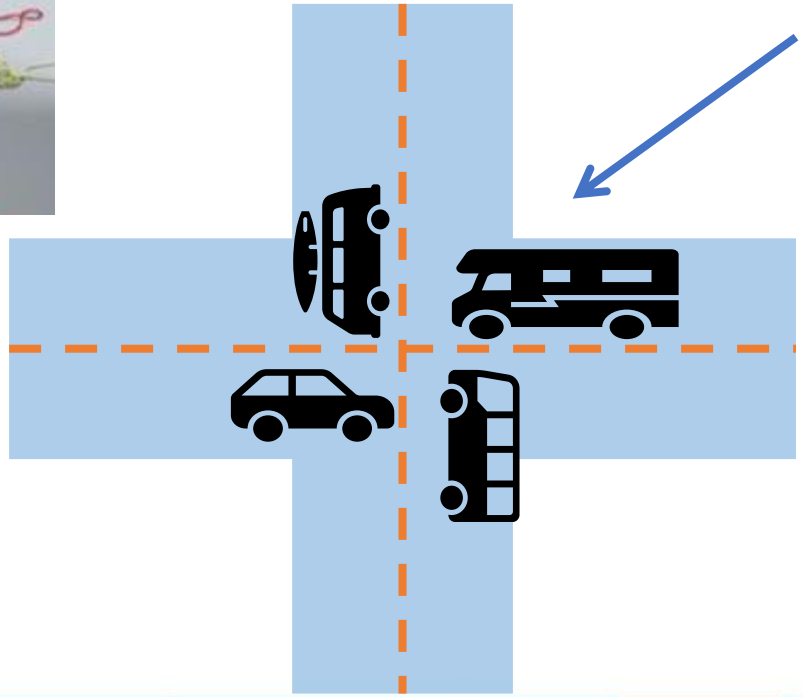
Multitasking

You could use flags / semaphores.




```
If (Flag1 == "End") then
        T3.start();
Else
        wait for T1 to
end!!!
```

Avoid Multiple Threads*



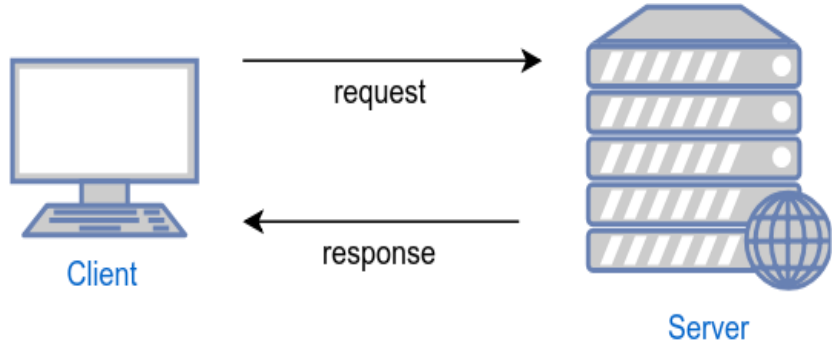
- Threads and Multitasking need very careful planning and management
- Can easily enter deadlock situation
- Can be difficult to debug
- Prefer fewer simultaneous tasks when possible

Delays – Dead Reckoning - Sensing

- Using **delay()**, can create blinking LED (on – delay – off – delay)
- Delays also used for switch debouncing
- **BUT** no other sensor reading, calculations, pin manipulation can happen during delay(), effectively bringing a single-thread programme to a halt.
- Most experienced programmers **avoid the use of delay()** for timing of events > 10s of milliseconds
- Tempting to ‘code to map’ **BUT** if there are changes to the map (e.g. FT301 vs FT307) or to the parameters of the mobot, code may not be robust
- **Better to use sensing** rather than delays or overly-prescribed programming
- **Dead-reckoning** (using speed, orientation & initial position) degrades over time, **should re-sense the environment** and re-start.

Communicating with the Cloud Server

- Summary of steps needed:
  1. Import Wi-Fi library
  2. Connect to Wi-Fi network using SSID
  3. Wait for connection to be established
  4. Wait for an IP address to be assigned by the network to your device
  5. Connect to server: IP Address: **3.250.38.184** TCIP port **8000**
  6. Send/receive message from server using the **HTTP protocol**

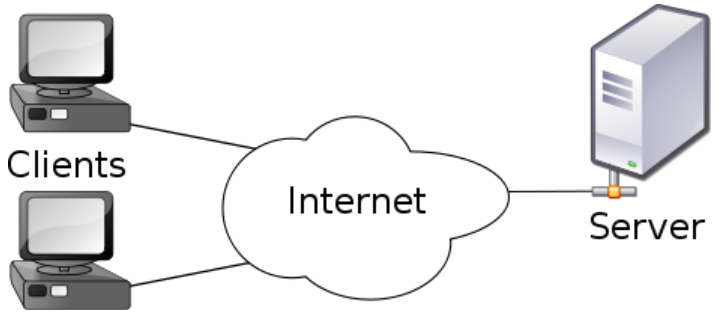


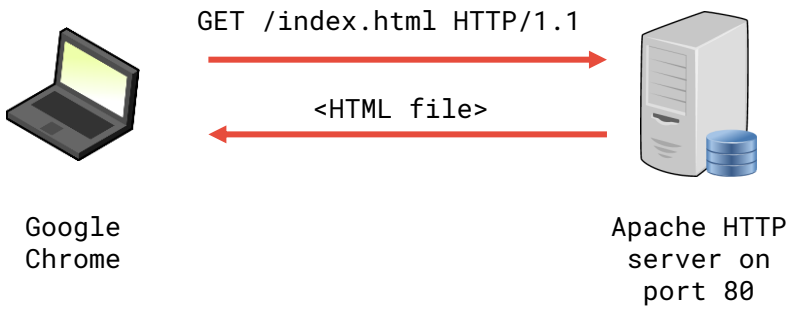
Connect to the Wi-Fi

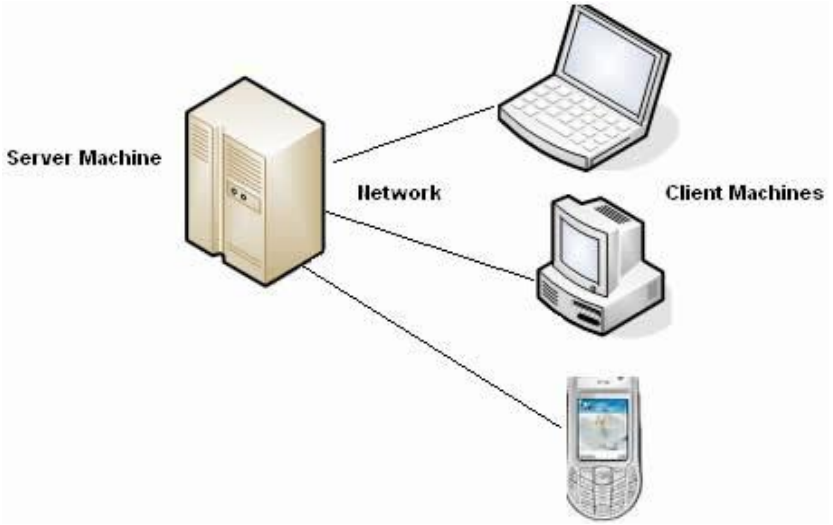
```
#include <WiFi.h>

// wi-fi details
char ssid[] = "***";
char password[] = "***";

WiFiClient client;

void connectToWiFi() {

  Serial.print("Connecting to network: ");
  Serial.print(ssid);
  Serial.flush();

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    Serial.flush();
    delay(300);
  }

  Serial.println("Connected");
  Serial.print("Obtaining IP address");
  Serial.flush();
  while (WiFi.localIP() ==
INADDR_NONE) {
    Serial.print(".");
    Serial.flush();
    delay(300);
  }

  Serial.println();
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}


void setup() {
  Serial.begin(9600);
  delay(1000);
  connectToWiFi();
}
```

Client-Server Technology

- Can have different configurations but the principles are the same.







ESP32 as a HTTP Client

ESP32-S3
DevKitC-1

```
POST /api/arrived/TEAM-ID
HTTP/1.1
```

3.250.38.184
Port 8000

```
HTTP/1.1 200
OK
…
…

1
```

ESP32 as a HTTP Server

192.168.x.x
Port 80

```
GET /index.html
HTTP/1.1
```

```
<HTML file>
```

ESP32-S3
DevKitC-1

HTTP Protocol

- An overview of HTTP
  - https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview
- The API documentation
  - http://ee303frontend.s3-website-eu-west-1.amazonaws.com/documentation

Note old project code URL is still valid

```
// server details
char server[] = "3.250.38.184";
int port = 8000;

…..

bool connect() {
  if (!client.connect(server, port)) {
    Serial.println("error connecting to server");
    return false;
  }
  return true;
}
```

Sending Messages

API documentation

The server that your robot will communicate with is at **http://3.250.38.184:8000**

Notify server of arrival at a destination

Send a post to **/api/arrived/TEAM-ID**, this request should contain your position in the body.

Example: To notify the server of arrival to position 0 the following post request would be sent.

POST /api/arrived/TEAM-ID HTTP/1.1

Content-Type: application/x-www-form-urlencoded
Content-Length: 10

position=0

```
  // post body
  String postBody("position=");
  postBody += position;

  // send post request and headers
  client.println("POST
/api/arrived/INSERT_YOUR_TEAM_ID_HERE
HTTP/1.1");
  client.println("Content-Type:
application/x-www-form-urlencoded");
  client.print("Content-Length: ");
  client.println(postBody.length());
  client.println();

  // send post body
  client.println(postBody);
```

Receiving Responses

where the `TEAM-ID` should be replaced by your own team ID. The server will respond with your next destination formatted in plain text after the HTTP headers in the response:

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Content-Type: text/plain; charset=utf-8
Content-Length: 1
ETag: W/"1-NWoZK3kTsExUV00Ywo1G5jlUKKs"
Date: Fri, 17 Jul 2020 16:16:57 GMT
Connection: keep-alive

1
```

Receiving Responses

```
// read buffer size for HTTP response
#define BUFSIZE 512

String readResponse() {
  char buffer[BUFSIZE];
  memset(buffer, 0, BUFSIZE);
  client.readBytes(buffer, BUFSIZE);
  String response(buffer);
  return response;
}

int getStatusCode(String& response) {
  String code = response.substring(9, 12);
  return code.toInt();
}

String getResponseBody(String& response) {
  int split =
response.indexOf("\r\n\r\n");
  String body =
response.substring(split+4,
response.length());
  body.trim();
  return body;
}
```

```
  int destination

  // read response
  String response = readResponse();

  // get status code
  int statusCode =
getStatusCode(response);
  if (statusCode == 200) {

    // success, read body
    String body =
getResponseBody(response);

    // check if at final destination
    if (!body.equals("Finished")) {
      destination = body.toInt();
    }
  }
```

Full Route & Disconnecting

```
// Get full route from Cloud Server

GET /api/getRoute/TEAM-ID HTTP/1.1     // replace TEAM-ID with your unique code
```

Note: you will still need to use the ‘Post’ command at each stage of the route so that the server car image and physical mobot correspond

```
// disconnect

  client.stop();
```

Your (2nd) Best Friend This Week Too!

Quick Start Guide