Using Android and Arduino to Develop a Universal, Smart Remote Control

Cassio Trindade Batista
cassio.batista.13@gmail.com

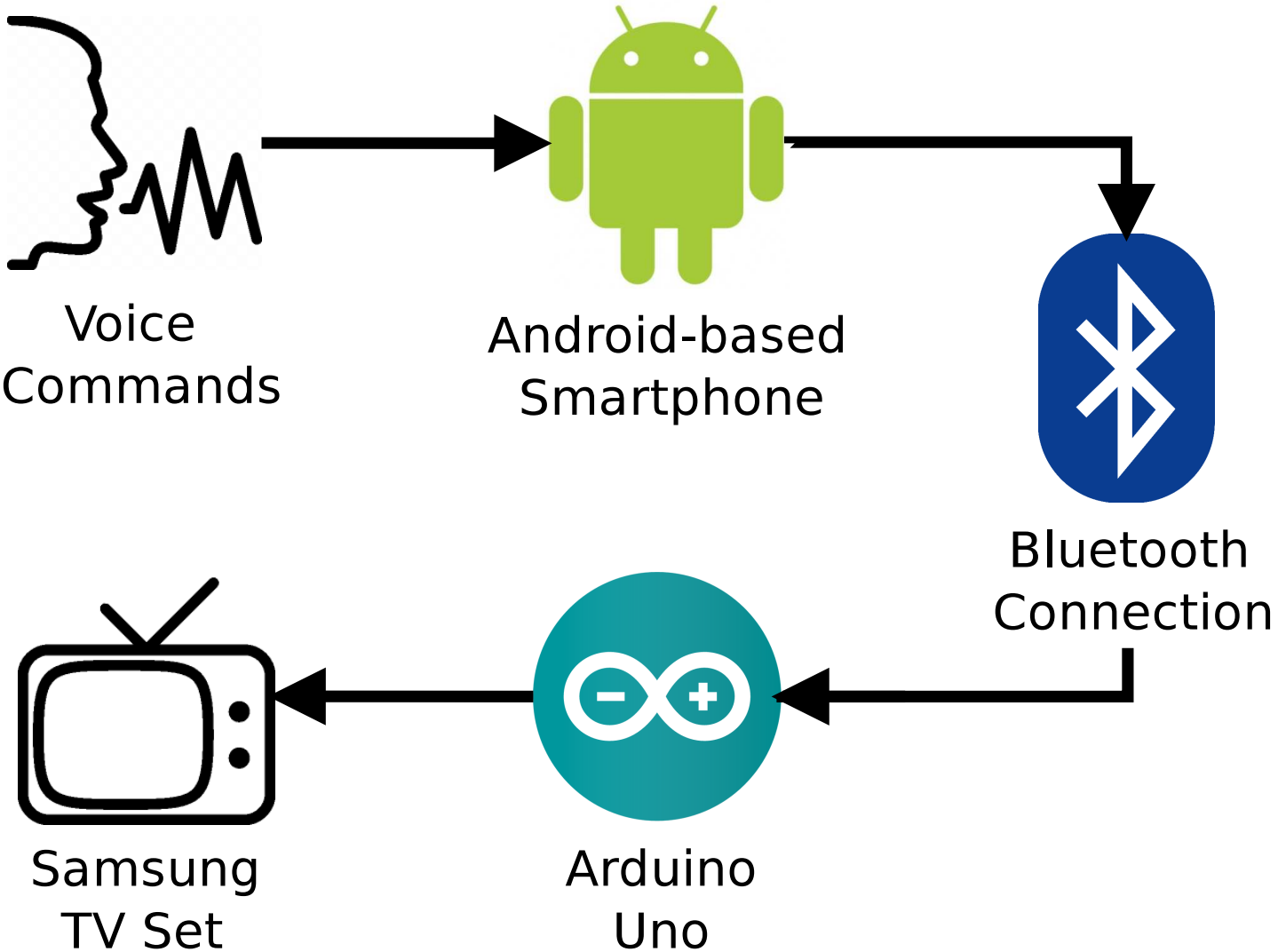
Universidade Federal do Pará
Programa de Pós-Graduação em Ciência da Computação
Centro de Estudos e Desenvolvimento dedicados à Engenharia
Arduino Day – Belém, PA – Brazil

April 1st, 2017

About.me

- ✓ Graduation in Computer Engineering (FCT, UFPA)
- MSc. student at Programa de Pós-Graduação em Ciência da Computação (PPGCC, UFPA)
- Fields of research: Speech Recognition, Alternative Communications, Assistive Technology

What am I gonna show?



Shall we test it already?

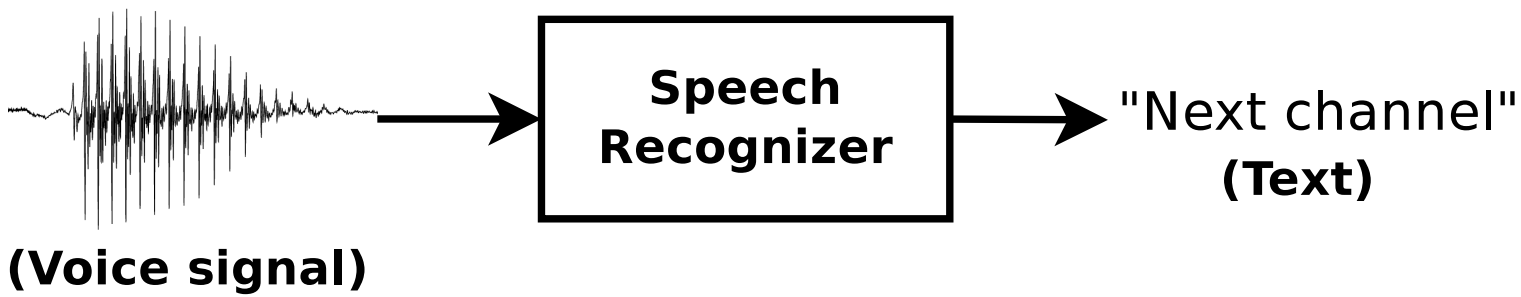
- “ligar televisão”
- “aumentar volume”
- “canal mais”
- “desligar televisão”
- “diminuir volume”
- “canal menos”

Cool, isn't it?

But how is it done?
How does it work?

Speech Recognition on Android: Native API

- Access native API via Intent calls
- `SpeechRecognizer` class
- `RecognitionListener` class



- Settings → My device → Language and input → Google voice typing → Offline speech recognition → Português (Brasil)

Bluetooth Tools

- Android:
  - `Bluetooth` API
  - Create a socket and send data through its OutputStream inside a thread
- Arduino:
  - `SoftwareSerial` library
  - Keeps checking whether is data on serial channel inside `loop()` function

Bluetooth on Arduino: HC-05 Module [1/3]

Bluetooth on Arduino: Config via AT+ [2/3]

```
1  pinMode(9, OUTPUT);
2  digitalWrite(9, HIGH); // set config pin HIGH
3
4  serialBT.write("AT+ORGL\r\n");
5  serialBT.write("AT+RESET\r\n");
6  serialBT.write("AT+RMAAD\r\n");
7  serialBT.write("AT+ROLE=0\r\n");
8  serialBT.write("AT+CMODE=1\r\n");
9  serialBT.write("AT+POLAR=0,1\r\n");
10 serialBT.write("AT+NAME=Tardis Blue\r\n");
11 serialBT.write("AT+PSWD=1234\r\n");
12 ...
13
14 digitalWrite(9, LOW);
```

Serial Port Bluetooth Module HC-05

Bluetooth on Arduino: receiving data [3/3]

```
1  #include <SoftwareSerial.h>
2
3  #define rxPin 2 //digital pin to receiver
4  #define txPin 5 //digital pin to transceiver
5
6  SoftwareSerial serialBT(rxPin, txPin); //define pins
7
8  void setup() {
9      pinMode(rxPin, INPUT);
10     pinMode(txPin, OUTPUT);
11
12     serialBT.begin(38400); //bauds to bluetooth
13 }
14
15 void loop() {
16     if(serialBT.available())
17         while (serialBT.available() > 0);
18             Serial.print((char) serialBT.read());
19     delay(200);
20 }
```

Infrared Communication: Samsung protocol

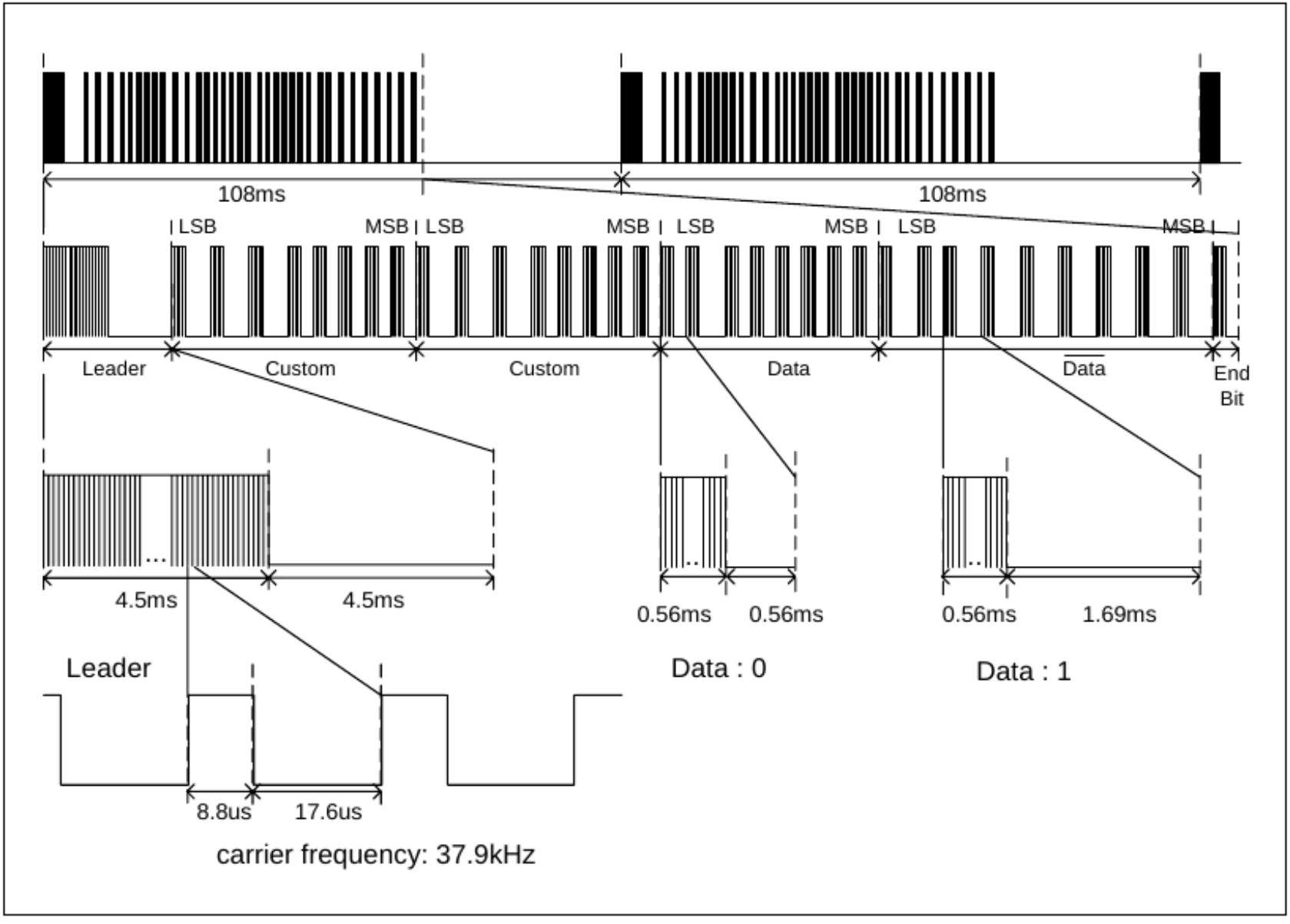
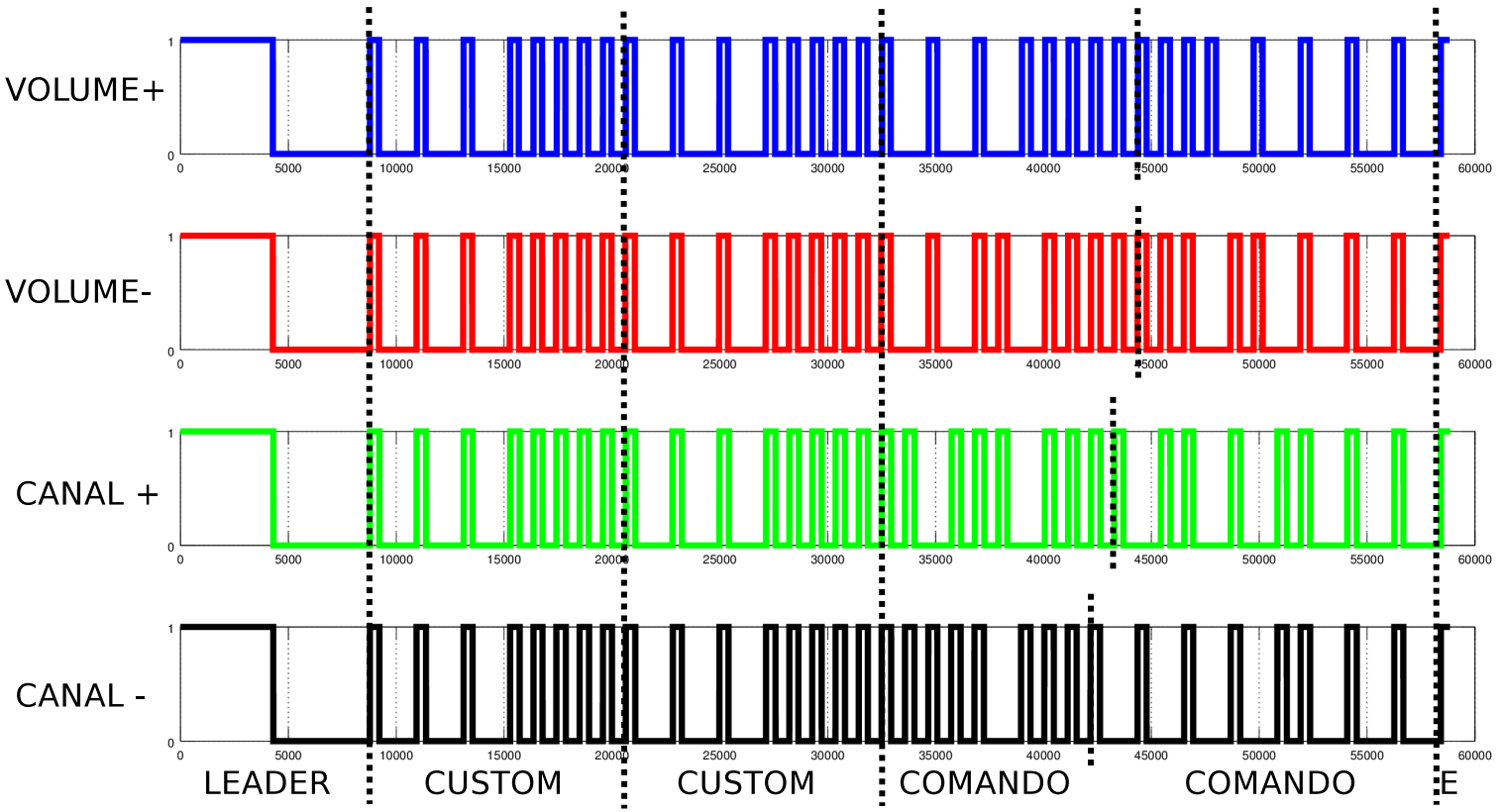


Figure 1. IR Signal

Infrared Communication on Arduino

- `IRremote` library
- Rx: Infrared sensor TSOP 18388



Infrared Communication on Arduino

- Tx: Infrared LED

```
1  #include <IRremote.h>
2
3  IRsend irsend;
4
5  void loop() {
6      for (int i=0; i<3; i++) {
7          irsend.sendSAMSUNG(0xE0E0D02F, 32); // volume--
8          delay(40);
9      }
10     delay(100);
11 }
```

- RC6 (Philips)
- NEC
- Sharp
- NEC
- Panasonic
- Sanyo
- Sony
- ...

Motivation: Naturalness

"it's hands-free", "that's cool", "it's more comfortable"

...

Real Motivation: Assistive Technology

For some people, that's the **only** way.

Thanks!

https://github.com/cassiobatista

cassio.batista.13@gmail.com