CSMFO LAB. CHARACTERIZATION and DEVELOPMENT of MATERIALS for PHOTONICS and OPTOELECTRONICS Laboratory

Consiglio Nazionale delle Ricerche

DSFTM
Dipartimento di Scienze Fisiche e Tecnologie della Materia

IFN
Unità di Trento
www.tn.ifn.cnr.it

FONDAZIONE BRUNO KESSLER

JUNIOR
RICERCA SCUOLA INNOVAZIONE

LEONARDO DA VINCI
LICEO SCIENTIFICO | TRENTO

Li-Fi Demonstration

NARDON GIANLUCA - BRIDI GIULIA

Liceo Scientifico L. da Vinci, Trento

Supervisors: Osman Sayginer and Maurizio Ferrari

Presentation Plan

1. **Introduction**
   - Telegraphy
   - Morse Alphabet
   - Arduino
   - MATLAB
   - Rot13
   - Li-Fi
2. **Methodology**
3. **Results**
4. **Conclusion and future works**

INTRODUCTION

TELEGRAPHY

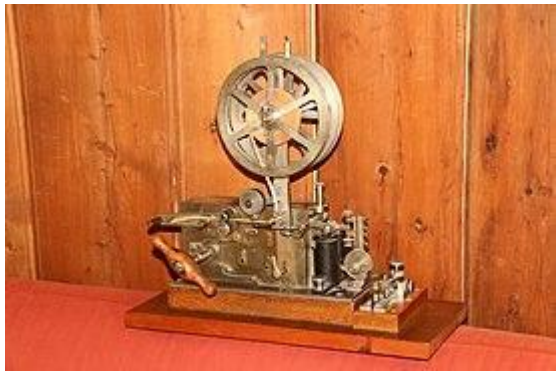
Long-distance transmission of messages with no physical exchange of an object, which bears the message

BUT

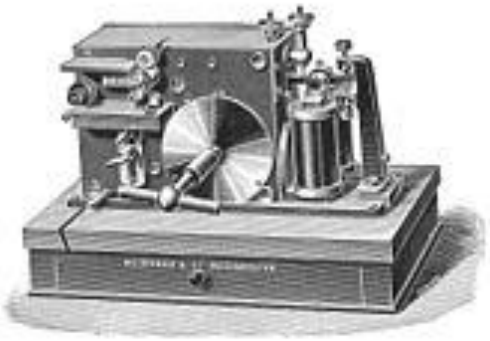
The method used for encoding must be known to both sender and receiver

OPTICAL TELEGRAPHY

ELECTRICAL TELEGRAPHY



WIRELESS TELEGRAPHY



OPTICAL TELEGRAPHY

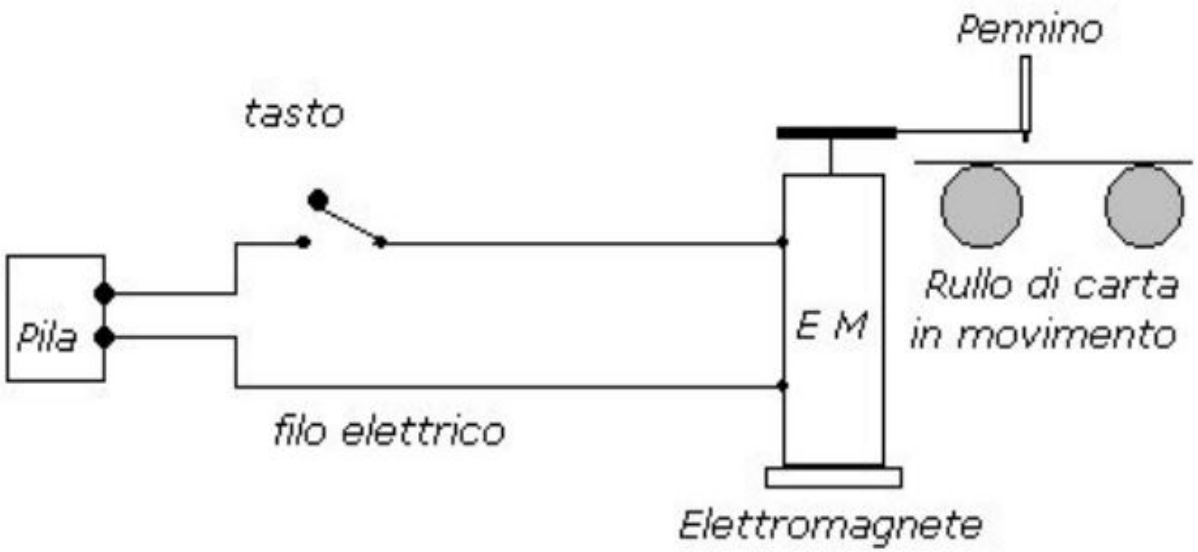
Invented by Claude Chappe in 1791, the first line, between Paris and Lila, was composed of 22 stations .
The system was built on elevated locations spaced one to the other for about a kilometer. These towers had a 10 meters pole, which enormous wooden arms were swirling on. The message was received by the telegraphist using the spyglass and decodified looking at the arms' positions.

ELECTRICAL TELEGRAPHY

SAMUEL MORSE

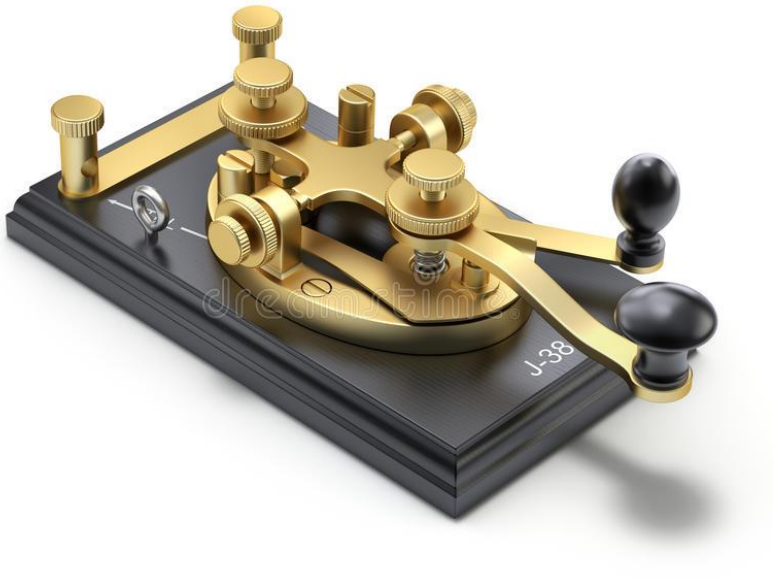
There are 3 parts:

- Battery
- Electromagnet
- Key to connect and disconnect those 2 parts



HOW DOES IT WORK?

Every time the key is tapped, an electrical pulse creates a beeping sound (short or long) or a pattern on a moving roll of paper (a dot or a dash)
These are letters and numbers, which can be used to create a message
This led the ability to send information and small text faraway at a very quick speed.



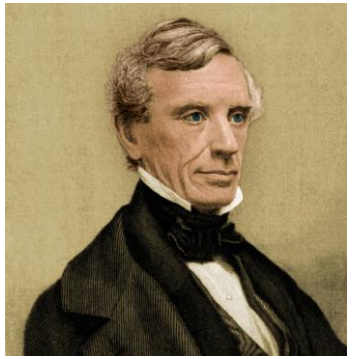
WIRELESS TELEGRAPHY

GUGLIELMO MARCONI

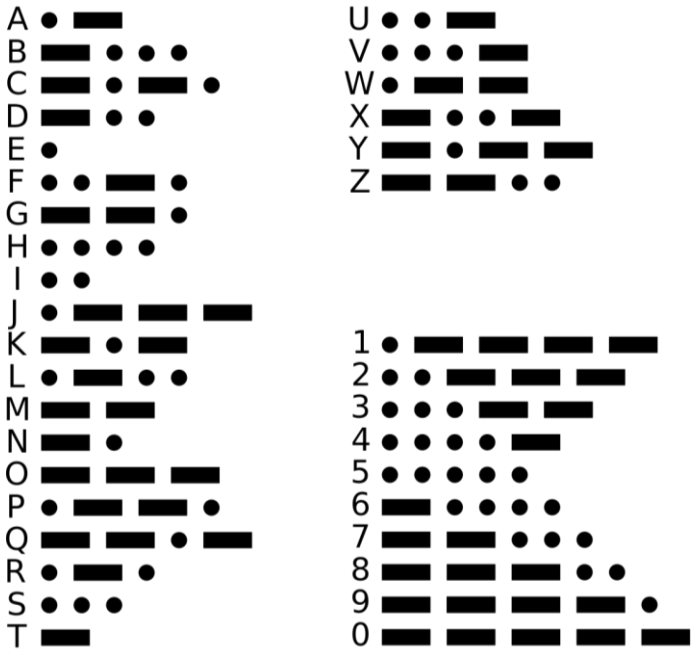
The system is very similar, but there are no cables which connect the transmitter with the receiver because wireless telegraph uses electromagnetic waves.

MORSE CODE

ORIGIN



- Was invented in 1835 by Samuel Morse, from whom Morse Code took its name, and was refined by his assistant Alfred Vail.



.... .- .-.. .-.. --- / .-- --- .-. .-.. -..

H E L L O W O R L D

HOW IT WORKS

- Each Morse code symbol represents either a letter or a number and is represented by a unique sequence of dots and dashes.

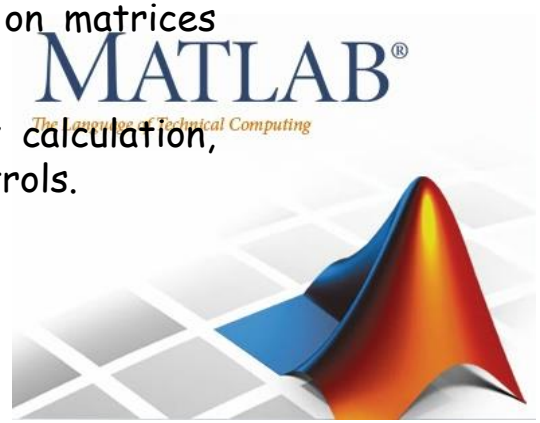
- The dot duration is the basic unit of time measurement in code transmission. The duration of a dash is three times the duration of a dot. Each dot or dash is followed by a short silence, equal to the dot duration. The letters of a word are separated by a space equal to three dots (one dash), and the words are separated by a space equal to seven dots.

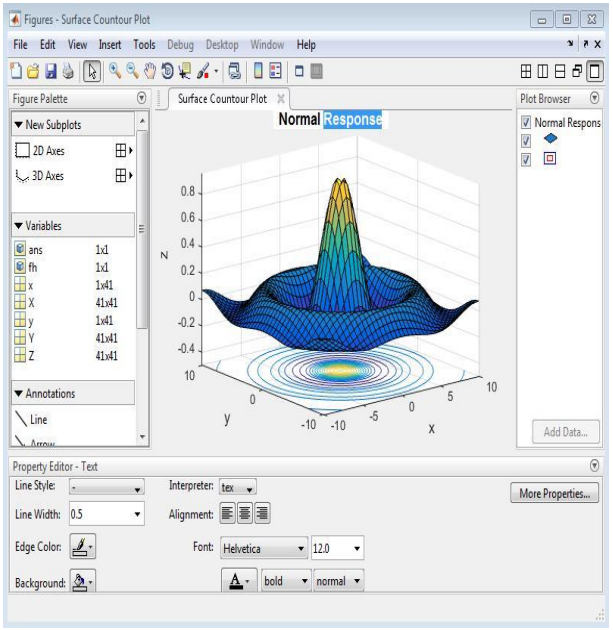
- To increase the speed of the communication, the code was designed so that the length of each character in Morse is approximately inverse to its frequency of occurrence in English. Thus the most common letter in English, the letter "E", has the shortest code, only a single dot

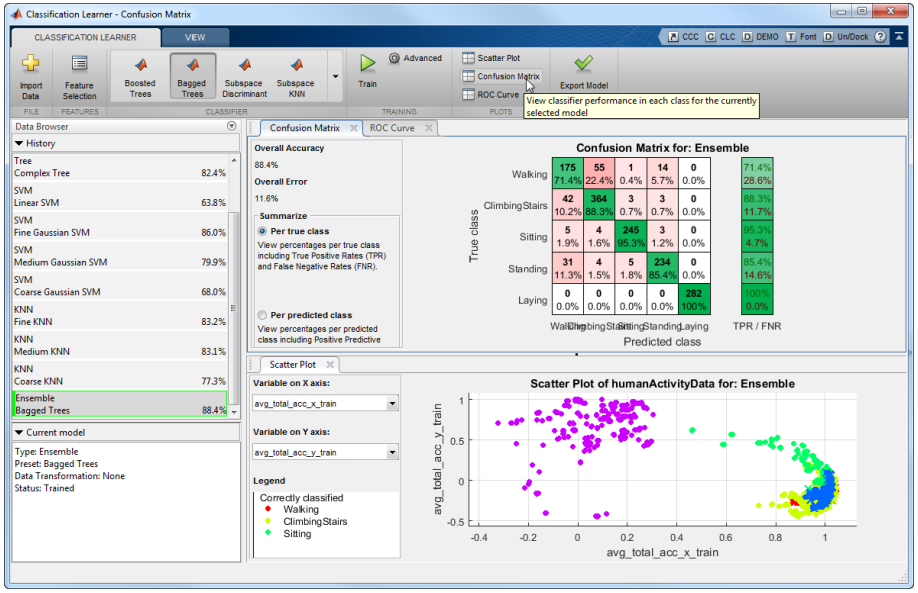
MATLAB

WHAT IS MATLAB?

- Matlab is an interactive program for numerical calculations based on matrices formed of both real and complex elements.
- It can be used for didactic and research in several fields: matrix calculation, implemented maths, bi and tri-dimensional graphics and automatic controls.
- It is written in C language







SOME EXAMPLES OF MATLAB PROGRAMS

ARDUINO

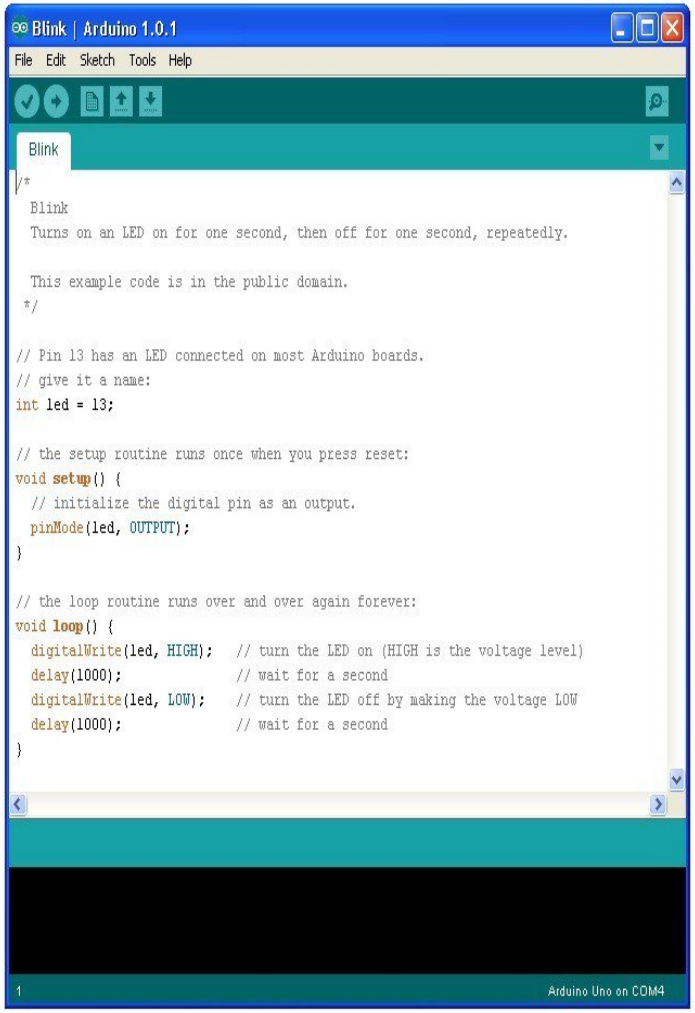
HISTORY

- The Arduino project started in 2003 as a program for students at the Interaction Design Institute Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.
- The name *Arduino* comes from a bar in Ivrea, Italy, where some of the founders of the project used to meet.

WHAT IS ARDUINO?

- Arduino is an open source computer hardware and software company, project, and user community that designs microcontroller kits for building digital devices and interactive objects that can sense and control other objects in the physical and digital world. The project's products are distributed as open-source hardware and software.
- Arduino board designs use a variety of microprocessors and controllers. The microcontrollers are typically programmed using the programming languages C and C++.
- Arduino's motherboard can interact with the environment in which it is by getting information from a huge variety of sensors. It can command lamps, motors and other actuators.

AN EXAMPLE OF ARDUINO PROGRAM



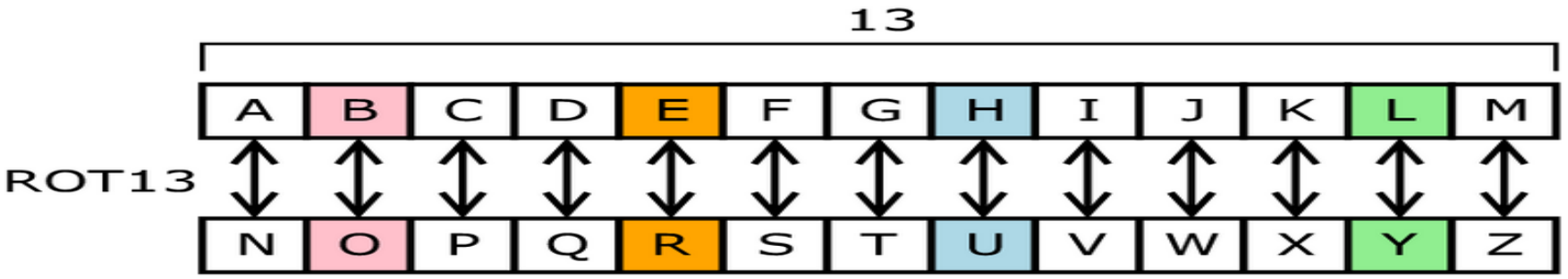
This basic program allows a led to blink for a determined period of time, which can be changed by increasing or decreasing the number in brackets

CRYPTOGRAPHY

- Cryptography is the process of **converting text** into unintelligible text and vice-versa
- It **stores and transmits data** in a particular form so that only those for whom it is intended can read and process it
- It **protects datas** from theft or alteration, but can also be used for authentication
- There are two main parts: the **algorithm** and the **key**. Without them, it is impossible to decrypt the message

ROT-13

- ROT13 is a simple **letter substitution cipher** that replaces a letter with the 13th letter after it, in the alphabet.
- Because there are 26 letters (2×13) in the Latin alphabet, ROT13 is its own inverse: that is - to undo ROT13 - the same algorithm is applied, so **the same action can be used for encoding and decoding**.
- However, the algorithm provides **no cryptographic security**, because of its weakness.
- ROT13 is nowadays used for hiding spoilers, punchlines and puzzle solutions in online forums



LI-FI

COMMUNICATION WITH LIGHT
DISCOVERED BY HARALD HAAS IN 2011

- The connection with Li-Fi (light fidelity) is 100 times **faster** than with Wi-Fi (Li-Fi has reached the connection speed of **224 Gigabits** per second)

- **Safer** communication (e.g. we can decide who may be connected to our Li-Fi network, avoiding people who want to steal our information)

- Li-Fi offers a **greater number of connected devices** into a smaller space (consequently, it brings to a higher speed of downloading and surfing than Wi-Fi)

PRINCIPAL ADVANTAGES

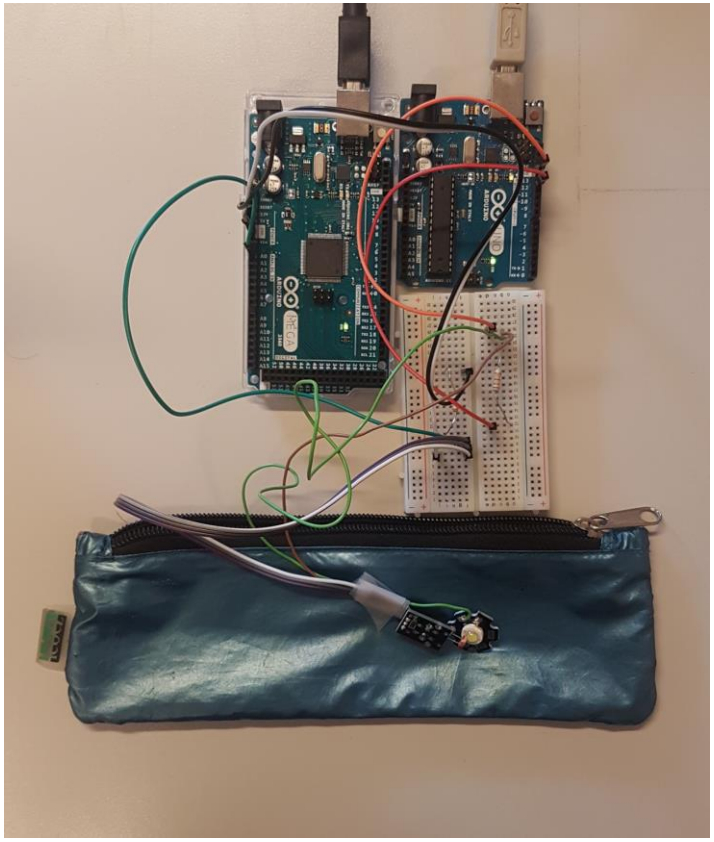
- It **does not interfere** with radio waves because the system uses the spectrum of visible light (this is, the frequency used is between 400 and 800 THz)
- Li-Fi uses light energy to work (it means **more data streams transmitted** at the same time)
- For this connection we have to use **special lamps** that emit in-phase waves, with a really fast blinking, undetectable by human eyes
- Because of the exponential growth of mobile devices, Li-Fi will became the **unique way to avoid unsustainable energy consumptions**

METHODOLOGY: OUR PROJECT

In this project we tried to communicate using light (the blinking of a LED)

We used:

- Arduino Uno
- Arduino Mega
- 2 computers
- MATLAB software
- Wires
- A breadboard
- Transistors
- A LED
- A photoresistor
- A case



1st PART: TRANSMITTER

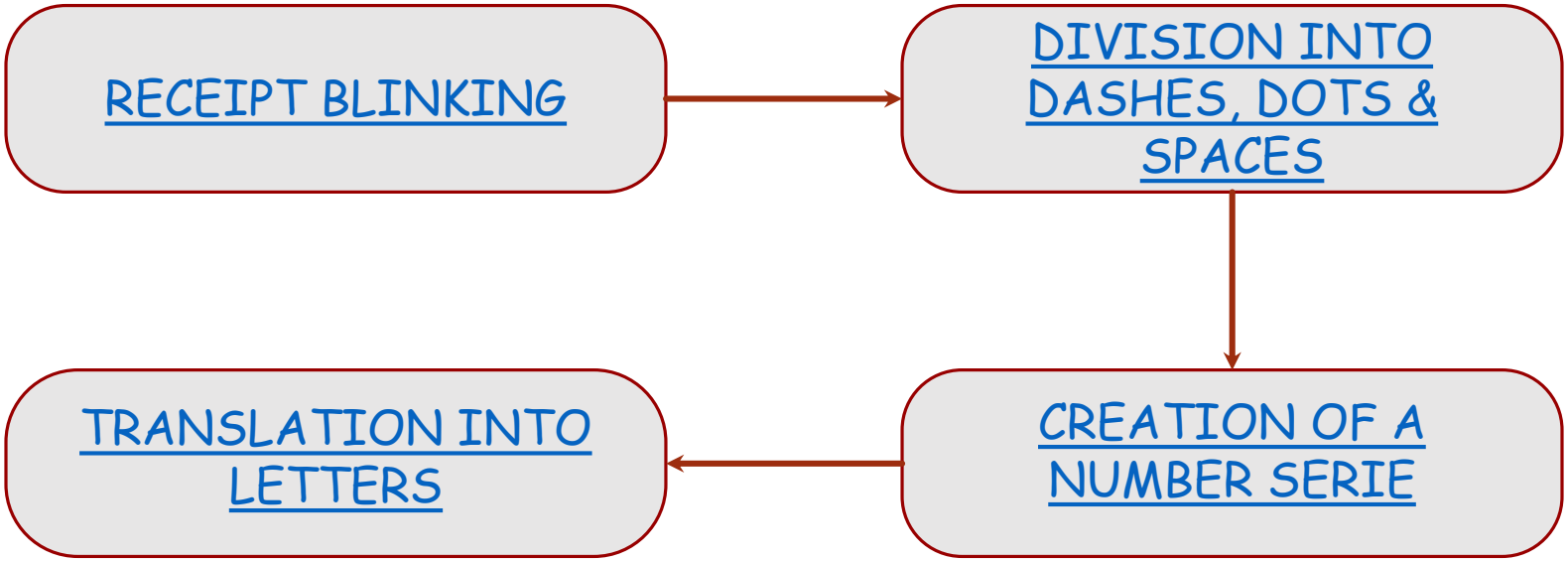
MESSAGE (LETTERS) → Morse Mapping → MORSE CODE

MORSE CODE → Integer Mapping → NUMBER SERIE

NUMBER SERIE → LED BLINKING

```
writeDigitalPin(a,'D12',1);
pause(1.5);
writeDigitalPin(a,'D12',0);
pause(1.5);
for k=1:length(NumberCoding)
    SeparateNumbers=NumberCoding(k);
    Ndigits = dec2base(SeparateNumbers,10)-'0';
    for j=1:length(Ndigits)
        Time=Ndigits(j)/8;
        if Ndigits(1)
            writeDigitalPin(a,'D12',1);
            pause(Time);
            writeDigitalPin(a,'D12',0);
            pause(Time);
        else
            writeDigitalPin(a,'D12',1);
            pause(Time*16);
            writeDigitaPin(a,'D12',0);
            pause(Time/2);
        end
    end
    writeDigitalPin(a,'D12',0);
    pause(1);
end
```

2nd PART: RECEIVER



CRYPTOGRAPHY

```
clear; clc;
a=arduino();
sensorPin = 'A0';
luceIniziale = readVoltage(a,sensorPin);
counter = 1;
while(1)
    valoreLuce = readVoltage(a,sensorPin);
    DataTable(counter,1) = valoreLuce;
    counter=counter+1;
end
```

This programme allows the **PHOTORESISTOR** to read the **Voltage** of our LED continuously.
In that way, we have a lot of values between 0 (completely bright) and 5 (totally dark)

```
WhiteSignalnfo = bwconncomp(~LightON);
for i=1:WhiteSignalnfo.NumObjects

    SignalToLetter=length(WhiteSignalnfo.PixelIdxList{1,i});

    if SignalToLetter <=17
        fprintf('DOT')
        WhiteSIGNAL(i,1)= 5;
    else
        fprintf('DASH')
        WhiteSIGNAL(i,1)= 1;
    end
end

BlackSignalInfo = bwconncomp(LightON);
for ii=1:BlackSignalInfo.NumObjects
    SignalToLetterBLACK=length(BlackSignalInfo.PixelIdxList{1,ii});
    if SignalToLetterBLACK >=40
        fprintf('SIGNAL SPACER');
        BlackSIGNAL(ii,1)= 2;
    else
        fprintf('NOISE');
        BlackSIGNAL(ii,1)= 3;
    end
end
```

Between dots and dashes (white signals) there is only one difference: the **length of the blinking**.
The same is also for short and long pauses.
With this programme we define **DOT** as less than 24 readings by the photoresistor, whereas **DASH** may assume all other values.
As far as black signals are concerned, **SIGNAL SPACER** (greater than 61) is the space between two letters, whereas all other signals are defined as **NOISE** (not useful for us)

```
WhiteSignalnfo = bwconncomp(~LightON);
for i=1:WhiteSignalnfo.NumObjects

    SignalToLetter=length(WhiteSignalnfo.PixelIdxList{1,i});

    if SignalToLetter <=17
        fprintf('DOT')
        WhiteSIGNAL(i,1)= 5;
    else
        fprintf('DASH')
        WhiteSIGNAL(i,1)= 1;
    end
end

BlackSignalInfo = bwconncomp(LightON);
for ii=1:BlackSignalInfo.NumObjects
    SignalToLetterBLACK=length(BlackSignalInfo.PixelIdxList{1,ii});
    if SignalToLetterBLACK >=40
        fprintf('SIGNAL SPACER');
        BlackSIGNAL(ii,1)= 2;
    else
        fprintf('NOISE');
        BlackSIGNAL(ii,1)= 3;
    end
end
```

If we want to translate the blinking into letters, it is necessary to divide the signals into dashes & dots (previous operation).

It is more important, though, to **distinguish** them **by** using **numbers:**

DOT = 5

DASH = 1

SIGNAL SPACER = 2

NOISE = 3

```
ALLSignals(ALLSignals == 3) = [];
SpacerLocations = [0 find(ALLSignals == 2)];
for ss = 1:length(SpacerLocations)-1
    start = SpacerLocations(ss)+1;
    stop = SpacerLocations(ss+1)-1;
    EachLetter = ALLSignals(start:stop);
    combineDigits = @(x) x*(10.^(length(x)-1:-1:0)).';
    MessageInNumbers(1,ss) = combineDigits(EachLetter);
end
```

This is also an important operation:
firstly, we **delete the values** we are not interested in (e.g. NOISE);
secondly, Matlab **finds SIGNAL SPACER locations** among all datas;
thirdly, we **define a letter** as all numbers between two SIGNAL SPACERS;
finally, we **split letters** and **save them** into different matrices.
In this way, every letter is separated and may easily understood by the software

The last operation allows Matlab to **recognise** the number series **and transform** them into letters.
There are 27 different possibilities and every string **must not be confused** with another letter (M and N are very similar also in Morse Code), otherwise there will be some problems with the translation

At the end of the whole process, **C** is the name of the variable, which **contains the words originally given**

```
newChr = num2str(MessageInNumbers);
C(MessageInNumbers==51)={'A'};
C(MessageInNumbers==1555)={'B'};
C(MessageInNumbers==1515)={'C'};
C(MessageInNumbers==155)={'D'};
C(MessageInNumbers==5)={'E'};
C(MessageInNumbers==5515)={'F'};
C(MessageInNumbers==115)={'G'};
C(MessageInNumbers==5555)={'H'};
C(MessageInNumbers==55)={'I'};
C(MessageInNumbers==5111)={'J'};
C(MessageInNumbers==151)={'K'};
C(MessageInNumbers==5155)={'L'};
C(MessageInNumbers==11)={'M'};
C(MessageInNumbers==15)={'N'};
C(MessageInNumbers==111)={'O'};
C(MessageInNumbers==5115)={'P'};
C(MessageInNumbers==1151)={'Q'};
C(MessageInNumbers==515)={'R'};
C(MessageInNumbers==555)={'S'};
C(MessageInNumbers==1)={'T'};
C(MessageInNumbers==551)={'U'};
C(MessageInNumbers==5551)={'V'};
C(MessageInNumbers==511)={'W'};
C(MessageInNumbers==1551)={'X'};
C(MessageInNumbers==1511)={'Y'};
C(MessageInNumbers==1155)={'Z'};
C(MessageInNumbers==55555)={'-'};
```

1x7 cell

	1	2	3	4	5	6	7
1	'C'	'N'	'R'	[]	'I'	'F'	'N'
2							

OUR CRYPTOGRAPHY PROJECT

```
Alphabet = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ-';


for EachletterOfMessage = 1: length(Message)
    k = strfind(Alphabet, Message(EachletterOfMessage),'ForceCellOutput',true);
    k = cell2mat(k);


    if k <=14
        Message(EachletterOfMessage)= Alphabet(k+13);
    else
        b = length(Alphabet)-k;
        Message(EachletterOfMessage)= Alphabet(13-b);
    end
end
```

We used cryptography (ROT-13) to send our message: in this way what we have written **remains secret until the final decryption**.
k represents the position of each letter in the alphabet: if k is less than 14, we **add 13**; otherwise, we **subtract 14** (this happens because we have 27 characters instead of 26).
Doing this, the sent message is apparently non-sense and can understood only with our key

Enter your Message:
"HELLO-WORLD"

Encryption with Rot13
"URYYAMIADYQ"

Morse Alphabet:
..- .-. -.-- -.-- .- -- .. .- -.. -.-- --.-

Numbers:
112 121 2122 2122 12 22 11 12 211 2122 2212

Blinking

Receiving the Signal With A Photoresistor

Noise Filtering

Classification of the Signal
(Dot, Dash, Space)

Extract Message

Decryption Message

Received Message

RESULTS

Thanks to this program, we are now able to **transmit** an encrypted message and **decrypt** it immediately after its arrival on the other computer.
It means that two computers can exchange messages and information using a code they only know.
Moreover, there is **no risk** for devices' message **to be intercepted** and translated (unless someone has the key used for encryption), because they are sent using light energy instead of travelling in the form of electromagnetic waves.

SOME POSSIBLE IMPROVEMENTS

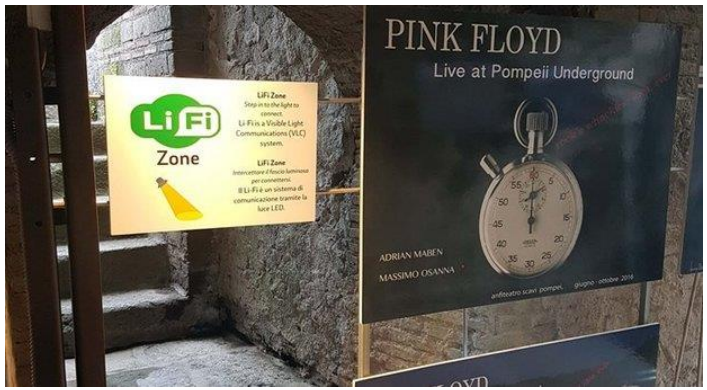
However, to work in a more satisfactory way must the system be improved a lot: for instance, if we had had a **more accurate photoresistor** and an **instant connection** between the software and Arduino, the speed of transmission could have been drastically reduced (instead of 40 seconds the entire operation would have taken less than 5).
Besides, the more the photoresistor is precise, the more the **message length might be increased** with no risks of misinterpretations.

Nonetheless, the system works satisfactorily for us although sometimes there are small inaccuracies either in data transmissions or in receipts, which inevitably bring to errors during the final translation.

FUTURE WORKS IN ITALY

Li-Fi network is now a reality, even though it could still be improved, an installation was made at **Pompeii**.
The Pompeii Li-Fi Project was developed by the **italian startup *ToBe*** to make visitors live a better experience into the archeological site thanks to Li-Fi LED Lamps placed near each point of interest.

Because of the possibility of modulating the incoming electricity, we can make some LED flicker very fast (so that we cannot see their blinking).
These impulses may be received with a photodiode that now we can find in our phones' camera and with the help of an application we are able to obtain information of what we have near us.



FUTURE PROJECTS

Oledcomm society presented **My Li-Fi** at CES 2018, a LED lamp that exploits the Li-Fi technology and spreads internet on users' computers. With this product will therefore be sufficient to turn on a little light to ensure internet at home.

Philips created office lamps entitled to Li-Fi, which give a faster and safer connection (30 Mbps).
There are several societies like **Icade** (it's carrying out a project for public places in Paris) and **Rod Lighting** that has developed a model of data communication by the use of bright bundles.

Recently, **Dubai** declared that in 2021 it will be the first city connected with Li-Fi, becoming one of the most advanced Smart City of the planet.