

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
KHOA KỸ THUẬT ĐIỆN TỬ I**



**BÁO CÁO BÀI TẬP LỚN  
MÔN HỌC XỬ LÝ ẢNH SỐ**

**Đề tài: Nghiên cứu và xây dựng mô hình truyền phát hình ảnh , video qua  
ESP8266**

**Giảng viên hướng dẫn: Phạm Văn Sư**

**Thực hiện: Nhóm DIPFL23PRJG33  
Mã đề : STC12**

<b>Triệu Tuấn Nghĩa</b>	<b>B20DCDT153</b>
<b>Nguyễn Tiến Duy</b>	<b>B20DCDT037</b>

*Hà Nội – 2023*

## Tự đánh giá

Họ và tên	Mã sinh viên	Công việc	Đánh giá
Triệu Tuấn Nghĩa	B20DCDT153	Làm Word, tìm hiểu mô hình chuẩn bị phần cứng	7/10
Nguyễn Tiến Duy	B20DCDT037	Lên ý tưởng , Tester , Lập trình code	8/10

## Giảng viên đánh giá

Sinh viên	Đánh giá
Triệu Tuấn Nghĩa B20DCDT153	
Nguyễn Tiến Duy-B20DCDT037	

# LỜI MỞ ĐẦU

Truyền phát hình ảnh và video qua mạng không dây là một trong những ứng dụng quan trọng và phổ biến của công nghệ thông tin và truyền thông. Tuy nhiên , việc truyền tin phát hình ảnh và video qua mạng không dây gặp nhiều yếu tố như băng thông hạn chế, độ trễ rất cao, mất gói tin, nhiễu sóng ....vv. Do đó , cần có những giải pháp tối ưu để đảm bảo chất lượng hình ảnh và video khi truyền phát qua mạng không dây.

ESP8266 là một module WiFi tích hợp vi điều khiển , có khả năng kết nối với mạng WiFi và thực hiện các chức năng như 1 máy tính nhỏ. ESP8266 có ưu điểm giá rẻ , kích thước nhỏ gọn, dễ sử dụng và lập trình. ESP8266 có thể được sử dụng để truyền phát hình ảnh và video qua mạng WiFi, tuy nhiên cũng cần phải giải quyết các vấn đề băng thông, độ trễ , mất gói tin, nhiễu sóng ....vv.

Đề tài nghiên cứu và xây dựng mô hình truyền phát hình ảnh , video qua ESP8266 nhằm mục đích:

- Nghiên cứu các nguyên lý cơ bản và các thuật toán liên quan đến truyền phát hình ảnh và video qua mạng không dây.
- Nghiên cứu các tính năng và khả năng của module ESP8266, cũng như các phương pháp lập trình và giao tiếp với module này.
- Xây dựng một mô hình truyền phát hình ảnh và video qua ESP8266 bao gồm các thiết bị phần cứng và phần mềm cần thiết.

Nội dung đề tài được trình bày theo các chương sau:

- Chương 1: Giới thiệu đề tài
- Chương 2: Cơ sở lý thuyết về phương pháp xử lý ảnh số
- Chương 3: Thiết kế và xây dựng mô hình truyền phát tín hình ảnh và video qua ESP8266
- Chương 4: Nguyên lý hoạt động của mô hình
- Chương 5 : Tổng kết

# MỤC LỤC

<b>LỜI MỞ ĐẦU</b>	<b>3</b>
<b>CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI</b>	<b>5</b>
<b>CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VỀ PHƯƠNG PHÁP XỬ LÝ ẢNH SỐ</b>	<b>6</b>
2.1 Ảnh số là gì?	6
2.2 Phương pháp xử lý ảnh từ webcam	6
2.2.1 Base64 là gì?	6
2.2.2 Cách hoạt động của Base64	6
2.2.3 Bảng mã Base64	7
2.2.4 Thuật toán mã hóa ảnh Base64	7
<b>CHƯƠNG 3 : THIẾT KẾ VÀ XÂY DỰNG MÔ HÌNH TRUYỀN PHÁT TÍN HIỆU HÌNH ẢNH VÀ VIDEO QUA ESP8266</b>	<b>8</b>
3.1 ESP 8266	8
3.2 LCD Nokia 5110	11
3.3 Giao thức truyền dẫn thông tin TCP Server	12
<b>CHƯƠNG 4 : NGUYÊN LÝ HOẠT ĐỘNG CỦA MÔ HÌNH</b>	<b>16</b>
4.1 Sơ đồ hệ thống	16
4.2 Khối Client Cam/Video	17
4.3 Khối SINK NODE	19
4.4 Khối NODE CLIENT	20
<b>CHƯƠNG 5: TỔNG KẾT</b>	<b>21</b>
5.1 Chạy thử	21
5.2 Tổng kết và đánh giá	22

## CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

Đề tài nghiên cứu và xây dựng mô hình truyền phát hình ảnh, video qua ESP8266 là một đề tài thú vị và có ý nghĩa thực tiễn. Đề tài nhằm mục đích nghiên cứu các nguyên lý và thuật toán liên quan đến truyền phát hình ảnh và video qua mạng không dây, sử dụng module ESP8266 làm thiết bị phần cứng chính. Đề tài cũng xây dựng một mô hình truyền phát hình ảnh và video qua ESP8266, bao gồm các thiết bị phần cứng và phần mềm cần thiết, và đánh giá hiệu năng và chất lượng của mô hình này. Đề tài gồm 4 chương, trình bày mục tiêu, nội dung, phương pháp, mô phỏng thực nghiệm và kết luận của đề tài.



Một hệ thống giám sát sử dụng webcam đã kết nối với Access Point (AP) để thu thập hình ảnh thông tin. Đầu tiên, hình ảnh dữ liệu được lấy từ webcam và chuyển đổi từ định dạng màu sang hình ảnh đen trắng 1-bit, trong đó mỗi ảnh chỉ có giá trị 0 hoặc 1, có thể hiển thị đen trắng của điểm ảnh đó.

Sau đó, kích thước của hình ảnh sẽ được thu nhỏ khoảng 48x84 pixel để phù hợp với màn hình điện thoại di động Nokia. Quá trình này có thể sử dụng các hình ảnh xử lý thuật toán như giảm, làm mờ hoặc lấy mẫu để giảm kích thước mà vẫn giữ được thông tin quan trọng.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VỀ PHƯƠNG PHÁP XỬ LÝ ẢNH SỐ

### 2.1 Ảnh số là gì?

Ảnh số là hình ảnh được ghi nhận bởi bộ cảm biến điện tử và lưu lại dưới dạng dữ liệu trong bộ nhớ của máy. Nó là khái niệm dùng để chỉ biểu diễn số học của hình ảnh trong máy tính, thường là biểu diễn nhị phân. Ảnh số bao gồm ảnh raster và ảnh vector.

Cấu trúc của ảnh số là tập hợp các điểm ảnh với mức xám phù hợp để mô tả ảnh gần với ảnh thực. Điểm ảnh (Pixel) chính là một phần tử của ảnh số tại tọa độ (x,y) với độ xám hoặc màu nhất định. Kích thước và khoảng cách giữa các pixel được lựa chọn thích hợp. Mật độ pixel được xác định trên một ảnh chính là độ phân giải của ảnh. Độ phân giải càng thấp thì hình ảnh hiển thị càng mờ, ngược lại độ phân giải càng cao thì ảnh càng sắc nét. Mức xám của ảnh số là kết quả của sự mã hóa tương ứng cường độ sáng của mỗi điểm ảnh với một giá trị số.

### 2.2 Phương pháp xử lý ảnh từ webcam

Ở đề tài sử dụng phương pháp Base64 để xử lý ảnh thu được qua webcam.

#### 2.2.1 Base64 là gì?

Base64 là một phương pháp mã hóa nhị phân dữ liệu thành một chuỗi ASCII sử dụng tập tin 64 ký tự (A-Z, a-z, 0-9, '+', '/'), giúp dữ liệu trở lại nên dễ đọc và truyền tải các văn bản giao thức như email, HTTP.

#### 2.2.2 Cách hoạt động của Base64

- Dữ liệu nhị phân được sắp xếp theo từng khối 24 bit (3 byte) liên tục.
- Mỗi đoạn 24 bit được nhóm thành bốn phần 6 bit mỗi phần.
- Mỗi nhóm 6 bit được chuyển đổi thành các giá trị ký tự Base64 tương ứng của chúng.
- Mã hóa Base64 chuyển đổi ba octet thành bốn ký tự được mã hóa. (mỗi octet có 8bits dữ liệu)
- Người nhận sẽ phải đảo ngược quá trình này để khôi phục thông điệp ban đầu.

Với ảnh, pixel giá trị của ảnh được chuyển đổi từ 8 bit sang 6 bit, tạo ra một ký tự ASCII chuỗi. Khi cần truyền hoặc nhận hình ảnh giữa các chương trình khác nhau, Base64 có thể được sử dụng. Bởi vì ảnh Base64 chỉ là văn bản nên nó có thể được nhúng trực tiếp vào HTML nguồn mã hóa.

### 2.2.3 Bảng mã Base64

Index	Binary	Char	Index	Binary	Char	Index	Binary	Char	Index	Binary	Char
0	000000	A	16	010000	Q	32	100000	g	48	110000	w
1	000001	B	17	010001	R	33	100001	h	49	110001	x
2	000010	C	18	010010	S	34	100010	i	50	110010	y
3	000011	D	19	010011	T	35	100011	j	51	110011	z
4	000100	E	20	010100	U	36	100100	k	52	110100	ø
5	000101	F	21	010101	V	37	100101	l	53	110101	1
6	000110	G	22	010110	W	38	100110	m	54	110110	2
7	000111	H	23	010111	X	39	100111	n	55	110111	3
8	001000	I	24	011000	Y	40	101000	o	56	111000	4
9	001001	J	25	011001	Z	41	101001	p	57	111001	5
10	001010	K	26	011010	a	42	101010	q	58	111010	6
11	001011	L	27	011011	b	43	101011	r	59	111011	7
12	001100	M	28	011100	c	44	101100	s	60	111100	8
13	001101	N	29	011101	d	45	101101	t	61	111101	9
14	001110	O	30	011110	e	46	101110	u	62	111110	+
15	001111	P	31	011111	f	47	101111	v	63	111111	/
Padding		=									

### 2.2.4 Thuật toán mã hóa ảnh Base64

Các giá trị từ 0 đến 64 tương ứng với các ký tự alphabet thông thường:

A-Z là từ 0 đến 25

a-z là từ 26 đến 51

0-9 là từ 52 đến 61

Dấu + là giá trị 62

Dấu / là giá trị 63

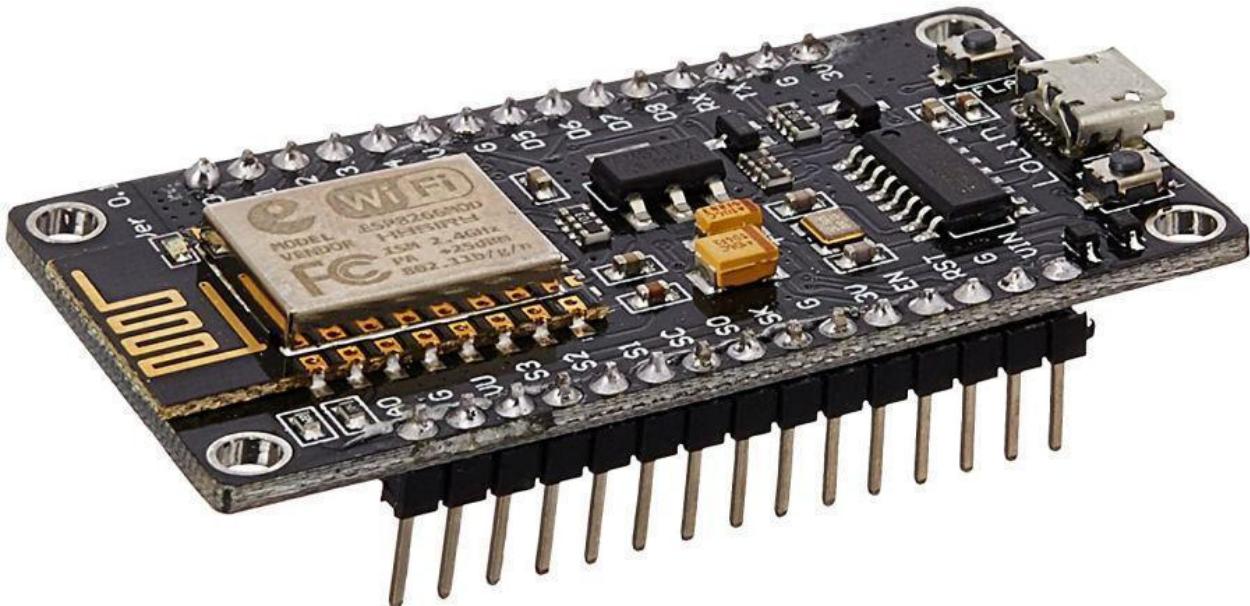
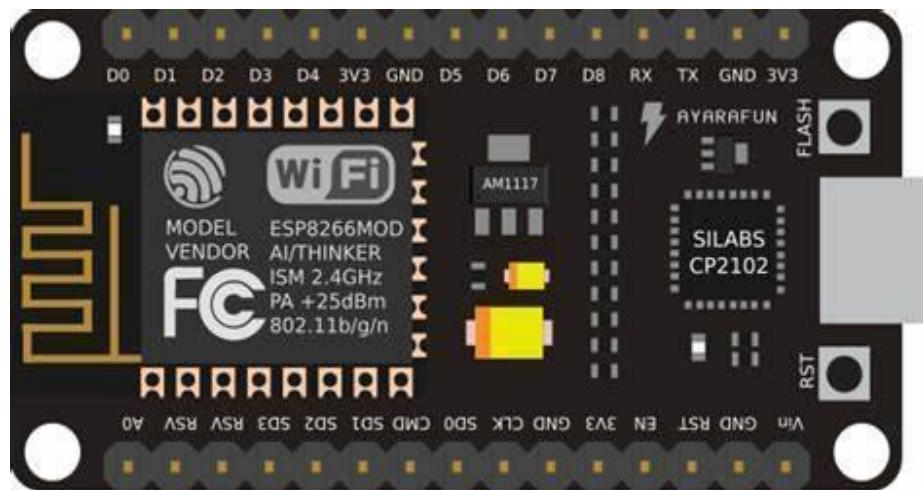
Như vậy thuật toán chuyển sang base64 sẽ đọc nội dung file binary, cứ gom cụm 6bit thành 1 nhóm rồi chuyển thành giá trị tương ứng. Kết quả là chuỗi text với các ký tự ở trên.

Và thuật toán giải mã đơn giản là làm ngược lại, cứ lấy các ký tự ghi lại thành nhị phân sẽ được ảnh ban đầu.

## CHƯƠNG 3 : THIẾT KẾ VÀ XÂY DỰNG MÔ HÌNH TRUYỀN PHÁT TÍN HIỆU HÌNH ẢNH VÀ VIDEO QUA ESP8266

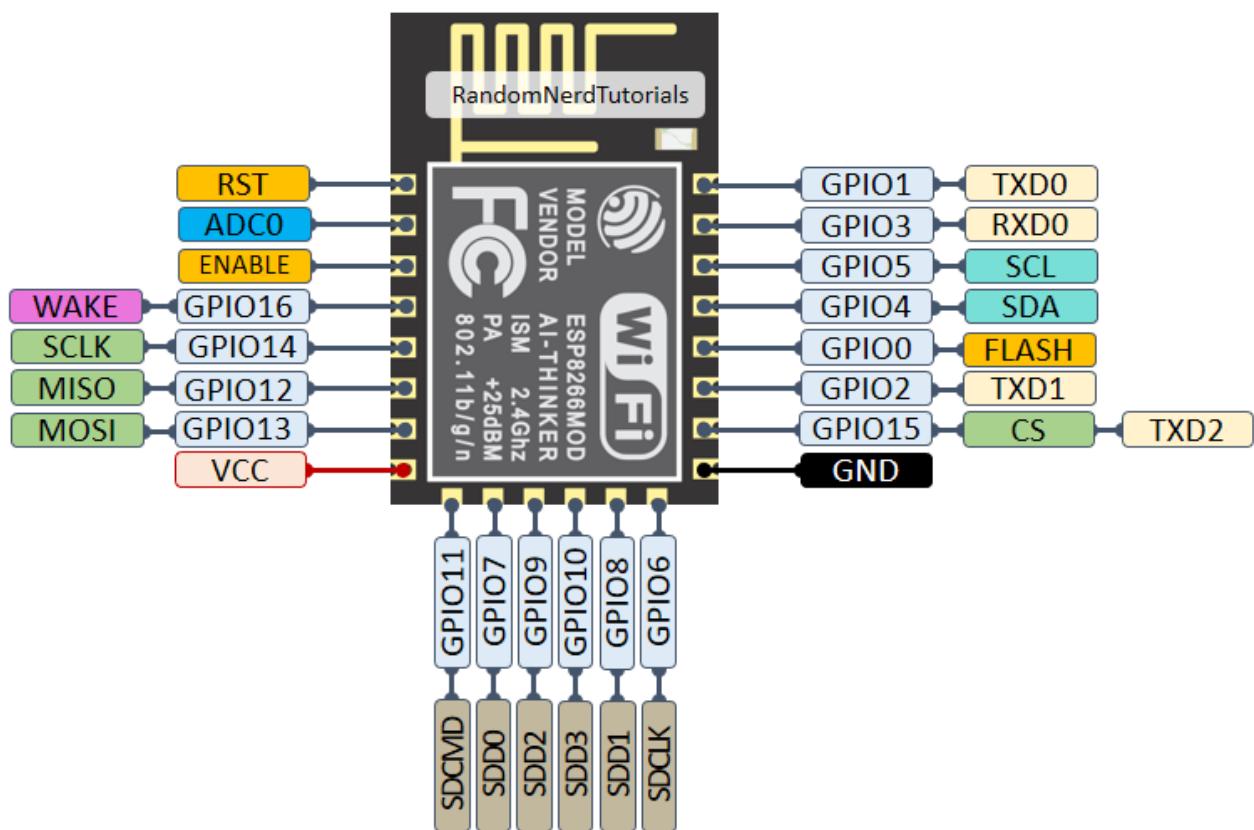
### 3.1 ESP 8266

**ESP8266** là một module Wi-Fi với khả năng kết nối Internet và được tích hợp sẵn trên một số board nhúng như NodeMCU, Wemos, và ESP-01. ESP8266 có thể hoạt động như một điểm truy cập (access point), một client kết nối đến một điểm truy cập khác, hoặc cả hai đều được. Nó được sử dụng rộng rãi trong các ứng dụng IoT (Internet of Things) như cảm biến thông minh, hệ thống kiểm soát thiết bị, hoặc các ứng dụng điều khiển từ xa. Module này có giá thành rẻ và rất dễ sử dụng, cùng với đó là khả năng tương thích với nhiều loại vi điều khiển khác nhau.

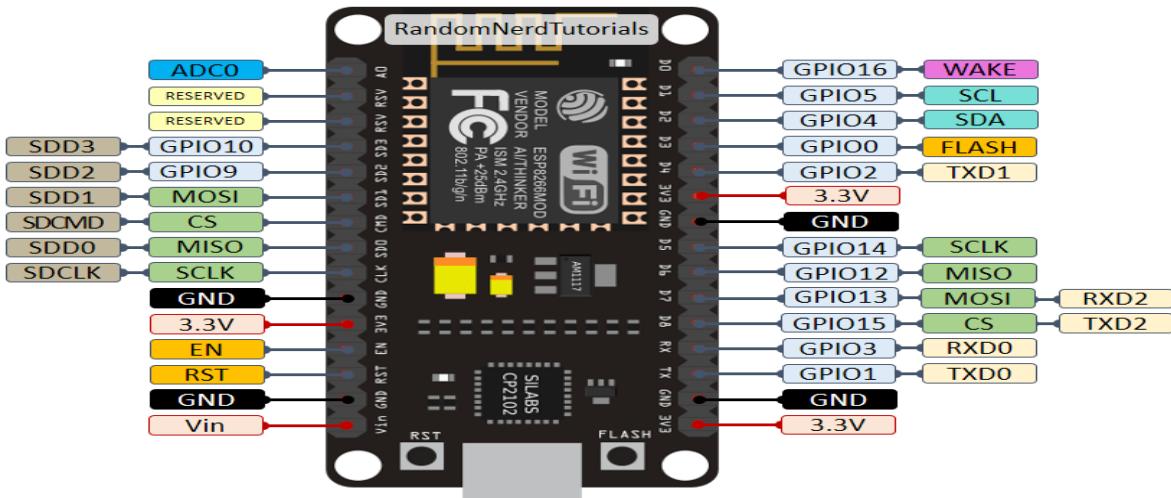


## Thông số kỹ thuật :

- IC chính : ESP 8266
- Điện áp hoạt động : 3.0 □ 3,6 VDC
- Dòng điện hoạt động : trung bình 80mA
- Giao tiếp: UART/ ADC / GPIO / PWM
- SPI Flash : 32Mbit
- Chân IO: 9
- Cổng nối tiếp : 300 □ 4608000 bps
- Wifi protocles : 802.11 b/g/n
- Dải tần số : 2.4GHz □ 2.5GHz ( 2400M □ 2483.5M)
- Nhiệt độ làm việc : -20 □ 85 độ C
- Nhiệt độ bảo quản : -40 □ 90 độ C
- Độ ẩm bảo quản : < 90RH
- Kích thước 24 x 16 x 3 mm
- Trọng lượng : 4g

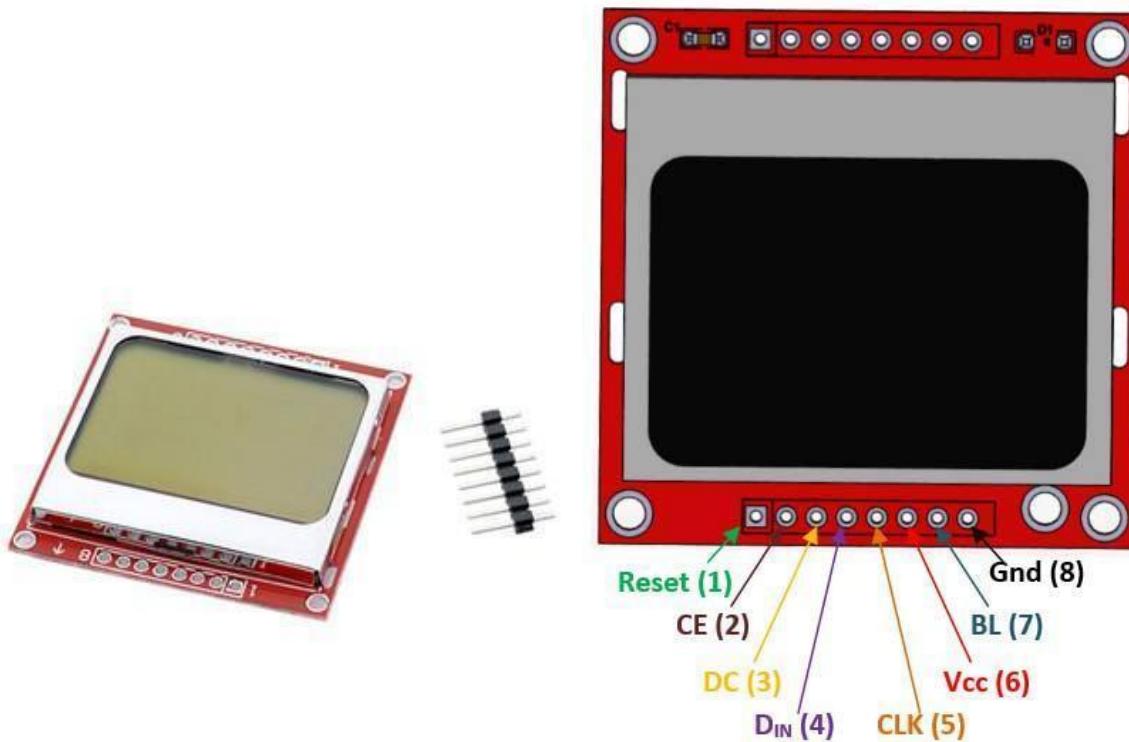


## Sơ đồ chân ESP8266



Nhãn	GPIO	Đầu vào	Đầu ra	Ghi chú
D0	GPIO16	không gián đoạn	không hỗ trợ PWM hoặc I2C	MÚC CAO khi khởi động Sử dụng để đánh thức khi ngủ sâu
D1	GPIO5	OK	OK	thường được sử dụng như SCL (I2C)
D2	GPIO4	OK	OK	thường được sử dụng như SDA (I2C)
D3	GPIO0	kéo lên	OK	kết nối với nút FLASH, khởi động không thành công nếu kéo MÚC THẤP
D4	GPIO2	Kéo lên	OK	MÚC CAO khi khởi động kết nối với đèn LED trên bo mạch, khởi động không thành công nếu kéo MÚC THẤP
D5	GPIO14	OK	OK	SPI (SCLK)
D6	GPIO12	OK	OK	SPI (MISO)
D7	GPIO13	OK	OK	SPI (MOSI)
D8	GPIO15	kéo đến GND	OK	SPI (CS)  Khởi động không thành công nếu kéo MÚC CAO
RX	GPIO3	OK	Chân RX	MÚC CAO khi khởi động
TX	GPIO1	Chân TX	OK	MÚC CAO khi khởi động đầu ra gõ lỗi khi khởi động, khởi động không thành công nếu kéo MÚC THẤP
A0	ADC0	Đầu vào analog	X	

### 3.2 LCD Nokia 5110



Màn hình LCD 5110 có tất cả  $84 \times 48$  điểm ảnh. Được điều khiển bởi Chip driver PD84544 thông qua giao thức SPI chế độ 0. Có 8 chân giao tiếp được mô tả như sau:

1. RST : Chân reset LCD
2. CE: Chân cho phép hoặc không cho phép LCD hoạt động ( 1 số màn ghi là CSE hoặc SE)
3. DC: Chân chọn dữ liệu gửi đến LCD là lệnh hay là dữ liệu để hiển thị ra màn hình
4. CLK: Chân truyền xung nhịp theo chuẩn SPI
5. DIN: Chân dữ liệu data
6. VCC: Chân cấp nguồn cho LCD
7. BL(LED): Chân cấp nguồn cho led nền màn hình LCD
8. GND: Chân mass, cấp nguồn 0V

#### Phương thức giao tiếp

- DC=0 : Chế độ gửi lệnh, tức là dữ liệu bạn gửi đến có nhiệm vụ điều khiển, cài đặt hoạt động của LCD chứ nó không in ra màn hình
- DC=1 : Chế độ gửi dữ liệu: Lúc này, tất cả mọi dữ liệu bạn gửi đến sẽ được hiển thị ra màn hình

LCD sẽ bị reset khi có chân reset ở mức thấp

### 3.3 Giao thức truyền dẫn thông tin TCP Server

TCP/IP là gì? Quá trình hình thành nên giao thức TCP/IP

*Giao thức TCP/IP* (Transmission Control Protocol/ Internet Protocol) đây là một bộ giao thức dùng để trao đổi các thông tin liên quan đến internet, ngoài ra nó còn biết đến là một mô hình các khái niệm và là một tập hợp tất cả các giao thức truyền thông được sử dụng trong internet và trong tất cả các hệ thống máy tính tương tự. giao thức này cũng là nền tảng của internet hiện nay.



#### Cách thức hoạt động của TCP / IP

Giao thức TCP/IP là một sự nghiên cứu phát triển nhờ sự kết hợp từ 2 giao thức lại với nhau, do đó mô hình của giao thức này cũng được hoạt động dựa trên 2 nền tảng chính, đối với giao thức IP (Giao thức liên mạng) thì chúng cho phép các gói tin sẽ được thực hiện và gửi đến các đích đã định sẵn bằng cách đó là thêm các thông tin dẫn đường vào trên gói tin đó gói tin.

Khi giao thức TCP/IP hoạt động dựa trên phương thức sử dụng mô hình giao tiếp của Host/Client (Máy khách/Máy chủ), thì trong đó người dùng hoặc là thiết bị (máy khách) sẽ được một máy tính khác (máy chủ) cung cấp các dịch vụ giống như việc gửi một trang web nào đó trong hệ thống mạng internet.

Đối với bộ giao thức TCP/IP thì chúng là một giao thức không có trạng thái, trong mỗi một yêu cầu sau nào đó của máy khách thì chúng đều tách biệt với các yêu cầu trước đó. Việc mà sử dụng các giao thức không có trạng thái này sẽ giúp giảm được việc trễ mạng, chính vì vậy chúng ta có thể yên tâm sử dụng liên tục mà không phải lo lắng đến việc bị mất kết nối hay lag.

Một điểm cần lưu ý trong giao thức này đó là trong các tầng vận chuyển của giao thức này vẫn có các trạng thái nhất định, nó sẽ thực hiện truyền đi một lệnh duy nhất và sẽ không có bất kỳ thay đổi nào cho đến khi mà tất cả các lệnh tập đều đã được trung tại một điểm cuối nào đó.



### Các loại giao thức TCP / IP phổ biến hiện nay

#### a. HTTP (HyperText Transfer Protocol):

Giao thức sử dụng để truyền dữ liệu không cần bảo mật giữa máy chủ (web server) và máy khách (web client).

Máy khách gửi yêu cầu tới máy chủ để nhận thông tin của trang web.

Dữ liệu truyền qua giao thức này không được mã hóa, không an toàn nếu có thông tin nhạy cảm.

#### b. HTTPS (HyperText Transfer Protocol Secure):

Giao thức được sử dụng để truyền dữ liệu có tính bảo mật giữa máy chủ và máy khách.

Sử dụng mô hình TCP/IP kết hợp với SSL và TLS để đảm bảo an toàn và bảo mật dữ liệu.

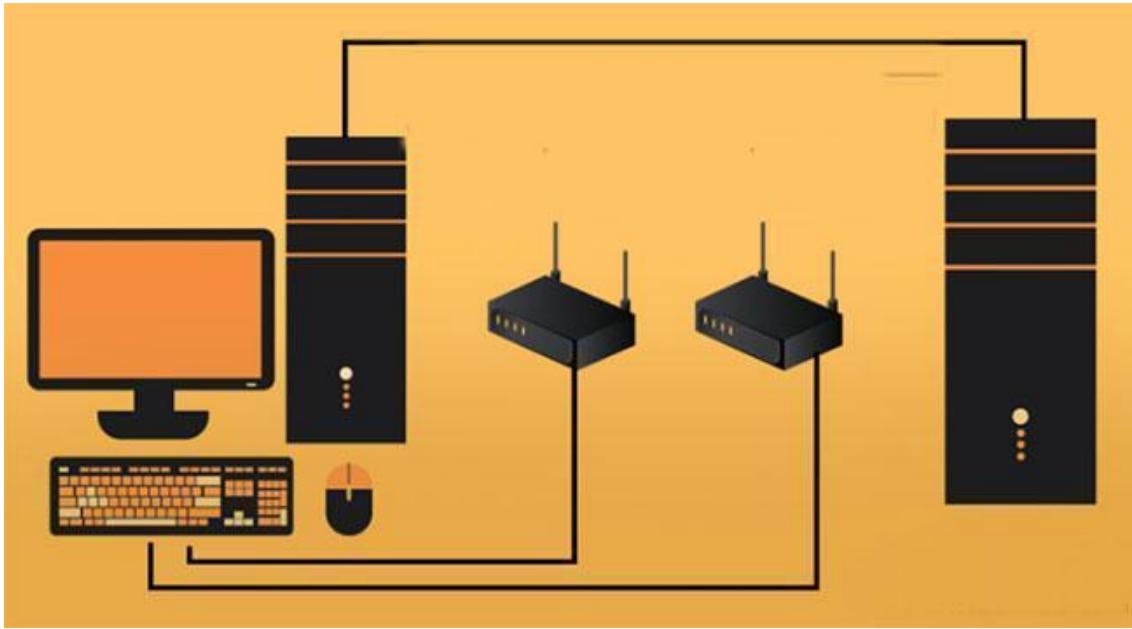
Thường được sử dụng cho các giao dịch tài chính và truyền thông tin cá nhân.

#### c. FTP (File Transfer Protocol):

Giao thức sử dụng để truyền dữ liệu giữa hai hoặc nhiều máy tính kết nối thông qua mạng Internet.

Cho phép máy khách truy cập vào máy chủ để gửi và nhận các dữ liệu, không quan trọng vị trí địa lý.

Dữ liệu truyền qua FTP có thể bao gồm các tệp tin và thư mục.



## A. Ưu điểm

- Giao thức này không phải chịu bất kỳ một sự kiểm soát nào đến từ các tổ chức nào. Chính vì điều này tạo ra cho người dùng sự tự do và thoải mái trong khi sử dụng.
- Giao thức TCP/IP này là loại giao thức có khả năng tương thích cao trên thị trường đối với tất cả các hệ điều hành, trên cả các phần cứng của thiết bị nên việc giao thức này hiển nhiên được hoạt động hiệu quả nhất trên nhiều những hệ thống khác nhau.
- Cuối cùng giao thức này có được một khả năng mở rộng cao, có thể thực hiện việc định tuyến nên khi thông qua mạng chúng có thể tự mình xác định được những đường dẫn hiệu quả nhất.

## B. Nhược điểm

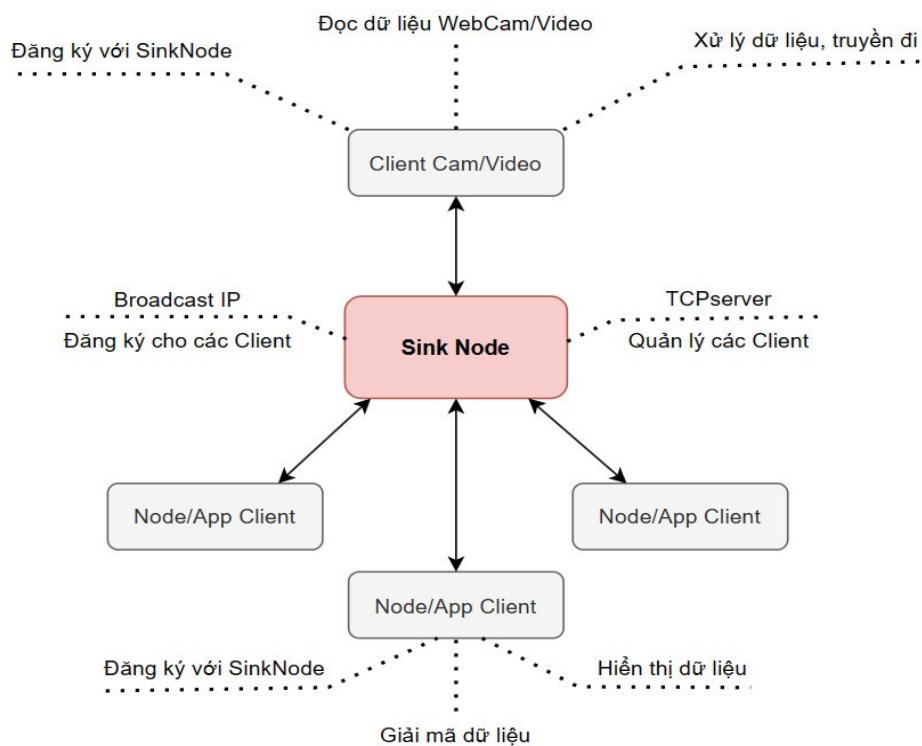
- Giao thức này là một giao thức rất khó để quản lý chúng, cũng như việc cài đặt để sử dụng được giao thức này khá phức tạp.
- Điều khó khăn tiếp theo của giao thức này là chúng không dễ để có thể thay thế được.
- Ngoài ra trên tầng transport của giao thức này cũng không đảm bảo cho việc phân phối được các gói tin.
- TCP/IP cũng không được hiệu quả khi mô tả các công nghệ trong hệ thống mạng mới mẻ do không thể tách biệt rõ ràng được giữa những khái niệm về dịch vụ, các giao diện và với giao thức.
- Nhược điểm cuối cùng của giao thức này là dễ bị tấn công SYN, đây là một kiểu tấn công từ chối các dịch vụ.

## CHƯƠNG 4 : NGUYÊN LÝ HOẠT ĐỘNG CỦA MÔ HÌNH

### Hệ thống chia làm 3 khối

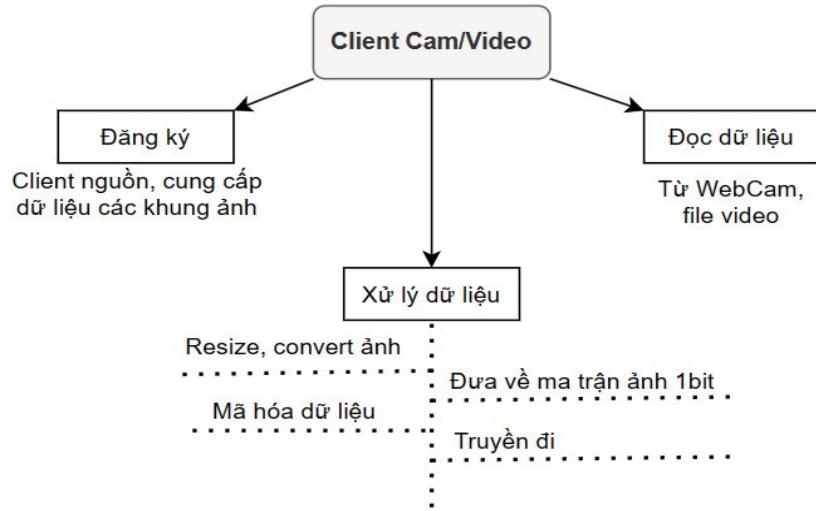
- 1 Khối Sinknode Broadcast IP hoạt động như 1 server đăng ký client để quản lý
- 2 Khối Client Cam/ Video đăng ký với sinknode để thực hiện nhiệm vụ
- 3 Khối Node /App client đăng ký với sinknode nhận dữ liệu từ sink node và hiển thị ra màn hình.

### 4.1 Sơ đồ hệ thống



Hệ thống bao gồm một webcam kết nối với máy tính, chuyển đổi dữ liệu hình ảnh thành ảnh đen trắng và thu nhỏ kích thước về 48x84 pixel. Quá trình này bao gồm việc chuyển đổi màu sắc sang độ xám và áp dụng ngưỡng để tạo ra ảnh đen trắng. Ảnh sau đó được mã hóa bằng Base64 encoding để truyền tải qua mạng. Dữ liệu ảnh mã hóa được chuyển đến Access Point (AP), thiết bị kết nối máy tính và màn hình Nokia qua mạng WiFi. Màn hình Nokia, sau khi nhận dữ liệu từ AP, giải mã ảnh từ định dạng Base64 về định dạng ảnh 1 bit và hiển thị trên màn hình. Qua chuỗi các bước này, hệ thống hoàn thành quá trình truyền tải hình ảnh từ webcam đến màn hình Nokia thông qua các quy trình xử lý và mã hóa.

## 4.2 Khối Client Cam/Video



### - Sử dụng Python và OpenCV

#### - Đọc dữ liệu :

Webcam chuyển đổi tín hiệu hình ảnh từ camera thành tín hiệu số thông qua bộ chuyển đổi analog-to-digital (ADC). Dữ liệu số sau đó được xử lý và chuyển đổi thành định dạng bit để truyền hoặc lưu trữ, đại diện cho màu sắc và độ sáng của hình ảnh.

OpenCV cung cấp các module mạnh mẽ để đọc dữ liệu webcam/video.

Xây dựng trong file (src/appclient/client\_stream\_src.py)

- **Đăng ký :** Đăng ký với sinknode, sinknode quyết định client này sẽ là nơi cung cấp dữ liệu
  - Gửi gói tin đăng ký với tiêu đề là Tab src
- **Xử lý dữ liệu bằng các hàm OpenCV cung cấp**

#### + **Resize (Thay đổi Kích Thước Ảnh):**

cv2.resize() trong OpenCV thực hiện thay đổi kích thước của hình ảnh và áp dụng hai kỹ thuật chính:

→ Kết quả thu được : ảnh thu nhỏ từ kích thước lớn về 48x84

#### a.Tự nhiên gần nhất (Nearest-neighbor interpolation):

- + Giữ nguyên giá trị pixel từ vị trí gần nhất.
- + Tốc độ nhanh, nhưng có thể tạo ra hình ảnh không mịn khi thay đổi kích thước.

## b.Nội suy tuyến tính (Bilinear interpolation):

- + Sử dụng thông tin từ 4 pixel lân cận để tạo hình ảnh mịn hơn.
- + Giữ nguyên một số đặc điểm cơ bản của hình ảnh.

### + Convert Ảnh:

Hàm `cv2.cvtColor()` trong thư viện OpenCV được sử dụng để chuyển đổi không gian màu của hình ảnh. Mã `cv2.COLOR\_BGR2GRAY` hoặc `cv2.COLOR\_RGB2GRAY` thường được áp dụng để chuyển đổi ảnh sang không gian màu độ xám. Ví dụ, khi chuyển đổi từ không gian màu BGR sang độ xám, mỗi pixel trong ảnh sẽ chỉ còn giá trị độ xám thay vì giữa ba kênh màu. Điều này giúp giảm chiều sâu màu và thuận tiện cho các phép toán xử lý ảnh đơn giản.

→ **Ảnh chuyển từ RGB về Gray chứa các mức xám từ 0 đến 255**

### + Đưa Ma Trận Ảnh 1 Bit:

Chức Năng: Chuyển đổi ảnh sang dạng ma trận 1 bit. Mỗi bit trong ma trận đại diện cho một pixel, thường được sử dụng để biểu diễn trạng thái pixel đen hoặc trắng.

- Sử dụng hàm CV\_THRESH\_BINARY đặt ngưỡng 127
- (lớn hơn 127 = 1 ; nhỏ hơn 127 = 0 )  
→ **Kích thước 48x84 bit = 4032:8 = 504( byte )**

### + Mã Hóa Dữ Liệu:

Chức Năng: Mã hóa dữ liệu ảnh thành định dạng phù hợp để truyền đi. Sử dụng các thuật toán mã hóa như Base64 để chuyển đổi dữ liệu thành dạng văn bản để thuận tiện trong quá trình truyền tải.

→ **Mỗi 3 byte dữ liệu mã hóa 4 ký tự Base 64**

Kích thước sau khi mã hóa sẽ bằng  $504 : 3 \times 4 = 672$  ký tự

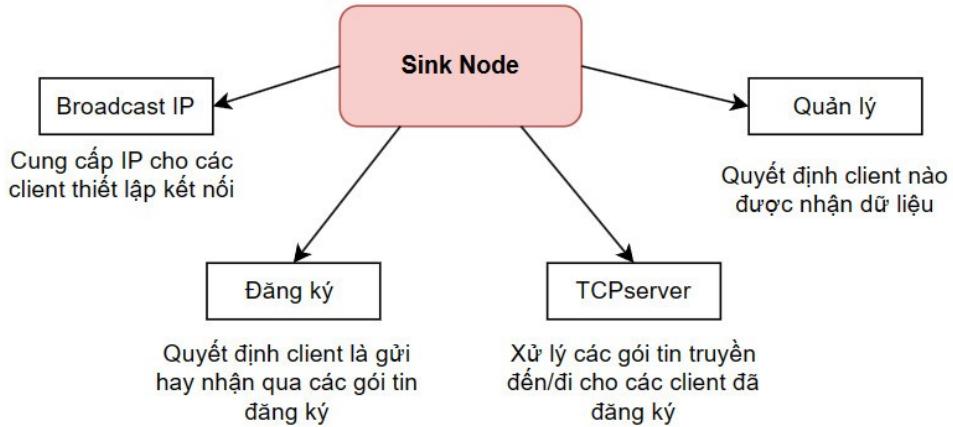
→ **Mỗi chuỗi mã hóa có 672 ký tự**

### + Truyền ĐI (Truyền Dữ Liệu):

Chức Năng: Sử dụng giao thức truyền tải TCPIP theo mô hình TCP-server-client thực hiện trao đổi các gói tin dạng văn bản. Truyền dữ liệu từ client này đến đích sink-node, sink-node nhận dữ liệu này và tiến hành truyền tiếp đến các client hiển thị.

⇒ **Xây dựng trong file mã nguồn (client\_stream\_src.py )**

### 4.3 Khối SINK NODE



**Sử dụng Esp8266 làm sink-node chạy trên môi trường lập trình của ESP IDF chạy hệ điều hành FreeRTOS**

#### Broadcast IP:

- ESP 8266 có một địa chỉ IP nhằm cung cấp địa chỉ IP cho các Client thiết lập kết nối.
- **Xây dựng trong file file mã nguồn src/sink/main.c**

#### Đăng ký :

- Xử lý các gói tin đăng ký từ các client gửi đến, xây dựng tác vụ xử lý riêng cho từng client theo nhiệm vụ mà nó đăng ký
- Đối với client là nguồn dữ liệu: SinkNode nhận các gói tin dữ liệu ảnh gửi về và duy trì kết nối với client, đảm bảo chỉ có 1 client là nguồn
- Đối với client là nơi hiển thị: SinkNode gửi các gói tin dữ liệu ảnh đi, duy trì kết nối
- **(Được xây dựng trong file mã nguồn tie\_softap.c và tie\_tcp\_server.c)**

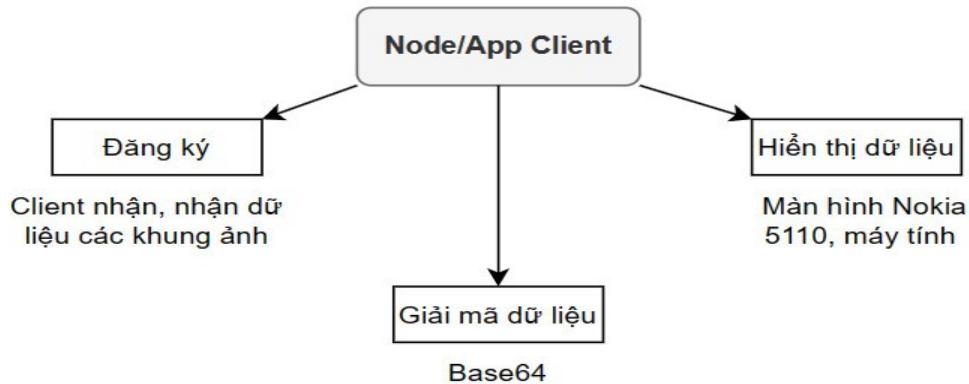
#### TCPserver

- Sinknode có thể hoạt động như một máy chủ TCP để lắng nghe và xử lý các yêu cầu từ các thiết bị khác trong mạng. Điều này giúp sinknode có thể nhận và xử lý dữ liệu từ webcam và các thiết bị khác thông qua giao thức truyền tải TCP.
- Xây dựng trong file mã nguồn main.c  
( hàm tcp\_src\_callback ,....)

#### Quản lý

- Quyết định dữ liệu ảnh sẽ được hiển thị tại client nào trong list các client đã đăng ký là nơi hiển thị
- Thao tác với Esp8266 để thực hiện quản lý thông qua nút nhấn
- Xây dựng trong file mã nguồn main.c ,  
Hàm BTN\_GPIO12\_Press\_CBack

#### 4.4 Khối NODE CLIENT



##### Đăng ký:

- Client đăng ký với SinkNode, SinkNode sẽ quyết định cho client này vào list các client sẽ hiển thị dữ liệu
- Đối với client là nodemcu hay là app thì việc đăng ký cũng như truyền nhận các gói tin là như nhau
- Xây dựng trong src/node client / main.c, tie\_tcp\_client.c

##### Giải mã:

- Để giải mã ảnh từ chuỗi Base64, bắt đầu bằng cách nhận chuỗi và chuyển đổi nó thành dữ liệu nhị phân. Dữ liệu này sau đó được tái tạo thành hình ảnh gốc, có thể hiển thị hoặc xử lý theo nhu cầu. Quá trình này giúp khôi phục ảnh từ dữ liệu Base64 đã được mã hóa trước đó.
- Xây dựng trong src/node client / inc/tie\_base64.h

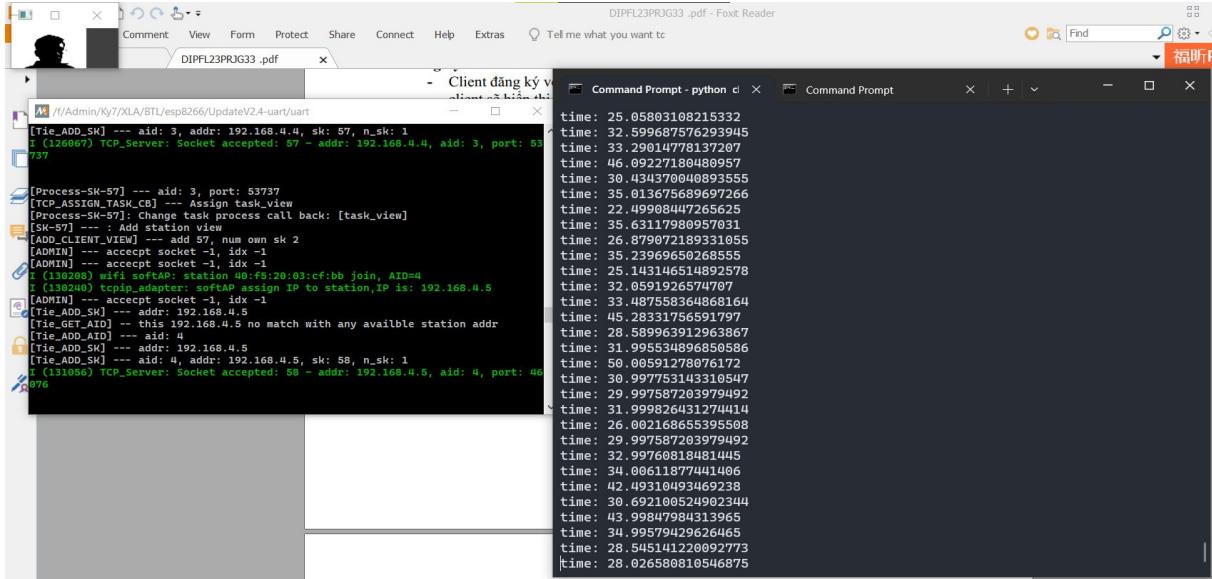
##### Hiển thị dữ liệu :

- Đối với client là nodemcu: Dữ liệu sau khi giải mã sẽ được hiển thị lên màn hình LCD 5110
- Đối với client là app: Dữ liệu sau khi giải mã sẽ được hiển thị lên màn hình máy tính, mỗi pixel là một ô vuông trong ma trận ô vuông hiển thị trong file srcappclient string view .py
- Xây dựng trong file mã nguồn src/node client/tie\_5110.c , main.c

## CHƯƠNG 5: TỔNG KẾT

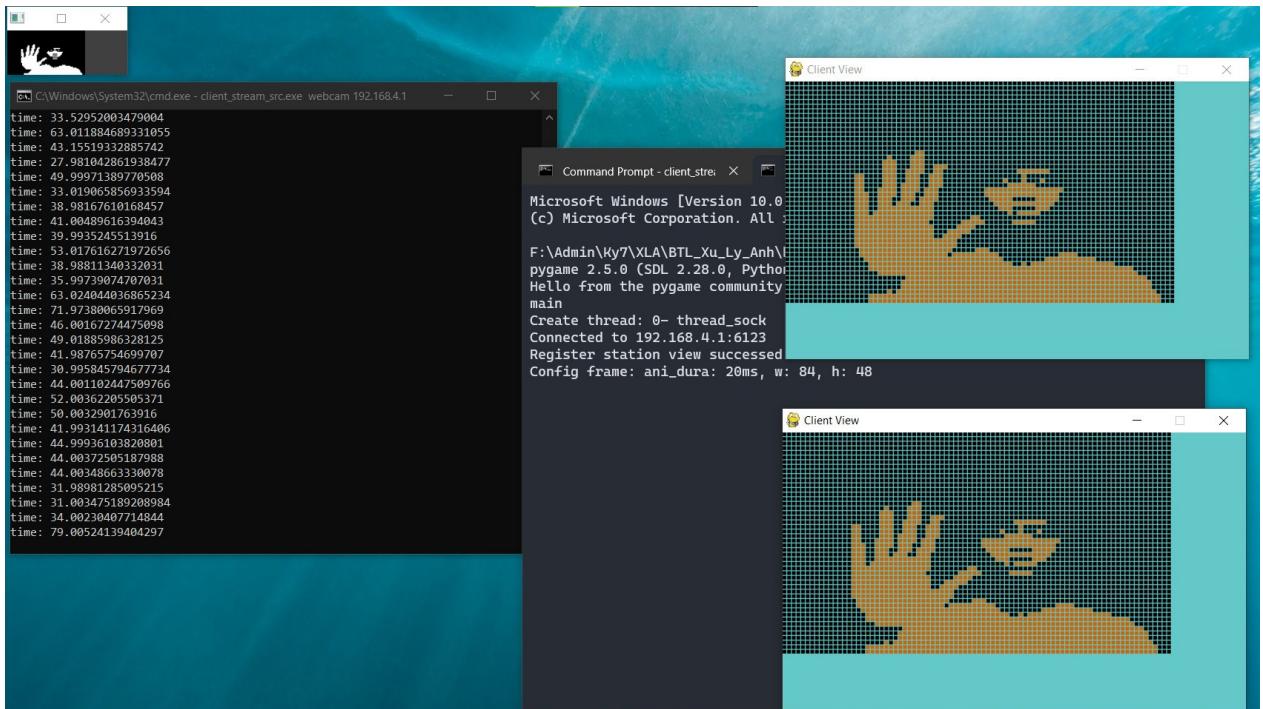
### 5.1 Chạy thử

Các bước cài đặt built thực hiện theo file readme.txt  
Mô phỏng phần mềm trên máy tính



```
DIPFL23PRJG33.pdf - Foxit Reader
Client đăng ký với station
I [126067] TCP_Server: Socket accepted: 57 - addr: 192.168.4.4, aid: 3, port: 53932
[SK-57] --- aid: 3, add: 192.168.4.4, sk: 57, n_sk: 1
I (126067) TCP_Server: Socket accepted: 57 - addr: 192.168.4.4, aid: 3, port: 53932
[ProtoCS-SK-57] --- aid: 3, port: 53932
[TCP_ASSIGN_TASK_CB] --- Assign task_view
[Process-SK-57]: Change task process call back: [task_view]
[SK-57] --- : Add station view
[ADD_CLIENT_VIEW] --- add: 57, num own sk 2
[ADMIN] --- accept socket -1, idx -1
[ADMIN] --- accept socket -1, idx -1
[ADMIN] --- accept socket -1, idx -1
I (130040) Client View: Register station successed: 20:03:cfc:bb join, AID=4
I (130040) laptop address softAP assign IP to station, IP is: 192.168.4.5
[ADMIN] --- accept socket -1, idx -1
[ADMIN] --- accept socket -1, idx -1
[Tie_GET_AID] --- this 192.168.4.5 no match with any available station addr
[Tie_ADD_AID] --- aid: 4
[Tie_ADD_SK] --- add: 192.168.4.5
[Tie_ADD_SK] --- aid: 4, add: 192.168.4.5, sk: 58, n_sk: 1
I (131056) TCP_Server: Socket accepted: 58 - addr: 192.168.4.5, aid: 4, port: 46076
time: 25.05803108215332
time: 32.599687576293945
time: 33.29014778137207
time: 46.09227180480957
time: 50.434370040893555
time: 55.013675689697266
time: 22.49908447265625
time: 35.63117980957031
time: 26.879072189331055
time: 35.23969650268555
time: 25.143146514892578
time: 32.0591926574707
time: 33.487558364868164
time: 45.28331756591797
time: 28.589963912963867
time: 31.99553489685086
time: 50.00591278876172
time: 30.997753143310547
time: 29.997587203979492
time: 31.999826431274414
time: 26.002168655395508
time: 29.997587203979492
time: 32.99760818481445
time: 34.00611877411406
time: 42.49310093469238
time: 30.692106524902344
time: 43.99847984313965
time: 34.99579429626465
time: 28.545141220092773
time: 28.026580810546875
```

Test SinkNode và Client\_Stream\_src



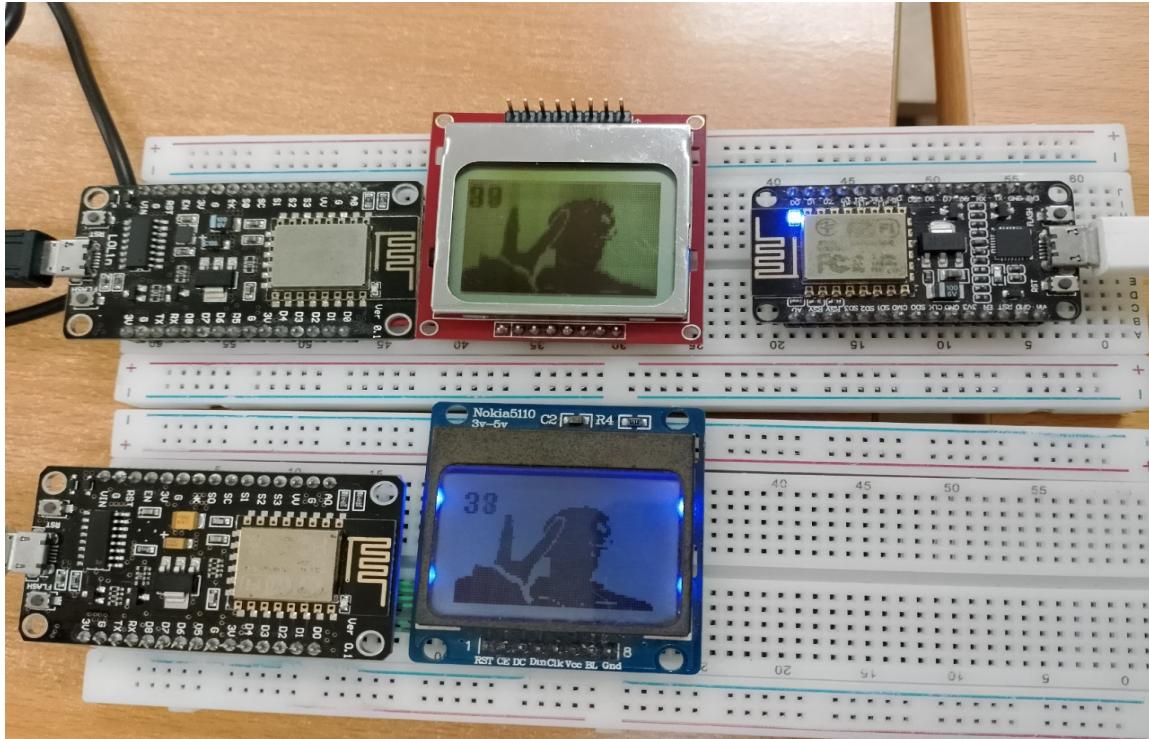
```
C:\Windows\System32\cmd.exe - client_stream_src.exe webcam 192.168.4.1
time: 33.52952003479004
time: 63.0118846589331055
time: 43.15519332085742
time: 27.981042861938477
time: 49.99971389770568
time: 33.019065856933594
time: 38.98167610168457
time: 41.00489616394043
time: 39.0935245512016
time: 53.017616271972656
time: 38.98811340332031
time: 35.99739074707031
time: 63.024044436865234
time: 71.97380065917969
time: 46.00167274475098
time: 49.01885986328125
time: 41.98765754699707
time: 30.995845794677734
time: 44.001102447509766
time: 52.003622095505371
time: 50.0032901763916
time: 41.993141174316406
time: 44.99936103820801
time: 44.00372595187988
time: 44.00348663330078
time: 31.98981285095215
time: 31.003475189208984
time: 34.00230407714844
time: 79.00524139404297
```

Microsoft Windows [Version 10.0]
(c) Microsoft Corporation. All rights reserved.

F:\Admin\Ky7\XLA\BTL\_Xu\_Ly\_Anh\client\_stream\_src.py
pygame 2.5.0 (SDL 2.28.0, Python 3.9.1)
Hello from the pygame community.
main
Create thread: 0- thread\_sock
Connected to 192.168.4.1:6123
Register station view succeeded
Config frame: ani\_dura: 20ms, w: 84, h: 48

Test AppClient

## Mô phỏng phần cứng thực tế



### 5.2 Tổng kết và đánh giá

Bài báo cáo này tập trung vào việc phát triển một hệ thống hiển thị hình ảnh từ webcam thông qua mô-đun ESP8266 và truyền tải chúng lên màn hình LCD Nokia 5110 sử dụng giao thức TCP/IP. Đầu tiên, nhóm tôi đã giới thiệu về ESP8266, một module Wi-Fi mạnh mẽ và linh hoạt, thích hợp cho ứng dụng IoT. Nhóm của tôi đã thực hiện kết nối ESP8266 với webcam, cho phép nó nhận dữ liệu hình ảnh từ camera.

Sau đó, nhóm tôi đã mô tả cách thiết lập giao thức TCP/IP để truyền dữ liệu từ ESP8266 đến màn hình LCD Nokia 5110. Việc này yêu cầu sự kết hợp chặt chẽ giữa các thư viện và nguồn mở mã có sẵn, chắc chắn bảo đảm liên kết ổn định giữa các thành phần của hệ thống. Nhóm tôi cũng đã giải quyết cách chuyển đổi hình ảnh dữ liệu thành định dạng phù hợp cho màn hình LCD.

Qua quá trình thực hiện, nhóm tôi đã đạt được kết quả thành công, khi màn hình LCD Nokia 5110 hiển thị hình ảnh từ webcam một cách ổn định thông tin qua ESP8266. Điều này mở ra nhiều ứng dụng tiềm năng trong quá trình theo dõi và hiển thị hình ảnh dữ liệu từ xa trong hệ thống giám sát và IoT. Tuy nhiên, có những quy trình cần được vượt qua, như hiệu suất tối ưu hóa và đảm bảo tính bảo mật của hệ thống.

Cuối cùng, bài báo cáo này là một bước quan trọng đầu tiên trong việc kết hợp các công nghệ nhúng, mạng và hiển thị để tạo ra một giải pháp giải pháp cho việc truyền tải và hiển thị hình ảnh từ webcam lên màn hình LCD, mở ra nhiều cơ hội ứng dụng trong thực tế.

