

ĐỒ ÁN: ĐỒNG HỒ SỨC KHỎE

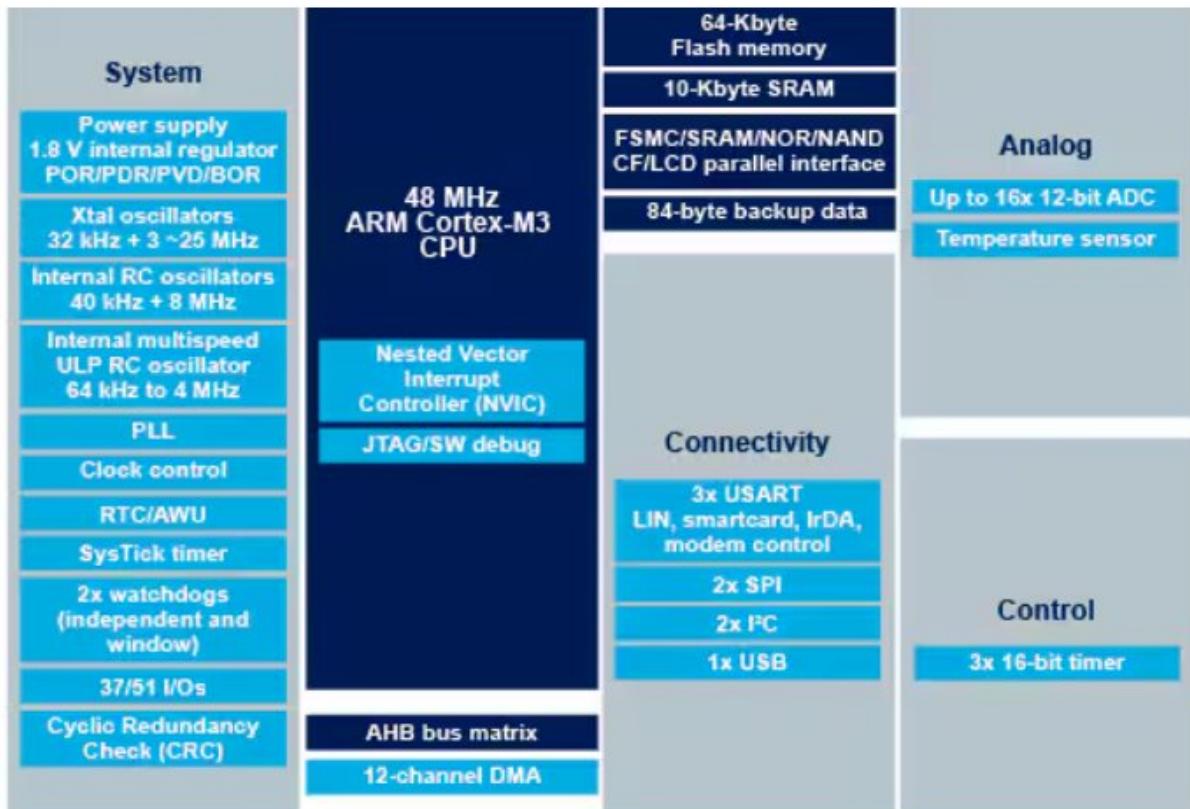
1. MỞ ĐẦU:

Theo thống kê của tổ chức y tế thế giới (WHO), trung bình cứ 2 giây lại có một người chết do bệnh tim mạch, cứ 3 người tử vong thì có một người mắc bệnh tim mạch.

Bệnh tim mạch hiện cũng là nguyên nhân hàng đầu của tàn phế sau đột quỵ. Bệnh tim mạch còn được dự đoán là nguyên nhân lớn nhất gây tàn phế trên thế giới vào năm 2020. Bởi vậy việc sử dụng một thiết bị theo dõi nhịp tim là rất cần thiết đối với tất cả mọi người, nhịp tim quá cao hoặc quá thấp có thể giúp mọi người sớm chuẩn đoán và phát hiện các bệnh về tim mạch, huyết áp. Thiết bị này có thể làm điều đó thông qua việc sử dụng cảm biến max30102 và hiển thị thông tin lên TFT ili9341. Thông số về nhịp tim mỗi phút và nồng độ oxy trong máu sẽ được hiển thị liên tục với độ chính xác cao. Thiết bị này còn có chức năng hiển thị thời gian, nhiệt độ, độ ẩm, tia UV cho người dùng và người nhà muốn quan sát sức khỏe của người đeo đồng hồ.

2. Lý thuyết :

- STM32 là một họ vi điều khiển 32-bit dựa trên bộ xử lý Arm Cortex-M 32-bit được phát triển bởi STMicroelectronics. STM32 cung cấp hiệu năng cao, khả năng xử lý thời gian thực, khả năng xử lý tín hiệu số, điện năng tiêu thụ thấp và khả năng kết nối (Wifi, Bluetooth, LoRa, Ethernet...) trong khi vẫn được sự hội nhập và dễ dàng phát triển. STM32F103 thuộc họ F1 với lõi là ARM COTEX M3, là vi điều khiển 32 bit, tốc độ tối đa là 72Mhz.
- STM32F103C8T6 trở nên hữu ích trong đề tài này hơn một số vi điều khiển khác bởi vì số lượng chân GPIO là đủ dùng cho cả TFT và MAX30102, các cảm biến khác. Tần số tối đa 72MHz đáp ứng được tốc độ tốt cho hệ thống hoạt động ổn định và nhanh. Khi phát triển thêm các chức năng thì vi điều khiển vẫn còn nhiều chân chức năng cũng như một bộ nhớ flash lớn(64KB) để có thể sử dụng cho các mục đích sau này.



Trong đề tài này, STM32F103C8T6 được sử dụng với một số chức năng như SPI,I2C,UART,TIMER,DMA,ADC.

2.1.1: RTC

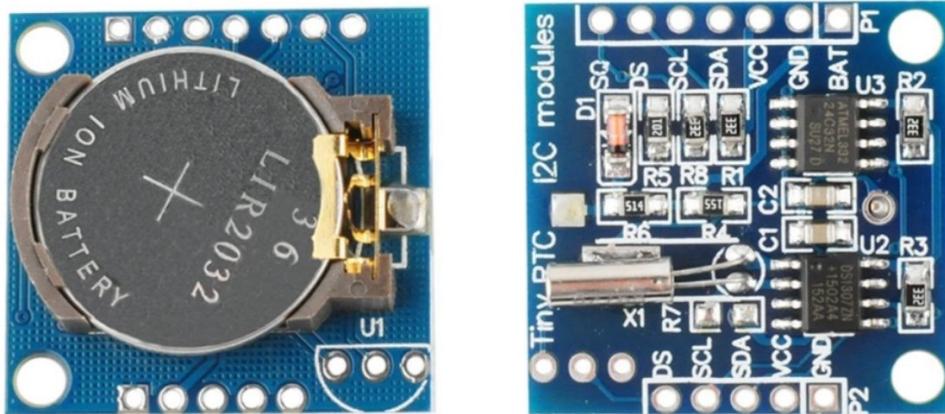
RTC hay Real Time Clock là ngoại vi sử dụng như một bộ đồng hồ – lịch thời gian thực. Thuật ngữ thời gian thực là chỉ thời gian trong cuộc sống của con người hàng ngày, chứ không phải thời gian trong tính toán như các Timer khác. Để bộ RTC có thể hoạt động chúng ta có thể sử dụng nguồn từ máy tính hoặc là Pin Cmos CR2032 để dùng cho RTC khi không có điện.

Ưu điểm của DS1307:

- Tiêu thụ thấp: Mô-đun này có hiệu suất cao và có thể hoạt động trong nhiều năm chỉ bằng một pin CR2032.
- Giao diện đơn giản: Nó sử dụng giao thức I²C, giảm số lượng chân cần thiết cho giao tiếp.
- Cải thiện độ chính xác: Mặc dù không chính xác bằng sự phát triển của DS3231, DS1307 vẫn đáng tin cậy hơn **quầy nội bộ** của một vi điều khiển tiêu chuẩn.
- Hoàn thiện và đa năng: Bao gồm các tính năng như tự động điều chỉnh cho những tháng ngắn và thêm ngày vào những năm nhuận.

Nhược điểm của DS1307:

- Không có bù nhiệt
- Tốc độ I2C thấp
- Pin backup CR2032 hao nhanh nếu mạch kém



Chân	Tên	Chức năng
1	X1	Kết nối đến thạch anh 32.768Khz làm nguồn dao động cho chip
2	X2	
3	VBat	Kết nối đến cực dương của Pin dự phòng, có điện áp tiêu chuẩn khoảng 3V
4	GND	Kết nối đến mass
5	SDA	Chân dữ liệu khi kết nối đến bus I2C
6	SCL	Chân nhận xung clock đồng bộ khi kết nối bus I2C
7	SQW/OUT	Ngõ xuất xung vuông, tần số có thể lập trình để thay đổi từ 1Hz, 4Khz, 8 Khz, 32 Khz
8	VCC	Nguồn cấp chính, khoảng 5VDC

Các thông tin về thời gian và lịch thu được bằng cách đọc các byte trong thanh ghi tương ứng. Thời gian và lịch được thiết lập hoặc khởi tạo bằng cách viết các byte thanh ghi thích hợp. Lưu ý dữ liệu lưu thời gian chứa trong thanh ghi đều theo định dạng BCD. Thứ trong tuần thay đổi tại lúc nửa đêm, lưu dưới dạng con số (Ví dụ 1 là Chủ Nhật, 2 là Thứ hai, 3 là thứ 3 ...). Khi lần đầu tiên IC được cấp nguồn các thanh ghi thời gian và lịch reset về 01/01/00 01 00:00:00.

Bảng chi tiết các thanh ghi:

00h	GIÂY
01h	PHÚT
02h	GIỜ
03h	THỨ
04h	NGÀY
05h	THÁNG
06h	NĂM
07h	THANH GHI ĐIỀU KHIỂN
08h–3Fh	VÙNG RAM LUU TRỮ 56 x 8

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE			
00h	CH	10 Seconds			Seconds				Seconds	00–59			
01h	0	10 Minutes			Minutes				Minutes	00–59			
02h	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23			
		24	PM/ AM										
03h	0	0	0	0	0	DAY			Day	01–07			
04h	0	0	10 Date		Date				Date	01–31			
05h	0	0	0	10 Month	Month				Month	01–12			
06h	10 Year				Year				Year	00–99			
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—			
08h–3Fh									RAM 56 x 8	00h–FFh			

THANH GHI CONTROL:

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

Bit 7: Output Control (OUT) thanh ghi điều khiển ngõ ra. Nó ảnh hưởng đến chân số 7 (chân SQW/OUT). Khi sóng vuông xuất ra chân này bị disable, nếu bit OUT=1 khi đó chân số 7 ở mức cao, ngược lại bit OUT=0 khi đó chân số 7 ở mức thấp

Bit 6: Luôn luôn đọc bằng 0

Bit 5: Luôn luôn đọc bằng 0

Bit 4: Square-Wave Enable (SQWE): khi bit này được thiết lập bằng 1, cho phép xuất ra xung vuông tại chân số 7., Tần số của xung vuông phụ thuộc vào bit RS0 và RS1. Khi tần số sóng vuông được thiết lập là 1 Hz. Các thanh ghi thời gian được cập nhật tại cạnh xuống của xung vuông. Khi lần đầu khởi tạo cấp nguồn đến thiết bị, bit này bằng 0.

Bits 1 and 0: Lựa chọn tốc độ(RS[1:0]). Những bit này điều khiển tần số của tần số sóng vuông, khi sóng vuông được cho phép

Thanh ghi giây (địa chỉ 00h): 4 bit thấp chứa hàng đơn vị, 3 bit cao chứa hàng chục của giây. Ngoài ra bit thứ 7 có tên là CH, nếu bit này được thiết lập bằng 1 thì đồng hồ không hoạt động. Vì vậy phải thiết lập bit này bằng không ngay từ đầu.

Thanh ghi phút (địa chỉ 01h): 4 bit thấp chứa hàng đơn vị, 3 bit cao chứa hàng chục của phút. Ngoài ra bit thứ 7 luôn bằng 0.

Thanh ghi giờ (địa chỉ 02h): 4 bit thấp của thanh ghi này chứa hàng đơn vị của giờ, bit thứ 6 quy định chế độ 12 h (bit6 =1) hoặc 24 h (bit 6=0). Nếu ở chế độ 24h thì bit 4 và bit 5 quy định hàng chục của giờ. Nếu chế độ 12h thì bit 4 quy định hàng chục, bit 5 khi đó quy định (PM: buổi chiều hoặc AM: buổi sáng)

Thanh ghi thứ (địa chỉ 03h): ba bit đầu quy định thứ trong tuần (Ví dụ 1 là Chủ Nhật, 2 là Thứ hai, 3 là thứ 3 ...). Các bit còn lại luôn bằng 0.

Thanh ghi ngày(địa chỉ 04h): 4 bit đầu lưu hàng đơn vị của ngày, bit 4 và bit 5 quy định hàng chục. Bit 6 và bit 7 luôn luôn bằng 0.

Thanh ghi tháng(địa chỉ 05h): 4 bit đầu lưu hàng đơn vị của tháng, bit thứ 4 quy định hàng chục. Các bit còn lại luôn bằng 0.

Thanh ghi năm (địa chỉ 06h): 4 bit thấp lưu hàng đơn vị và 4 bit cao hàng chục của năm (từ 00 - 99).

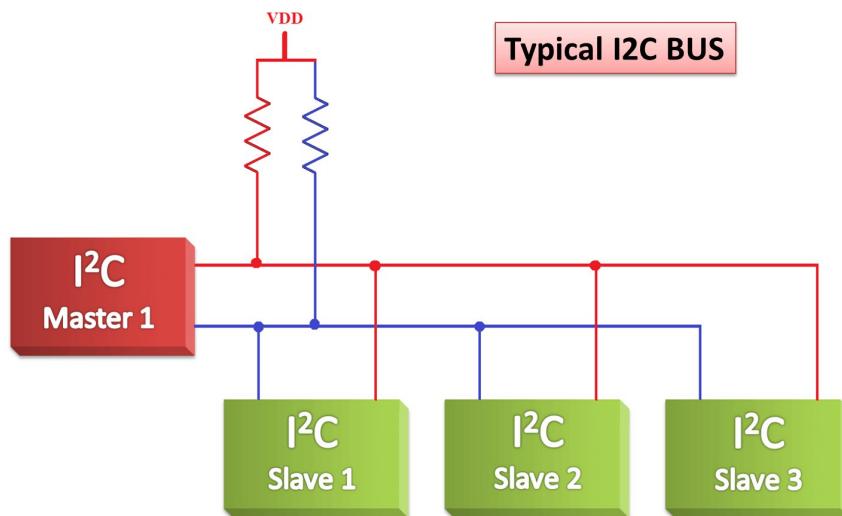
2.1.2: I2C

I2C (Inter-Integrated Circuit): là một giao thức giao tiếp được phát triển bởi Philips Semiconductors (1982) để truyền dữ liệu giữa một bộ xử lý trung tâm với nhiều thiết bị (~128 thiết bị) chỉ sử dụng hai đường truyền tín hiệu. I2C là một loại giao thức giao tiếp nối tiếp đồng bộ. Các bit dữ liệu được truyền từng bit một theo các khoảng thời gian đều đặn được thiết lập bởi một xung Clock tham chiếu.

Đặc điểm:

- Chỉ cần có hai đường bus chung để điều khiển bất kỳ thiết bị nào trên mạng I2C.
- Không cần thỏa thuận trước về tốc độ truyền dữ liệu như trong giao tiếp UART. Vì vậy, tốc độ truyền dữ liệu có thể được điều chỉnh bất cứ khi nào cần thiết.
- Cơ chế đơn giản để xác thực dữ liệu được truyền
- Sử dụng hệ thống địa chỉ 7-bit để xác định một thiết bị cụ thể trên bus I2C.
- Các mạng I2C dễ dàng mở rộng. Các thiết bị mới có thể được kết nối đơn giản với hai đường dây chung I2C.
- Cấu trúc: Bus I2C (dây giao tiếp) chỉ gồm hai dây: Serial Clock Line (SCL), Serial Data Line (SDA).
- Dữ liệu được truyền đi được gửi qua dây SDA và được đồng bộ với tín hiệu đồng hồ (clock) từ SCL.

Trong đề tài, master là STM32F103C8T6 và slave là DS1307, MAX30102. Dữ liệu được truyền giữa thiết bị Master và các thiết bị Slave thông qua đường dữ liệu SDA, thông qua các chuỗi có cấu trúc gồm các số 0 và 1 (bit).



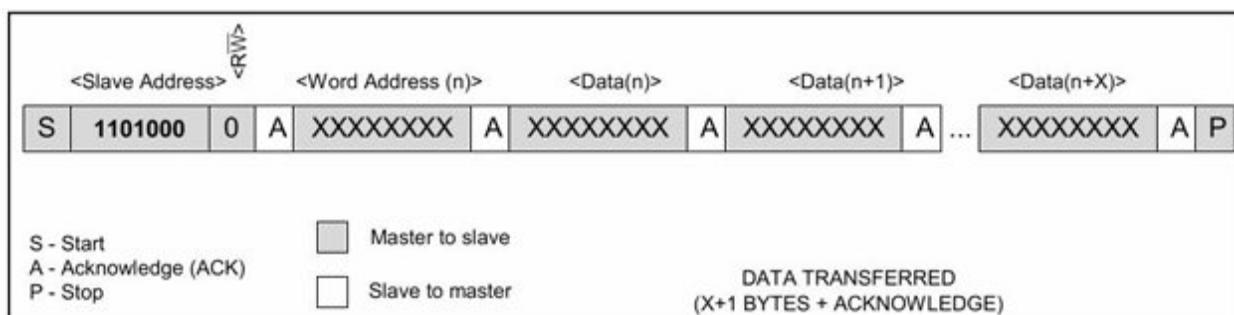
DS1307:

DS1307 có 7 bit địa chỉ cố định là 1101000. Tiếp theo là bit quy định hướng truyền dữ liệu. Nếu bit=0, thì byte dữ liệu truyền từ vi điều khiển -> DS1307. Ngược lại bit=1 thì hướng dữ liệu sẽ là DS1307-> Vi điều khiển.

Ở chế độ ghi dữ liệu:

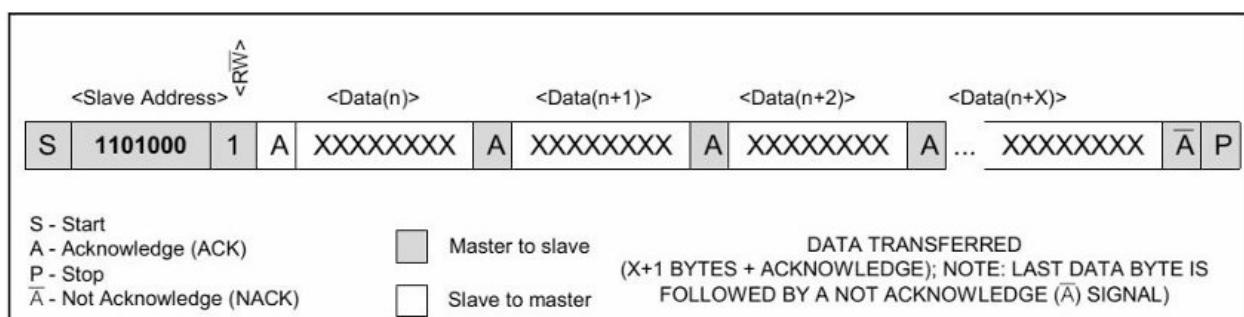
- Vi điều khiển tạo tín hiệu Start lên bus để báo bắt đầu muôn giao tiếp.

- Vi điều khiển gửi địa chỉ của thiết bị muốn giao tiếp. Ở đây DS 1307 là 1101000 + bit hướng = 0. Do vậy byte được gửi là 11010000.
- DS 1307 tạo tín hiệu ACK
- Vi điều khiển gửi địa chỉ của thanh ghi muốn viết đến.
- DS 1307 tạo tín hiệu ACK báo đã nhận thành công.
- Tiếp theo vi điều khiển gửi dữ liệu muốn ghi
- Con trỏ địa chỉ của DS 1307 tự tăng lên đến địa chỉ thanh ghi tiếp theo
- Vi điều khiển gửi dữ liệu muốn ghi đến thanh ghi tiếp theo.
- Vi điều khiển muốn kết thúc qua trình truyền dữ liệu bằng cách gửi tín hiệu Stop.



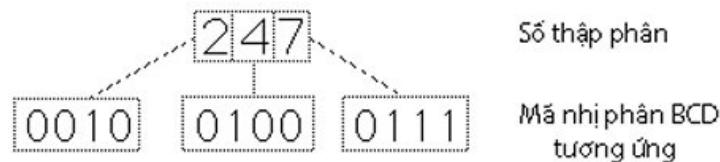
Ở chế độ đọc dữ liệu:

- Vi điều khiển tạo tín hiệu Start lên bus để báo bắt đầu muốn giao tiếp.
- Vi điều khiển gửi địa chỉ của thiết bị muốn giao tiếp. Ở đây DS 1307 là 1101000 + bit hướng = 1. Do vậy byte là 1101001.
- DS 1307 tạo tín hiệu ACK.
- DS 1307 gửi dữ liệu của thanh ghi đầu tiên
- Vi điều khiển gửi tín hiệu ACK báo nhận thành công.
- DS 1307 gửi dữ liệu của thanh ghi tiếp theo.
- Vi điều khiển gửi tín hiệu ACK báo nhận thành công
- Vi điều khiển gửi tín hiệu NACK khi không muốn nhận thêm dữ liệu



Mã nhị phân BCD và DCB:

- Mã nhị phân BCD, trong đó mỗi số nguyên của một số thập phân(0-9) được biểu diễn bởi một số nhị phân 4-bit.



- Do dữ liệu thanh ghi thời gian thực của DS1307 được biểu thị dưới dạng mã nhị phân nên ta cần ghi dữ liệu vào DS1307 là phải chuyển từ thập phân sang nhị phân và đọc dữ liệu của DS1307 thì phải chuyển từ nhị phân sang thập phân.

MAX30102:

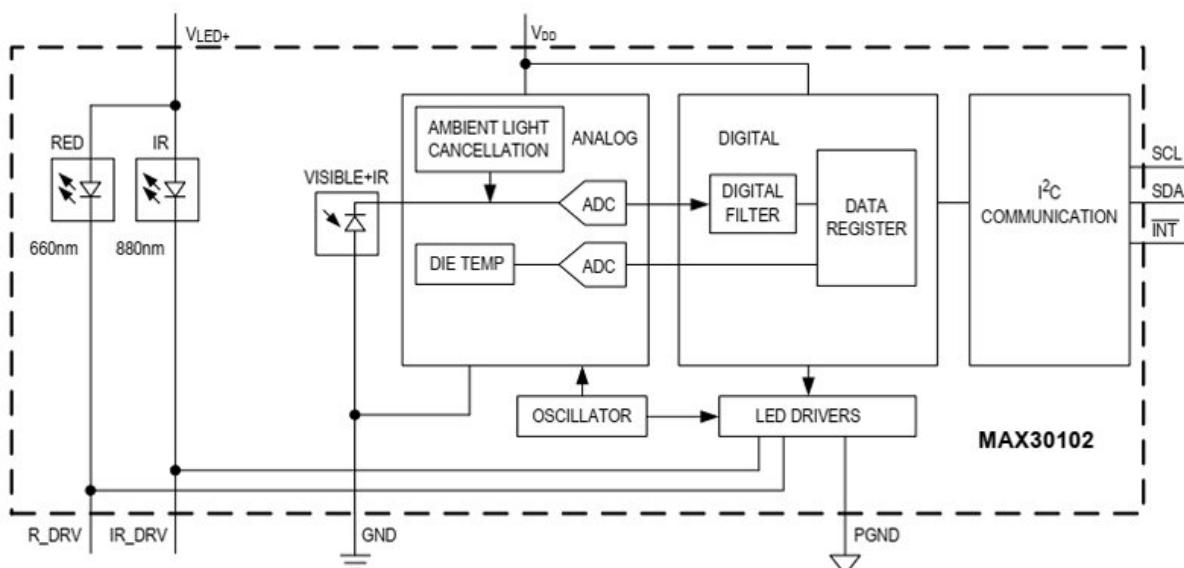
Là một mô-đun đo nhịp tim và oxy trong máu tích hợp. Nó bao gồm đèn LED bên trong, bộ tách sóng quang, các bộ phận quang học và triệt nhiễu với khả năng loại bỏ ánh sáng nhiễu xung quanh. Nhỏ gọn, siêu tiết kiệm năng lượng, thích hợp cho các thiết bị đo di động.



- Giao tiếp: I2C
- Vin: 3.1 ~ 5.25 V
- ADC Resolution: 18 bits
- Giá trị ADC của IR và RED LED: 55536 ~ 75536, ổn định ở 65536
- Nhiệt độ: -40 ~ 85 °c

Nguyên lý hoạt động:

Khi có ngón tay hoặc cổ tay đặt vào vị trí led ir và led red, giá trị analog được truyền về led driver và có một bộ triệt nhiễu bên trong làm nhiệm vụ triệt tiêu các loại ánh sáng nhiễu, giá trị sau khi đã triệt nhiễu được đưa vào một bộ lọc và được lưu vào thanh ghi FIFO có sẵn trong cảm biến. Giá trị có thể được đọc qua giao tiếp I2C và được tính toán trong phần mềm để cho ra giá trị nhịp tim và SpO2 chính xác nhất.



Địa chỉ của các thanh ghi là yếu tố duy nhất cần quan tâm để cấu hình mọi chức năng trong cảm biến MAX30102. Mỗi thanh ghi sẽ có một địa chỉ, trải dài từ 0x00 đến 0xFF với các chức năng cho phép đọc ghi tùy từng thanh ghi. Các giá trị từ B0 đến B7 là 8 bit để điều chỉnh giá trị cho thanh ghi, từ đó cảm biến sẽ lấy mẫu, điều chỉnh cường độ dòng điện hoặc bật/tắt tùy theo giá trị đã được cố định trong datasheet.

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
STATUS											
Interrupt Status 1	A_FULL	PPG_RDY	ALC_OVF	PROX_INT				PWR_RDY	0x00	0X00	R
Interrupt Status 2							DIE_TEMP_RDY		0x01	0x00	R
Interrupt Enable 1	A_FULL_EN	PPG_RDY_EN	ALC_OVF_EN	PROX_INT_EN					0x02	0X00	R/W
Interrupt Enable 2							DIE_TEMP_RDY_EN		0x03	0x00	R/W
FIFO											
FIFO Write Pointer					FIFO_WR_PTR[4:0]				0x04	0x00	R/W
Overflow Counter					OVF_COUNTER[4:0]				0x05	0x00	R/W
FIFO Read Pointer					FIFO_RD_PTR[4:0]				0x06	0x00	R/W
FIFO Data Register					FIFO_DATA[7:0]				0x07	0x00	R/W
CONFIGURATION											
FIFO Configuration		SMP_AVE[2:0]		FIFO_ROLL_OVER_EN		FIFO_A_FULL[3:0]			0x08	0x00	R/W
Mode Configuration	SHDN	RESET				MODE[2:0]			0x09	0x00	R/W
SpO ₂ Configuration	0 (Reserved)	SPO2_ADC_RGE[1:0]			SPO2_SR[2:0]		LED_PW[1:0]		0x0A	0x00	R/W
RESERVED									0x0B	0x00	R/W
LED Pulse Amplitude		LED1_PA[7:0]							0x0C	0x00	R/W
		LED2_PA[7:0]							0x0D	0x00	R/W
RESERVED									0x0E	0x00	R/W
RESERVED									0x0F	0x00	R/W
Proximity Mode LED Pulse Amplitude		PILOT_PA[7:0]							0x10	0x00	R/W
Multi-LED Mode Control Registers		SLOT2[2:0]			SLOT1[2:0]				0x11	0x00	R/W
		SLOT4[2:0]			SLOT3[2:0]				0x12	0x00	R/W

Có một số thanh ghi quan trọng sẽ được trình bày như bên dưới:

- Interrupt Status: 0x00 – 0x01

Chức năng hay dùng nhất ở thanh ghi ngắn này là PPG_RDY nằm ở bit thứ 6 (B6). Khi có một mẫu được ghi vào thanh ghi FIFO giá trị ngắn lập tức thay đổi và sẽ được xóa khi gọi hàm đọc data ở thanh ghi FIFO, điều này đảm bảo chỉ khi có mẫu được đọc, giá trị mới được lấy vào và tính toán để đáp ứng tính chính xác nhất.

- FIFO: thanh ghi chứa các địa chỉ để đọc, ghi và lưu giá trị ở mỗi lần lấy mẫu.

FIFO Configuration (0x08)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
FIFO Configuration		SMP_AVE[2:0]		FIFO_ROL LOVER_EN		FIFO_A_FULL[3:0]			0x08	0x00	R/W

Bit 7-5: Sample Averaging /SMP_AVE/

Ngoài ra, ở địa chỉ thanh ghi 0x08, có một bit B4 có vai trò xóa dữ liệu hoặc giữ lại dữ liệu mà không đọc tiếp khi thanh ghi FIFO đầy.

- Mode Configuration:

Đây là thanh ghi quan trọng nhất, bit thứ B7 được set là 0 thì cảm biến sẽ shutdown nhưng các giá trị trong cảm biến vẫn giữ nguyên, cờ ngắt được xóa. Bit thứ B6 khi được set là 1, toàn bộ cảm biến sẽ được reset về trạng thái ban đầu, bit sẽ tự xóa về 0 sau khi hoàn thành reset. Các bit 2:0 cho phép bật tắt số led cần thiết, 010 sẽ bật chỉ mỗi led red để đo nhịp tim, 011 sẽ bật cả 2 led để đo nhịp tim và spo2.

Ví dụ: khi dùng cả 2 led để đo spo2 và nhịp tim và không shutdown, không reset thì mã hex ghi vào cho thanh ghi này sẽ là 0x03, tương ứng 00 000 011 ở mã nhị phân

Table 4. Mode Control

MODE[2:0]	MODE	ACTIVE LED CHANNELS
000		Do not use
001		Do not use
010	Heart Rate mode	Red only
011	SpO2 mode	Red and IR
100-110		Do not use
111	Multi-LED Mode	Red and IR

- Thanh ghi spo2 configuration: có địa chỉ là 0x0A, điều chỉnh số mẫu và ADC range

Các bit B4 B3 B2 cấu hình tốc độ lấy mẫu trên mỗi lần chuyển đổi của led ir và led red. 2 bit B1 B0 là độ phân giải của ADC cần dùng, độ phân giải càng lớn thì led phát ra cường độ càng cao thích hợp cho di động vì vùng chiếu sáng và nhận tín hiệu của 2 led sẽ rộng hơn.

SpO₂ Configuration (0x0A)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
SpO ₂ Configuration		SPO2_ADC_RGE[1:0]		SPO2_SR[2:0]		LED_PW[1:0]		0x0A	0x00	R/W	

SPO2_ADC_RGE[1:0]	LSB SIZE (pA)	FULL SCALE (nA)
00	7.81	2048
01	15.63	4096
10	31.25	8192
11	62.5	16384

Table 6. SpO₂ Sample Rate Control

SPO2_SR[2:0]	SAMPLES PER SECOND
000	50
001	100
010	200
011	400
100	800
101	1000
110	1600
111	3200

Table 7. LED Pulse Width Control

LED_PW[1:0]	PULSE WIDTH (μs)	ADC RESOLUTION (bits)
00	69 (68.95)	15
01	118 (117.78)	16
10	215 (215.44)	17
11	411 (410.75)	18

Mã hex ghi vào thanh ghi 0x0A sẽ là 0x23 cho ra hiệu năng mạnh để có độ chính xác cao khi đo nhịp tim và spo2.

- Địa chỉ để đọc và ghi dùng cho slave như sau:

B7	B6	B5	B4	B3	B2	B1	B0	WRITE ADDRESS	READ ADDRESS
1	0	1	0	1	1	1	R/W	0xAE	0xAF

Hình 15. Địa chỉ đọc ghi MAX30102

Một số thông số và thanh ghi khác như nhiệt độ, multiled không được dùng trong đề tài này. Đề tài chỉ tập trung đọc giá trị ir và red led sau đó cấu hình spo2 và lọc để ra giá trị chính xác nhất.

2.1.3 UART:

- UART (Universal Asynchronous Receiver-Transmitter – Bộ truyền nhận dữ liệu không đồng bộ) là một giao thức truyền thông phân cứng dùng giao tiếp nối tiếp không đồng bộ và có thể cấu hình được tốc độ
- Giao thức UART là một giao thức đơn giản và phổ biến, bao gồm hai đường truyền dữ liệu độc lập là TX (truyền) và RX (nhận). Dữ liệu được truyền và nhận qua các đường truyền này dưới dạng các khung dữ liệu (data frame) có cấu trúc chuẩn, với một bit bắt đầu (start bit), một số bit dữ liệu (data bits), một bit kiểm tra chẵn lẻ (parity bit) và một hoặc nhiều bit dừng (stop bit).

Đặc điểm :

- Chỉ cần dùng 2 dây truyền dữ liệu
- Không cần đến tín hiệu clock
- Có 2 bit chẵn lẻ nên có thể kiểm tra lỗi dễ dàng
- Cấu trúc gói dữ liệu có thể thay đổi được miễn là cả 2 bên đều được thiết lập để giao tiếp với nhau

HC05:



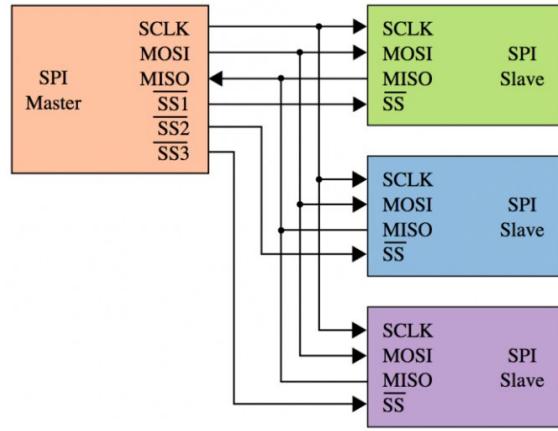
Thông số :

- Điện áp hoạt động: 3.3~5VDC
- Baudrate UART có thể chọn được: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
- Frequency: 2.4 GHz ISM band
Rate: Asynchronous: 2.1 Mbps (max.)/160 kbps
- Synchronous: 1 Mbps/1 Mbps
- Kích thước: 37 x 15.6mm

Là module sử dụng Bluetooth 2.0 sử dụng cho việc tương tác giữa thiết bị của máy tính với các thiết bị khác hoặc việc truyền dữ liệu giữa các thiết bị có sẵn. Nó có thể chạy trong chế độ Master hoặc Slave và hỗ trợ baud rate từ 9600 đến 115200 bps. HC-05 có thể kết nối với các thiết bị sử dụng cả truyền nối vật lý và giao thức số. Chúng ta có thể nhập mật khẩu:1234 và kết nối vào app Bluetooth để điều khiển trên điện thoại.

2.1.4 SPI:

- Giao tiếp SPI (Serial Peripheral Interface) là một giao thức truyền thông nối tiếp được sử dụng phổ biến trong việc kết nối giữa vi điều khiển và các thiết bị ngoại vi như EEPROM, ADC, DAC, các màn hình LCD, hoặc các cảm biến. Giao thức SPI được biết đến với tính đơn giản, tốc độ truyền cao, và khả năng truyền nhạy đồng dữ liệu.
- SPI là một giao thức truyền thông dựa trên kiểu truyền nhị phân (đồng bộ). Trong SPI, có 4 tín hiệu chính:
 - SCLK (Serial Clock): Tín hiệu xung nhịp do Master tạo ra.
 - MOSI (Master Output/Slave Input) – Master gửi dữ liệu đến Slave.
 - MISO (Master In Slave Out): Dòng dữ liệu truyền từ Slave đến Master.
 - SS (Slave Select): Tín hiệu chọn Slave (còn được gọi là CS – Chip Select).



Ưu điểm :

- Truyền nhị phân: SPI truyền dữ liệu nhị phân, cho phép trao đổi song song giữa Master và Slave.
- tốc độ cao : giao tiếp SPI có tốc độ cao hơn so với I2C do không có ACK
- Hỗ trợ nhiều Slave: Bằng cách sử dụng nhiều tín hiệu SS, một Master có thể giao tiếp với nhiều Slave.
- Chiều dài cáp giới hạn: SPI phù hợp cho kết nối ngắn do cơ chế truyền song song đòi hỏi đồng bộ cao.

TFT ILI9341:



Thông số:

- Điện áp hoạt động: 3.3V / 5V
- Độ phân giải: 240 x 320
- Hỗ trợ hiển thị 16BIT RGB với 65K màu
- Tích hợp khe cắm thẻ SD
- Khối lượng: 52g

Nguyên lý hoạt động:

ILI9341 sử dụng bộ nhớ đồ họa GRAM (Graphic RAM) để lưu trữ thông tin màu của từng pixel. Mỗi pixel trên màn hình tương ứng với một ô nhớ trong GRAM. Mỗi pixel được biểu diễn bằng 16bit màu và có tọa độ X và Y.

Dưới đây là các lệnh chính dùng để điều khiển TFT ili9341:

Column Address Mode Set - 2AH

Column Address Data Set – (00H – FFH) (vị trí cột bắt đầu và kết thúc)

Page Address Mode Set - 2BH

Page Address Data Set – (00H – FFH) (vị trí hàng bắt đầu và kết thúc)

Write GRAM Mode Set - 2CH

Software Reset - 01H

Sleep OUT - 11H

Sleep IN - 10H

Display ON - 29H

Display OFF - 28H

Memory Access Control Mode Set - 36H

Pixel Format Mode Set - 3AH

Pixel Format Data Set - 55H

Frame Rate Control Mode Set - B1H

Frame Rate Data Set – (00H – FFH)

Display Function Control Mode Set - B6H

Power Control A – CBH

Power Control B – CFH

Power Control 1 - C0H

🎬 Power Control 2 – C1H

Power On Sequence Control – EDH

Pump Ratio Control - F7H

VCOM Control 1 - C5H

VCOM Control 2 - C7H

Driver Timing Control A - E8H

Driver Timing Control B – EAH

Gamma Curve Select - 26H

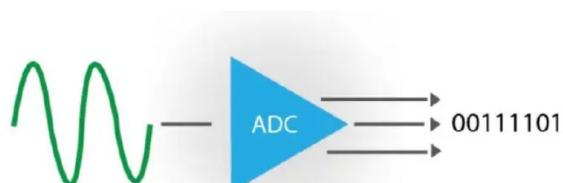
3Gamma Function Disable - F2H

Positive Gamma Correction - E0H

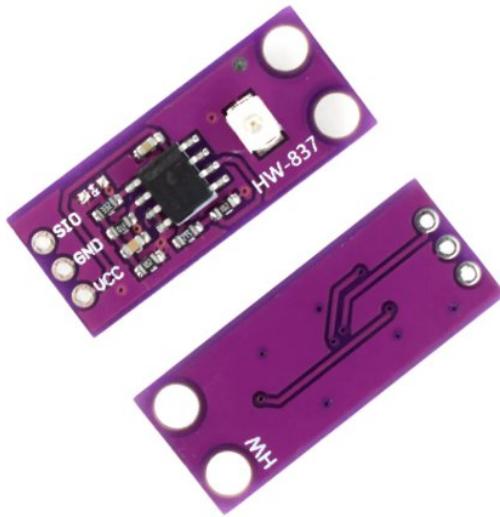
Negative Gamma Correction - E1H

2.1.5 ADC:

ADC là từ viết tắt của Analog-to-Digital Converter trong tiếng Anh, có nghĩa là bộ chuyển đổi. Nó là một thiết bị được sử dụng để chuyển đổi tín hiệu analog thành tín hiệu số. ADC là một phần quan trọng trong hầu hết các hệ thống điện tử hiện đại, nhất là trong lĩnh vực điện tử, viễn thông và điều khiển tự động. Nó cho phép hệ thống xử lý và hiểu được các tín hiệu tương tự từ các nguồn khác nhau như cảm biến, vòng lặp điều khiển và tín hiệu âm thanh.



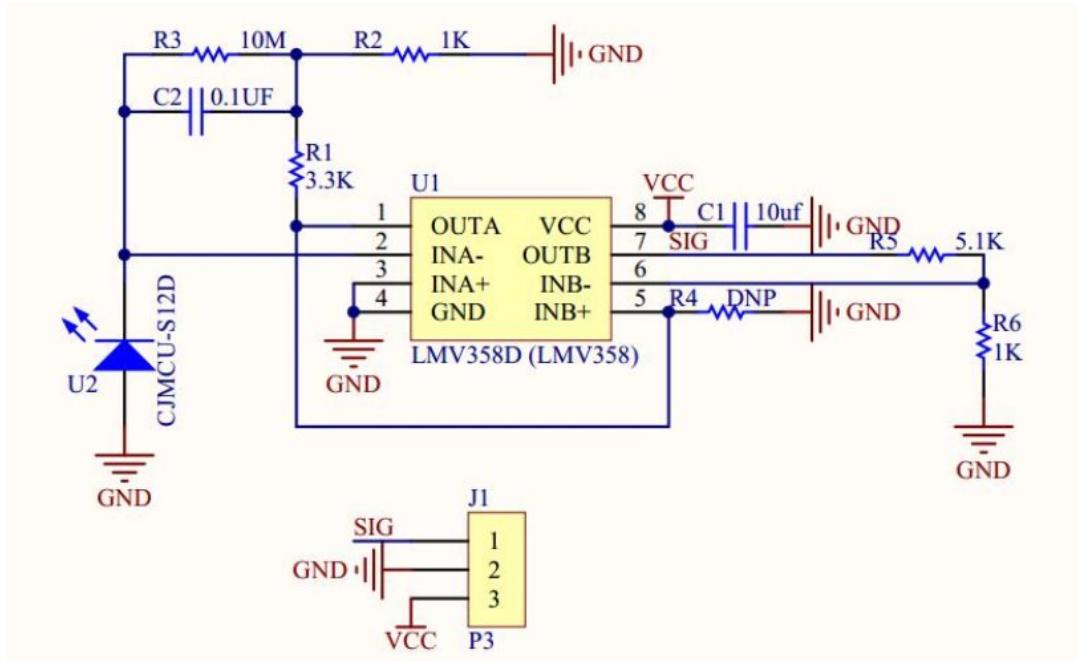
UV CJMCU-GUVA-S12SD:



Thông số:

- Phát hiện tia UV trong phạm vi 240nm – 370nm
- Đầu ra điện áp tương tự tỷ lệ với chỉ số UV của tia UV rơi trên máy dò
- Điện áp 3,3 và 5V
- Độ nhạy cao
- Độ ổn định cao

Sơ đồ mạch cảm biến:



Công thức :

$$\text{Voltage} = (\text{ADC} * 3.3) / 1024$$

UV = voltage * 10

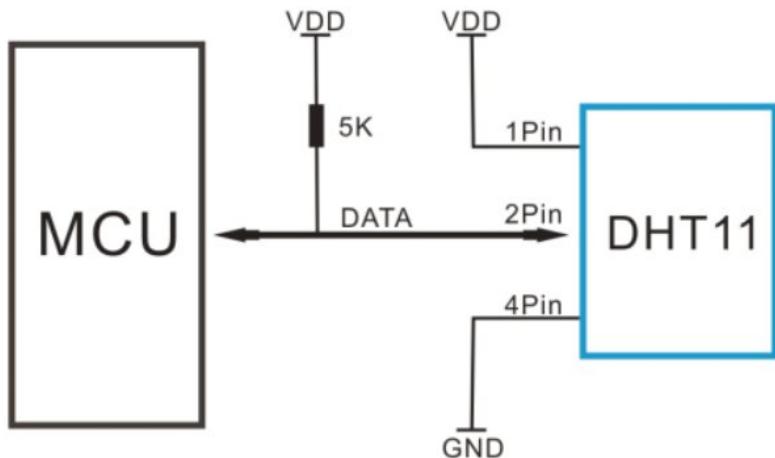
DHT11:

là một cảm biến đơn giản để đo nhiệt độ và độ ẩm. có độ tin cậy và độ ổn định cao do tín hiệu kỹ thuật số được hiệu chỉnh. Cảm biến này có thể dễ dàng giao tiếp với bất kỳ bộ vi điều khiển vi nào như Arduino, Raspberry Pi, ... để đo độ ẩm và nhiệt độ ngay lập tức.



Thông số:

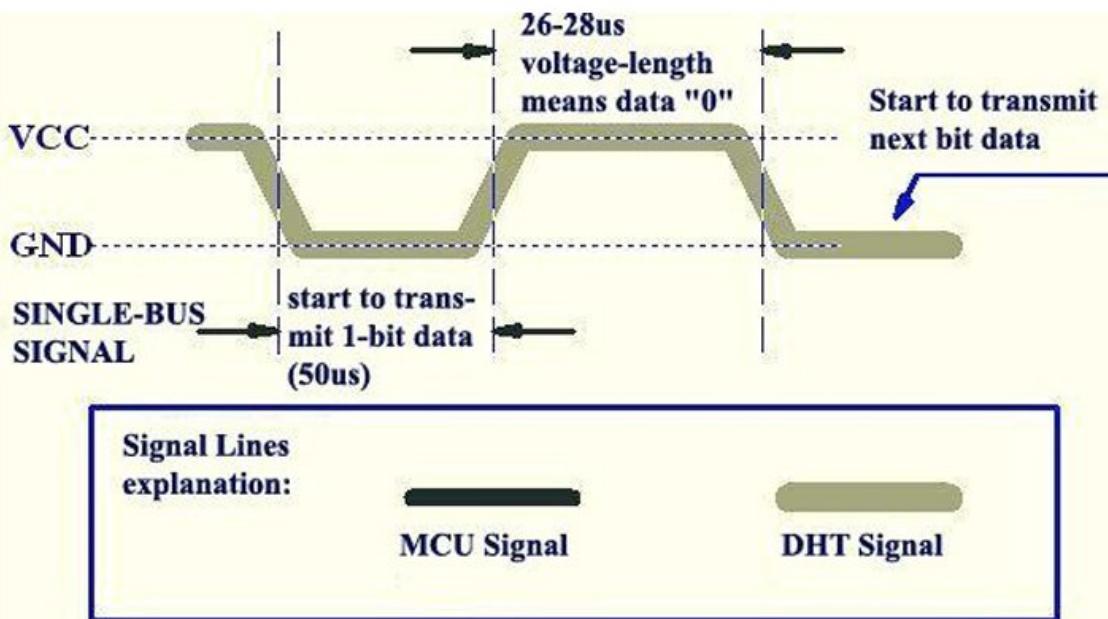
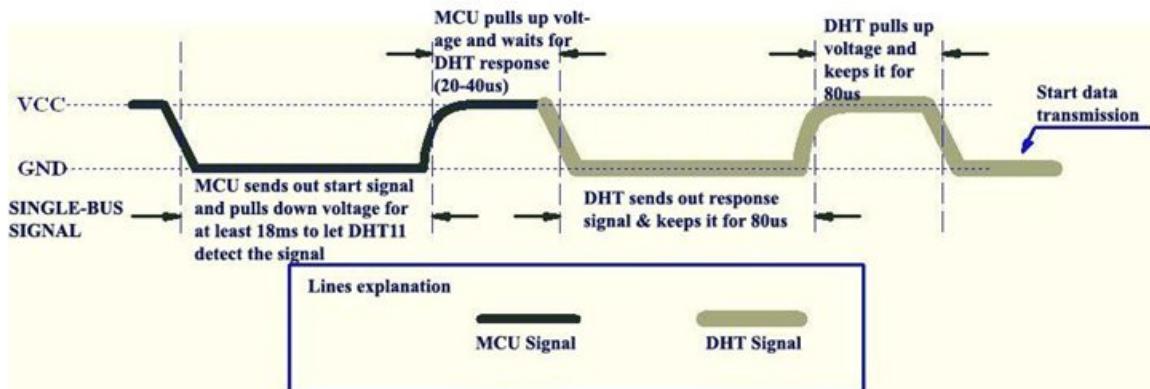
- Điện áp hoạt động: 3V – 5V (DC)
- Dải độ ẩm hoạt động: 20% – 90% RH, sai số $\pm 5\%$ RH
- Dải nhiệt độ hoạt động: 0°C ~ 50°C, sai số $\pm 2\%$ C
- Khoảng cách truyền tối đa: 20m

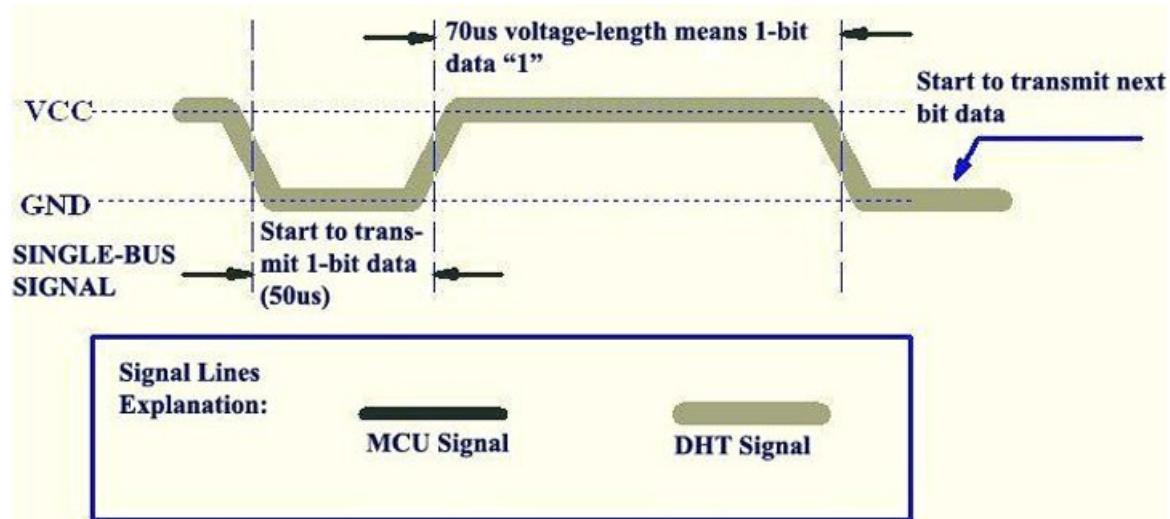


Nguyên lý :

Khi vào giai đoạn start : chân MCU khi kết nối với DHT11 lúc đầu sẽ cấu hình là output và sẽ kéo xuống mức thấp trong khoảng 18ms sau đó sẽ kéo lên mức cao từ 20 đến 40 ms sau đó sẽ đổi thành input để nhận tín hiệu từ DHT11 lúc này DHT11 sẽ gửi về MCU tín hiệu check đó là mức thấp 80us rồi kéo lên mức cao 80us sau đó kéo xuống mức thấp bắt đầu tín hiệu data. Vào giai đoạn đọc data DHT11 sẽ kéo xuống mức thấp 50us và nếu nó

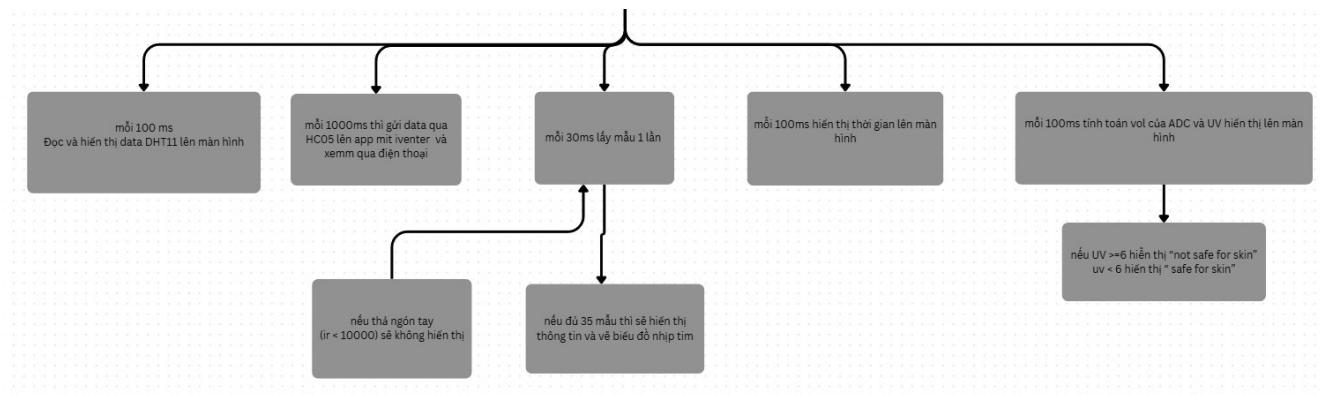
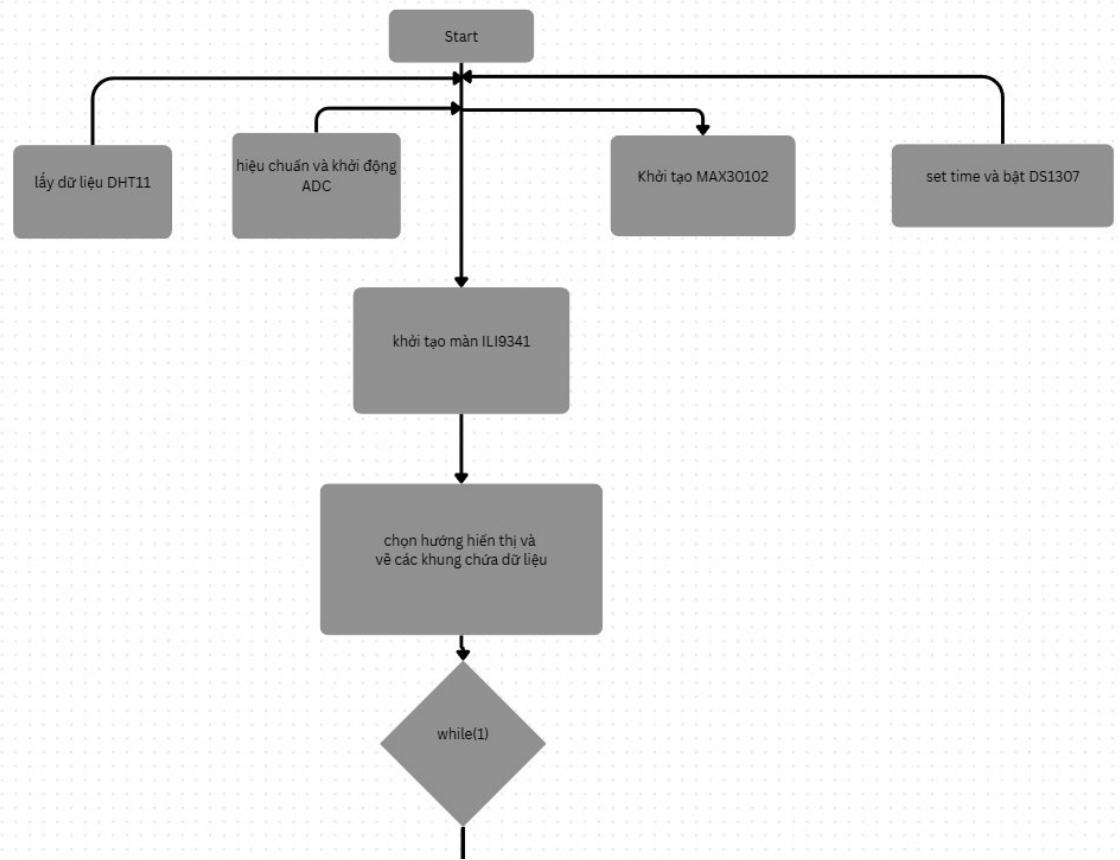
kéo lên cao từ 26 -28us là data = 0 còn nếu là kéo lên cao 70us là data =1. Cứ như vậy cho đến khi đọc đủ 40 bit.



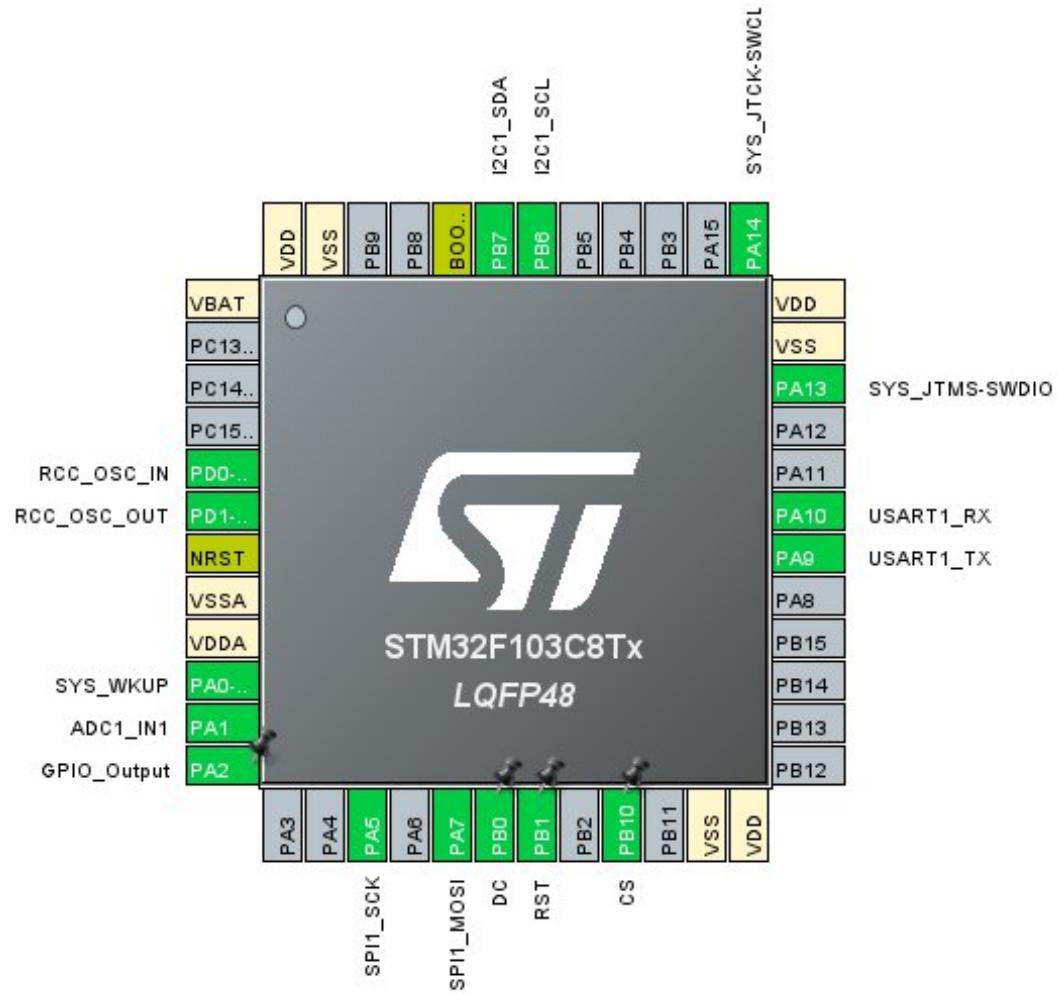


3. SƠ ĐỒ KHỐI VÀ HOẠT ĐỘNG

3.1 Sơ đồ khối



Cấu hình chân STM32:



Cấu hình các chân trên CubeMX:

- PA1: là chân ADC cho việc sử dụng GUVA-S12SD.
 - PA2: là chân kết nối với DHT11.
 - PA10, PA9: là hai chân kết nối với module HC05 sử dụng UART.
 - PA6,PA7: là 2 chân kết nối với DS1307, MAX30102 sử dụng I2C.
 - PA5,PA7,PB0,PB1,PB10: là các chân để kết nối với TFT ili9341 sử dụng SPI
 - PA13, PA14: Serial debug cho STM32.

3.2 Hoạt động:

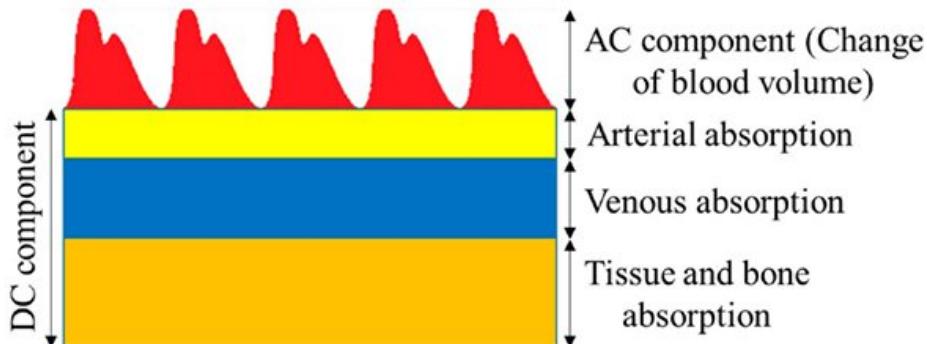
- a) Tính cường độ tia uv

Để tính được tia UV ta cần phải tính được:

- $\text{voltage} = (\text{ADC} * 3.3)/1024$
 - $\text{UV} = \text{VOLTAGE} * 10$

UV Index	0	1	2	3	4	5
Vout(mV)	<50	227	318	408	503	606
Analog Value	<10	46	65	83	103	124
UV Index	6	7	8	9	10	11 ⁺
Vout(mV)	696	795	881	976	1079	1170+
Analog Value	142	162	180	200	221	240

b) Tính nhịp tim và nồng độ oxy trong máu



Hình 19. Các tín hiệu qua phân tích sóng ánh sáng

- Định luật Lambert-Beer là cơ sở phân tích quang học dùng để thực hiện phép đo SpO₂ trong máu, định luật này biểu thị cách đo độ hấp thụ trong một sóng ánh sáng nhất định trên một chất, điều này sẽ phụ thuộc vào nồng độ của hợp chất hấp thụ, khoảng cách và loại sóng ánh sáng.
- PHOTOPLÉTHYOGRAPHIC: là phương pháp phân tích quang học dựa vào định luật trên. Giá trị thu được bằng cách chiếu ánh sáng vào một bên ngón tay với tần số tương ứng trong các khoảng thời gian khác nhau; ở đầu kia là một cảm biến thu nhận sự thay đổi của tín hiệu. Sóng

ánh sáng đi qua máu động mạch, được gọi là phần động (AC) và phần tĩnh (DC). Được tạo thành bởi các mô, xương, móng và da. Tín hiệu AC được phân tích và sử dụng, loại bỏ thành phần tĩnh DC.

- Công thức tính nhịp tim mỗi phút sẽ bằng số đỉnh của một khoảng thời gian chia cho khoảng thời gian đó và nhân với 60 để đổi ra đơn vị nhịp tim mỗi phút.

$$\text{Nhịp tim mỗi phút} = \frac{\text{Số đỉnh AC}}{\text{Thời gian lấy mẫu}} * 60$$

Hoặc:

Nhịp tim = (tần số lấy mẫu *60)/ khoảng cách trung bình giữa các đỉnh.

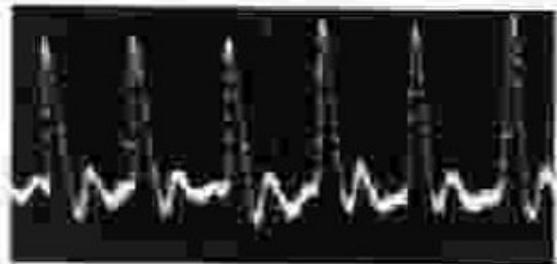
- Công thức tính nồng độ oxy trong máu (spo2):

$$R = \frac{\frac{AC_{red}}{DC_{red}}}{\frac{AC_{infrared}}{DC_{infrared}}} \quad \text{Spo2} = 107 - 14 * R$$

Công thức này đúng cho các sản phẩm đo nhịp tim và spo2 của dòng MAX30

c) Vẽ đồ thị nhịp tim

Mô phỏng lại PPG được sử dụng để vẽ nhịp tim. Tín hiệu PPG là Photoplethysmography là một công nghệ sử dụng ánh sáng để đo những biến đổi nhỏ trong lưu thông máu. Khi ánh sáng được chiếu lên da và cảm biến theo dõi sự hấp thụ ánh sáng trở lại, nó phân tích sự biến đổi trong lưu lượng máu đang chảy qua. Dựa trên thông tin này, cảm biến có thể đo lường và phân tích các thông số như nhịp tim và số lượng nhịp tim trong một khoảng thời gian cụ thể.



Hình 6. Tin hiệu sóng PPG

ở đề tài này chỉ là vẽ mô phỏng nên có thể không chính xác như khi có thiết bị. Biên độ tối đa của sóng là 50. Nếu giá trị red tăng đường tín hiệu đi xuống và ngược lại mỗi lần vẽ là 2 pixel nối với nhau và không được vượt quá khung vẽ.

d) Chương trình chính (nằm trong vòng lặp while (1))

Chạy 5 loại module khác nhau mỗi modul sẽ có delay khác nhau tuy nhiên e dùng systick nên gần như không có delay trong chương trình làm tránh xảy ra lỗi :

- 100ms sẽ hiện dữ liệu DHT11
- 100ms sẽ hiện giá trị ADC của GUVA-S12D
- 100ms sẽ hiện thời gian của DS1307
- 1000ms sẽ gửi giá trị về HC05 hiện lên điện thoại
- 30ms sẽ lấy mẫu 1 lần của max30102

4. Hình ảnh thực tế:

