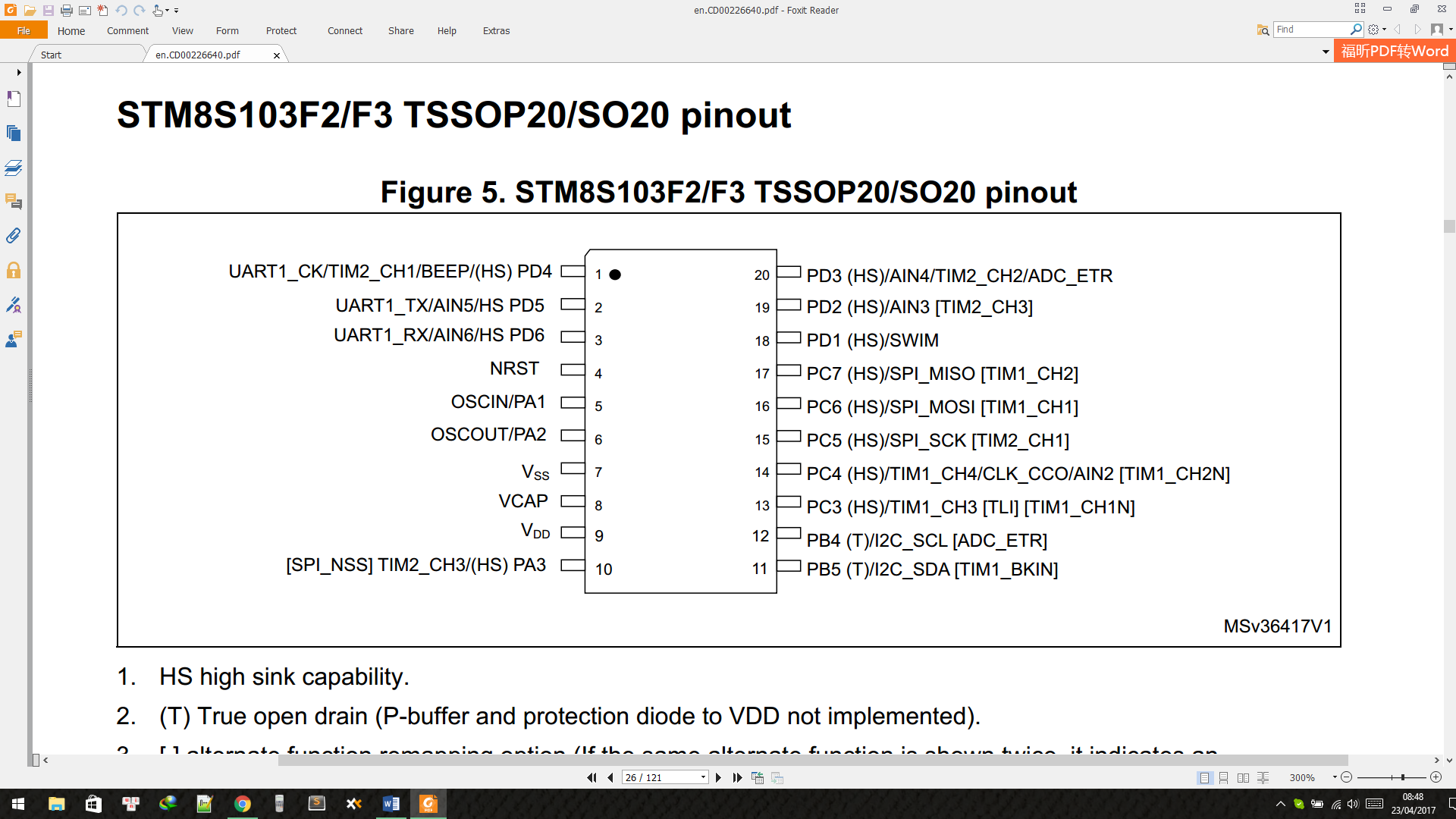
**GPIO OUTPUT**

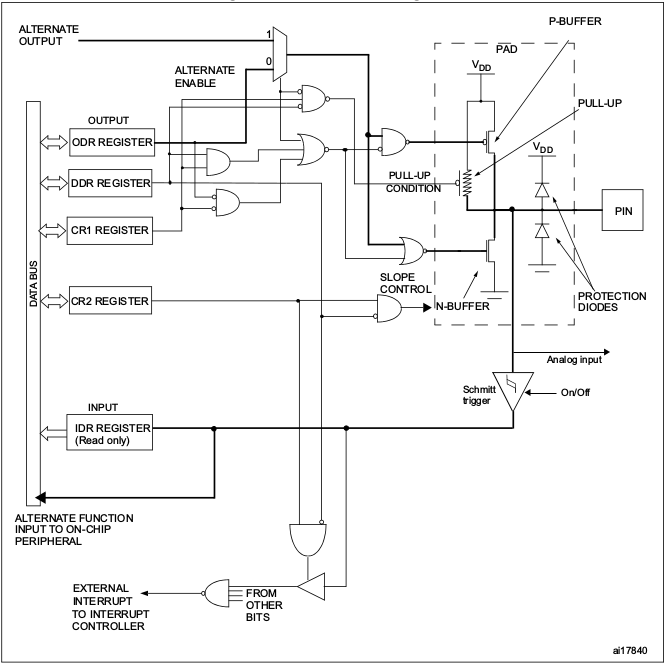
**1.Khái niệm**

General Purpose Input/Output (GPIO) là khối chức năng cơ bản với mọi loại vi điều khiển, nắm vai trò tương tác với bên ngoài. Một port I/O gồm tối đa 8 chân. Mỗi chân có thể được lập trình riêng là một đầu vào kỹ thuật số (digital input) hoặc đầu ra kỹ thuật số (digital output). Ngoài ra, một số port có thể có chức năng thay thế (alternative function – **AF**) như đầu vào tương tự (**ADC**), ngắt ngoài (**EXTI**), đầu vào / đầu ra cho **ngoại vi** trên chip. Chỉ có một chức năng thay thế có thể được ánh xạ tới một chân tại một thời điểm, việc ánh xạ chức năng thay thế được điều khiển bởi byte tùy chọn.



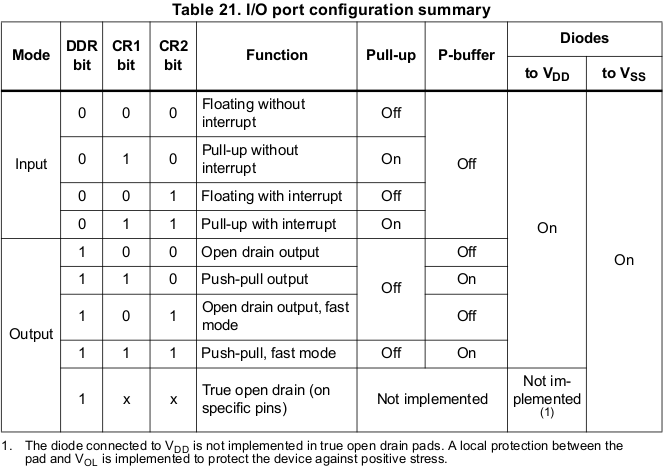
Mô tả vị trí các chân trên STM8S103F3P

**2. Nguyên lý hoạt động**

Sơ đồ một khối GPIO

Các bạn có thể thấy rõ các thanh ghi trong khối GPIO, trong đó, khác với các thanh ghi còn lại, thanh ghi Px\_IDR là thanh ghi chỉ đọc. Thiết kế bên trong của STM8S đã có trở kéo nội, các bạn không cần thiết phải sử dụng thêm trở kéo bên ngoài. Một số chân có kết nối tới ngoại vi sẽ có thể lựa chọn làm chân của giao tiếp của ngoại vi tương ứng hoặc giữ chức năng như GPIO thông qua một **bit cho phép**(Alternate Enable) trong thanh ghi cấu hình của khối ngoại vi đó. Các chân sẽ có đầu vào chịu được mức điện áp tối đa là 5V Danh sách các chân I/O và các chức năng thay thế tương ứng, các bạn tham khảo tragn **datasheet trang** **26**.

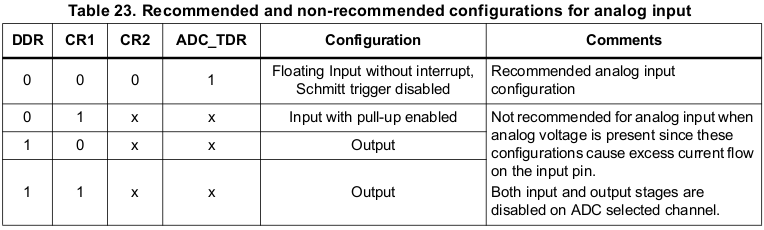
Về cấu hình hoạt động của các chân dưới vai trò của GPIO, các bạn tham khảo trong **bảng 17, trang 101 – tài liệu hướng dẫn tham chiếu**.

Bảng tổng hợp cấu hình I/O

Một vài khái niệm có thể khiến nhiều bạn băn khoăn:

* Với cấu hình là đầu vào:
  + Pull-up: bạn sẽ sử dụng trở kéo trong, mặc định ở trạng thái kéo cao. Đây là hình thức được sử dụng phổ biến. Schmitt trigger luôn bật.
  + Floating: một số trường hợp có thể gọi là High Impedance Input (đầu vào trở kháng cao) chân bị thả nổi mức tín hiệu, mức tín hiệu trên chân vi điều khiển mặc định ở trạng thái không xác định. Mức tín hiệu phụ thuộc vào nguồn ở bên ngoài
* Với cấu hình là đầu ra:
  + Push-pull: đầu ra mức logic luôn nằm trong hai lựa chọn 0 hoặc 1.
  + Open-drain: tương tự floating input, đầu ra mức tín hiệu phụ thuộc vào nguồn ở bên ngoài.

Floating input được dùng trong trường hợp phụ thuộc hoàn toàn vào nguồn tác nhân bên ngoài, tiêu biểu là ADC. Open-drain ouput được sử dụng trong trường hợp muốn điều khiển nhiều thiết bị tiêu thụ dòng với chỉ 1 vi điều khiển, kèm thiết kế trở treo bên ngoài. Cũng vì vậy, Open-drain ouput hay xuất hiện trong các ứng dụng yêu cầu mạch lái, tiêu biểu nhất là bus I2C.



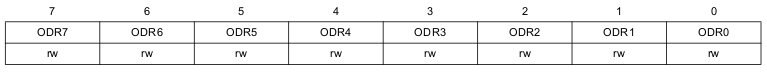
Cấu hình khuyến nghị đối với analog input.

**3.Các thanh ghi liên quan**

Các bit trong mỗi thanh ghi của một cổng sẽ điều khiển chân ứng của cổng đó.

a.Port x output data register (Px\_ODR)

Đây là thanh ghi dữ liệu đầu ra của cổng. Mô tả cụ thể như sau:

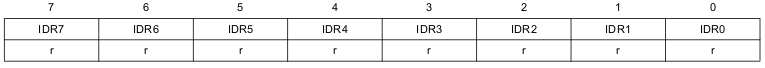
Thanh ghi dữ liệu đầu ra Px\_ODR

Giá trị ghi vào thanh ghi ODR trong chế độ output được áp dụng trên I/O thông qua mạch chốt (latch). Giá trị đọc ra từ thanh ghi ODR là giá trị lần chốt dữ liệu trước đó. Trong chế độ input, dữ liệu vẫn ghi được vào thanh ghi ODR nhưng không được áp dụng trên I/O.

Giá trị sau reset: 0x00

b.Port x input data register (Px\_IDR)

Đây là thanh ghi dữ liệu đầu vào của cổng, là thanh ghi chỉ đọc. Có thể sử dụng để đọc giá trị trên chân I/O dù ở chế độ input hay output. Mô tả cụ thể như sau:

Thanh ghi dữ liệu đầu vào Px\_IDR

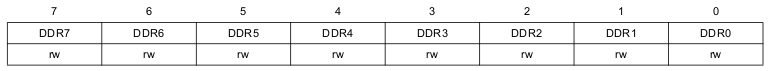
Giá trị sau reset: phụ thuộc vào bên ngoài

c. Port x data direction register (Px\_DDR)

Đây là thanh ghi định hướng dữ liệu. Các bit trong thanh ghi được cấu hình từ phần mềm để lựa chọn hướng dữ liệu của từng chân I/O cụ thể.

* 0: input
* 1: output

Mô tả cụ thể như sau:

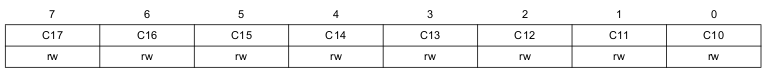
Thanh ghi định hướng PxDDR

Giá trị sau reset: phụ thuộc vào bên ngoài

d. Port x control register 1 (Px\_CR1)

Đây là một trong 2 thanh ghi cấu hình của khối GPIO. Nhiệm vụ của nó là lựa chọn cấu hình làm việc của chân I/O đã đề cập tới trong **bảng 21, trang 106 – tài liệu hướng dẫn tham chiếu**, các bạn có thể xem lại ở bên trên hoặc trong tài liệu. Các bit được thiết lập và xóa bởi phần mềm.

* Input mode (DDR = 0):  
  0: Floating input  
  1: Input với trở kéo cao
* Output mode (DDR = 1):  
  0: Giả open-drain  
  1: Push-pull, tần số hoạt động tối đa phụ thuộc vào bit tương ứng của CR2

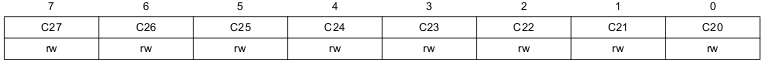
Thanh ghi cấu hình 1 Px\_CR1

Giá trị sau reset: 0x00, **riêng PD\_CR1 là 0x02**

e. Port x control register 2 (Px\_CR2)

Đây là thanh ghi cấu hình thứ 2 của khối GPIO. Nhiệm vụ của nó có sự khác biệt giữa 2 chế độ input và output. Ở chế độ input, đây là các bit cấu hình ngắt ngoài. Nếu chân I/O không hỗ trợ ngắt ngoài, CR2 không có tác dụng. Ở chế độ output, đây là các bit thiết lập tần số làm việc tối đa trên I/O. Các bit được thiết lập và xóa bởi phần mềm.

* Input mode (DDR = 0):  
  0: Không cho phép ngắt ngoài  
  1: Cho phép ngắt ngoài
* Output mode (DDR = 1):  
  0: tốc độ lên đến 2 MHz  
  1: tốc độ lên đến 10 MHz

Thanh ghi cấu hình 2 Px\_CR2

Giá trị sau reset: 0x00.

**4.Các hàm liên quan**

|  |  |  |
| --- | --- | --- |
| GPIO\_Init(GPIOx, GPIO\_Pin, GPIO\_Mode) | | |
| cấu hình chân | | |
| GPIOx | GPIOA | Port A của vi điều khiển |
|  | GPIOB |  |
|  | GPIOC |  |
|  | GPIOD |  |
| GPIO\_Pin | GPIO\_PIN\_0 | Chân 0 của vi điều khiển |
|  | ………………………………….. |  |
|  | GPIO\_PIN\_7 |  |
|  | GPIO\_PIN\_ALL |  |
| GPIO\_Mode | GPIO\_MODE\_IN\_FL\_NO\_IT | Input floating, no external interrupt |
|  | GPIO\_MODE\_IN\_PU\_NO\_IT | Input pull-up, no external interrupt |
|  | GPIO\_MODE\_IN\_FL\_IT | Input floating, external interrupt |
|  | GPIO\_MODE\_IN\_PU\_IT | Input pull-up, external interrupt |
|  | GPIO\_MODE\_OUT\_PP\_LOW\_SLOW | Output push-pull, low level, 2MHz |
|  | ……………………………………….. |  |

|  |  |  |
| --- | --- | --- |
| GPIO\_WriteReverse(GPIOx, PortPins) | | |
| Đảo trạng thái chân hoặc port của vi điều khiển | | |
| GPIOx | GPIOA | Port A của vi điều khiển |
|  | GPIOB |  |
|  | GPIOC |  |
|  | GPIOD |  |
| PortPins | GPIO\_PIN\_0 | Chân 0 của vi điều khiển |
|  | ………………………………….. |  |
|  | GPIO\_PIN\_7 |  |
|  | GPIO\_PIN\_ALL |  |

|  |  |  |
| --- | --- | --- |
| GPIO\_WriteHigh (GPIOx, PortPins)  GPIO\_WriteLow(GPIOx, PortPins) | | |
| Xuất chân hoặc port của vi điều khiển | | |
| GPIOx | GPIOA | Port A của vi điều khiển |
|  | GPIOB |  |
|  | GPIOC |  |
|  | GPIOD |  |
| PortPins | GPIO\_PIN\_0 | Chân 0 của vi điều khiển |
|  | ………………………………….. |  |
|  | GPIO\_PIN\_7 |  |
|  | GPIO\_PIN\_ALL |  |

|  |  |  |
| --- | --- | --- |
| GPIO\_ReadInputData (GPIOx) | | |
| Đọc tín hiệu port của vi điều khiển | | |
| GPIOx | GPIOA | Port A của vi điều khiển |
|  | GPIOB |  |
|  | GPIOC |  |
|  | GPIOD |  |

**5.Quy trình khởi tạo một chân GPIO output**

--Cấu hình chân ngõ ra push pull.

GPIO\_Init(GPIOD,(GPIO\_Pin\_TypeDef)GPIO\_PIN\_2,GPIO\_MODE\_OUT\_PP\_LOW\_FAST);

--Viết chương trình.

**6.Code mẫu**

VD: code mẫu chớp tắt led chân D2

|  |
| --- |
| #include "stm8s.h"  //#define USE\_RES  void Delay (uint16\_t nCount);  void main(void)  {  #ifdef USE\_RES  /\* cau hinh PD7 la output  \* che do push-pull  \* tan so lam viec toi da 2MHz After reset, the microcontroller restarts by default with an internal 2 MHz clock (HSI/8)  \*Trang 15 datasheet STM8S  \*/  GPIOD->DDR |= GPIO\_PIN\_2;//data direction registor thanh ghi dinh huong du lieu cua tung chan i/o cu the, th nay la out put  GPIOD->CR1 |= GPIO\_PIN\_2;//lua chon cau hinh lam viec cua chan io che do push pull do DDR=1,  /\*CR2 o che do input cac bit duoc cau hinh ngat ngoai,output cac bit duoc lam viec toi da do bit CR2=0 nen  chay max 2Mhz con neu CR2=1 chay max 10Mhz reference 101 page\*/  GPIOD->CR2 &= ~GPIO\_PIN\_2;//=0 chon o che do output push pull xem bang 17/page101 reference  #else  GPIO\_Init(GPIOD, (GPIO\_Pin\_TypeDef)GPIO\_PIN\_2, GPIO\_MODE\_OUT\_PP\_LOW\_FAST);  #endif    while(1)  {  Delay(30000);  #ifdef USE\_RES  GPIOD->ODR ^= GPIO\_PIN\_2;  #else  GPIO\_WriteReverse(GPIOD, (GPIO\_Pin\_TypeDef)GPIO\_PIN\_2);  #endif  }    }  /\*Ham delay \*/  void Delay(uint16\_t nCount)  {  /\* Decrement nCount value \*/  while (nCount != 0)  {  nCount--;  }  } |