



UNIVERSIDADE FEDERAL DE MINAS GERAIS – UFMG  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA - DEE  
Programação de Sistemas Embarcados - Prof. Ricardo Duarte

## **AN1 - APPLICATION NOTE**

### **ANALOG TO DIGITAL CONVERTER**

Nander Santos do Carmo  
Marcos Vinícios Ribeiro

Belo Horizonte, março de 2021

## SUMÁRIO

INFORMAÇÕES GERAIS.....	4
GUIA DE USO.....	6
Ambiente.....	6
Hardware.....	6
Configurando o Modo de Operação.....	7
Habilitando o DMA.....	10
A Aplicação.....	10
BIBLIOGRAFIA.....	12

## LISTA DE IMAGENS

Figura 1: Diagrama de Blocos do ADC.....	5
Figura 2: YwRobot Easy Module Shield V1.....	6
Figura 3: NUCLEO-G474RE Pinouts.....	7
Figura 4: ADC e Canal Conectados no Pino PA0.....	7
Figura 5: Configuração do ADC1.....	9
Figura 6: Configuração do DMA.....	10
Figura 7: Valores de Tensão do Potenciômetro Lidos pelo ADC.....	11

## INFORMAÇÕES GERAIS

Essa *application note* tem como objetivo fornecer o passo a passo de como configurar e utilizar o conversor analógico-digital presentes nos kits STM32 presentes na família NUCLEO-G4. Algumas informações iniciais básicas acerca do ADC presente no kit que serão tomadas como conhecidas neste documento são:

- O NUCLEO-G474RE possui até 5 ADC's
  - ADC1 e ADC2 são acoplados e podem operar em modo duplo (ADC1 é o mestre)
  - ADC3 e ADC4 são acoplados e podem operar em modo duplo (ADC3 é o mestre)
  - ADC5 é controlado de forma independente
- Cada ADC possui até 12 bits de resolução, podendo ser configurados para operar com 6, 8, 10 ou 12 bits
- Os ADC's são implementados usando o método de aproximação sucessivas
- Cada ADC possui até 19 canais multiplexados
- Os ADC's podem ser configurados para funcionar no modo de conversão única, conversão contínua, varredura ou modo descontínuo
- O valor de leitura de cada ADC armazenado no registrador pode ser alinhado à direita ou à esquerda
- O tempo de conversão do ADC depende da frequência de *clock* do barramento AHB
- Os ADC's permitem expandir a resolução para até 16bits
- Os ADC's permitem fazer um ajuste de ganho e *offset* das leituras

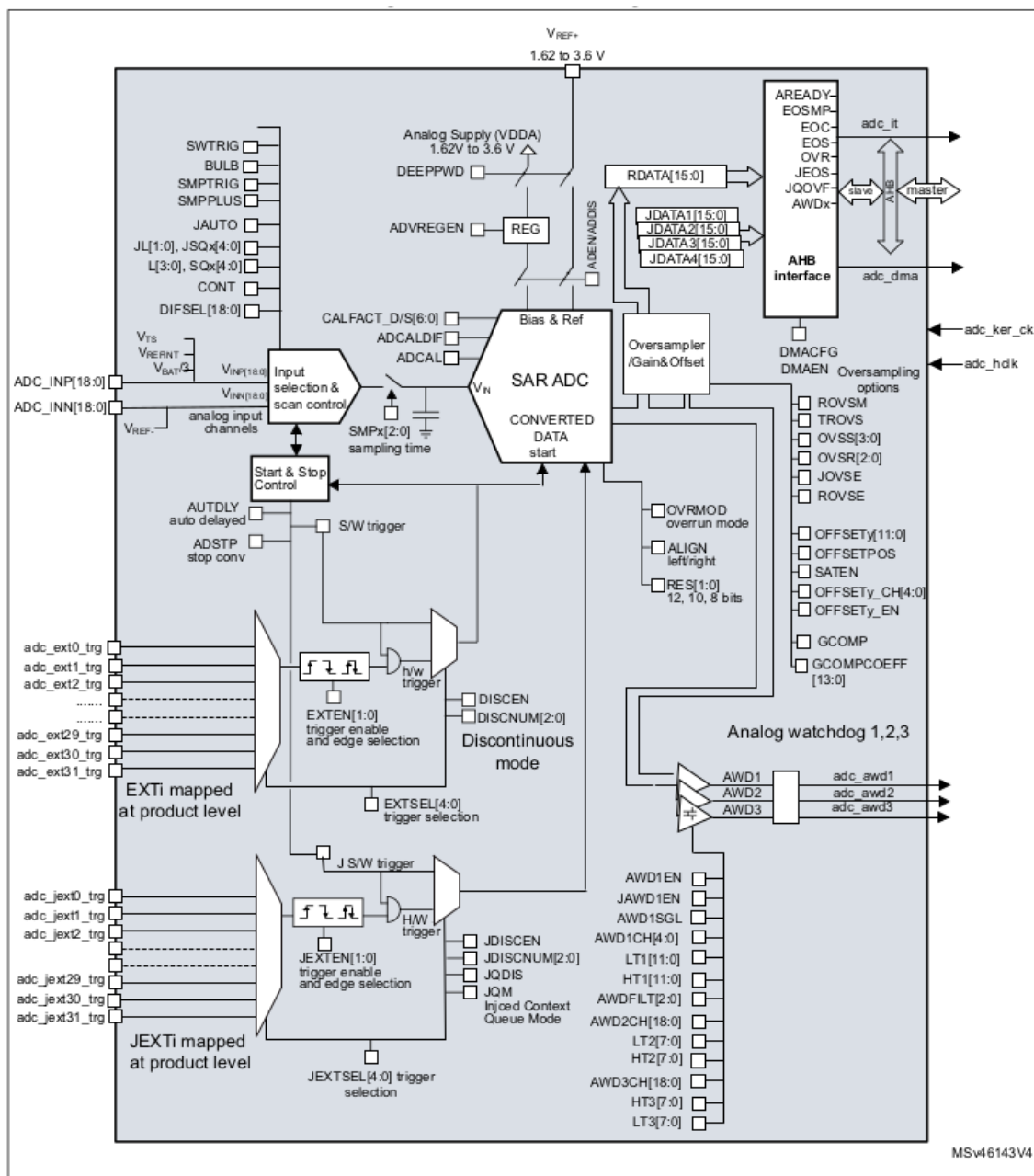


Figura 1: Diagrama de Blocos do ADC

A Figura 1 apresenta o diagrama de blocos de um ADC, onde é possível observar a unidade principal de conversão de dados SAR ADC, a interface AHB e o controlador de início e parada de leitura, *Start & Stop Control*, o controlador do canal a ser lido (modo single ou em varredura), o controlador de *oversampling* e de ganho/offset, dentre outras unidades importantes.

## GUIA DE USO

### Ambiente

Esse documento fornece um guia, passo a passo, de como configurar e utilizar o ADC do kit NUCLEO-G474RE, utilizando a plataforma STMCubeMX e a IDE SW4STM32, logo o código mostrado e as figuras das janelas de configuração apresentadas são referentes a um projeto desenvolvido utilizando essas ferramentas.

### Hardware

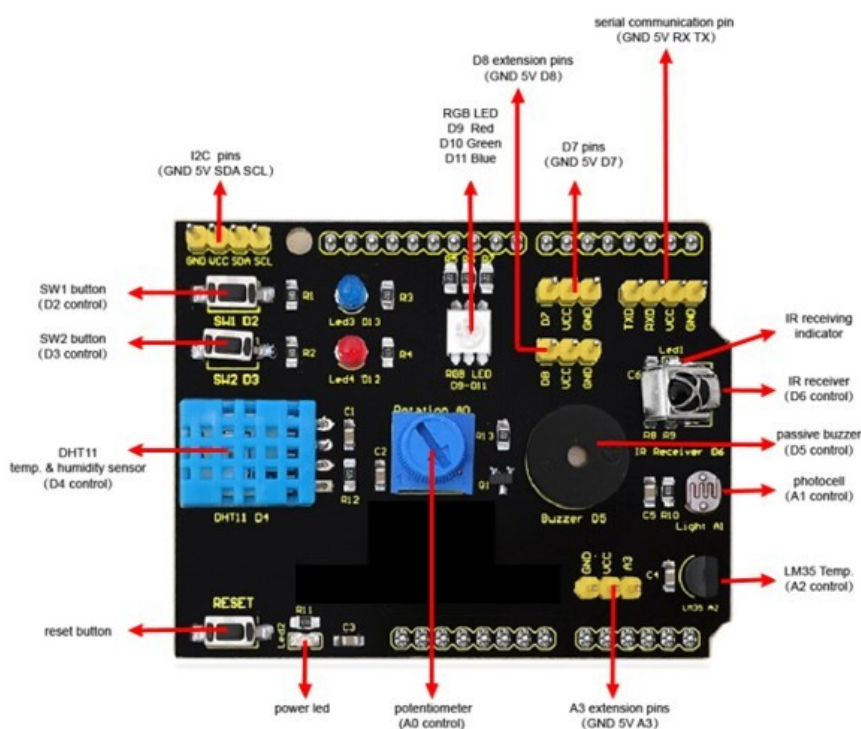


Figura 2: YwRobot Easy Module Shield V1

De forma a simplificar a utilização do ADC será considerado o uso da *shield* para Arduino YwRobot Easy Module Shield V. A Figura 2 mostra os sensores e componentes presentes nessa *shield* e os respectivos pinos aos quais eles estão conectados. Esses pinos são referentes aos conectores

Arduino do kit da STM, logo, através da é possível saber quais os pinos do kit serão utilizados. Para demonstração optou-se por utilizar o potenciômetro que está conectado no pino A0, logo, PA0, do kit.

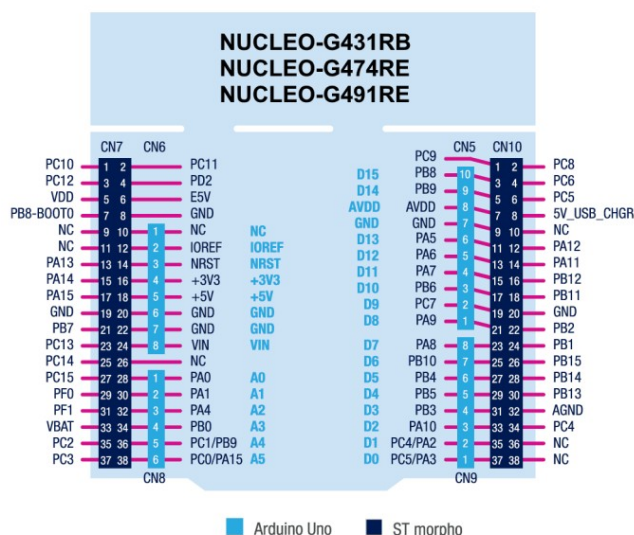


Figura 3: NUCLEO-G474RE Pinouts

## Configurando o Modo de Operação

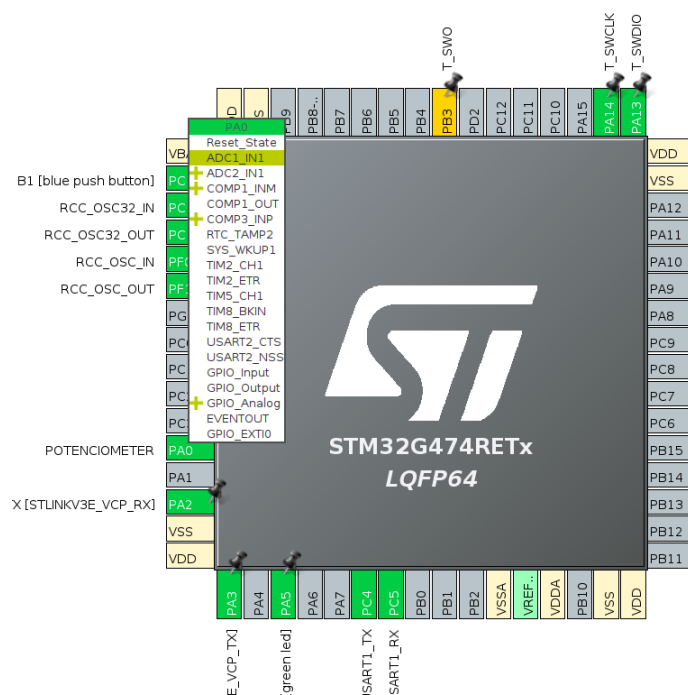


Figura 4: ADC e Canal Conectados no Pino PA0

Este documento, visa exemplificar e demonstrar a operação do ADC no modo de conversão contínua, para um único canal com uso de DMA. Dessa forma, é preciso primeiro, definir qual ADC e qual canal será utilizado para fazer a aquisição dos dados. Isso é feito através da escolha do tipo de comportamento do pino PA0 como ADC1\_1N, como pode ser visto na Figura 4, definindo assim que esse pino está conectado ao canal 1 do ADC1 e no modo de entrada *Single Ended*, o que resetou o bit correspondente ao canal 1 do registrador ADC\_DIFSEL.

Na sequência é preciso definir os parâmetros que vão setar os registradores de controle do ADC, de forma a definir o comportamento de leitura que se quer obter. Para isso, no *CubeMX*, na seção *Analog*, escolhendo o ADC1 e o canal 1, na seção *Parameter Settings*, foram definidos os seguintes parâmetros de interesse:

- Clock Prescaler
  - Divide o clock padrão do ADC, no caso 170Mhz, por 256, de forma a diminuir a taxa de amostragem pelo maior valor possível
- Resolution
  - Define a resolução do ADC como 12 bits, logo definindo o tempo de conversão como 12.5 ciclos de clock do ADC
- Data Aligment
  - Define o alinhamento dos dados do registrador como alinhados à direita
- End of Conversion Selection
  - Como irá ser utilizado apenas um canal e pretende-se monitorar o final de cada conversão, esse parâmetro foi definido de forma a gerar uma interrupção sempre que uma conversão for finalizada, disparando o sinal EOC, indicando que um novo dado está disponível no registrador ADC\_DR

- Continuous Conversion Mode
  - Esse parâmetro habilita o modo de operação de leitura contínua, setando o bit CONT no registrador ADC\_CR
- DMA Continuous Requests
  - Habilita o uso de DMA no sentido Periférico → Memória, de forma a realizar o armazenamento dos dados lidos em um buffer na memória de forma mais eficiente com relação ao uso da CPU. Isso seta o bit DMAEN do registrador ADC\_CFGR

Esses parâmetros de maior interesse, assim como os demais que não foram mencionados podem ser observados na Figura 5.

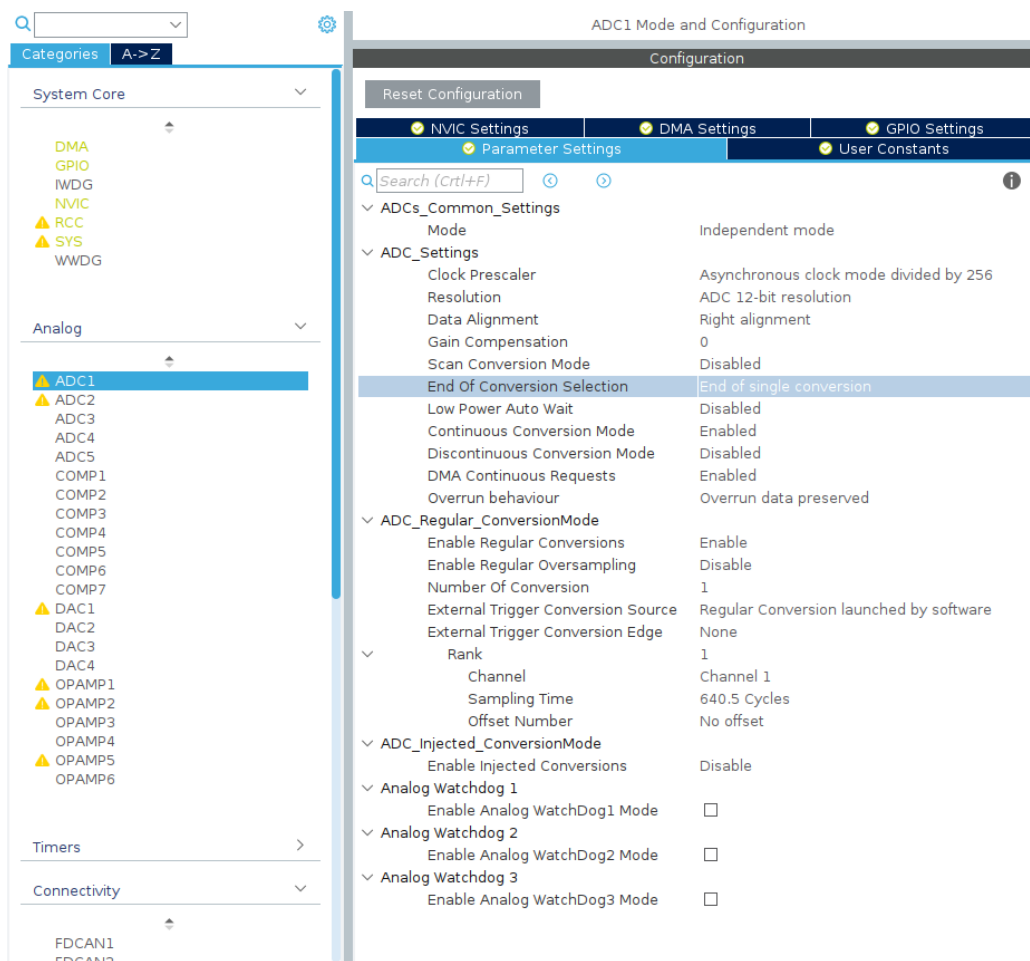


Figura 5: Configuração do ADC1



## Habilitando o DMA

Como já mencionado, de forma a otimizar o uso da CPU, como será utilizado o modo de conversão contínua, será necessário configurar o DMA, de forma a armazenar os valores lidos em um buffer na memória, de forma a serem lidos e tratados. Para isso é preciso selecionar, dentro da seção *System Core* do CubeMX, a opção DMA e adicionar um novo canal DMA associado ao ADC1, conforme mostrado na Figura 6.

DMA Request	Channel	Direction	Priority
ADC1	DMA1 Channel 2	Peripheral To Memory	Low

**Add** **Delete**

**DMA Request Settings**

	Peripheral	Memory
Mode	Circular	
Increment Address	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Data Width	Half Word	Half Word

**DMA Request Synchronization Settings**

Enable synchronization	<input type="checkbox"/>
Synchronization signal	
Signal polarity	
Enable event	<input type="checkbox"/>
Request number	

Figura 6: Configuração do DMA

Explicar o princípio de funcionamento do DMA foge do escopo deste documento e não será abordado aqui. Porém, dessa forma, o canal entre o ADC1 e a memória foi configurado de forma a ter um tamanho de dado de 16bits, compatível com o tamanho do dado coletado pelo ADC e para operar de forma circular, permitindo que o buffer seja constantemente preenchido.

## A Aplicação

Para demonstrar a operação do ADC operando em modo de conversão contínua utilizando DMA com apenas um canal, foi desenvolvido um pequeno

projeto demonstrativo, disponível em: <https://github.com/NanderSantos/My-STM32-ADC-AN>.

Essa aplicação, basicamente coleta os valores lidos pelo ADC e os armazena em um buffer. Sempre que a interrupção indicando que uma conversão terminou acontece o valor armazenado no buffer é enviado através da serial para um computador onde o dado pode ser observado.

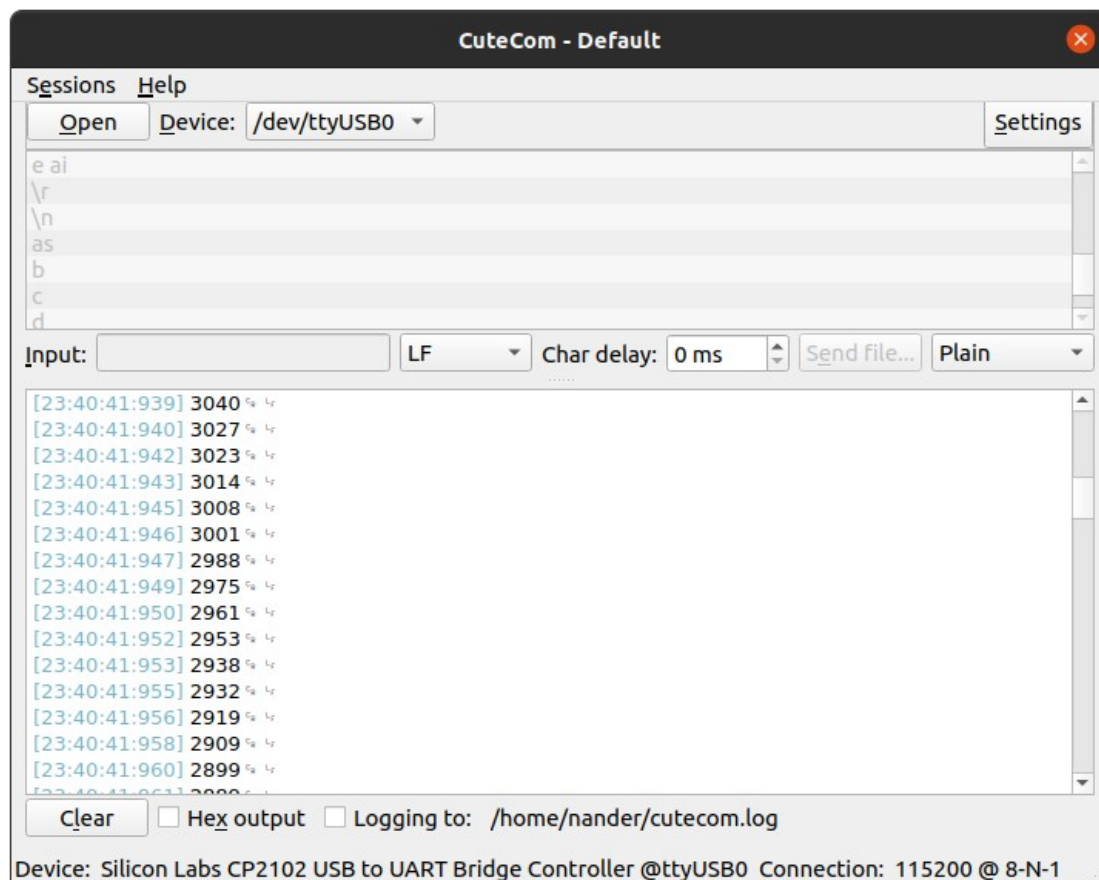


Figura 7: Valores de Tensão do Potenciômetro Lidos pelo ADC

O código do programa desenvolvido deve ser observado no repositório citado acima.

---

## BIBLIOGRAFIA

ST. **RM0440 Reference Manual.** Disponível em:  
<[https://www.st.com/resource/en/reference\\_manual/dm00355726-stm32g4-series-advanced-armbased-32bit-mcus-stmicroelectronics.pdf](https://www.st.com/resource/en/reference_manual/dm00355726-stm32g4-series-advanced-armbased-32bit-mcus-stmicroelectronics.pdf)>. Acesso em março de 2021.

ST. **UM2505 User Manual.** Disponível em:  
<[https://www.st.com/content/ccc/resource/technical/document/user\\_manual/group1/b2/79/ff/9e/f2/ea/44/cb/DM00556337/files/DM00556337.pdf/jcr:content/translations/en.DM00556337.pdf](https://www.st.com/content/ccc/resource/technical/document/user_manual/group1/b2/79/ff/9e/f2/ea/44/cb/DM00556337/files/DM00556337.pdf/jcr:content/translations/en.DM00556337.pdf)>. Acesso em março de 2021.

DeepBlue. **STM32 ADC Tutorial – Complete Guide With Examples.** Disponível em: <<https://deepbluembedded.com/stm32-adc-tutorial-complete-guide-with-examples/>>. Acesso em março de 2021.