

04 ADC

5 Mayıs 2021 Çarşamba 08:02

03 ADC

Giriş

- Doğada var olan bütün fiziksel büyüklükler (ısı, ışık, ses, zaman vs.) analog büyüklük kavramına girer.
- Dünyadaki herhangi bir şeyi dijital sistemlerimiz ile ölçmek, değerlendirmek, işlemek ve bu değerlere göre işlem yapabilmek için ADC (Analog Digital Converter) ihtiyaç vardır.
- ADC modülleri gerek harici, gerek dahili olsun hepsi bir referans voltaja ihtiyaç duyarlar. Genellikle mikro işlemcilerde referans voltajı işlemcinin besleme gerilimidir. Bu değer aynı zamanda ayarlar yapılarak harici olarak verilebilir.
- STM32'de 12-bit ADC, ardışık yaklaşım prensibine dayanan bir analog-dijital çeviricidir. Bu çevirici, 16 harici kaynaktan, iki dahili kaynaktan ve VBAT kanalından gelen sinyalleri ölçebilmek için en fazla 19 multiplexli kanala sahiptir. Kanalların A/D dönüşümü single, continuous, scan veya discontinuous modda gerçekleştirilebilir. ADC'nin sonucu, sola ya da sağa hizalanmış 16-bit veri kaydına depolanır.
- Analog watchdog özelliği, uygulamanın giriş voltajının kullanıcı tanımlı üst veya alt sınırları aşmasını algılamasına olanak tanır.

Çözünürlük

- ADC'ler 10, 12, 16, 24 vb. bit çözünürlükte bulunurlar.
- STM32F407'de ADC'ler 6, 8, 10 ve 12 bit çözünürlükte çalışabilirler ve referans voltajı default 3.3V'dur.
- ADC modülün 10 bit olduğunu düşünelim. $2^{10} = 1024$ değeri okunacak maksimum değerdir yani $0V = 0$, $3.3V = 1023$ değeri bize döner. Buradan her bit değer alacağı voltaj değerini $3.3 / 1024 = 0.0032$ olarak buluruz. Buradan da biz ADC modülünden okuduğumuz değeri bu ifade ile çarparsak voltaj değerini buluruz. $640 \text{ değeri için } 640 * 0.0032 = 2.048 \text{ V}$ olarak buluruz.
- STM32F407'de 0 - 3.6V aralığında ölçümler yapılabilir. Buradaki voltaj aralığında ADC birimin beslemesi (VDDA-VSSA) ile ilgili bir durumdur.
- ADC birimin besleme voltajı (VDD) ve referans gerilimi (VREF), ADC birimin ölçebileceği gerilim aralığını belirler.
- Her ne olursa olsun ADC birimi 3.6V'dan fazlasını ölçemez.
- Analog bir değerden dijital bir değer dönüşüm yapılırken dikkat edilmesi gereken hususlar vardır. Bunlardan en önemlisi, ölçülecek analog gerilim değerinin dönüşümü yapacak çipin **ölçüm aralığında** olması gerekir. Diğer en önemli nokta, ölçüm yapılacak **hassasiyetin belirlenmesi** ve buna uygun bir **genişliğinde** bir dönüştürücü seçilmelidir.
- Ölçüm hassasiyetinde önemli olan dönüşüm yapacak sistemin bir **çözünürlüğüdür**.
- $\text{Resolution} = VREF / (2^n - 1)$
- Örneğin 0 - 3.3V aralığı arası ölçüm yapabilen bir ADC ölçüm ünitesinin ölçebileceği minimum değer yaklaşık olarak formülden 8 bit çözünürlük için 12mV, 10 bit çözünürlük için 3,2mV, 12 bit çözünürlük için 805uV'tur.
- Çözünürlük arttıkça (bit sayısı arttıkça), ADC'nin ölçebileceği minimum voltaj değeri küçülür ve bu da daha hassas ölçümler yapabilmenizi sağlar.

Çevrim Süresi

- <https://controllerstech.com/adc-conversion-time-frequency-calculation-in-stm32/> linkten ADC için çevrim süresinin nasıl hesaplandığı ile ilgili yazıyı okuyabiliriz.
- STM32F407'de ADC birimin ulaşabileceği maximum hız 36 MHz'dir. Bu hız aynı zamanda ADC çözünürlüğü ile ters orantılıdır. Çözünürlük **arttıkça** ADC birimin ölçüm hızı **düşmektedir**.
- Çevrim süresi hesabı için üç değere ihtiyaç var. Bunlar **Cycles**, **Sampling Time** ve **Clock**'tur.
- Cycles değeri seçilen Resolution değerine bağlıdır.
- Sampling Time ve Clock değerleri ise istediğimiz çevrim süresine göre değiştirebiliriz.
- ADC'de örnekleme süresi genellikle 3, 15, 28, 56, 84, 112, 144, 480 ADC saat çevrimi (cycles) olarak seçilebilir. Bu, sinyali yeterince doğru bir şekilde yakalamak için önemlidir.
- Clock değeri ADC'nin bağlı olduğu clock hattına bağlıdır.
- Tüm işlemcilerde aynı mantıktır fakat formül işlemciye göre farklılık gösterebilir bunun için kaynaklardan bakılması gerekir.

- **Tconv = Sampling Time + Cycles / ADC Clock**
 - **Tconv**: Dönüşüm süresi (conversion time)
 - **Sampling Time**: ADC'nin sinyali örnekleme süresi (örnekleme çevrimleri ile belirtilir)
 - **Cycles**: ADC çözünürlüğüne bağlı olarak kullanılan ek çevrimler. 12-bit ADC için bu değer genellikle 12 çevrimdir.
 - **ADC Clock**: ADC'nin saat frekansı
- Eğer hızlı bir sinyali ölçmeniz gerekiyorsa, yüksek bir dönüşüm frekansı seçmelisiniz. Örneğin, ses sinyalleri veya hızlı değişen analog sinyaller için dönüşüm frekansı yüksek olmalıdır. Ancak çok yüksek frekanslar, özellikle gürültülü ortamlarda sinyal doğruluğunu azaltabilir, bu nedenle filtreleme ve örnekleme süresi dikkatle seçilmelidir.
- Daha yavaş sinyaller için (örneğin sıcaklık ölçümü gibi) düşük bir dönüşüm frekansı yeterli olabilir. Daha uzun örnekleme süresi kullanarak sinyali daha doğru bir şekilde ölçebilirsiniz. Ayrıca düşük dönüşüm frekansı güç tüketimini azaltabilir. Yüksek Çözünürlük, 12-bit çözünürlük gibi daha hassas ölçümler sağlar, ancak dönüşüm süresini artırır.
- Örnekleme süresi ne kadar düşükse, dönüşüm o kadar hızlı olur, ancak ölçüm hassasiyeti ve doğruluğu azalabilir.
- Genel olarak belirli bir sinyal frekansını doğru şekilde örneklemek için **Nyquist kriterine** uymak gerekir. Bu kriter gereği, örnekleme frekansı, ölçülen sinyalin en yüksek frekans bileşeninin en az iki katı olmalıdır.
 - **Yavaş değişen sinyaller** (örneğin sıcaklık sensörleri, basınç sensörleri, potansiyometreler), dönüşüm frekansı birkaç kHz olabilir (1-10 kHz aralığı genellikle yeterlidir).
 - **Orta hızlı sinyaller** (örneğin ses sinyalleri, motor hız sensörleri), dönüşüm frekansı genellikle 10 kHz - 100 kHz aralığında olmalıdır.
 - **Hızlı sinyaller** (örneğin RF sinyalleri, yüksek frekanslı analog sinyaller), dönüşüm frekansı 100 kHz ile MHz mertebesinde olabilir.

Ancak çoğu uygulama için 10 kHz - 1 MHz aralığı yeterlidir.

Çalışma Modları

- **Single Conversion Mode**, bir **tek** dönüşüm gerçekleştirildikten sonra ADC'nin otomatik olarak durmasını sağlar. Her dönüşüm, başlatma komutu ile başlatılır ve tamamlandığında ADC otomatik olarak durur.
- **Continuous Conversion Mode**, başlatıldığı andan itibaren **sürekli** olarak dönüşümler gerçekleştirir. Otomatik durma olmadığı için dönüşümler devam eder, kullanıcı tarafından durdurulana kadar devam eder.
- **Scan Mode**, belirli bir kanal listesini otomatik olarak **tarama** yeteneğine sahiptir. Tarama modu, birden fazla kanalı tek bir dönüşüm başlatma komutu ile sırayla ölçmeyi sağlar.
- **Discontinuous Mode**, kullanıcı belirli bir kanal listesinin **ardışık** olarak ölçülmesini sağlayabilir. Ancak, kanal arasında belirli bir gecikme bulunabilir.

Ölçüm Yöntemleri

- ADC ölçümlerini almak için kullanılan farklı yöntemler şunlardır: Polling, Interrupt ve DMA
- <http://www.elektrobot.net/stm32-adc-kullanimi-polling-interrupt-ve-dma/> ile <https://controllerstech.com/stm32-adc-single-channel/> linkten Polling, Interrupt ve DMA metodu kullanarak yapılan örnekleri inceleyebiliriz.
- **Polling** yöntemi, mikrodenetleyici ADC'nin çevrim süresince farklı bir işlem yapmaz ve çevrimin bitmesini bekler. Yapılacak ölçümün çok hızlı olmasının gerekmediği yada uzun zaman aralıklarında tek ölçüm yapılmasının yeterli olduğu durumlarda sıklıkla kullanılır.
- **Interrupt** yöntemi, ADC dönüşümü tamamlandığında bir kesme çağrısı gerçekleşir. Böylece mikrodenetleyicinin başka işlerle meşgulken dahi ADC verilerini işlemesine izin verir. Daha karmaşık uygulamalarda, dönüşüm tamamlandığında hemen yanıt verilmesi gereken durumlar için uygundur. Verimli kullanım, mikrodenetleyicinin diğer görevlere odaklanmasını sağlar.
- **DMA** yöntemi, ADC sonuçları doğrudan belleğe kopyalanır, bu da CPU'nun dahil olmadan çalışmasına olanak tanır. Büyük veri setlerini hızlı bir şekilde işlemek ve mikrodenetleyicinin CPU'sunu diğer görevlere odaklamak için uygundur. Bellek yönetimi konusunda dikkatlice ele alınması gerekebilir.
- DMA'nın Interrupt ile kullanımından en büyük farkı, ADC'nin çevrimi tamamladıktan sonra elde ettiği değeri hafıza bölgesine DMA tarafından yazılmasıdır. Böylece mikrodenetleyici hiç bir şekilde ADC işlemleri ile meşgul olmaz. Özellikle çok sayıda ölçümün ard arda ve hızlı yapılmasının istendiği durumlarda DMA kullanılır.

Birim Yapısı

- **ADC_SR** (Status Register), ADC durumunu izleyen bu register, dönüşüm tamamlandığında, taşma veya analog bekçi olaylarının gerçekleştiğini belirten bayrakları içerir.
- **ADC_CR1** (Control Register 1), dönüşüm kesmelerini etkinleştirme, scan modunu kontrol etme, discontinuous modu ve enjekte dönüşümleri yönetme gibi temel ADC kontrol ayarlarını içerir.
- **ADC_CR2** (Control Register 2), ADC'nin genel kontrolünü sağlayan bu register, ADC'nin etkinleştirilmesi, continuous conversion modu, DMA modu, kalibrasyon ve harici tetikleme seçenekleri gibi ayarları içerir.
- **ADC_SMPR1** ve **ADC_SMPR2** (Sampling Time Register 1 ve 2), örnekleme süresini belirleyen bu registerlar, her bir kanalın örnekleme süresini ayarlamayı sağlar.
- **ADC_DR** (Data Register), Dönüşüm sonuçlarını depolar; yani ADC tarafından ölçülen analog sinyalin dijital karşılığını içerir.

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0x00	ADC_CSR	Reserved										OVR	STRT	JSTRT	JEOC	EOC	AWD	Reserved			OVR	STRT	JSTRT	JEOC	EOC	AWD	Reserved			OVR	STRT	JSTRT	JEOC	EOC	AWD
	Reset value											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	ADC_CCR	Reserved										TSVREFE			VBATE			Reserved			ADC3			ADC2			Reserved			ADC1					
	Reset value											0	0	0	0	0	0				0	0	0	0	0	0				0	0	0	0	0	0
0x08	ADC_CDR	Regular DATA2[15:0]															Regular DATA1[15:0]																		
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

- ADC'deki **common registerlar**, birden fazla ADC modülünün ortak kullanıldığı durumlar için genel ayarları ve durumu izlemek için tasarlanmış registerlardır. Bu registerlar, birden fazla ADC'nin ortak özelliklerini kontrol etmek ve izlemek için kullanılır.
- **ADC_CSR** (Common Status Register), modülünün genel durumunu gösterir. Özellikle birden fazla ADC'nin kullanıldığı durumlarda ortak durumu izlemek için kullanılır.
- **ADC_CCR** (Common Control Register), ortak ayarları içerir. Örneğin, referans voltajlarını (VREF+ ve VREF-) belirlemek gibi genel ADC kontrol parametrelerini içerir.
- **ADC_CDR** (Common Data Register), birden fazla ADC kullanıldığında, çeşitli ADC'lerden gelen verileri depolar.