

Câu chuyện bắt đầu từ một cậu bé,
và một ý tưởng
có thể
làm thay đổi thế giới...

PAY IT FORWARD

Đó là khi bạn giúp đỡ 3 người bạn không quen biết,
đủ là bằng thời gian,
hay công sức,
hay kinh nghiệm,
hay kiến thức,
hay tiền bạc, ...
của mình.



Mà không chờ đợi một sự báo ân nào.

Chỉ cần mỗi người trong 3 người đó,
lại đem những gì mình có, mà người khác cần,
tiếp tục giúp đỡ thêm 3 người nữa.

Chính những người-giúp-đỡ, và người-được-giúp-đỡ,
sẽ là những người góp phần thay đổi thế giới...

Một thế giới sẽ chia kiến thức - và yêu thương ...

PAY IT FORWARD ...

Chúng tôi không sáng tạo ra câu nói này.

Pay it forward...

Hãy tri ân người giúp mình bằng cách giúp đỡ người khác
Cho đi không phải để nhận lại.

PAY IT FORWARD



Analog Digital Converter



22/12/2018

 payitforward.edu.vn

1

- Giới thiệu ADC

2

- Module ADC trong MSP430G2553

3

- Thực hành

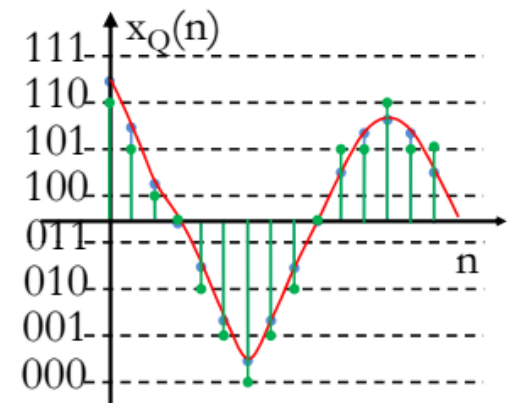
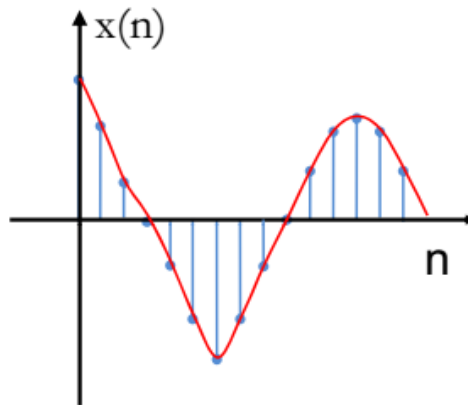
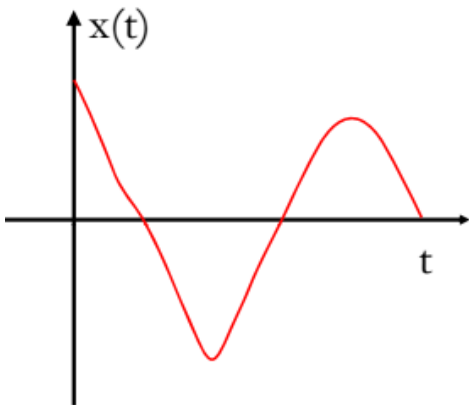
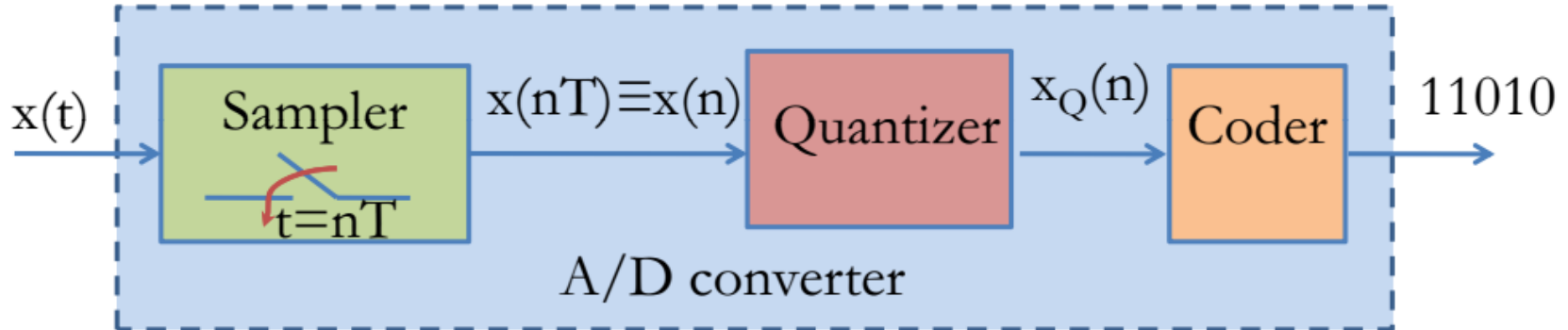
PAY IT FORWARD



1

Giới thiệu ADC

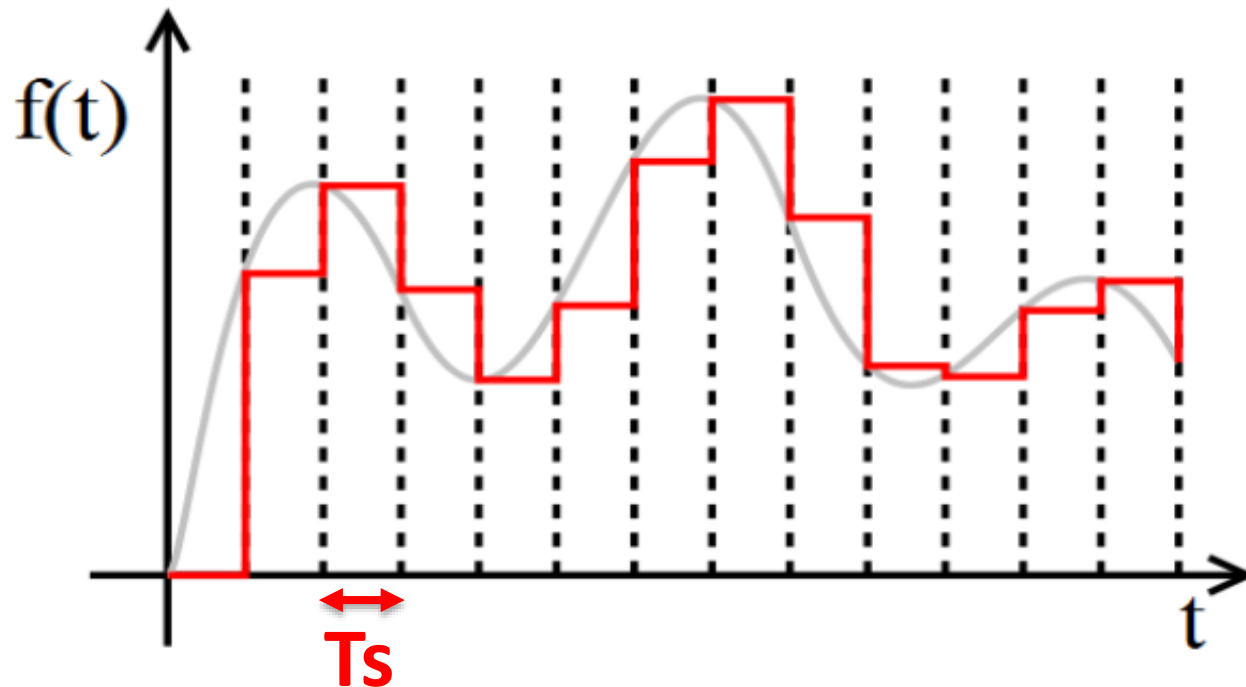
- ADC là bộ chuyển đổi tín hiệu **tương tự** sang tín hiệu **số**.



1

Giới thiệu ADC

- Các thông số cơ bản:
 - Sample Time:

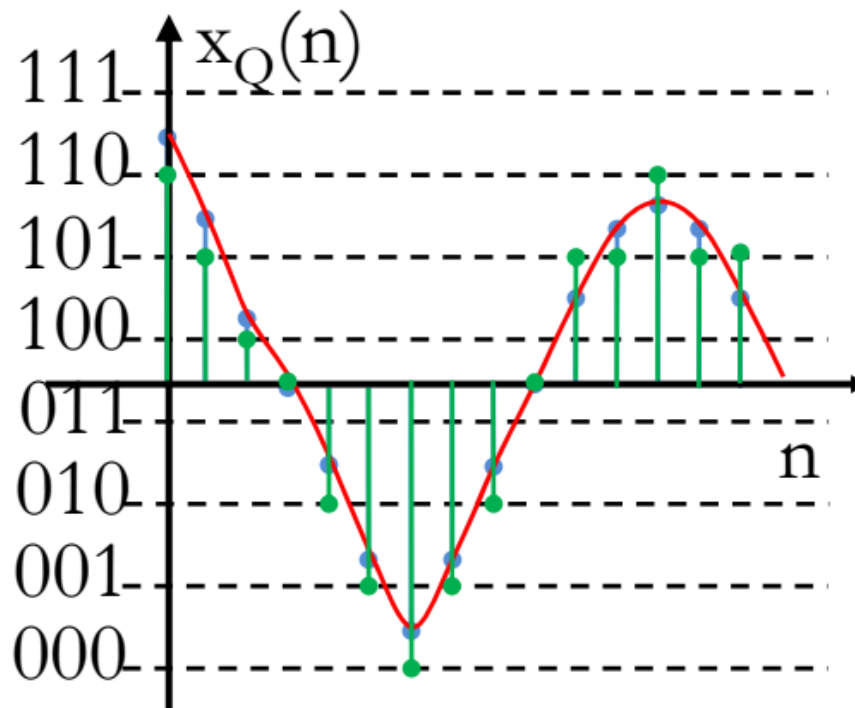


1

Giới thiệu ADC

■ Các thông số cơ bản:

- **Resolution:** độ phân giải n bits thì ADC có thể biểu diễn được 2^n giá trị.



1

Giới thiệu ADC

- **Các thông số cơ bản:**

- **Vref:**

- Là giá trị điện áp tham chiếu cho module ADC.
 - Nếu $V_{in} \geq V_{ref}$ thì giá trị chuyển đổi bằng $2^n - 1$.
 - Vref cần phải là nguồn áp có chất lượng tốt và ổn định.

2

Module ADC trong MSP430G2553

- MSP430G2553 có 2 module ADC là **ADC 10bits** và **ADC 12bits**.
- Giới thiệu ADC 10bits:
 - Có 8 kênh chuyển đổi ngoài và 4 kênh liên kết nội.
 - Vref có thể được cấp nguồn ngoài hoặc nguồn nội (2.5V/1.5V).
 - Kết quả lưu vào thanh ghi **ADC10MEM**.
 - Phát ra ngắt khi chuyển đổi hoàn tất (**ADC10IFG**).

2

Module ADC trong MSP430G2553

- Giá trị ADC đọc được.

$$N_{\text{ADC}} = 1023 \times \frac{V_{\text{in}} - V_{\text{R-}}}{V_{\text{R+}} - V_{\text{R-}}}$$

2

Module ADC trong MSP430G2553

- Khi sử dụng điện áp tham chiếu là VCC.

ADC10CTL0 =

SREF_0 +  **Vref = VCC**

ADC10SHT_1 +  **ADC sample and hold
time = 8 ADC clocks**

ADC10ON +  **Turn on ADC10**

ADC10IE;  **Enable ADC10 Interrupt**

2

Module ADC trong MSP430G2553

- Khi sử dụng điện áp tham chiếu nội.

ADC10CTL0 =

SREF_0 +  **Vref = VCC**

ADC10SHT_1 +  **ADC sample and hold time = 8 ADC clocks**

ADC10ON +  **Turn on ADC10**

ADC10IE  **Enable ADC10 Interrupt**

REFON;  **Turn on reference generator → 1.5V**

2

Module ADC trong MSP430G2553

- Chọn kênh chuyển đổi và config ADC10Clock.

ADC10CTL1 =

INCH_1+  **Select Input chanel 1**

ADC10DIV_1+  **ADC10 Clock divider: 2**

ADC10SSEL_3;  **ADC10 clock source
select: SMCLK**

2

Module ADC trong MSP430G2553

- Enable ngõ vào analog.

ADC10AE0 |=

BIT1;



**Enable analog input on
A1 channel**

2

Module ADC trong MSP430G2553

- Bắt đầu quá trình chuyển đổi.

ADC10CTL0 |=

ENC +  **ADC10 Enable**

ADC10SC;  **Start sample - conversion**

3

Thực hành

- Đọc giá trị biến trở:

```
51 #include <msp430g2553.h>
52 #include <stdio.h>
53 #include <stdint.h>
54 #include "lib/UART.h"
55 #include "lib/Basic config.h"

76 uint16_t adc_result;
77 uint16_t volt;
78 uint8_t UART_Buf[25];
```

3

Thực hành

▪ Đọc giá trị biến trở:

```

57 void ADC10_Config(void)
58 {
59     ADC10CTL0 = SREF_0 + ADC10SHT_1 + ADC10ON + ADC10IE;
60     /*
61      * Vref = VCC
62      * ADC sample and hold time = 8 ADC clocks
63      * Turn on ADC10
64      * Enable ADC10 Interrupt
65      */
66     ADC10CTL1 = INCH_0 + ADC10DIV_1 + ADC10SSEL_3;
67     /*
68      * Select Input channel 0
69      * ADC10DF = 0: The 10-bit conversion results are right justified
70      * ADC10 Clock divider: 2
71      * ADC10 clock source select: SMCLK
72      */
73     ADC10AE0 |= BIT0;    //Enable analog input on A0 channel
74 }

```


3

Thực hành

- Đọc giá trị biến trở:

```
33 int main(void) {
34     WDTCTL = WDTPW | WDTHOLD;    // Stop watchdog timer
35
36     P2SEL = 0;                    //Turn off external crystal
37     P2SEL2 = 0;
38
39     Config_Clocks();
40     uart_init();
41     ADC10_Init();
42
43     while(1)
44     {
45         ADC10CTL0 |= ENC + ADC10SC;
46         __bis_SR_register(GIE);
47
48         volt = adc_result * (3300 / 1023);
49         sprintf((char*) UART_Buf, "Volt: %d mV\n", volt);
50
51 //         sprintf((char*) UART_Buf, "ADC: %d\n", adc_result);
52         uart_puts((char*) UART_Buf);
53         __delay_cycles(500000);
54     }
55 }
```

3

Thực hành

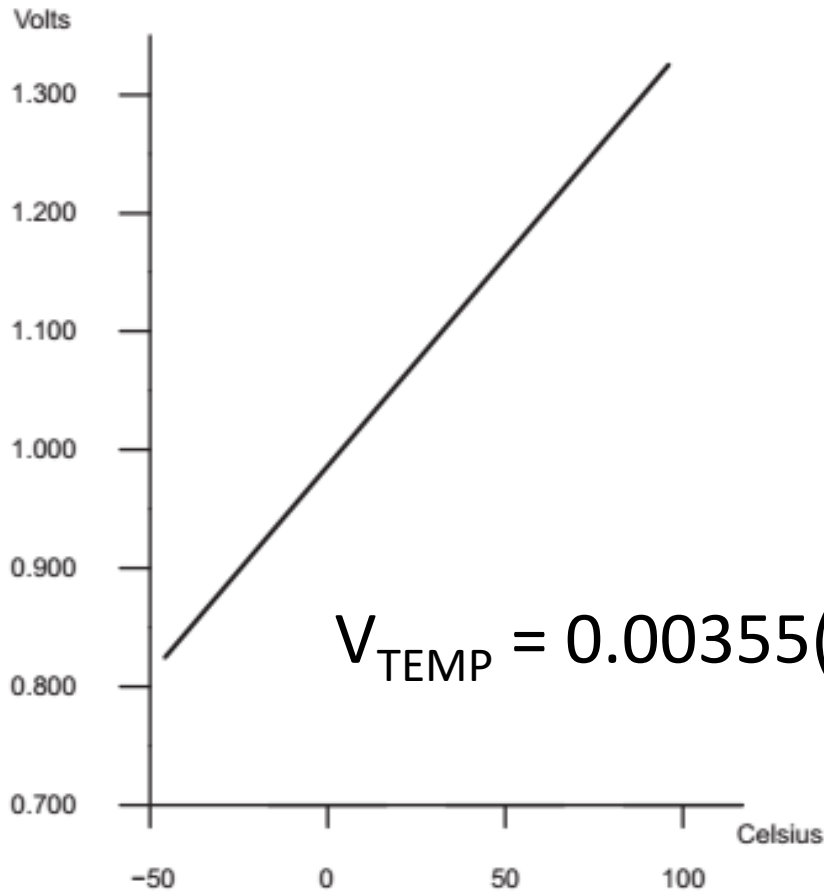
- Đọc giá trị biến trở:

```
57 #pragma vector = ADC10_VECTOR
58 __interrupt void ADC10_Interrupt(void)
59 {
60     adc_result = ADC10MEM;
61 }
```

3

Thực hành

- Đọc giá trị cảm biến nhiệt độ nội:



$$V_{\text{TEMP}} = 0.00355(\text{TEMP}_c) + 0.986$$

3

Thực hành

- Đọc giá trị cảm biến nhiệt độ nội:

```
14 void ADC10_Config(void)
15 {
16     ADC10CTL0 = SREF_1 + ADC10SHT_3 + ADC10ON + ADC10IE + REFON;
17     /*
18      * Vr+ = Vref+
19      * ADC sample and hold time = 64 ADC clocks: When using the temperature
20      * sensor, the sample period must be greater than 30 µs
21      * Turn on ADC10
22      * Enable ADC10 Interrupt
23      * Turn on reference generator
24      */
25     ADC10CTL1 = INCH_10 + ADC10DIV_1 + ADC10SSEL_3 ;
26     /*
27      * Select Input channel: Temperature Sensor
28      * ADC10DF = 0: The 10-bit conversion results are right justified
29      * ADC10 Clock divider: 4
30      * ADC10 clock source select: SMCLK
31      */
32 }
```

3

Thực hành

- Đọc giá trị cảm biến nhiệt độ nội:

```
38 void main(void)
39 {
40     Config_stop_WDT();
41     P2SEL = 0;           //Turn off external crystal
42     P2SEL2 = 0;
43
44     Config_Clocks();
45     uart_init();
46     ADC10_Config();
47
48     while (1)
49     {
50         ADC10CTL0 |= ENC + ADC10SC;           //Enable ADC10, Start sample - conversion
51         _bis_SR_register(GIE);                 //Enter LPM0, wait for sample-conversion finish
52         //Ref in http://forum.43oh.com/topic/1954-using-the-internal-temperature-sensor/
53         temp = ((adc_result - 673) * 423) / 1024;
54         sprintf((char*)UART_Buf, "Temp: %d\n", temp);
55         uart_puts((char*)UART_Buf);
56         _delay_cycles(500000);
57     }
58 }
```

PAY IT FORWARD



payitforward.edu.vn