A/D and D/A converters

R. Ferrero , P. Bernardi

Politecnico di Torino

Dipartimento di Automatica e Informatica (DAUIN)

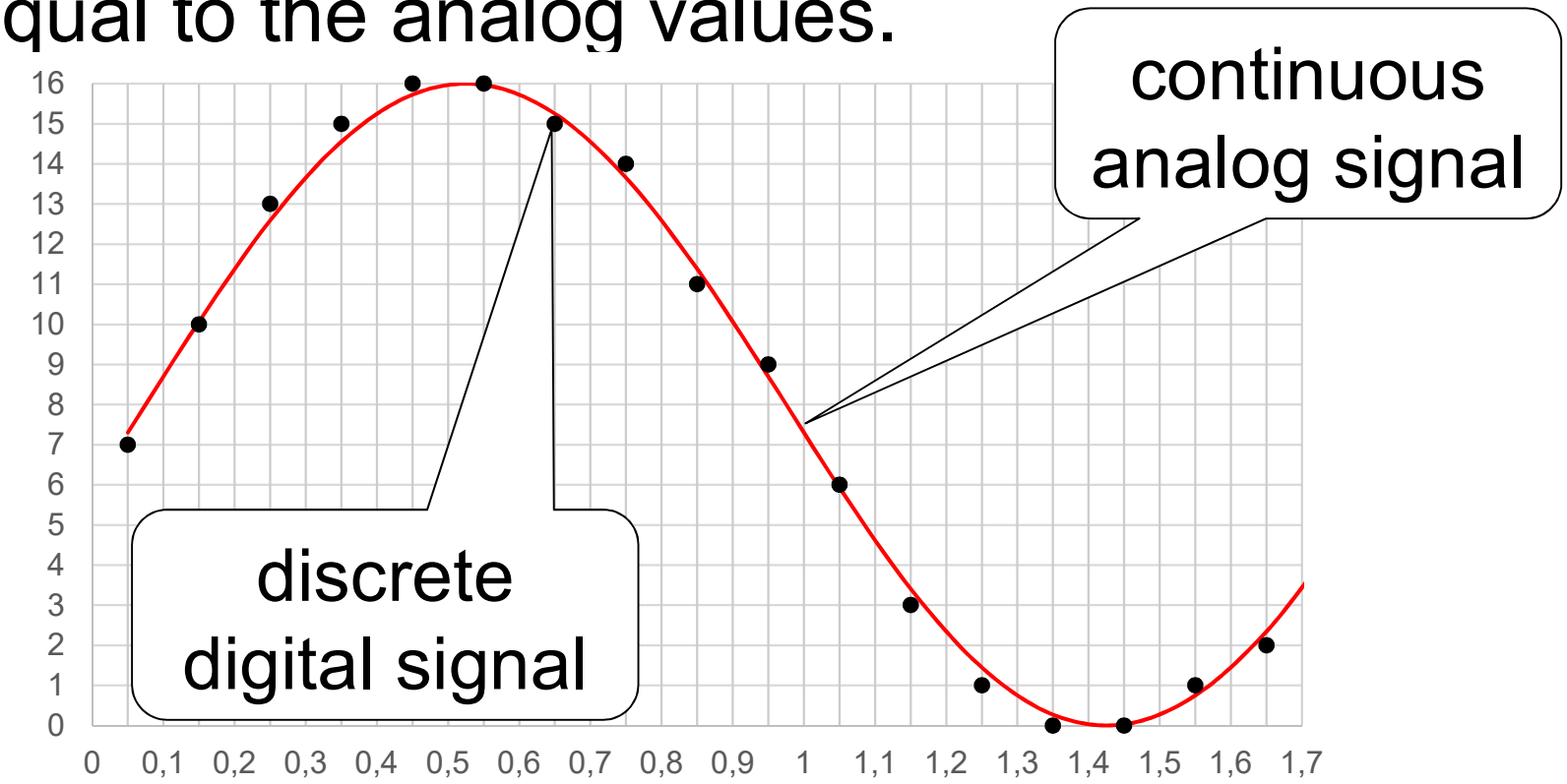
Torino - Italy

Analog and Digital conversion

- Computers interact with real world
  - analog signals in input (e.g.: temperature measured by a sensor)
  - analog signals in output (e.g.: playing a song).
- Processed values are always digital.
- An analog-to-digital converter (ADC) receives a signal with an infinite amount of values and convert it into a binary representation.
- A digital-to-analog converter (DAC) turns a binary code into an analog voltage.

Digitization of analog signals

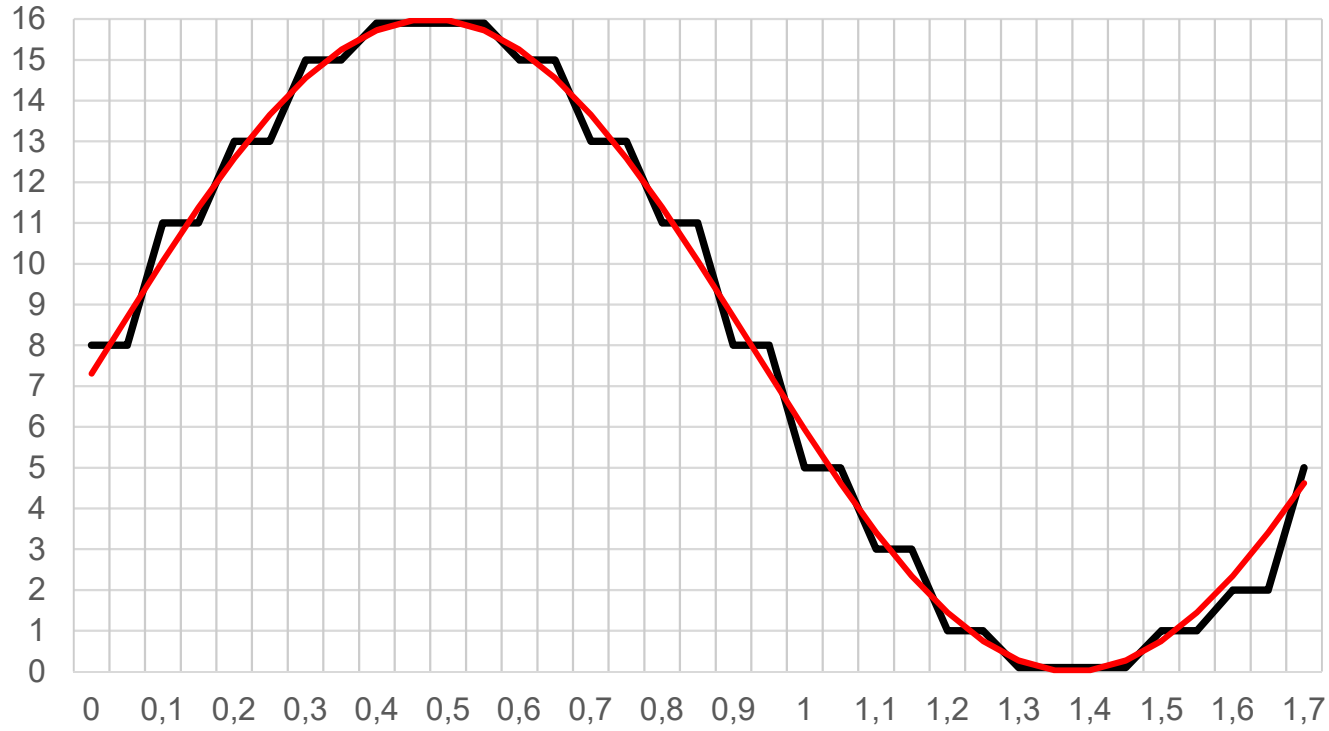
1. Sampling: few points are chosen on the analog signal.
2. Rounding: digital values are approximately equal to the analog values.



Digitization of analog signals

3. Digital values are joined to round off the analog signal to near stabilized values.

- Such a process is called as Time and Amplitude Quantization.

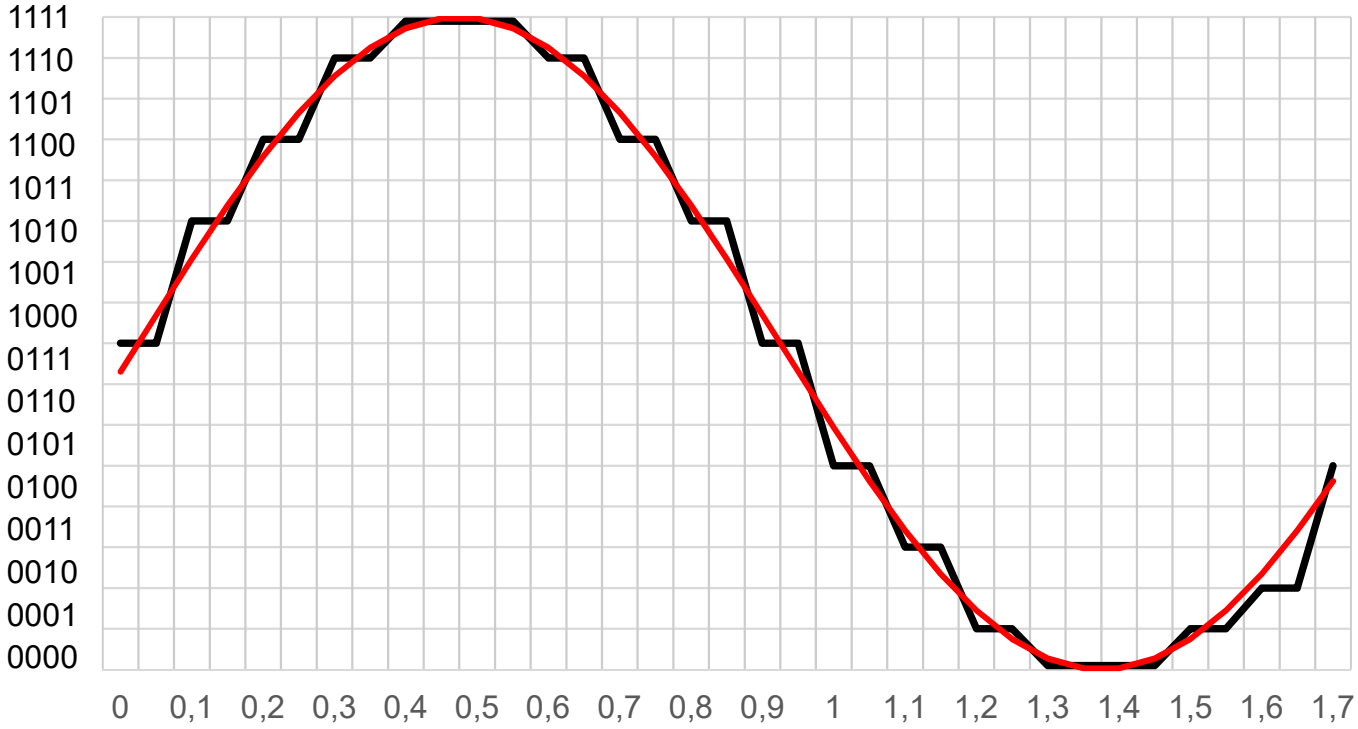


Time quantization

- The analog signal is sampled at discrete points in time.
- The time between two discrete points is referred to as the sampling interval.
- The minimum sample interval is the inverse of the maximum conversion rate of the ADC; it depends on the specific ADC circuit.

Amplitude quantization

- The analog signal is discretized with a fixed number of quantization levels.
- ADC resolution is specified in bits; it indicates the number of distinct output codes (2^n).



LPC1768 Analog-to-Digital Converter

- 12-bit successive approximation ADC
- Input multiplexing among 8 pins
- Power-down mode
- Measurement range VREFN to VREFP (3 V)
- 12-bit conversion rate of 200 kHz
- Some advanced usage modes
  - Burst conversion for single or multiple inputs.
  - Optional conversion on transition on input pin or Timer Match signal.

ADC registers (page 587)

Table 531. ADC registers

Generic Name	Description	Access	Reset value[1]	AD0 Name & Address
ADCR	A/D Control Register. The ADCR register must be written to select the operating mode before A/D conversion can occur.	R/W	1	AD0CR - 0x4003 4000
ADGDR	A/D Global Data Register. This register contains the ADC's DONE bit and the result of the most recent A/D conversion.	R/W	NA	AD0GDR - 0x4003 4004
ADINTEN	A/D Interrupt Enable Register. This register contains enable bits that allow the DONE flag of each A/D channel to be included or excluded from contributing to the generation of an A/D interrupt.	R/W	0x100	AD0INTEN - 0x4003 400C
ADDR0	A/D Channel 0 Data Register. This register contains the result of the most recent conversion completed on channel 0.	RO	NA	AD0DR0 - 0x4003 4010
ADDR1	A/D Channel 1 Data Register. This register contains the result of the most recent conversion completed on channel 1.	RO	NA	AD0DR1 - 0x4003 4014
ADDR2	A/D Channel 2 Data Register. This register contains the result of the most recent conversion completed on channel 2.	RO	NA	AD0DR2 - 0x4003 4018
ADDR3	A/D Channel 3 Data Register. This register contains the result of the most recent conversion completed on channel 3.	RO	NA	AD0DR3 - 0x4003 401C
ADDR4	A/D Channel 4 Data Register. This register contains the result of the most recent conversion completed on channel 4.	RO	NA	AD0DR4 - 0x4003 4020
ADDR5	A/D Channel 5 Data Register. This register contains the result of the most recent conversion completed on channel 5.	RO	NA	AD0DR5 - 0x4003 4024
ADDR6	A/D Channel 6 Data Register. This register contains the result of the most recent conversion completed on channel 6.	RO	NA	AD0DR6 - 0x4003 4028
ADDR7	A/D Channel 7 Data Register. This register contains the result of the most recent conversion completed on channel 7.	RO	NA	AD0DR7 - 0x4003 402C
ADSTAT	A/D Status Register. This register contains DONE and OVERRUN flags for all of the A/D channels, as well as the A/D interrupt/DMA flag.	RO	0	AD0STAT - 0x4003 4030
ADTRM	ADC trim register.	R/W	0x0000 0F00	AD0TRM - 0x4003 4034

A/D Control Register (page 588)

Table 532: A/D Control Register (AD0CR - address 0x4003 4000) bit description

Bit	Symbol	Value	Description	Reset value
7:0	SEL		Selects which of the AD0.7:0 pins is (are) to be sampled and converted. For AD0, bit 0 selects Pin AD0.0, and bit 7 selects pin AD0.7. In software-controlled mode, only one of these bits should be 1. In hardware scan mode, any value containing 1 to 8 ones is allowed. All zeroes is equivalent to 0x01.	0x01
15:8	CLKDIV		The APB clock (PCLK_ADC0) is divided by (this value plus one) to produce the clock for the A/D converter, which should be less than or equal to 13 MHz. Typically, software should program the smallest value in this field that yields a clock of 13 MHz or slightly less, but in certain cases (such as a high-impedance analog source) a slower clock may be desirable.	0
16	BURST	1	The AD converter does repeated conversions at up to 200 kHz, scanning (if necessary) through the pins selected by bits set to ones in the SEL field. The first conversion after the start corresponds to the least-significant 1 in the SEL field, then higher numbered 1-bits (pins) if applicable. Repeated conversions can be terminated by clearing this bit, but the conversion that's in progress when this bit is cleared will be completed. **Remark:** START bits must be 000 when BURST = 1 or conversions will not start. If BURST is set to 1, the ADGINTEN bit in the AD0INTEN register (Table 534) must be set to 0.	0
		0	Conversions are software controlled and require 65 clocks.	
20:17	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
21	PDN	1	The A/D converter is operational.	0
		0	The A/D converter is in power-down mode.	
23:22	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
26:24	START		When the BURST bit is 0, these bits control whether and when an A/D conversion is started:	0
		000	No start (this value should be used when clearing PDN to 0).	
		001	Start conversion now.	

A/D (Global) Data Registers

- 8 A/D Data Registers: each one holds the result of the last conversion on the corresponding channel.
- 1 A/D Global Data Register: it holds the result of the most recent A/D conversion that has completed (regardless of the channel).
- All registers have also a DONE flag: this bit is set to 1 when a conversion completes; it is cleared when the register is read.

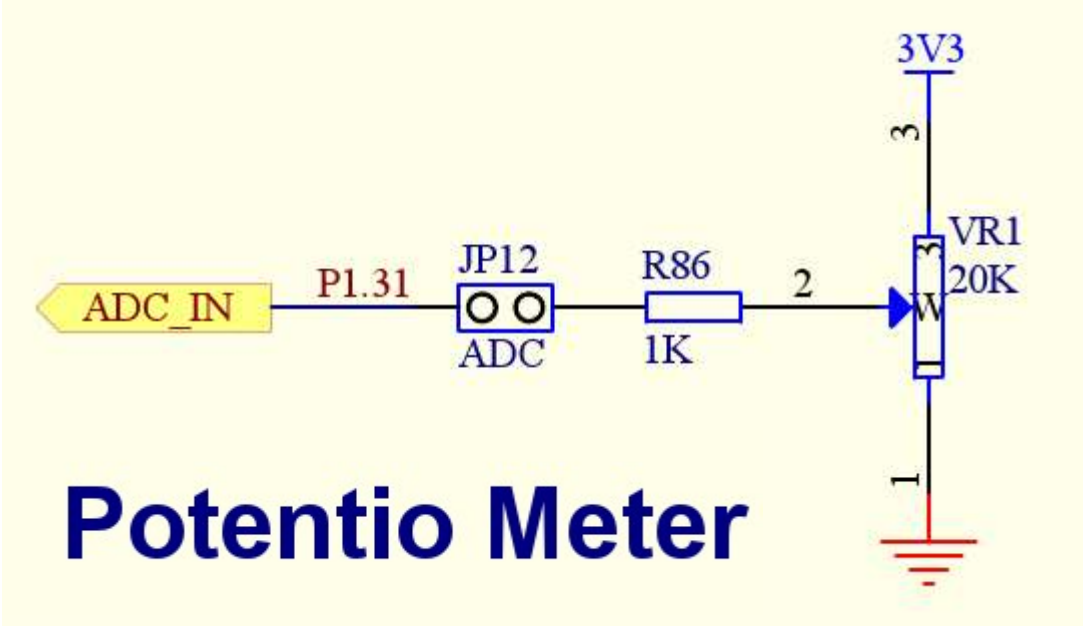
A/D Interrupt Enable register (p. 589)

Table 534: A/D Interrupt Enable register (AD0INTEN - address 0x4003 400C) bit description

Bit	Symbol	Value	Description	Reset value
0	ADINTEN0	0	Completion of a conversion on ADC channel 0 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 0 will generate an interrupt.	
1	ADINTEN1	0	Completion of a conversion on ADC channel 1 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 1 will generate an interrupt.	
2	ADINTEN2	0	Completion of a conversion on ADC channel 2 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 2 will generate an interrupt.	
3	ADINTEN3	0	Completion of a conversion on ADC channel 3 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 3 will generate an interrupt.	
4	ADINTEN4	0	Completion of a conversion on ADC channel 4 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 4 will generate an interrupt.	
5	ADINTEN5	0	Completion of a conversion on ADC channel 5 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 5 will generate an interrupt.	
6	ADINTEN6	0	Completion of a conversion on ADC channel 6 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 6 will generate an interrupt.	
7	ADINTEN7	0	Completion of a conversion on ADC channel 7 will not generate an interrupt.	0
		1	Completion of a conversion on ADC channel 7 will generate an interrupt.	
8	ADGINTEN	0	Only the individual ADC channels enabled by ADINTEN7:0 will generate interrupts. **Remark:** This bit must be set to 0 in burst mode (BURST = 1 in the AD0CR register).	1
		1	Only the global DONE flag in ADDR is enabled to generate an interrupt.	
31:17	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

ADC connection on board

- Adjustable potentiometer VR1 is connected to analog channel P1.31 (AD0.5).
- JP12 jumper enables the potentiometer input.
- VR1 setting provides input voltages between 0V and 3V to the ADC.



init() and start_conversion()

```
 7  void ADC_init (void)
 8  {
 9    LPC_PINCON->PINSEL3 |=  (3UL<<30);     /* P1.31 is AD0.5                    */
10
11    LPC_SC->PCONP       |=  (1<<12);       /* Enable power to ADC block         */
12
13    LPC_ADC->ADCR        =  (1<< 5) |      /* select AD0.5 pin                  */
14                            (4<< 8) |      /* ADC clock is 25MHz/5              */
15                            (1<<21);       /* enable ADC                        */
16
17    /* Use either one of these instructions to enable interrupt */
18    // LPC_ADC->ADINTEN = (1 << 8); /* global enable interrupt  */
19    LPC_ADC->ADINTEN = (1 << 5);    /* interrupt generated after completing a conversion on channel 5 */
20
21    NVIC_EnableIRQ(ADC_IRQn);              /* enable ADC Interrupt              */
22  }
23
24  void ADC_start_conversion (void) {
25    /* Use either one of these instructions */
26    //LPC_ADC->ADCR |=  (1<<24);           /* Start one A/D Conversion       */
27    LPC_ADC->ADCR |=  (1<<16);             /* Start burst A/D Conversion     */
28  }
```

ADC_IRQHandler()

```
37  void ADC_IRQHandler(void) {
38    unsigned short led_last, led_current;
39    /* Use either one of these instructions to read conversion result */
40    // AD_current = ((LPC_ADC->ADGDR>>4) & 0xFFF);  /* if global interrupts are enabled */
41    AD_current = ((LPC_ADC->ADDR5 >> 4) & 0xFFF); /* if interrupts on channel 5 are enabled */
42
43    led_last = AD_last * 7/ 0xFFF;        // ad_last : AD_max = x : 7
44    led_current = AD_current * 7/ 0xFFF;  // ad_current : AD_max = x : 7
45    if(led_current != led_last){
46      LED_Off(led_last);
47      LED_On(led_current);
48      disable_timer(0);
49      reset_timer(0);
50      init_timer(0,freqs[led_current]);
51      enable_timer(0);
52
53      AD_last = AD_current;
54    }
55
56  }
```

LPC1768 Digital-to-Analog Converter

- 10-bit DAC
- Precision = 1024 levels between 0 and 3.3V
- Maximum update rate of 1 MHz.

D/A Converter Register (page 594)

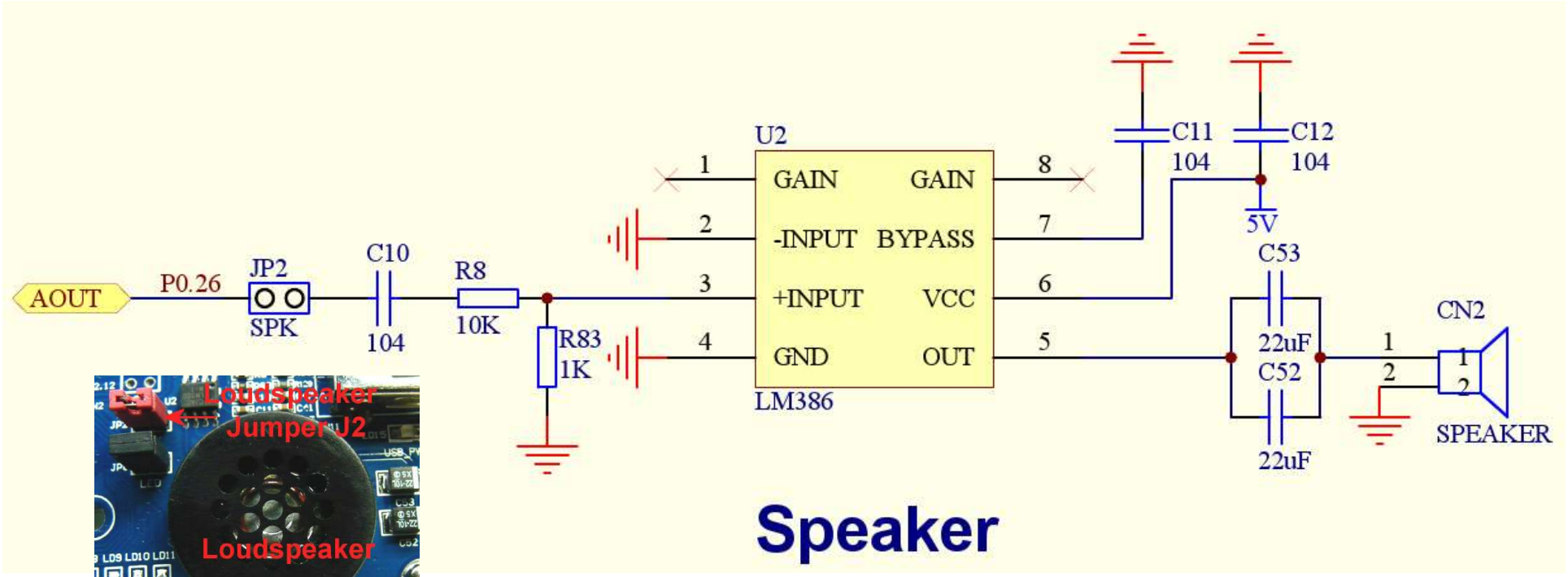
- It contains the 10-bit digital value to be converted to analog.

Table 540: D/A Converter Register (DACR - address 0x4008 C000) bit description

Bit	Symbol	Value	Description	Reset Value
5:0	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:6	VALUE		After the selected settling time after this field is written with a new VALUE, the voltage on the AOUT pin (with respect to V_{SSA}) is VALUE × ((V_{REFP} - V_{REFN})/1024) + V_{REFN}.	0
16	BIAS[1]	0	The settling time of the DAC is 1 μs max, and the maximum current is 700 μA. This allows a maximum update rate of 1 MHz.	0
		1	The settling time of the DAC is 2.5 μs and the maximum current is 350 μA. This allows a maximum update rate of 400 kHz.	
31:17	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

DAC connection on board

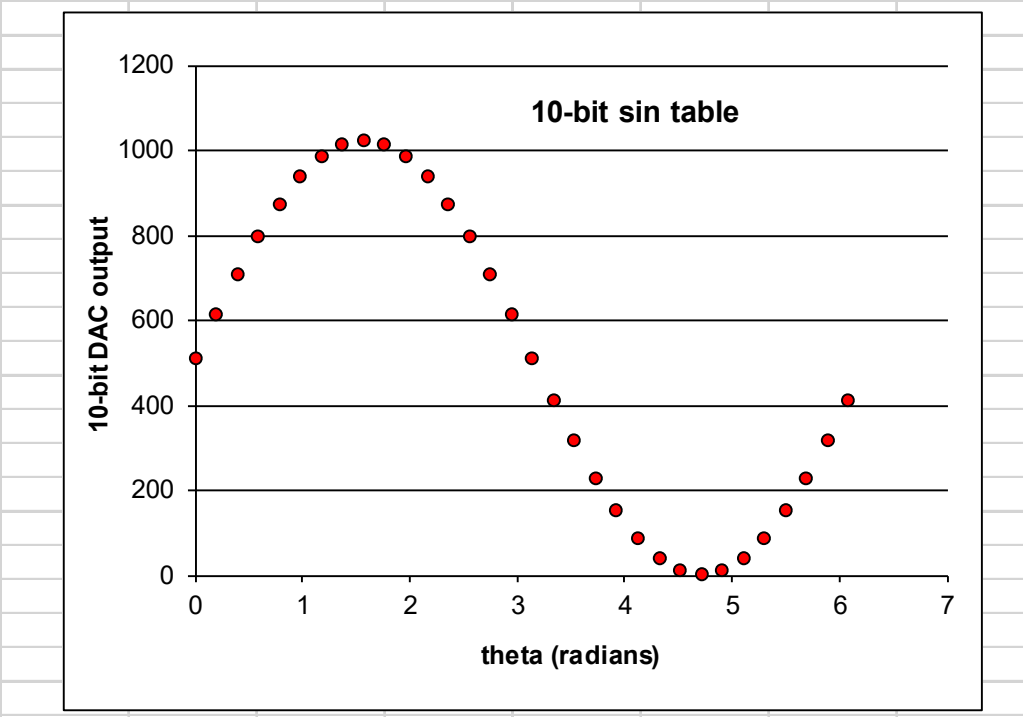
- External speaker is connected to DAC output pin P0.26.
- The DAC output is enabled by JP2 jumper.z



10-bit Sinusoid Table

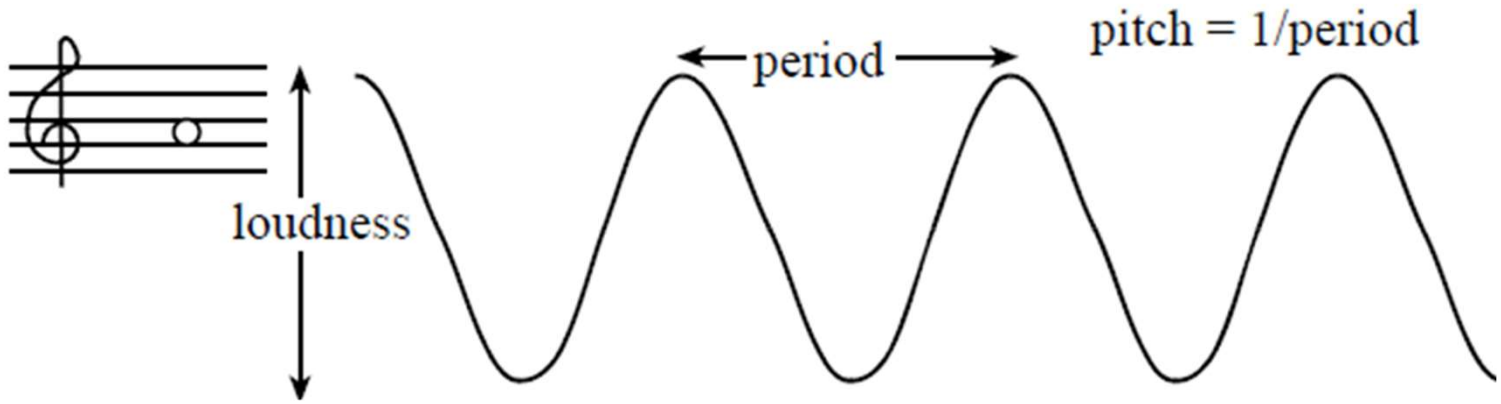
- Sound can be obtained by repeatedly feeding the loudspeaker with a sampled sinusoid.

```
uint16_t SinTable[45] =
{ 410, 467, 523, 576,
627, 673, 714, 749, 778,
799, 813, 819, 817, 807,
789, 764, 732, 694, 650,
602, 550, 495, 438, 381,
324, 270, 217, 169, 125,
87, 55, 30, 12, 2, 0, 6,
20, 41, 70, 105, 146,
193, 243, 297, 353};
```



Loudness and pitch

- They are controlled by amplitude and frequency, respectively.



- Humans can hear from about 25 to 20,000 Hz.
- A musical tone (note) is produced by a sinusoid of a particular frequency.

Notes

- Middle A is 440 Hz.
- The sinusoid is completely reproduced in 2.27 ms.

Note	f	T (ms)
C	523	1.91
B	494	2.02
B^b	466	2.15
A	*440*	*2.27*
A^b	415	2.41
G	392	2.55
G^b	370	2.70
F	349	2.87
E	330	3.03
E^b	311	3.22
D	294	3.40
D^b	277	3.61
C	262	3.82

Synthesizing digital music

- A musical tone is a sinusoid of a particular frequency.
- The samples are fixed point numbers.
- A resistor network receives appropriate digital values and it effectively produces a pseudo-analog signal.
- A programmable timer regulates the music frequency, by determining when the next value has to be put in output.

Use of timer handler

```
11  uint16_t SinTable[45] =
12  {
13      410, 467, 523, 576, 627, 673, 714, 749, 778,
14      799, 813, 819, 817, 807, 789, 764, 732, 694,
15      650, 602, 550, 495, 438, 381, 324, 270, 217,
16      169, 125, 87 , 55 , 30 , 12 , 2  , 0  , 6  ,
17      20 , 41 , 70 , 105, 146, 193, 243, 297, 353
18  };
19
20  void TIMER0_IRQHandler (void)
21  {
22    static int ticks = 0;
23    /* DAC management */
24    LPC_DAC->DACR = (SinTable[ticks]) << 6;
25    ticks ++;
26    if (ticks == 45)
27      ticks=0;
28
29    LPC_TIM0->IR = 1;     /* clear interrupt flag */
30    return;
31  }
32
```