Analog input and output

P.Bernardi

Analog and Digital conversion

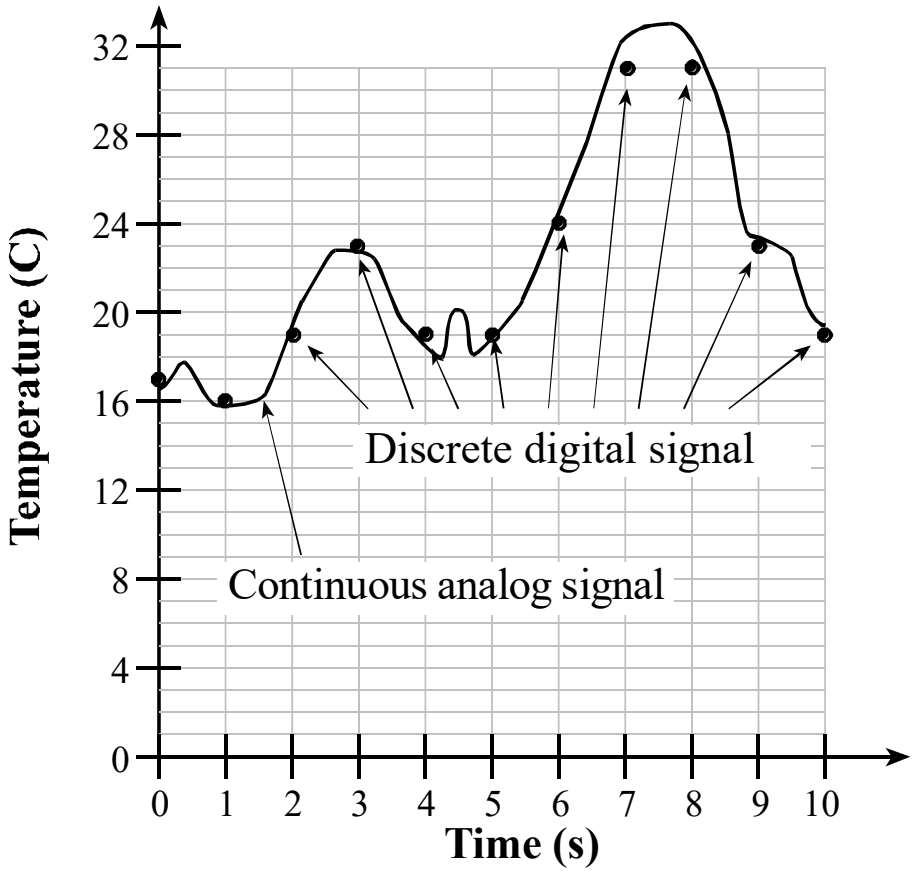
- Imagine the engineer tasked with controlling a Heating, Ventilation and Air Conditioning (HVAC) unit.
- If planning to use any kind of microcontroller or microprocessor, it will be necessary to be able to read an analog temperature
  - which has an infinite number of values and
  - convert that into a binary representation laid out into discrete steps
- When dealing with an analog value that needs to be processed by a digital system, an analog-to-digital converter (**ADC**) will be imperative.

Analog and Digital conversion (II)

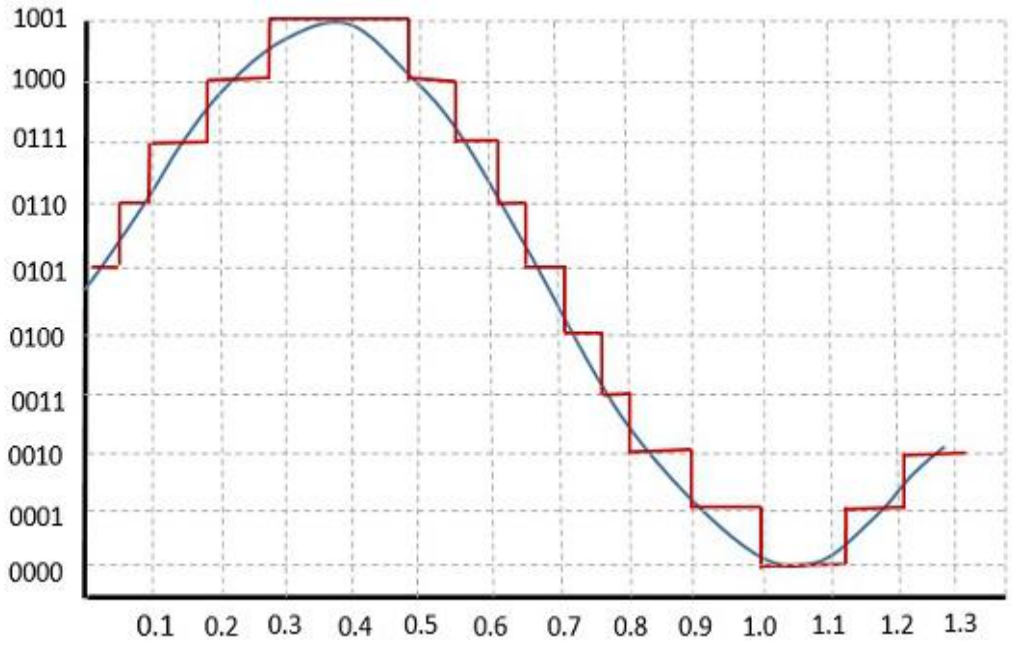
- The same theory can be applied backwards to a digital signal that needs to be converted into an analog signal.
- Streaming a song online involves a few different steps that use the conversion of digital signals to analog ones.
- The signal that the host device receives from the server will be a binary representation of the original analog signal.
- The original signal was analog so the final representation of that will also need to be analog.
- This objective is obtained by the use of the digital-to-analog converter (**DAC**).
- This type of device takes a binary code that could have been encoded by an analog to digital converter and turns it back into an analog voltage.

Digital Representation of Analog Signals

- The digitization of analog signals involves quantizing the sampled values to the nearest discrete level.
- The method of sampling chooses a few points on the analog signal and then these points are joined to round off the value to a near stabilized value.
- Such a process is called as Time and Amplitude Quantization.

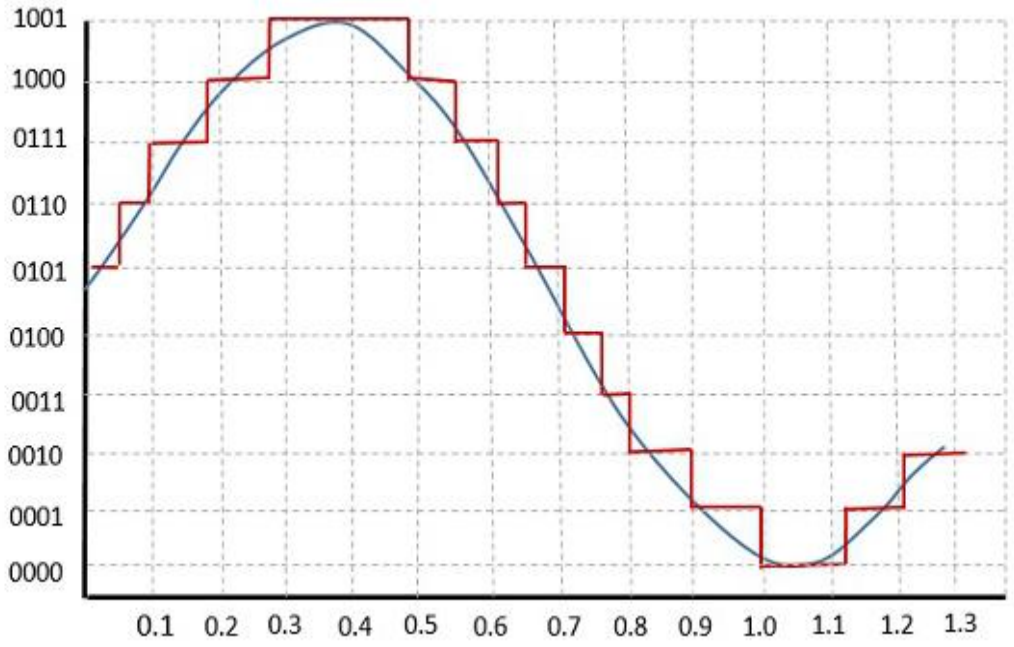


ADC conversion rate



- Time Quantization is obtained by sampling the analogue signal at discrete points in time
- These points are usually evenly spaced in time, with the time between being usually referred to as the sampling interval
- The minimum sample interval or maximum conversion rate depends on the specific ADC circuit.

ADC resolution



- The amplitude quantization of an analog signal is done by discretizing the signal with a number of quantization levels
- The spacing between the two adjacent representation levels is called a quantum or step-size
- The resolution of an A/D converter (ADC) is specified in bits and determines how many distinct output codes (2^n) the converter can produce.
  - For example, an ADC with a resolution of 4 bits can encode an analog input to one in 16 different levels ($2^4 = 16$)
  - This means that the ADC assumes 3.3V is 15, and anything less than 3.3V will be a ratio between 3.3V and 0
  - Quantum = 3.3V/15 = 0.22V.

LPC1768 12-bit Analog to Digital Converter

- A 12-bit ADC is on-chip in the LPC1768 device
  - 12-bit successive approximation analog to digital converter
  - 8 multiplexed Input channels
  - Power-down mode
  - Measurement range **VREFN** (Ground voltage) to **VREFP** (typically 3.3 V)
  - 12-bit conversion rate of 200 kHz (T= 5 µs)
  - Some advanced usage mode like
    - Burst conversion mode for single or multiple inputs
    - Optional conversion on transition on input pin or Timer Match signal.

Additional details

- Input clock frequency: Maximum 13MHz
- A conversion requires 65 clock cycles

ADC registers

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

Control Register
Select operation mode functionalities

Holds the result of the most recent A/D conversion and a status flags

Allows control over which A/D channels generate an interrupt when a conversion is complete

The Data Registers hold the result of the last conversion for each A/D channel

The A/D Status register allows checking the status of all A/D channels.

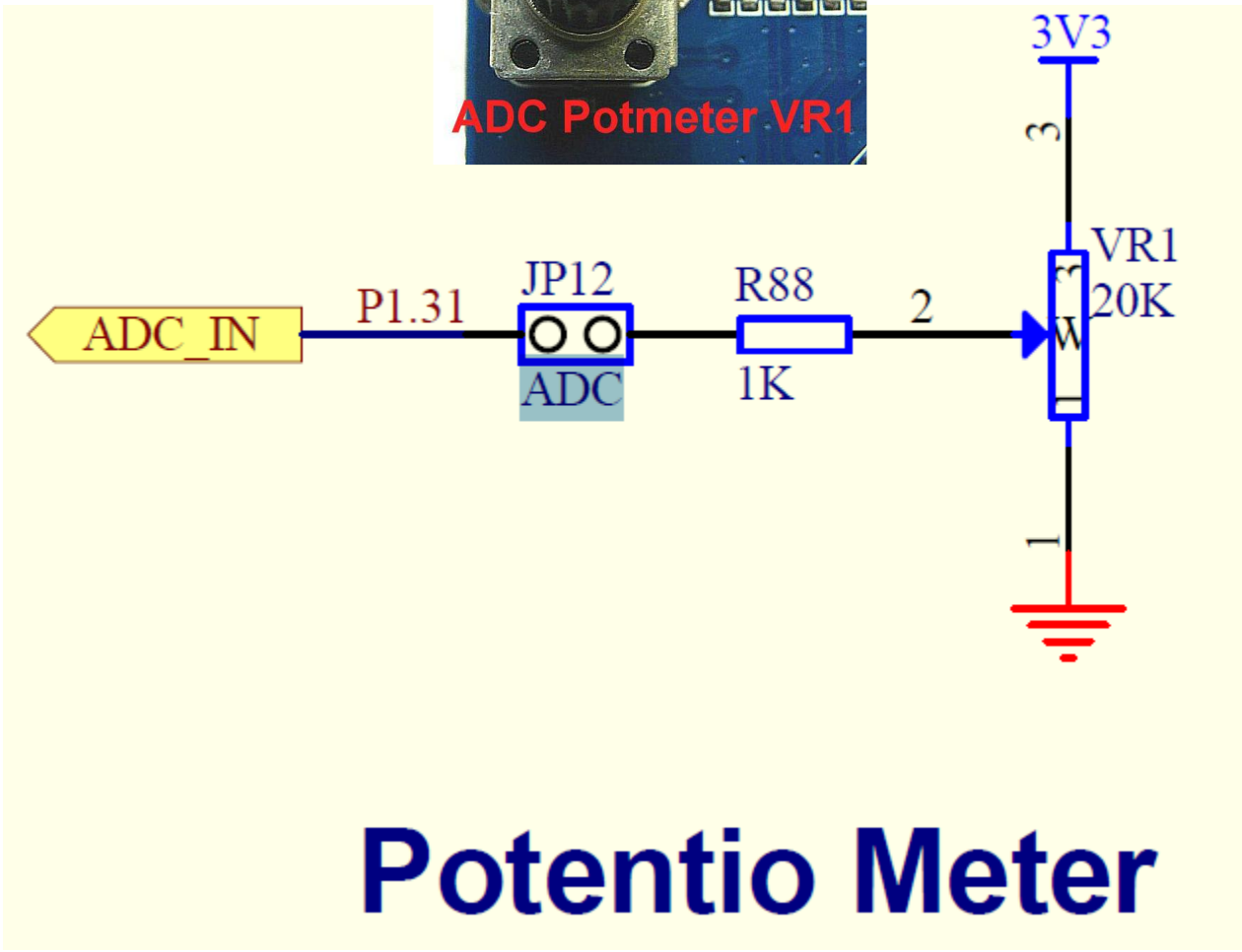
It contains the trim values for the DAC and the ADC

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

ADC connection on board

- Adjustable potentiometer VR1 is connected to analog channel P1.31 (AD0.5). JP12 jumper is used to enable the potentiometer input.
- VR1 provides input voltages between 0V and 3.3 V to the ADC.



ADC_init() and ADC_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    LPC_ADC->ADINTEN     =   (1<< 8);       /* global enable interrupt
18
19    NVIC_EnableIRQ(ADC_IRQn);               /* enable ADC Interrupt
20  }
21
22  void ADC_start_conversion (void) {
23    LPC_ADC->ADCR |=  (1<<24);              /* Start A/D Conversion
24  }
```

ADC_init() and ADC_start_conversion()

```
 7  void ADC_init (void) {
 8
 9    LPC_PINCON->PINSEL3 |=   (3UL<<30);      /* P1.31 is ADC
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
15                             (1<<21);       /* enable ADC
16
17    LPC_ADC->ADINTEN     =   (1<< 8);       /* global enable i
18
19    NVIC_EnableIRQ(ADC_IRQn);               /* enable ADC Interrupt
20  }
21
22  void ADC_start_conversion (void) {
23    LPC_ADC->ADCR |=  (1<<24);              /* Start A/D Conversion
24  }
```

Setup PIN function to ADC

Power on the ADC block

Setup ADC behavior

Enable ADC interrupt

Enable NVIC channel reserved to ADC interrupt

Start the conversion

A/D Control Register

```
 7  void ADC_init (void) {
 8
 9    LPC_PINCON->PINSEL3 |=   (3UL<<30);
10
11    LPC_SC->PCONP       |=   (1<<12);
12
13    LPC_ADC->ADCR        =   (1<< 5) |
14                             (4<< 8) |
15                             (1<<21);
16
17    LPC_ADC->ADINTEN     =   (1<< 8);
18
19    NVIC_EnableIRQ(ADC_IRQn);
20  }
21
22  void ADC_start_conversion (void) {
23    LPC_ADC->ADCR |=   (1<<24);
24  }
```

select AD0.5 pin

25MHz/5

enable ADC

Start conversion

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.	
		010	Start conversion when the edge selected by bit 27 occurs on the P2.10 / EINT0 / NMI pin.	
		011	Start conversion when the edge selected by bit 27 occurs on the P1.27 / CLKOUT / USB_OVRCRn / CAP0.1 pin.	
		100	Start conversion when the edge selected by bit 27 occurs on MAT0.1. Note that this does not require that the MAT0.1 function appear on a device pin.	
		101	Start conversion when the edge selected by bit 27 occurs on MAT0.3. Note that it is not possible to cause the MAT0.3 function to appear on a device pin.	
		110	Start conversion when the edge selected by bit 27 occurs on MAT1.0. Note that this does not require that the MAT1.0 function appear on a device pin.	
		111	Start conversion when the edge selected by bit 27 occurs on MAT1.1. Note that this does not require that the MAT1.1 function appear on a device pin.	
27	EDGE		This bit is significant only when the START field contains 010-111. In these cases:	0
		1	Start conversion on a falling edge on the selected CAP/MAT signal.	
		0	Start conversion on a rising edge on the selected CAP/MAT signal.	
31:28	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

A/D Interrupt Enable register

```
 7  void ADC_init (void) {
 8
 9    LPC_PINCON->PINSEL3 |=   (3UL<<30);
10
11    LPC_SC->PCONP       |=   (1<<12);
12
13    LPC_ADC->ADCR        =   (1<< 5) |
14                             (4<< 8) |
15                             (1<<21);
16
17    LPC_ADC->ADINTEN     =   (1<< 8);
18
19    NVIC_EnableIRQ(ADC_IRQn);
20  }
```

Global interrupt enabled

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_IRQHandler()

```
50  void ADC_IRQHandler(void) {
51
52      unsigned short AD_current;
53      unsigned short AD_last = 0xFF;      /* Last converted value
54
55      AD_current = ((LPC_ADC->ADGDR>>4) & 0xFFF);/* Read Conversion Result
56      if(AD_current != AD_last){
57          /* your action here */
58          AD_last = AD_current;
59      }
60
61  }
```

According to previous initializations

The sampled value is read, cleaning the interrupt

A/D Global Data Register

```
50  void ADC_IRQHandler(void) {
51
52      unsigned short AD_current;
53      unsigned short AD_last = 0xFF;      /* Last converted value
54
55      AD_current = ((LPC_ADC->ADGDR>>4) & 0xFFF);/* Read Conversion Result
56      if(AD_current != AD_last){
57          /* your action here */
58          AD_last = AD_current;
59      }
60
61  }
```

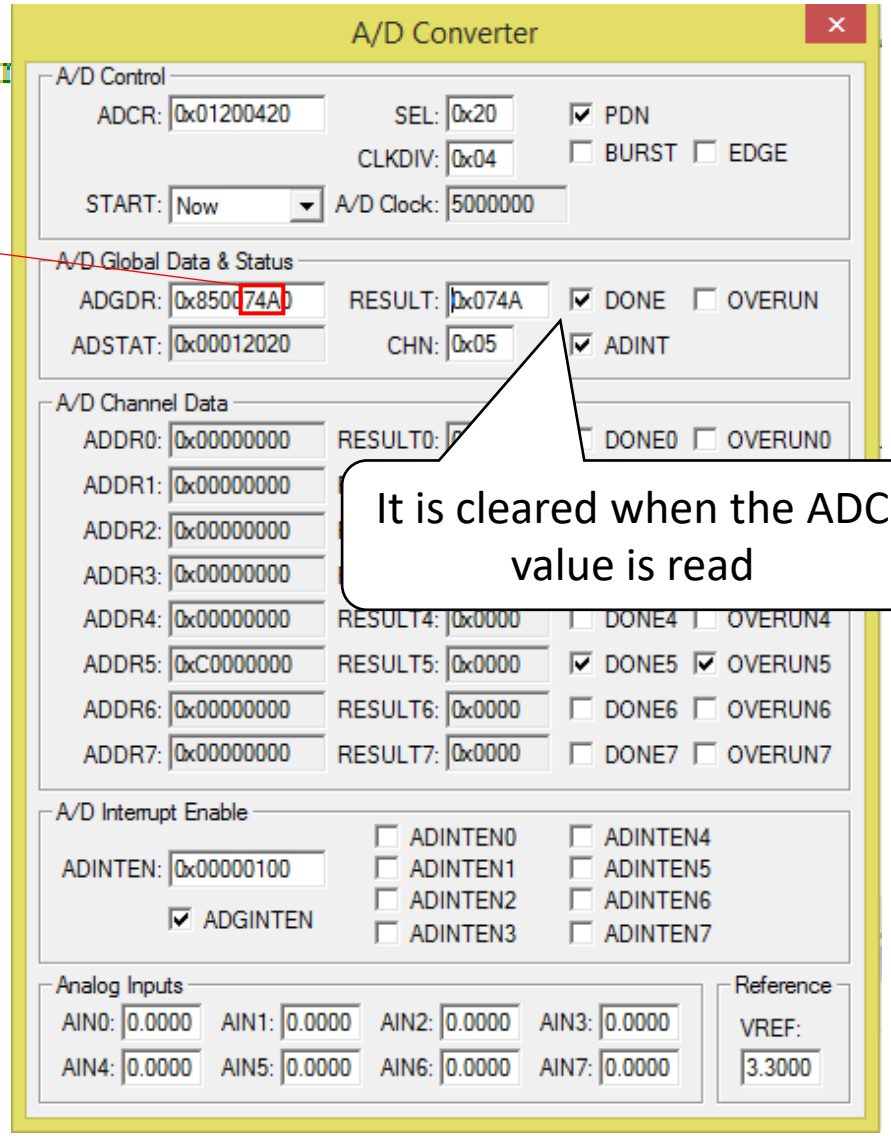
Table 533: A/D Global Data Register (AD0GDR - address 0x4003 4004) bit description

Bit	Symbol	Description	Reset value
3:0	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:4	RESULT	When DONE is 1, this field contains a binary fraction representing the voltage on the AD0[n] pin selected by the SEL field, as it falls within the range of V_{REFP} to V_{REFN}. Zero in the field indicates that the voltage on the input pin was less than, equal to, or close to that on V_{REFN}, while 0xFFF indicates that the voltage on the input was close to, equal to, or greater than that on V_{REFP}.	NA
23:16	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
26:24	CHN	These bits contain the channel from which the RESULT bits were converted (e.g. 000 identifies channel 0, 001 channel 1...).	NA
29:27	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
30	OVERRUN	This bit is 1 in burst mode if the results of one or more conversions was (were) lost and overwritten before the conversion that produced the result in the RESULT bits. This bit is cleared by reading this register.	0
31	DONE	This bit is set to 1 when an A/D conversion completes. It is cleared when this register is read and when the ADCR is written. If the ADCR is written while a conversion is still in progress, this bit is set and a new conversion is started.	0

It is cleared when the ADC value is read

A/D Global Data Register

```
50 void ADC_IRQHandler(void) {
51
52    unsigned short AD_current;
53    unsigned short AD_last = 0xFF;      /* Last co
54
55    AD_current = ((LPC_ADC->ADGDR>>4) & 0xFFF);/*
56    if(AD_current != AD_last){
57        /* your action here */
58        AD_last = AD_current;
59    }
60
61 }
```



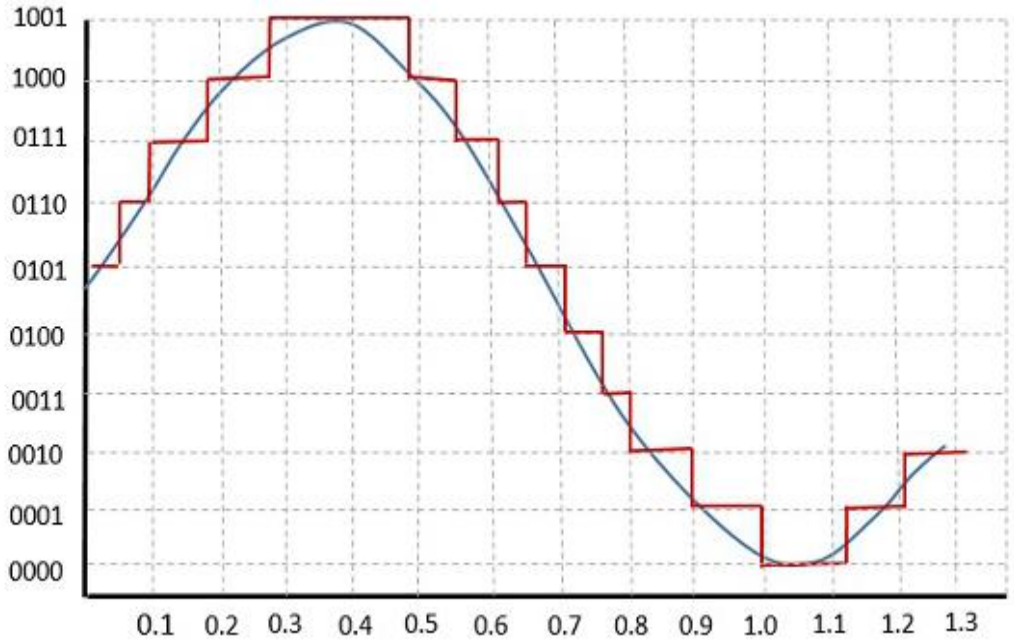
It is cleared when the ADC value is read

ADC value read can be triggered by a counter
e.g., the RIT starts a conversion every
50ms = $50*10^{-4}$s = 200Hz

```
26 void RIT_IRQHandler (void)
27 {
28
29   /* ADC management */
30   ADC_start_conversion();
31
32   LPC_RIT->RICTRL |= 0x1; /* clear interrupt flag */
33 }
```

Conversion from Digital to Analog

- Similarly to A/D Conversion, a Digital value can be converted into an analog one using a DAC
- Range
  - 0 to 3.3V
- Resolution
  - Depends on the number of bits of the DAC
  - 3.3V/15 = 0.22V
- Precision
  - n bits
  - 2^n levels
  - 2^4 = 16 levels

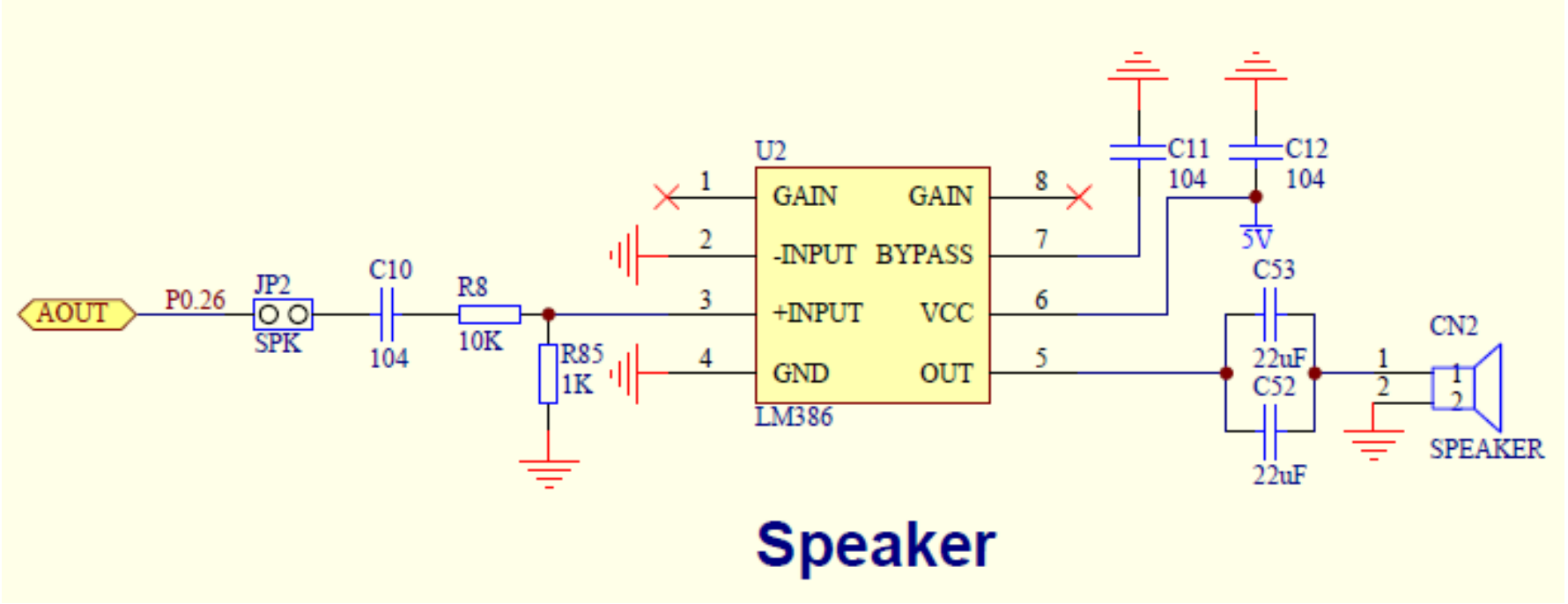


LPC1768 10-bit Digital to Analog Converter

- A 10-bit DAC is on-chip in the LPC1768 device
  - Precision = 1024 levels between 0 and 3.3V
  - Maximum update rate of 1 MHz.

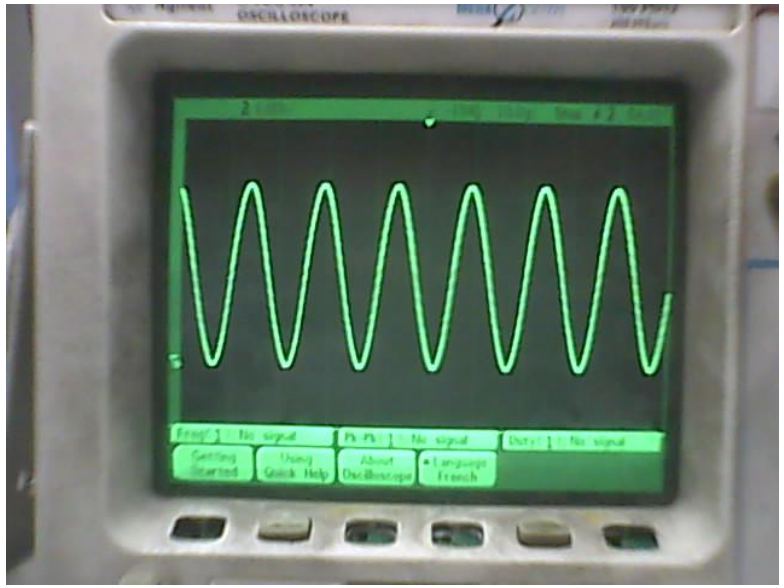
DAC connection on board

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

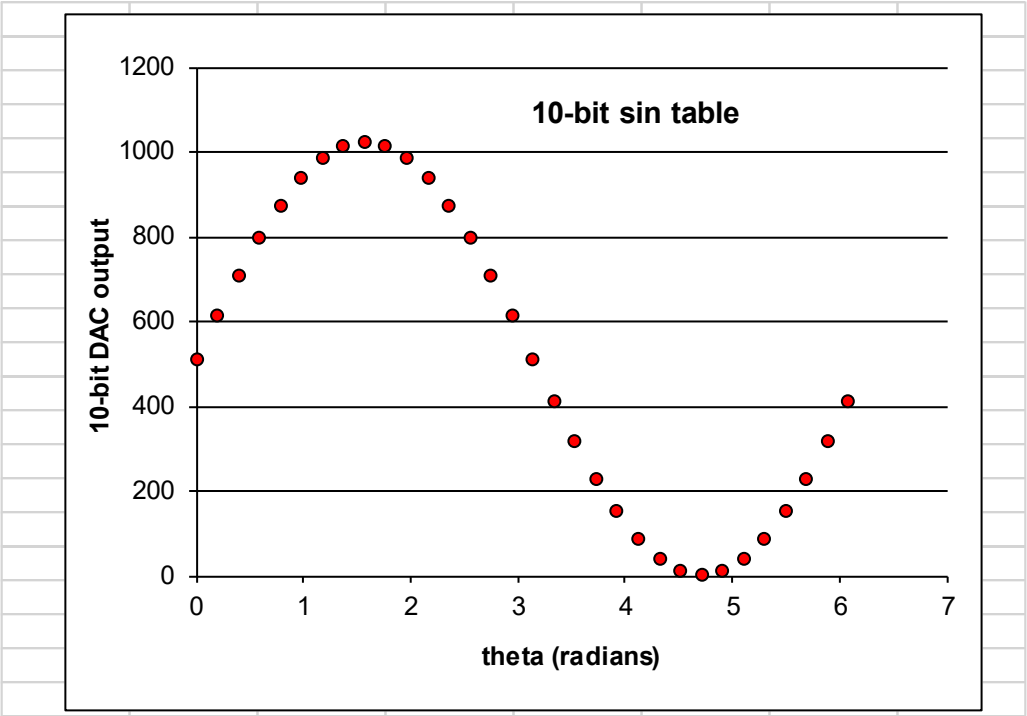


10-bit Sinusoid Table

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



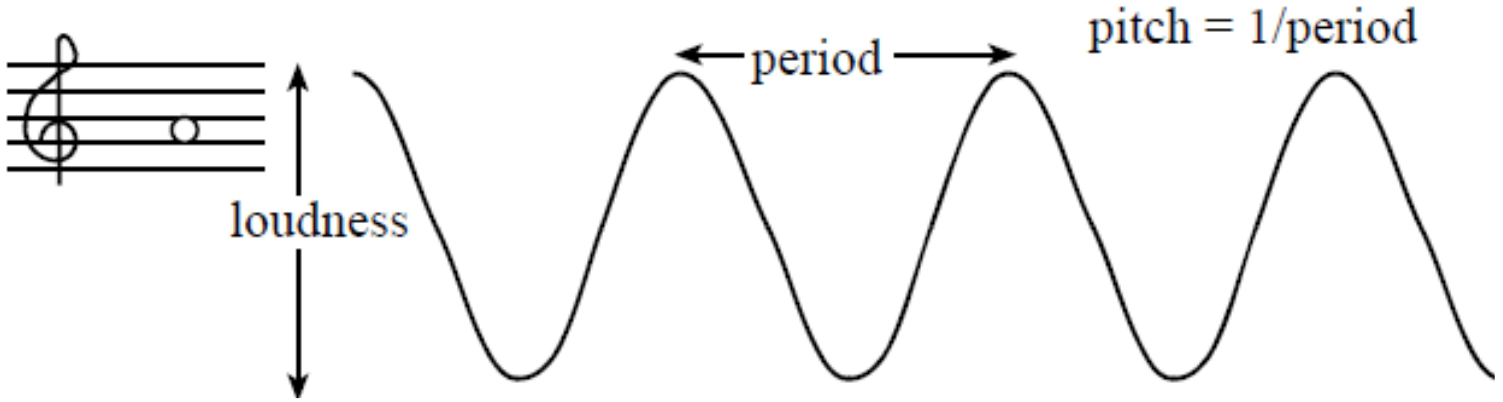
32 value sinusoid




```
int SinTab[32] =
{512,612,708,796,873,937,984,1013,1023,1013,98
4,937,873,796,708,612,512,412,316,228,151,87,4
0,11,1,11,40,87,151,228,316,412};
```

Sound

- Loudness and pitch
  - Controlled by amplitude and frequency



  - Humans can hear from about 25 to 20,000 Hz.
  - A musical tone (note) produced by a sinusoid of a particular frequency
    - Middle A is 440 Hz
    - The sinusoid is completely reproduced every 2.27ms

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

Use of timer handler

- To reproduce the sinosoid, a timer can be used
- Timer frequency determines the note

```
23  volatile uint16_t SinTable[45] =
24  {
25      410, 467, 523, 576, 627, 673, 714, 749, 778,
26      799, 813, 819, 817, 807, 789, 764, 732, 694,
27      650, 602, 550, 495, 438, 381, 324, 270, 217,
28      169, 125, 87 , 55 , 30 , 12 , 2  , 0  , 6  ,
29      20 , 41 , 70 , 105, 146, 193, 243, 297, 353
30  };
31
32  void TIMER0_IRQHandler (void)
33  {
34    static int ticks=0;
35    /* DAC management */
36    LPC_DAC->DACR = SinTable[ticks]<<6;
37    ticks++;
 8    if(ticks==45) ticks=0;
 9
 0
 1    LPC_TIM0->IR = 1;      /* clear interrupt flag */
 2    return;
 3  }
```

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

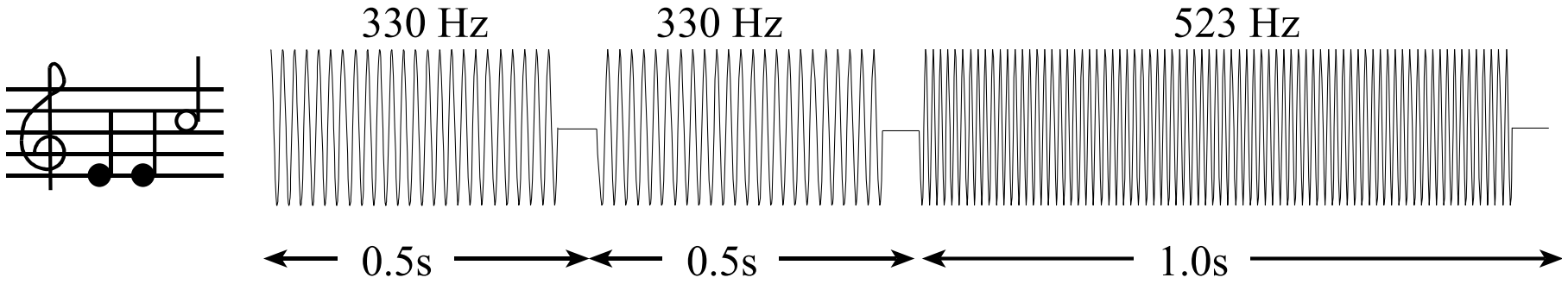
Bit	Symbol	Value	Description	Reset Value
5:0	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:6	VALUE		After the selected settling time after this field is written with a new VALUE, the voltage on the AOUT pin (with respect to 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

Synthesizing Digital Music (cont.)

- What is a musical tone?
  - A sinusoid of a particular frequency
  - Notes vary by twelfth root of 2 ~ 1.059
- What would the samples be?
  - Fixed point numbers
- How do we generate a sinusoid?
  - Output appropriate digital values via a resistor network that effectively produces a *pseudo*-analog signal
- What about frequency?
  - Employ a programmable timer to tell us when to output the next value

Tempo

- Tempo defines note duration
- Quarter note = 1 beat
- 120 beats/min => ½ s duration



Chord

- Two notes at the same time
  - Superimposed waveforms
  - 262 Hz (low C) and a 392 Hz (G)