EECS 452 – Lecture 1

Digital Signal Processing Laboratory

Course Description (from catalog)

EECS 452. Digital Signal Processing Design Laboratory Prerequisite: EECS 216, EECS 280 and EECS 451 or graduate standing. I, II (4 credits) Architectures of single-chip DSP processors. Laboratory exercises using two state-of-the-art fixed-point processors; A/D and D/A conversion, digital wave-form generators, and real-time FIR and IIR filters. Central to this course is a team project in real-time DSP design (including software and hardware).

Course webpages: `http://www.eecs.umich.edu/courses/eecs452/`

EECS 452 – Lecture 1

Digital Signal Processing Laboratory

Lecture: Tue, Thurs 10:30-12:00 in 1311 EECS
Labs: Tue (Sec 2), Wed (Sec 1) 3:30-6:30 in 4341 EECS

Instructors: Prof. Alfred Hero (hero) 4417 EECS.
Prof. Greg Wakefield (ghw) 3637 CSE
GSIs: Mr. Jonathan Kurzer (adamyang) 4341 EECS
Instructor emeritus: Dr. Kurt Metzger

Focus: embedded real-time DSP (using TI C5515 eZDSP stick and Altera DE2-70 FPGA).

Today:
- overview of EECS 452,
- review of DSP

One must learn by doing the thing; for though you think you know it, you have no certainty, until you try. — Sophocles

EECS 452 – Office Hours

Prof. Hero
Tue 12-1:30PM and Fri 10:30-12:00PM in EECS 4417

Prof. Wakefield
One time only: Mon 9/8 from 10-1PM in CSE 3637.
Starting on 9/22: Mon 1-4PM in CSE 3637.

Jonathan Kurzer
Monday 2:00-4:00PM; Tuesday and Wednesday 2:00-3:30PM and 6:30-8:00PM; in EECS 4341

Dr. Metzger
When in office and by appointment.

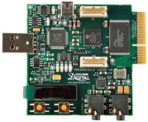
EECS 452 organization

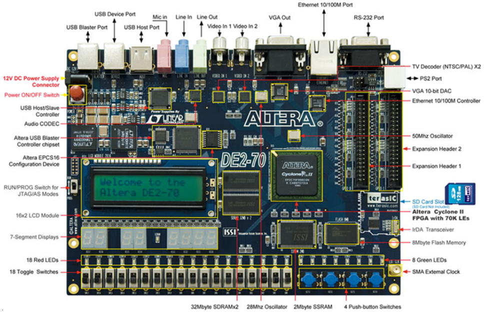
Goal: to provide a Major Design Experience (MDE); apply concepts from signals (216, 451) to real applications in an embedded environment.

- First half: you will learn embedded DSP from lectures and labs
  - Lab exercise 1: Introduction to the C5515 eZDSP Stick
  - Lab exercise 2: Basic DSP Using the C5515 eZDSP Stick
  - Lab exercise 3: Introduction to the DE2-70 FPGA board
  - Lab exercise 4: DSP on the DE2-70 FPGA board
  - Lab exercise 5: IIR filters
  - Lab exercise 6: Interrupts, FFT, and Graphics
    - Fall break Oct 13-14: T lab section shifts to Th
  - Lab exercise 7: (Optional) SPI
- Last half: you will design, implement and demonstrate a project (MDE).

EECS452 labs will use DSP and FPGA boards

TI C5515 DSP eZDSP stick – programmed via Matlab/C.
Altera DE2 FPGA – programmed via Verilog.
Custom boards to connect the two (designed by Dr. Metzger).





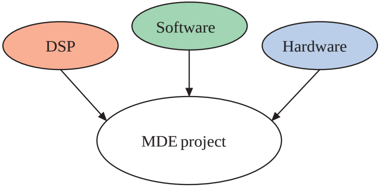
EECS452 organization

In the last half:

- Lab sessions will be devoted to your project; no more programmed lab assignments.
- Project process
  - you develop pre-project ideas (PPI) and turn them in (Ctools) as a graded assignment (due Sept 11)
  - you rank PPI's in Ctools Test Center between Sept 13 and Sept 16
  - you form project teams at a team formation meeting (Sept 18)
  - your team turns in a proposal (Sept 26)
  - your team orally presents the project proposal (Sept 29)
  - your team participates in 2 milestone meetings (Nov 6 and Nov 25)
  - your team demos your project at the COE Design Expo (Dec 4)
  - your team makes a final presentation to the class (Dec 10)
  - your team turns in a final project report (due Dec 12)

Team projects

- Student defined and executed.
- Teams of 3-4 students.
- Targeted to eventually become a commercial product.
  - Often a “proof-of-concept” or an “enabling” technology.
  - Demonstrate a working prototype at the semester end.
- Must involve digital signal processing concepts.
  - Real time DSP implementation
  - Signal/image manipulation (I/O, filters, transforms,. . .)
- Harris Inc will sponsor a project this semester



Some recent projects

- Vision, motion, control
  - Camera-directed robot (following prespecified color/path).
  - Obstacle avoiding robot
  - Automated music reverse transcription and playback system
- Audio, sound processing, and control
  - Shape controlled synthesizer
  - Touch screen synthesizer
  - Guitar auto-tuner
- Sensing, communication and networking
  - Wireless body-worn soldier health monitoring.
  - Stroke-Pro: racing boat stroke monitoring.
  - Feel the music: a glove that turns music into vibration
- DSP and communication fundamentals
  - Audio steganography: hiding messages behind an audio signal
  - OFDM modem
  - Active noise cancellation headset.

Harris' projects ("hwk/projects" webpage)

- **Automated garage parking**
- **Noise-canceling automobile**
- **Smart elevator**
- **Water bottle drinking meter**
- **Object localization and tracking**

If you elect to work on a Harris project your team will

- define (with instructors and Harris) a DSP project related to one of the above
- have periodic telecons with an engineer assigned by Harris to work with you
- have access to other experts within Harris
- build a relationship with a major Fortune 500 company in DSP/communications industry

Harris projects will otherwise work as any other project: same deadlines and reporting requirements.

References

Textbooks on reserve:

- Proakis and Manolakis, *Digital Signal Processing*, 4th ed 2006.
- Lyons, *Understanding Digital Signal Processing*, 3rd Ed, 2011.
- Dutoit & Marques *Applied Signal Processing - A MATLAB-Based Proof of Concept*, 2009.
- Schilling and Harris *Fundamentals of Digital Signal Processing Using MATLAB*, 2nd Edition, 2011
- Welsh, Cameron and Morrow, *Real-Time Digital Signal Processing from MATLAB to C with the TMS320C6x DSPs*, Second Edition, Taylor and Francis, 2011.
- Kuo and Gan, *Digital Signal Processing: Architectures, Implementations and Applications*, 2nd ed. 2005.

Hardware documentation and technical notes

- TI C55xx user manuals and other documentation
- Altera DE2 user manual and other documentation
- Analog Devices tutorials and technical notes
- Maxim/AD tutorials and technical notes

Grading policy and course requirements

Lab	20%	prelab (25%) and report (75%)
Homework	10%	5 of them + PPI
Midterm	20%	open book and notes
Project	50%	final presentation (50%) and report (50%)

Midterm: 7-9PM on Oct 23. Covers lecture, lab, and homework.

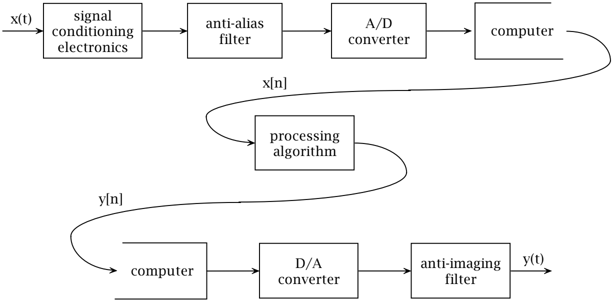
Homework is to be done **individually**. You may discuss the homework but the final submission is to be individual work.

Homework is due at the beginning of lecture unless otherwise indicated. No late homework is accepted.

Prelabs are to be done individually and turned in before the designated lab session. Lab reports are to be turned in one week from your lab session, one report per per partner pair, and all work should be the **original** work of that partner pair. Any violation will be reported to the Honor Council.

Unless otherwise specified lab reports are **not** to be hand written! Homeworks can be neatly handwritten.

DSP Review: The basic DSP paradigm



Cts time/amp signal → Digital signal → Cts time/amp signal

What is a Digital Signal Processor?

- (wikipedia) A digital signal processor is a specialized microprocessor designed specifically for digital signal processing, generally in real-time computing.
- Optimized to handle things like FIR, FFT, and other similar algorithms using as little power/cost/formfactor as possible.
  - Memory mapped I/O (MMIO): actuates physical devices and acquires signals;
  - Very fast multiply-accumulates (MACs), critical to FIR filters, matrix operations, and FFT;
  - Fixed point (Integer) arithmetic: faster than floating point.
- Incorporates analog-digital and digital-analog conversion (CODEC)
- Frequently *embedded* in a larger system; interacts with physical world and processes signals/images in real time.

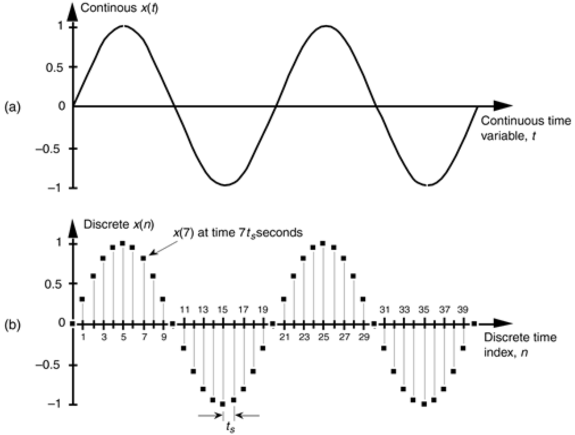
Digitization errors and distortions

Shannon's data processing theorem: Digitization (sampling/quantization) of signals entails loss of information

- Distortion due to sampling
  - Aliasing distortion → increase sample rate
  - Non-ideal sampling distortion → increase sampling bandwidth
  - Non-ideal reconstruction → increase sample rate, improve interpolation
  - Clock jitter → stabilize clock rates.
- Distortion due to quantization
  - Round-off error → increase resolution (# bits)
  - Saturation/overload error → proper scaling/companding
  - Roundoff error propagation → careful balancing of arithmetic opns
- Other types of errors: thermal noise, non-linear transducers, latency, drift.

It especially important to understand and mitigate these for embedded DSP. We will study these major sources of error over the semester.

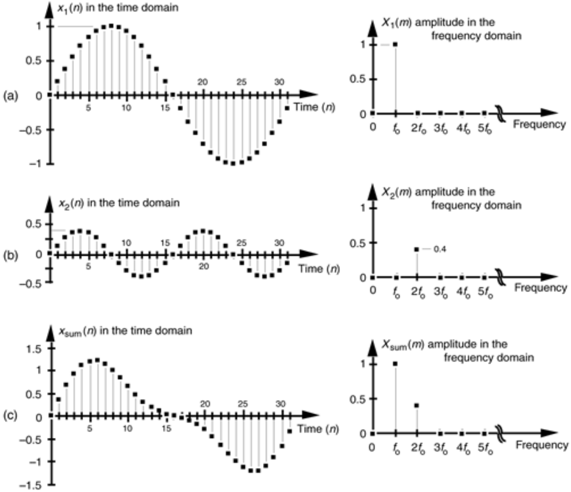
Continuous and discrete signals (Lyons Ch 1)



If done right, sampled signal inherits properties of cts time signal

To do DSP need understand relation between sampled and original cts time signals.

Utility of Fourier signal analysis (Lyons Ch 1)



Fourier analysis decomposes signals into sinusoidal components

Review of basic DSP concepts

Topics we will review in signal processing

- Canonical cts signals: sinusoids, complex exponentials, delta functions.
- Passing a cts time signal through a linear time invariant system
- Frequency domain representations
  - Fourier transform and Fourier series.
  - Linear time invariant systems (LTI)
- Discrete time signals and filters
- Lowpass and bandpass signals and filters
  - Frequency conversion.
  - Sampling and reconstruction.
  - Frequency aliasing and anti-aliasing filter.

Continous time signals

Finite power periodic signals:

- Sinusoidal signal: $s(t) = A\cos(2\pi F_0 t + \phi)$
  - $A > 0$, F_0 and ϕ: amplitude, frequency (Hz), and phase of s
  - Power $\frac{1}{T_0}\int_0^{T_0} |s(t)|^2 dt = A^2/2$, $T_0 = 1/F_0$ period of s
- Complex exponential signal: $x(t) = Ae^{j\phi}e^{j2\pi F_0 t}$
- Euler relations:

$$e^{j\theta} = \cos(\theta) + j\sin(\theta), \qquad \cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

Finite power pulsatile signals:

- Symmetric pulse signal: $p_\tau(t) = \begin{cases} 1/\tau, & -\tau/2 \le t \le \tau/2 \\ 0, & o.w. \end{cases}$
  - Power $\frac{1}{\tau}\int_{-\infty}^{\infty} |p_\tau(t)|^2 dt = 1$
- Dirac delta function signal: $\delta(t) = \lim_{\tau\to\infty} p_\tau(t)$

Linear time invariant (LTI) filters

A *linear system* h takes an input x and produces an output y

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau$$

Fundamental property of a linear filter is *superposition*:

$$\text{if} \quad x_1(t) \xrightarrow{h} y_1(t) \text{ and } x_2(t) \xrightarrow{h} y_2(t)$$

$$\text{then} \quad ax_1(t) + bx_2(t) \xrightarrow{h} ay_1(t) + by_2(t)$$

We can decompose a signal or filter into components, solve for the responses to the individual components and then construct the overall response by adding up the individual responses.

Nonlinear systems create additional modulation cross-products between x_1 and x_2 and do not satisfy superposition principle.

Time invariance, impulse response, causal filters

Fundamental time invariance property of an LTI: $x(t) \xrightarrow{h} y(t)$ implies

$$x(t-\tau) \xrightarrow{h} y(t-\tau)$$

The *impulse response* of an LTI is the output $y(t)$ when $x(t) = \delta(t)$.

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)\delta(\tau)dt = \int_{-\infty}^{\infty} h(\tau)\delta(t-\tau) = h(t)$$

A LTI system is *causal* if output $y(t)$ only depends on past inputs $x(\tau), \tau \leq t$. Equivalently

$$h(t) = 0, \;\; t < 0$$

LTI with sinusoidal inputs

Linear time-invariant systems (LTI):

- If $h(t)$ is the impulse response of the LTI, then the input-output relationship is given by a convolution:

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau = \int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau$$

- Not usually easy to compute. However, if input is a periodic complex exponential then output is also with same frequency.

$$\begin{aligned} x(t) &= Ae^{j2\pi F_o t}, \\ y(t) &= \int_{-\infty}^{\infty} Ae^{j2\pi F_o (t-\tau)} h(\tau)d\tau \\ &= \int_{-\infty}^{\infty} h(\tau)e^{-j2\pi F_o \tau} d\tau \cdot Ae^{j2\pi F_o t} \\ &= H(F_o) \cdot Ae^{j2\pi F_o t} \end{aligned}$$

- $H(F_o)$ is called the system *transfer function* at frequency F_o (Hz).

So...

Computing the response $y(t)$ of LTI systems to exponential inputs $x(t) = Ae^{j2\pi F_o t}$ is *very easy*.

- It is thus both natural and desirable in linear systems analysis to derive methods for expanding signals as sums of complex exponentials.
- We know two such methods:
  - Fourier series (FS);
  - Fourier transform (FT).

Fourier series and Fourier transform (CFT)

Fourier series: for a finite power signal that is periodic with period T_o, i.e., fundamental frequency $F_o = 1/T_o$:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{j2\pi n F_o t}$$

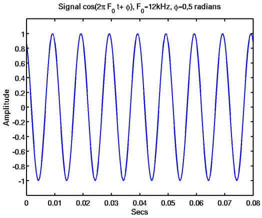
$$c_n = \frac{1}{T_o} \int_0^{T_o} x(t) e^{-j2\pi n F_o t} dt, \quad \text{where } -\infty < n < \infty.$$

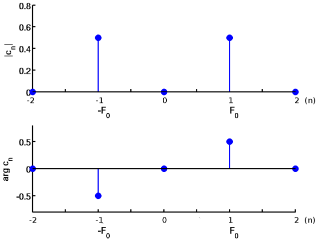
Continuous-time Fourier transform: for a finite energy signal:

$$X(F) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi F t} dt \quad \text{where } -\infty < F < \infty,$$

$$x(t) = \mathcal{F}^{-1}\{X(F)\} = \int_{-\infty}^{+\infty} X(F) e^{j2\pi F t} dF \quad \text{where } -\infty < t < \infty.$$

Example: FS sinusoid $s(t) = A\cos(2\pi F_0 t + \phi)$





Sinusoid at $F_0 = 100$Hz

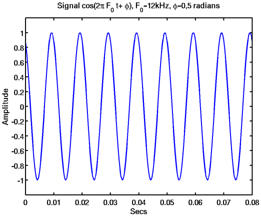
Fourier series ($T_0 = 1/F_0$=period)

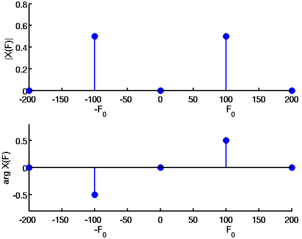
$$s(t) = \sum_{n=-\infty}^{\infty} c_n e^{j2\pi n F_0 t}, \quad c_n = \frac{1}{T_0}\int_0^{T_0} s(t) e^{-j2\pi n F_0 t} dt$$

Find c_n by inspection using Euler $\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$

$$s(t) = \underbrace{\frac{A}{2}e^{j\phi}}_{c_1} \cdot e^{j2\pi F_0 t} + \underbrace{\frac{A}{2}e^{-j\phi}}_{c_{-1}} \cdot e^{-j2\pi F_0 t} + \underbrace{0 + 0 + 0}_{c_n,\ n \notin \{-1,1\}}$$

Example: CFT sinusoid $s(t) = A\cos(2\pi F_0 t + \phi)$





Sinusoid at $F_0 = 100$Hz   Fourier transform ($T_0 = 1/F_0$=period)

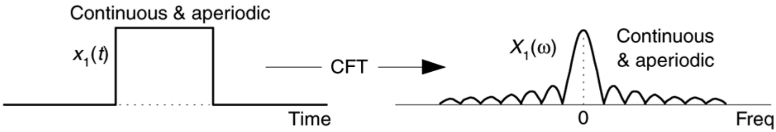
$$S(F) = \mathcal{F}\{s(t)\} = \int_{-\infty}^{\infty} s(t)e^{-j2\pi Ft}dt$$

Recall Fourier transform identity: $\mathcal{F}\left\{e^{j2\pi F_0 t}\right\} = \delta(F - F_0)$:

From the FS expression we just derived

$$S(F) = \frac{A}{2}e^{j\phi} \cdot \delta(F - F_0) + \frac{A}{2}e^{-j\phi} \cdot \delta(F + F_0)$$

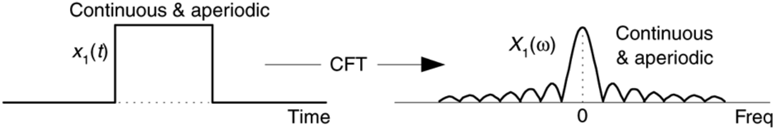
Example: pulsatile signal of width τ



Fourier transform of symmetric pulse signal $x_1(t) = p_\tau(t)$ given above

$$X_1(F) = \int_{-\infty}^{\infty} x_1(t) e^{-j2\pi F t} dt = \mathrm{sinc}(\pi F \tau) = \frac{\sin(\pi F \tau)}{\pi F \tau}$$

Example: pulsatile signal of width τ



Note: $\lim_{\tau\to 0} X_1(F) = 1$ and $\lim_{\tau\to\infty} X_1(F) = \delta(F)$.

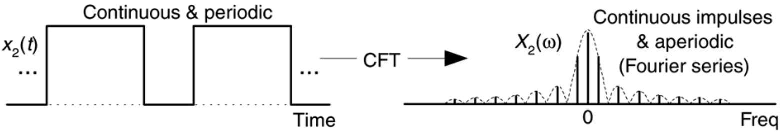
Fourier transform of delayed pulse $z(t) = p_\tau(t-u)$?

$$Z(F) = X_1(F)e^{-j2\pi F u}$$

Hence, we recover well known facts

$$\mathcal{F}\{\delta(t-u)\} = e^{-j2\pi F u}, \qquad \mathcal{F}^{-1}\{e^{-j2\pi F u}\} = \delta(t-u)$$

Example: periodic pulse train $\tau \ll T_0$=period



Fourier series of $x_2(t) = \sum_{n=-\infty}^{\infty} p(t - nT_0)$, $T_0 = 1/F_0$

$$x_2(t) = \sum_{n=-\infty}^{\infty} c_n e^{j2\pi n F_0 t}$$

$$c_n = \frac{1}{T_0} \int_0^{T_0} x_2(t) e^{-j2\pi n F_0 t} dt = \frac{1}{T_0} \mathrm{sinc}(\pi n F_0 \tau) = \frac{1}{T_0} X_1(nF_0)$$

Discrete time signal transforms

Let $\{x[n]\}_{n=-\infty}^{\infty}$ be discrete-time signal

Discrete-time Fourier series: for finite power periodic signal with period N:

$$x[n] = \sum_{k=0}^{N-1} c_k e^{j2\pi nk/N}$$

$$c_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}$$

Discrete-time Fourier transform (DTFT): for a finite energy signal:

$$X(f) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi fn}$$

$$x[n] = \int_{-1/2}^{1/2} X(f) e^{j2\pi fn} df.$$

$X(f)$ is periodic in digital frequency f with period 1.

Discrete Fourier Transform (DFT) (1/2)

DTFT is not useful for DSP computations:

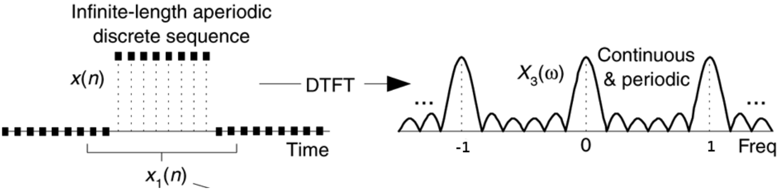
- DTFT is a continuous function of frequency.
- It requires an infinite number of time-domain samples for calculation.
- Not computationally feasible using DSP hardware.
- Idea: sample the continuous spectrum $X(f)$.

(An N-point) Discrete Fourier Transform (DFT):

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \quad \text{where } k = 0, 1, 2, \ldots, N-1.$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi kn/N} \quad \text{where } n = 0, 1, 2, \ldots, N-1.$$

Example: DTFT for sampled pulsatile signal



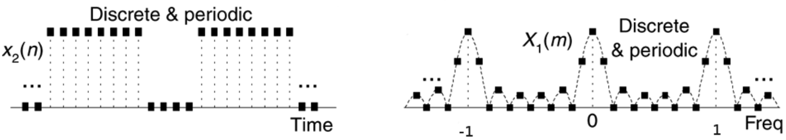
Periodically sampled version of pulse at sampling freq. $F_s = 1/T_s$:

$$x[n] = p_\tau(nT_s), \quad n = \ldots - 1, 0, 1$$

We set the sample period T_s such that $\tau/T_s = M$ (odd)
Then DTFT is:

$$X_{DTFT}(f) = \sum_{n=-\infty}^{\infty} x[n]e^{j2\pi fn} = \frac{1}{T_s}\underbrace{\frac{\sin(\pi f M)}{M\sin(\pi f)}}_{Dirichlet} = \frac{\sin(\pi f F_s \tau)}{\sin(\pi f F_s)\tau}$$

Example: DFT for finitely sampled pulsatile signal



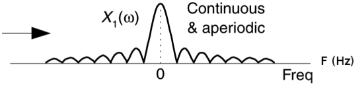
Consider a window of N of these samples containing the pulse

$$x_1[n] = p_\tau(nT_s - \tau/2), \quad n = \dots 0, \dots, N-1$$

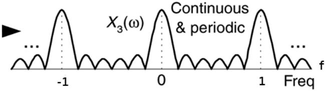
where $N \geq M$ and, as before, $\tau/T_s = M$ (odd).
Then DFT is:

$$X_{DFT}[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N} n} = \frac{1}{T_s} \underbrace{\frac{\sin(\pi \frac{k}{N} M)}{M \sin(\pi \frac{k}{N})}}_{Dirichlet}$$

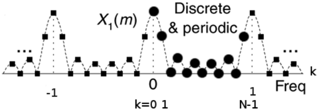
Freq conversions btwn CFT, DTFT, and DFT



$$X_{CFT}(F) = \frac{\sin(\pi F \tau)}{\pi F \tau}$$



$$X_{DTFT}(f) = \frac{\sin(\pi(fF_s)\tau)}{\sin(\pi(fF_s))\tau}$$



$$X_{DFT}[k] = \frac{\sin\left(\pi\left(\frac{k}{N}F_s\right)\tau\right)}{\sin\left(\pi\left(\frac{k}{N}F_s\right)\right)\tau}$$

Conversion formulas:

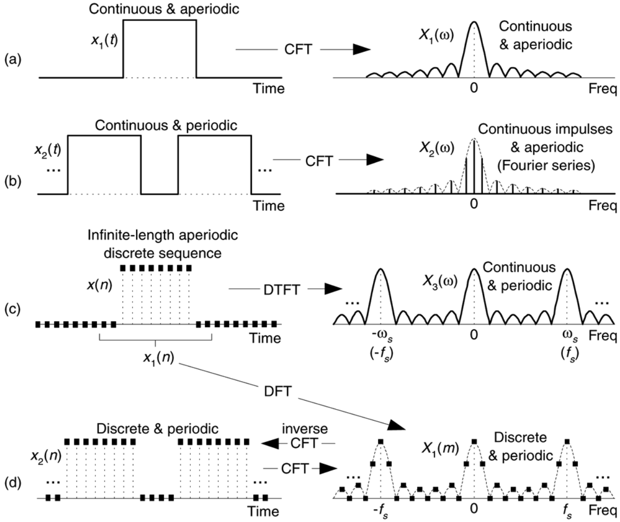
- Conversion from DTFT freq f to CFT freq F: $F = fF_s$
- Conversion from DFT freq k to DTFT freq f: $f = \frac{k}{N}$
- Conversion from DFT freq i to CFT freq k: $F = \frac{k}{N}F_s$

Continuous time vs discrete time signals

Series representations and transforms

- For continuous time signals:
  - Fourier series (for periodic signals).
  - Continuous time Fourier transform (for non-periodic signals).
  - (The above two may be unified through the use of impulse functions).
- For discrete time signals:
  - Discrete time Fourier series (for periodic signals).
  - Discrete time Fourier transform (for non-periodic signals).
  - The two can be unified using impulse functions.
  - Discrete Fourier transform (DFT) (for finite duration signals).
- Other transforms: Laplace, Z-transform, Wavelet, etc.

Summary: cts and discrete transform relations



The z-transform

The z-transform of a discrete set of values, $x[n]$, $-\infty < n < \infty$, is defined as

$$X_Z(z) = \mathcal{Z}(x[n]) = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$$

where z is complex valued. The z transform only exists for those values of z where the series converge. z can be written in polar form as $z = re^{j\theta}$.

r is the magnitude of z and θ is the angle of z. When $r = 1$, $|z| = 1$ is the unit circle in the z-plane.

When $x[n] = 0$ for $n < 0$, $X(z)$ reduces to single-sided z-transform

$$X_Z(z) = \mathcal{Z}(x[n]) = \sum_{n=0}^{\infty} x[n] z^{-n}$$

The inverse z-transform

$$x[n] = \mathcal{Z}^{-1}[X(z)] = \frac{1}{2\pi j} \oint_C X_Z(z) z^{n-1} dz$$

Here C is the closed contour of $X(z)$ of the region of convergence (ROC).

We will never use the contour integral to compute the inverse z-transform

Alternative methods of computation:

- Long division method.
- Partial fraction expansion method.
- Use of residues.

See Proakis or a similar text (or Wikipedia) for details.

The z-transform and the DTFT/DFT

- Z-transform over complex plane

$$X_Z(z) = \mathcal{Z}(x[n]) = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$$

- DTFT over digital frequency $f \in [0, 1]$

$$X_{DTFT}(f) = \text{DTFT}(x[n]) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi f n}$$

- DFT over integers $k = 0, \ldots, N$

$$X_{DFT}[k] = \text{DFT}(x[n]) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N} n}$$

Conclude: $X_{DTFT}\,(f) = X_Z\left(e^{j2\pi f}\right)$ and

$$X_{DFT}[k] = X_{DTFT}\left(\frac{k}{N}\right) = X_Z\left(e^{j2\pi \frac{k}{N}}\right)$$

Transfer functions: cts time signals

The waveform $y(t)$ obtained by processing a waveform, $x(t)$, by a LTI system having impulse response $h(t)$ is the convolution

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau\,.$$

Take Fourier Transform to obtain equivalent frequency domain relation:

$$Y(F) = H(F)X(F)\,.$$

where $Y(F), X(F), H(F)$ are FTs of $y(t), x(t), h(t)$ evaluated at Hertzian freq. F Hz.

- It is often easier to think of the effects of LTI in the (frequency) domain than in the time domain.
- It is sometimes easier to operate on a waveform in the transform domain than it is in the time domain, in spite of the computational costs of going between domains.

Transfer functions: discrete time signals

The sequence $y[n]$ obtained by processing a sequence, $x[n]$, by a discrete time LTI system having impulse response $h[n]$ is the convolution

$$y[n] = \sum_{k=-\infty}^{\infty} h[n-k]x[k]\,.$$

Take DTFT to obtain equivalent frequency domain relation:

$$Y(f) = H(f)X(f)\,.$$

where $Y(f), X(f), H(f)$ are DTFTs of $y[n], x[n], h[n]$ evaluated at digital freq. f.

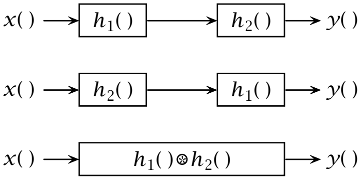
- It is often easier to think of the effects of LTI in the (frequency) domain than in the time domain.
- It is sometimes easier to operate on a waveform in the transform domain than it is in the time domain, in spite of the computational costs of going between domains.

How are transforms used in this course

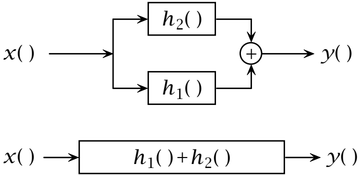
The z-transform will be used to model filter transfer functions in the frequency domain.

The DFT will be used as a computational tool for implementing filters and for visualizing spectra.

LTI system connections



cascade connection



parallel connection

Summary of what we covered today

- Course overview
- DSP review
  - Linear time invariant systems
  - Transfer functions
  - Fourier series
  - Fourier transforms DFT
- Next: Sampling and reconstruction: aliasing, anti-aliasing, anti-imaging

Reference: Proakis and Manolakis, *Digital Signal Processing*, 4th ed 2006.