Section #05330 / #05359 Winter 2010

Study Guide and Listening Lists for

Music 413/513

The History and Practice of Electronic Music

Professor:
Dr. Mark Phillips

Graduate Assistant:
Roderick Gorby

Music 413/513: The History and Practice of Electronic Music

Winter 2010 / Sec. #05330 /#05359
Tues/Thurs 1:10–2:00 p.m.
Music Bldg. Rm. 475A

Mark Phillips, Professor
Office: rm. 475B Glidden Hall
phillipm@ohio.edu

This course is designed to give an overview of the history of electronic music (including representative musical compositions), a solid background understanding of its technology and terminology, and "hands on" experience with the equipment and software available in the lab. Important components of the class will deal with MIDI (Musical Instrument Digital Interface), digital audio, and digital synthesis. It is assumed that students have taken the prerequisite class, Music 178/178A: Computer Skills for Musicians.

Class Packets: *Music 413: The History and Practice of Electronic Music* plus *Basic FM Synthesis*
> both available in ".pdf" format as Course Documents on the Music 413/513 Blackboard™ site.

Texts: *Electric Sound: The Past and Promise of Electronic Music,* Joel Chadabe, Prentice-Hall, 1997.
>available in bookstores and on reserve in the library. **(required)**

Introduction to Electro-Acoustic Music, Barry Schrader, Prentice-Hall, 1982.
>available on library reserve. **NOTE:** Also on reserve is a photocopy of excerpts from this book which deal specifically with the listening examples on your tapes (see Listening Guides 1 & 2). It is strongly recommended that you make yourself a copy of this material.

Modern Recording Techniques, Robert E. Runstein, Howard W. Smith & Co., (various editions)
>on reserve: Chapter 2 suggested reading on sound waves

In addition to attending Tuesday/Thursday lectures you will work in the lab on individual projects.

Class grades will be determined solely by the totaling the following:

Exam I (25%) + Exam II (25%) + Exam III (25%) + lab projects (25%)

- No make-up exams will be given without prior arrangements.
- Academic dishonesty on exams will result in a class grade of "F"

Course Overview

I. Introduction — Early History & Musique Concrète

January 5 — 28. Exam I on (or near) January 26 or 28. (ca. 100 pts.)

Readings: Schrader: pp. 1-15, 16-38, 39-58 Chadabe: Chapters 2, 3, & 4
Runstein: Chapter 2 — suggested

Listening: Exam 1 listening — mp3 files available on *Blackboard*™

>Lab work: mini-project 1 — *musique concrète*–style composition/soundscape (ca. 30 pts)**

II. Basic Principles of Electronic Sound Synthesis/Early Electronic Instruments

February 2 — February 18. Exam II on (or near) February 18. (ca. 100 pts.)

Readings: Schrader: pp. 61-69, 75-119, 122-159 Chadabe: Chapters 1, 6

Listening: Exam 2 listening — mp3 files available on *Blackboard*™

>Lab work: mini–project 2 — analog-style synthesis project (ca. 30 pts.)**

III. Digital Audio, Digital Sampling, Computer Synthesis Techniques, Other Uses for MIDI

February 23 — March 11. Exam III — During Final Exam Week (ca. 100 pts.)

- **Monday, March 15, at 2:30 p.m.**
- Lab Projects due by 5 PM, Thursday, March 18 — *but early submission is recommended* (40 pts.)

Readings: Chadabe: Chapters 5, 7, 8 & 9

Listening: Exam 3 listening — mp3 files available on *Blackboard*™

>Lab work: final project **

*** See details for lab projects on following pages of the course packet.*

Course Outline (tentative lecture & lab topics — both subject to change!)

Week 1

1/5 Course Intro — Early History of Electronic Music & Musique Concrète

1/7 Intro to mini-project 1 (musique concrète–style using DP4, Pierre's Playhouse, etc.)

**lab: mini-project 1

Week 2

1/12 Physical Properties & Human Perception of Sound as a Waveform Phenomenon

1/14 Continuation on Sound Waves;Lab Project Help Session

**lab: mini-project 1 (continued)

Week 3

1/19 Organized Sound, the Art of Noise, and the Origins of Electronic Music Esthetics

1/21 Early Musique Concrète "Classics" (Exam 1 listening examples)

**lab: mini-project 1 (continued)

Week 4

1/26 More Musique Concrète "Classics" / Exam 1 Review

1/28 **Exam 1**

**lab: mini-project 1 (completed)

Week 5

2/2 Early Electronic Music Instruments & Early Electronic Studio Tape Music

2/4 Pioneers of Analog Synthesis (Moog, Buchla, etc.); Basic Principles of Sound Synthesis

**lab: mini-project 2 (analog-style synthesis using E-A_MusicPrimer, etc.)

Week 6

2/9 More on Electronic Sound Synthesis Techniques: Subtractive & Additive Synthesis

2/11 Early "Classics" of Electronic Tape Music, Analog Synthesizers, and the RCA Mark II (Exam 2 listening examples)

Week 7

2/16 More on Electronic Sound Synthesis Techniques; review for Exam 2

2/18 **Exam 2**

**lab: mini-project 2 (completed)

Week 8

2/23 Intro to Digital Audio / Direct Digital Synthesis (Csound)

2/25 Basic FM synthesis

**lab: final project

Week 9

3/2 More Basic FM synthesis

3/4 Intro to MIDI: history & origins, technology standards, applications, MIDI controllers, etc.

**lab: final project (continued)

Week 10

3/9 More Digital Audio (granular synthesis, analysis, and re-synthesis, etc.)

3/11 Early "Classics" of Direct Digital Synthesis, Computer Music, & recent live/electronic music — (Exam 3 listening examples) / Review for Final Written Exam

**lab: final project (continued)

Week 11 **Exam 3: Monday, March 15, at 2:30 p.m.** (scheduled final exam period)

Final Project: Due Thursday, March 18 at 5:00 p.m. — *but early submission is recommended!*

*** See details for lab projects on the following pages of the course packet.*

Lab Projects Guidelines — See class Blackboard™ site for more details

For all your projects, you should turn in a stereo mix audio file as well as a "working copy" of your entire project, including all audio files. Refer to "Post Production Guidelines" at the end of this document for more details.

Lab Mini-Project 1: Musique Concrète-style Project — due: by 5:00 p.m. Friday, 1/29

Create a musique concrète-style composition or soundscape (ca. 2 minutes). Use only recorded audio as your sound source ... no synthesizers, please. You should experiment with various programs that allow serious mangling of audio, especially those that approximate the traditional musique conrète style of analog tape manipulation — things like speed and direction change, small fragments, loops, etc.. A good starting point is Pierre'sPlayhouse, which is one of the modules contained on the "ProjectSynthPack" page of my E-A_MusicPrimer. SoundHack is another option. Digital Performer (DP6), itself, also has considerable sound mangling resources.

You can use any full-featured, multi-channel digital audio program to assemble your final mix. In the MIDI Lab you can use Digital Performer (DP4.6), Audacity, or Garage Band (but w/o using MIDI synthesizers and built-in loops!). Other suitable programs include such programs as Pro Tools, Cubase, Acid, Ableton Live, Apple Logic, FruityLoops, and Cakewalk Sonar. (Built-in sample players are fine, but please do not use any built-in software synthesizers for this project.)

Mini-Project 1 — a step-by-step approach

1. Import some audio material from the "SoundFX folder" on the server — AND/OR ...
2. Import or record (field recording?) small sound-bites of your own "non-musical" audio (≤ 20"/sound-bite)
   Note: The Music Dance Library has the entire BBC sound effects library on CD.
3. Make LOTS of edits (copy/cut/paste, reverse, gain changes, etc.) — But be careful to avoid "clicks"
4. Use several examples of speed change — slower, faster, and especially variable speed change.
   Note: Currently in the lab, variable speed change can be accomplished with SoundHack or Pierre'sPlayhouse.
5. Create an "automated mix" (i.e. fade in/out, pan, etc.) and "bounce" or record a stereo file to disk.

Lab Mini-Project 2: Analog-Style Synthesis Project — due: by 5:00 p.m. Friday, 2/19

Various options available — including the following:

MAX/MSP

- Explore the various synthesis modules in E-A_MusicPrimer especially the following:
  Filters&LFO_Modulation: all submodules
  ProjectSynthPack: AdditiveSynthesizer, AdditiveSynthesizer+, NoiseySynthesizer, SoundSculptor, ModWheelMayhem, and StepSequencer
- Explore the Forbidden Planet module.
- Design some interesting sounds with these modules and record the audio output.
- Create a short composition (ca. 2 min.) by compiling these audio files in DP6, or some other multi-track audio editor software. (You can use the same program for assembling and mixing that you used in your 1st project.)

Digital Performer (DP6) — MIDI/audio project using analog-style synthesizer plug-ins

There are several analog synthesizer plug-ins available in the lab to use with DP6.

- DP6, itself has some new analog modeling synths that you can use:
  -- check out Bassline, Modulo, and Polysynth
- BuzZer, Additive Synth, Crystal, (and probably others)
- plug-in versions of some of my ProjectSynthPack modules
- last, but definitely not least are all the many analog modeling synths in the Pluggo collection (see below for examples of appropriate Pluggo plug-ins)

  [additive heaven, AdditiveSynth, analogue drums, analogue percussion, bassline, Comber, deep bass, filtered drums, FilterTaps, Flange-o-tron, flying waves, harmonic dreamz, Harmonic Filter, HF Ring Mod, LFO, Mangle Filter, ModWheelMayhem, ModWheelScales, Monstercrunch, Mouse Mod, Moving Filters, moving waves, Multi-Filter, Multi-Filter[S], NoiseySynth2, Noyzckippr, PluggoFuzz, Resonation, Resosweep, Ring Modulator, shape synth, Sine Bank, Spectral Filter, Step Sequencer, Tremellow, Vibrato Cauldron, Waveshaper, wavy waves]

Garage Band

- You can use Garage Band ... but use only the analog-style software synth tracks (i.e. no "real" tracks) and avoid the pre-programmed loops.

Other software:

- If you have access to other analog-style synths (software or hardware) you are welcome to use them, but check with me first. Many of the software packages listed for mini-project 1 will also be appropriate for this project, provided you have appropriate synth plug-ins (Logic & Reason are particularly good choices).

Final Project Guidelines: **due: by 5:00 p.m. Thursday, March 18**

Your Final Project should represent your best efforts to create an interesting piece of electronic music/computer music, given the skills and knowledge you've acquired in this and other courses. Set interesting and challenging (but attainable!) goals for yourself. Ask for ideas, help, guidance, extra reference materials, etc., from professor or lab assistants. Your final project will be a 2-4 minute original composition. Imagination and original sounds are more important than length.

Final Project — a step-by-step approach

A. New Sounds — create some sounds for your project using FM and/or granular synthesis (and/or Csound?!?) using one or more of the following methods. You do not need to use all of these, but you do need to use at least one of the following.

1. Use the FM synthesizers in *E-A_MusicPrimer* to record some digital audio files.
2. Use Max/MSP to create some granular synthesized digital audio files in real-time.
3. Use thOnk (requires Mac "Classic" OS) to generate some digital audio files (non-real-time).
4. Use software synths plug-ins in DP6 (Crystal, etc.)
5. Pluggo plug-ins in DP6 (especially the ones in bold)

[big ben bell, Convolver, **fm 4-op, FM-Synth, Granular-to-Go**, Randomizer, Slice-n-Dice, Stutterer, Vocoder 10-band, Vocoder 16-band]

B. "Old" Sounds — create some sounds using any of the techniques we've covered earlier in the quarter

1. Musique Concrète-style digital audio manipulation. (*Pierre's Playhouse, SoundHack*, etc.) from project 1
2. Additive and Subtractive synthesis (*E-A_MusicPrimer*, various analog modeling synths, etc.) from project 2

C. Composition

Use DP6 or other suitable software to assemble your project (import your newly created audio files; record and edit your MIDI data; record digital audio; combine, mix, EQ, add effects, and blend your audio. Do a "Bounce to Disk" (or mix) c the completed project.

<u>Post-Production Guidelines</u> — on all lab projects, follow these basic post production steps.

Once you have assembled and mixed your project, you should create a stereo file of it ("Bounce to Disk" or "Export" are th common menu commands.) PLEASE! Be sure to listen to your "bounce" file to make sure it sounds the way you want it to. If appropriate, "normalize" the resulting file to achieve maximum output. In the lab this can be done in DP6 or Peak. Acceptable formats for submission include and AIFF, WAV, MP3, or MP4 files (via Data CD or Blackboard™) or Audio CD. Then complete the project by doing one of the following:

- **Option 1:** Blackboard uploading.
  Put all your session files and your clearly labled "bounce" file in a folder with your name on it. Create a compressed (or archived) version of it so that it ends with the ".zip" file type identifier. Go to the class Blackboard™ site; to access the Digital Dropbox page and upload "zipped" file. (Only attempt this with a high-speed connection. Even so it will take several minutes to complete. Be patient!) Once it has been uploaded, be sure to hit the "send" button.

- **Option 2**: CD submission

  NOTE: If you are working in the Midi Lab, DO NOT try to burn CD from files on the server! After labeling it clearly and testing that it is good, you should turn it in to me. Hand it to me personally, or slide it under my office door.

  - **Option 2a:** Create a single data CD
    Create a data CD with all your session files PLUS a clearly labled copy of your stereo "bounce" file. You can use the Mac's OS finder features or Adaptec Toast (under the Format menu select both "Mac files and folders" and "Audio files.)

  - **Option 2b:** Create two separate CDs:
    1) and audio CD of your stereo "bounce" file ... PLUS ... 2) a data CD with all your session files.

DO NOT send me your project as an email attachment!!!!!

Introduction

Electronic Music — or — Electro-Acoustic Music

(What exactly are we dealing with in this class?)

Electro-Acoustic Music "Tree"

- Electro-Acoustic Music
  - tape music
    - "pure" electronic
      - analog synthesis
      - digital synthesis (computer)
    - musique concrète
  - live/electronic music
    - live elec. perf.
    - tape & instr.
    - live elec. & instr.

Musique Concrète — Pierre Schaeffer, Études (1948) (Library CD# 1347, vol. I)

play work in class

Etude aux chemins de fer ("pathway" or (ribbons) of steel") radio broadcast (1948)
Paris railyard sounds recorded on phonograph disks; Locked-groove recording ("record skip")

Traditional Composition:
1) abstract idea in composer's imagination 2) notation 3) realization by performers
Musique Concrète:
1) "real" (concrete) acoustic sounds 2) manipulation to create work 3) electromechanical playback

Used and developed techniques that foreshadow tape editing and manipulation:
1. Locked-groove —> Tape loops
2. Editing & juxtaposing —> Tape cutting and splicing
3. Speed Change 4. Direction Change

Used phonograph disks until 1951, when tape recorders using magnetic tape took over. Schaeffer then had to "relearn" all his techniques when he adopted the new technology. Among his later accomplishments was inventing devices made tape technology even more flexible for use in creating musique concrète (for example: a device that could quickly and easily "transpose" recordings, essentially a sort of forerunner to the Mellotron.)

From 1948 onward, beginning with Pierre Schaeffer, electronic music has a well-documented history and literature. In addition to championing new esthetic concepts, this new type of music is distinguished from all previous music — including earlier live performance on early electronic instruments — by one critical fact. The new genre of music has an absolutely fixed form, and is not subject to performer interpretation.

Basic Properties of Sound

Sound is a waveform phenomenon and exhibits eight distinct (8) wave form properties.

1) frequency 2) amplitude 3) velocity 4) wavelength
5) waveform 6) phase 7) harmonic content 8) envelope

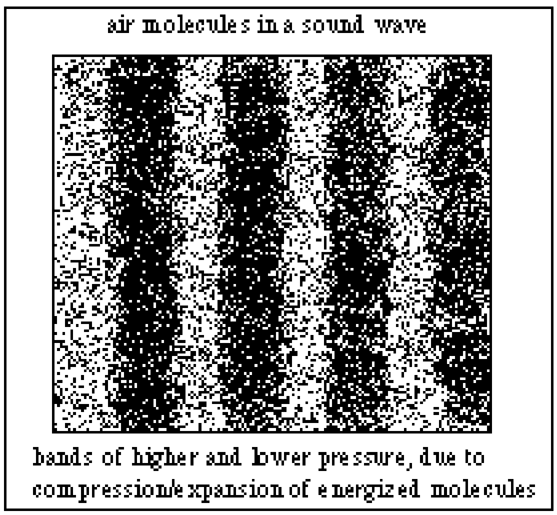
Sound waves require an "elastic" medium to pass through. They are longitudinal, because the motion of the particles is parallel to the direction of propagation of the wave.

[Note: All electromagnetic waves are transverse waves; they do not require a medium and can travel in a vacuum. In transverse waves oscillations occur perpendicular to the direction of the wave. (example: shaking a rope w/ one end fixed to a post)]

Frequency: unit of measure is Hertz (Hz) = cycles / second [= **pitch**]

Amplitude: unit of measure is decibels (db) [= **volume** or **loudness** or **dynamics**]

Note: Sound waves are the result of increases and decreases of atmospheric pressure the amount of those change; or the degree of compression and decompression determines amplitude. The greater the compression the more amplitude.



Velocity: sound travels at 1130 ft./sec … through air … at 70º … at sea level
…as temperature goes up — speed goes up
…as density goes up — speed goes up

Notes:
1) molecules do NOT move far
2) velocity measures the speed of the "chain reaction" of molecules transferring the energy to their neighboring molecules
3) Electron movement replaces molecular movement in electronic signals
4) Velocity accounts for stereo/binaural hearing, echo, and reverb

Wavelength: affects hearing in important way — accounts for bending properties
- •longer wavelengths can bend around obstacles
- •shorter wavelengths simply reflect off of surfaces

velocity, frequency, and wavelength are related...

•**Formula:** velocity = frequency x wavelength

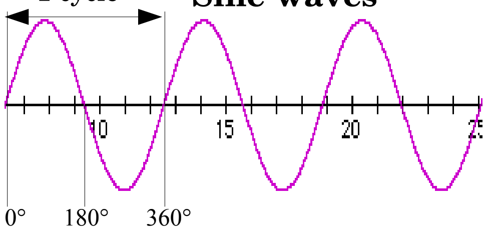
Shape: 4 main types of simple waveforms [=**timbre**] *(See diagrams below)*

•sine •triangle •sawtooth (or ramp) •rectangle (…pulse…square)

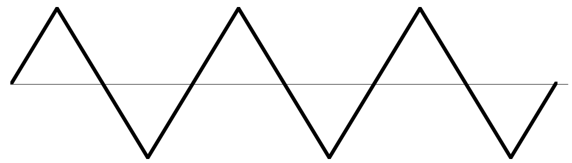
E-A_MusicPrimer Demos: Waveforms (>*Music 413.app*)

K2000 Demos

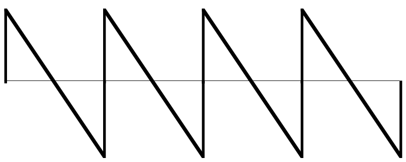
Sine waves



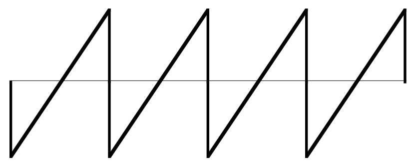
Triangle waves



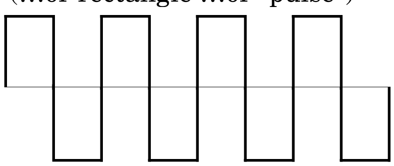
Sawtooth waves
(negative going)



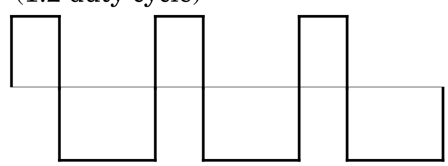
Sawtooth waves
(positive going)



Square waves
(…or rectangle …or "pulse")



Rectangle waves
(1:2 duty cycle)

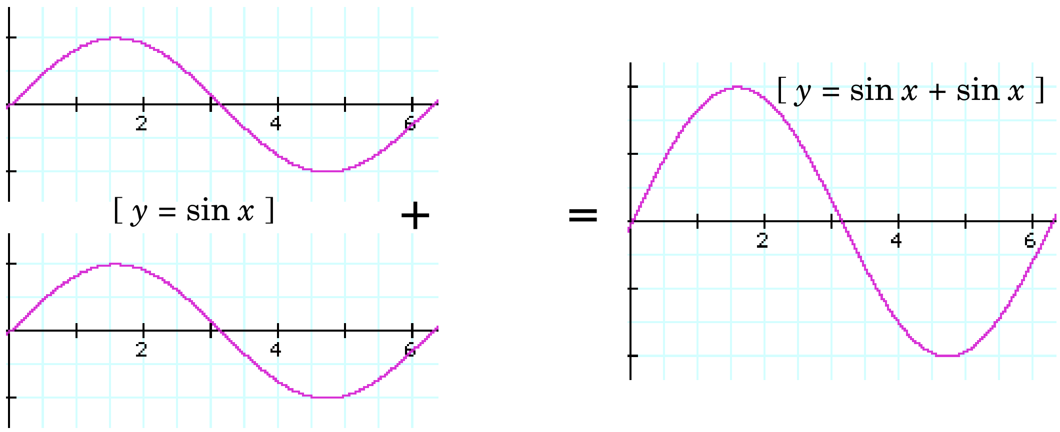


Schedule Lab Help Sessions !

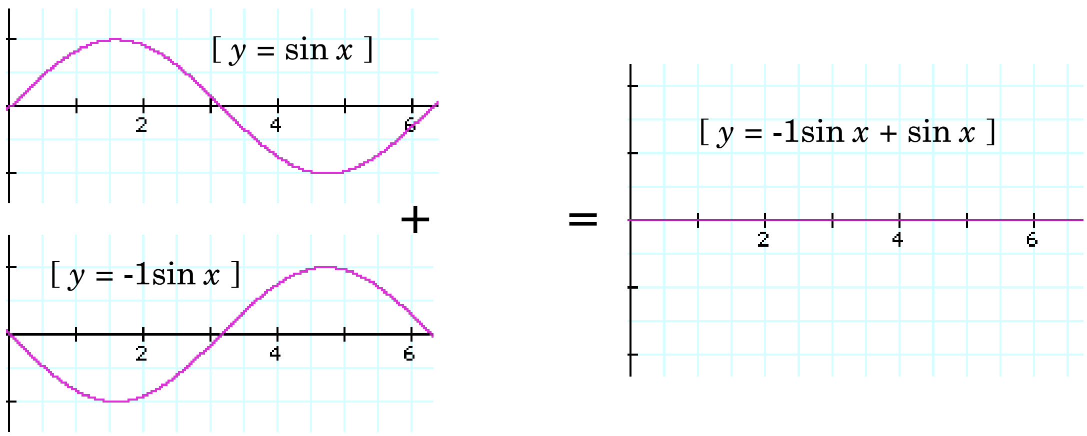
Phase: Constructive and Destructive Interference *(See diagrams below)*

Describes the phenomenon that 2 waves with identical frequency, shape, amplitude, & wavelength, may not appear identical as they pass a specific point is space at a given time.

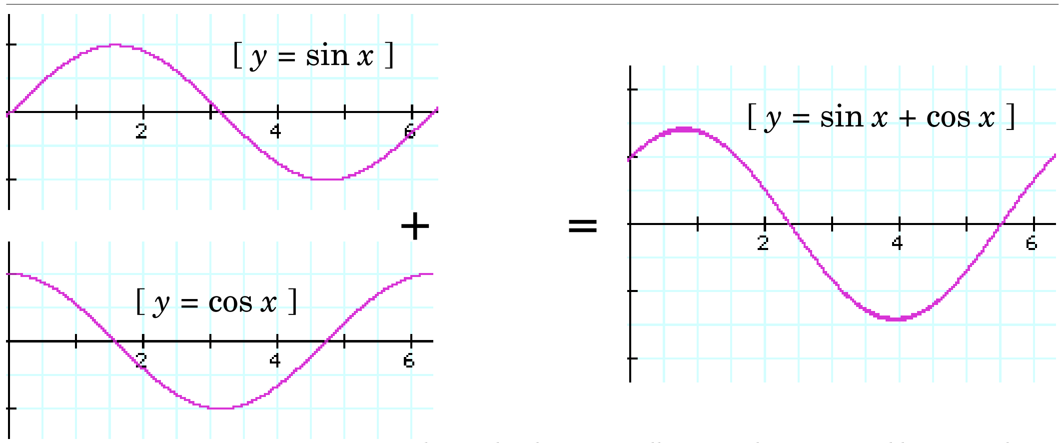
•measured in "degrees out-of-phase" from 0° —> 360° (which is back "in-phase")



Constructive Interference: the amplitudes of in-phase waves add when they are mixed.



Destructive Interference: waves of equal amplitude cancel completely when mixed 180° out-of-phase.



Constructive & Destructive Interference: the amplitudes of partially out-of-phase waves add in some places and subtract in other places when they are mixed..

diagrams above produced using the "Graphing Calculator" — a similar software application called "Grapher" is included in current version of the Macintosh OS X.

E-A_MusicPrimer Demos: DestructInt & ConstructInt (>*Music 413.app*)

Harmonics: waveforms related by integral multiple of another waveform

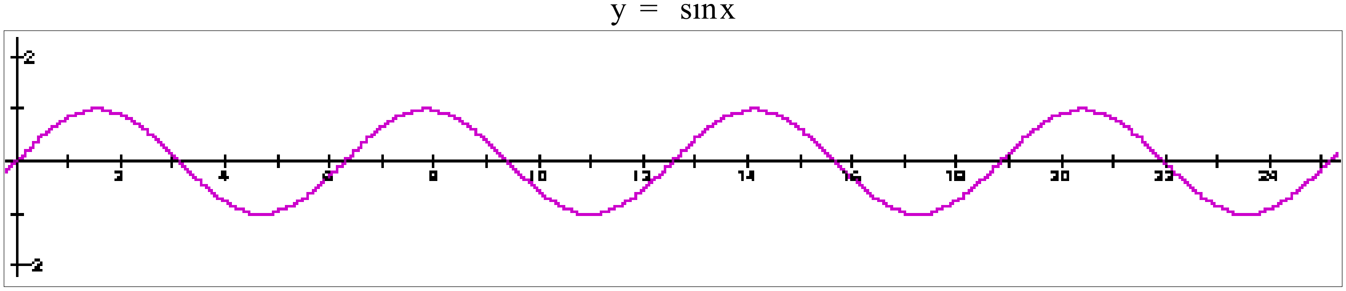
2000 Hz	x4 =	4th harmonic	[2 octaves higher … or one 8ve higher than 2nd harm.]
1500 Hz	x3 =	3rd harmonic	[octave + P5th]
1000 Hz	x2 =	2nd harmonic	[one octave higher]
500 Hz	x1 =	1st harmonic	[fundamental]

>>the concept of harmonic relationship is used extensively in "additive sythesis"

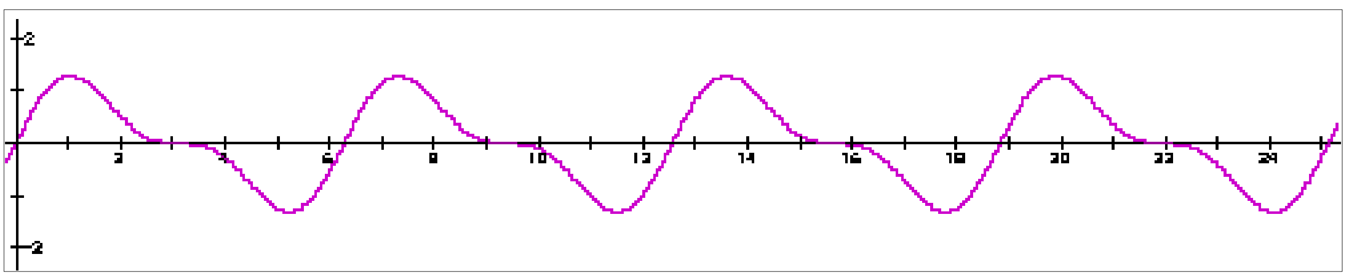
Additive Synthesis

Graphic Representations of Additive synthesis using "Graphic Calculator"

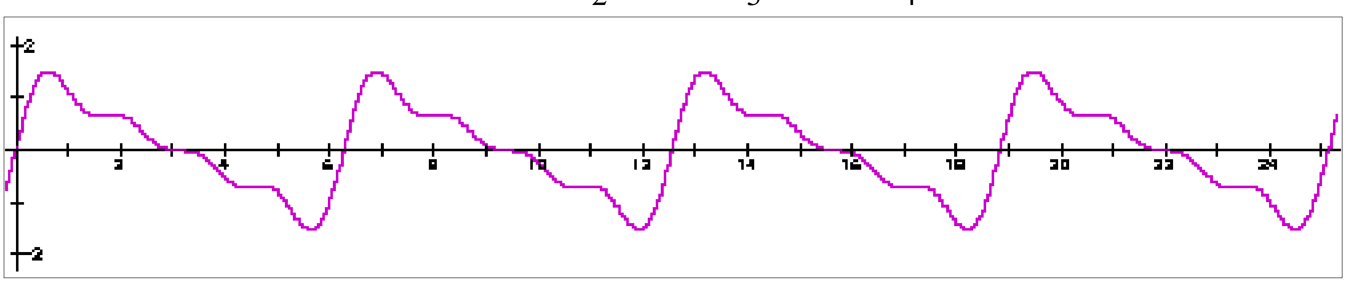
$$y = \sin x$$



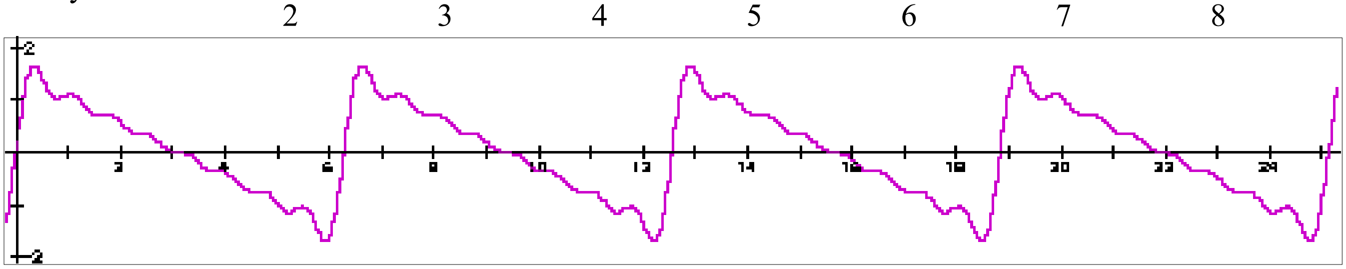
$$y = \sin x + \frac{(\sin 2x)}{2}$$



$$y = \sin x + \frac{(\sin 2x)}{2} + \frac{(\sin 3x)}{3} + \frac{(\sin 4x)}{4}$$

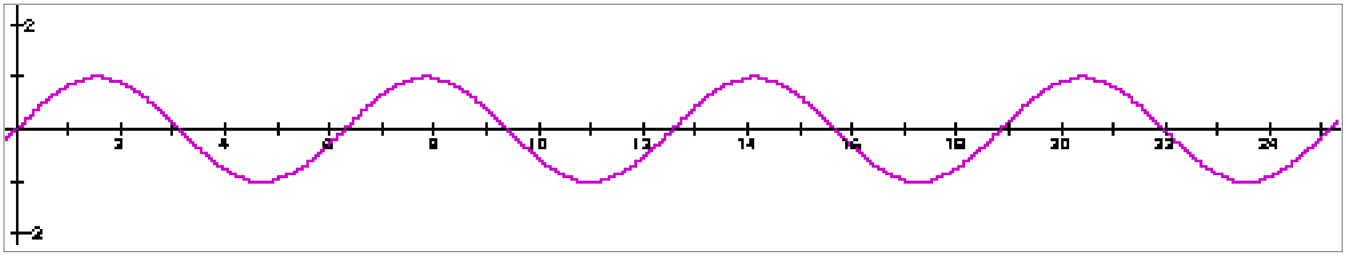


$$y = \sin x + \frac{(\sin 2x)}{2} + \frac{(\sin 3x)}{3} + \frac{(\sin 4x)}{4} + \frac{(\sin 5x)}{5} + \frac{(\sin 6x)}{6} + \frac{(\sin 7x)}{7} + \frac{(\sin 8x)}{8}$$

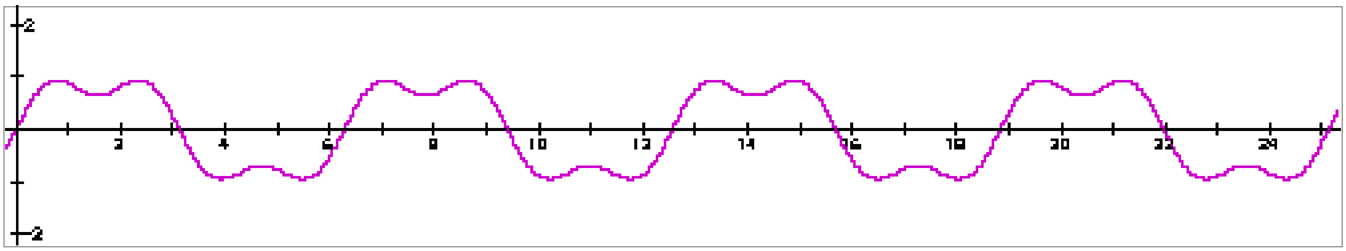


E-A_MusicPrimer Demos: AddSynth1.Saw (>*Music 413.app*)

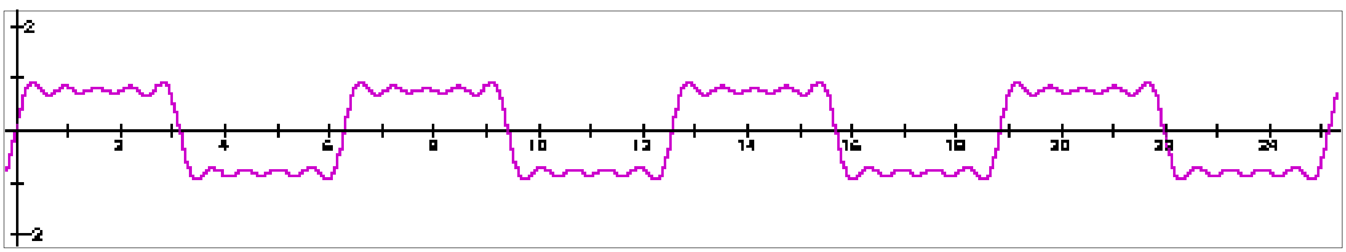
$$y = \sin x$$



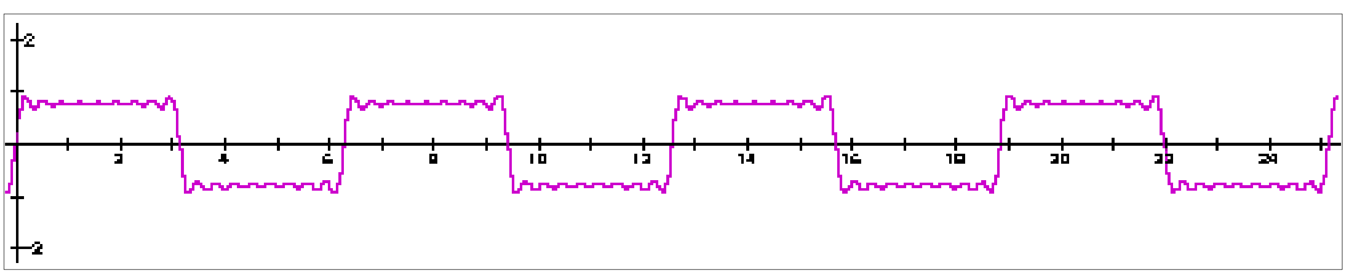
$$y = \sin x + \frac{(\sin 3x)}{3}$$



$$y = \sin x + \frac{(\sin 3x)}{3} + \frac{(\sin 5x)}{5} + \frac{(\sin 7x)}{7} + \frac{(\sin 9x)}{9}$$



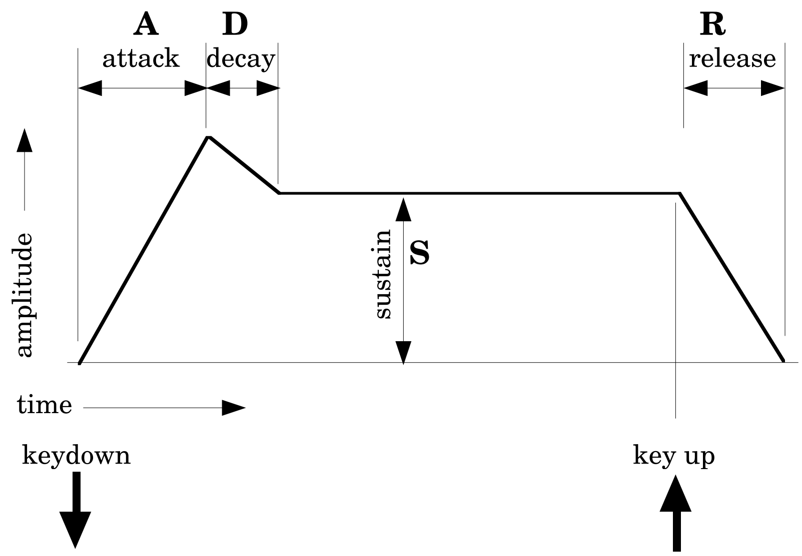
$$= \sin x + \frac{(\sin 3x)}{3} + \frac{(\sin 5x)}{5} + \frac{(\sin 7x)}{7} + \frac{(\sin 9x)}{9} + \frac{(\sin 11x)}{11} + \frac{(\sin 13x)}{13} + \frac{(\sin 15x)}{15} + \frac{(\sin 17x)}{17}$$



E-A_MusicPrimer Demos: AddSynth2.SQ (>*Music 413.app*)

Envelope: describes the way a waveform varies in intensity over time

Simple ADSR Envelope Diagram



Basic Principles of Human Hearing & Perception of Sound

Non-Linear Perception

K2000 Demos

Frequency Response: (reliable frequency range) ....

- ca.. 15 Hz — 20kHz (…or 20,000 Hz) >>roughly 10.5 octaves
- younger ears can hear higher frequencies
- older ears have a lower limit on high frequencies

E-A_MusicPrimer Demos: SubAudio & VHF-sine (>*Music 413.app*)

- octave = 2:1 ratio of frequencies … therefore the higher the frequency the greater the change required for equal perception of that change to occur.

Dynamic Range: one trillion-to-one ratio of perceivable energy levels (from threshold to pain)

- measurement unit = decibel (db) [after Alexander Graham Bell] allows only relative comparisons, does not define an absolute scale.
  >>db = smallest perceivable loudness difference [spl = sound pressure level]
- equal changes in perceivable loudness require increasingly greater amounts of energy as loudness level increase.
- logarithmic ratio: $\text{db} = 10 \log P_1/P_2 = 10 \log spl_1^2/spl_2^2 = 10 \log (spl_1/spl_2)^2 = 20 \log spl_1/spl_2$
- for amplifiers 2x power = ca. 3db power gain
- for spl 2:1 = 6 db ; 3:1 = 10 db ; 10:1 = 20 db ; 100:1 = 40 db ; 1000:1 = 60 db ; etc.
- Middle C played in air as loudly as the ear can tolerate results in density changes about 0.02 percent above and below the density of quiet air; the maximum displacement of air molecules is less than one-tenth of a millimeter (0.004 in)

Non-Linear Frequency Response of the Human Ear:

- extreme sensitivity from 1000 Hz — 4000 Hz
- thus…equal power at different frequencies may produce different perceptions of loudness

- examples from orchestra/band (i.e. piccolo, bass drum, tuba, etc.)

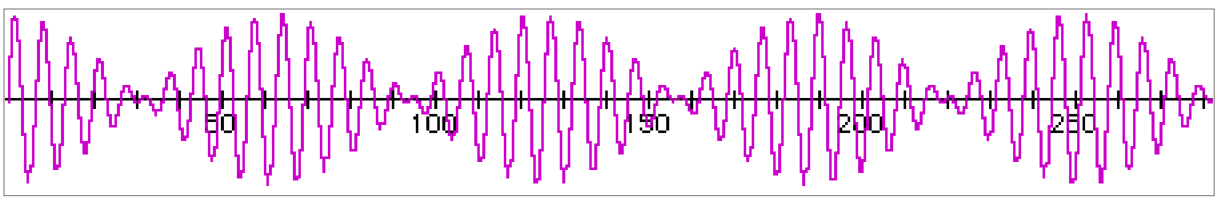
- ”edge” harmonics (above the 7th harmonic) usually fall within this register; important for identifying instrument timbres.

- trumpet: no difference in amplitude of “fundamental” (1st harmonic) between soft and loud notes >> amplitude changes for the upper harmonics cause entire difference in perception.

Beats: caused when 2 tones are almost of equal pitch or frequency

(i.e. they are less than 40Hz to 50Hz apart)............................... **K2000 demos**

[Graphing Calculator formula: $y=10\ \sin x + 10\ \sin(.9x)$]



E-A_MusicPrimer Demos: beats (>*Music 413.app*)

Combination Tones: caused when (loud) 2 tones differ by more than 50 Hz

>> summation & difference tones formed [K2000 “<u>BeatsDemo</u>” with higher pitch]

Masking: results when one (loud) sound prevents a softer sound from being heard.

E-A_MusicPrimer Demos: Masking (>*Music 413.app*)

K2000 Demos

SineWaveDemo	220 BeatsDemo	240 SineFMDemo	260 SawAmpEG
201 TriangleDemo	221 MaskingDemo	241 SineAMDemo	261 Saw-LPFilter
202 RisingSawDemo	222 MaskingDemo2	242 SawAMDemo	262 SawFilterEG1
203 FallingSawDemo	223 LFO Sawtooth	243 SawTMDemo	263 AdditiveSine
204 SquareWavDemo	224 LFO Square	244 NoiseTMDemo1	264 AdditiveOrgan
205 Saw-LPFilter	225 LFO SineWave	245 NoiseTM-kbd	265 SawFreqEG
206 Squar-LPFilter	226 VHF Sine Wave	246	266 SawKybdFilter
207 Noise-LPFilter	227 VHF Sawtooth	247	267 NoizKybdFilter
208 Noise-HPFilter	228	248	268 SawKybd+Amp
209 Noise-BPFilter	229	249	269 SawKybd-Amp
210 Noise-BRFilter	230	250	270

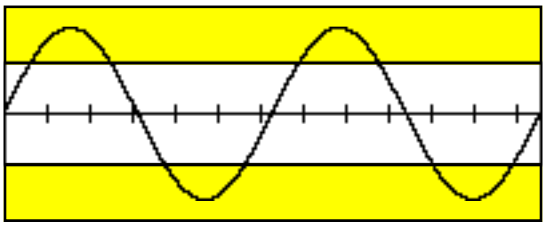
Some Basic Audio Recording Terms & Concepts

Transducers: microphones, speakers, tape recorders, electric guitar pick-up, phono cartridge
- transfers energy from one form to another [i.e. Microphone: physical sound –> electrical energy]

>**demo:** "microphone as speaker"

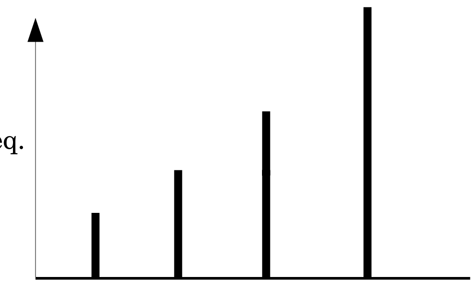
Distortion: **E-A_MusicPrimer Demos: Clipping (>*Music 413.app*)**

Harmonic Distortion (…or Clipping)
- amplitude related problem…affects tone color



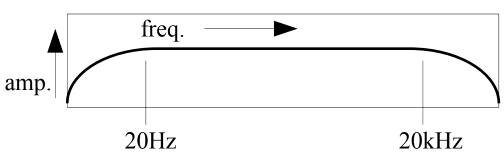
- If system can't handle the amplitude of a sine wave, it "clips" away what won't fit in its dynamic window (represented by the shaded area). Sine waves begin to look (and <u>sound</u>!) like square waves
- rather common …sometimes desirable (especially among guitarists!)

Intermodulation (frequency related problem)
- Frequencies "a" and "b" interact to form <u>audible</u> summation and difference tones

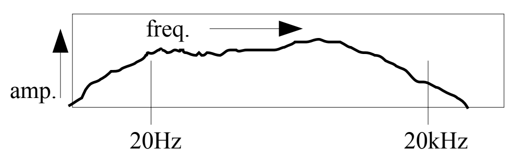


- adds new pitches that where not present
- rare, uncommon; but REALLY problematic when it does occur

Frequency Response:
- The ability to transmit all desired frequencies at the original amplitude. In other words how accurately does the system transmit (or reproduce) amplitude over its given frequency spectrum.
- The "ideal" frequency response is <u>FLAT</u> —between 20Hz and 20kHz (…not usually a complementary description in the music world, but it is in the audio technician's world.)

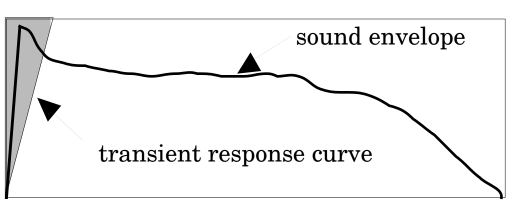


"ideal"



"real world" example

Transient response:
- The ability to transmit (…or track) sudden changes of amplitude. (Also known as "slew" rate.)



 = part of attack envelope missed by "theoretically" slow transient response

History of "Noise" Esthetics that Lead to Electronic Music & Musique Concrète

purpose: …to try to discover the aesthetic origins of electronic music and musique concrète

There are several threads running through the course of Western Music which become entwined in the early part of this century into an esthetic responsible for the birth and continued interest in electronic music. Today we will examine some of the main strands.

I. The use of percussion instruments — usually borrowed originally from another culture, often used to achieve an exotic effect.

II. The use of ordinary (standard Western orchestral) instruments to depict real world sonic events, not normally associated with the more abstract qualities of music expression. —> "Programmatic Music"

III. Next…logically…is the use of actual noisemakers themselves to lend authenticity to these same types of "programmatic" compositions

IV. Later…the influence of the Industrial Revolution.
 > Ferruccio Busoni published "Sketch of a New Esthetic in Music" (1907) which discusses electronic and other sound sources in futuristic music — had profound impact on Edgard Varèse (whom we will listen to later today)
 > Other Italian "futurists" Ballila Pratella and Luigi Russolo began investigating, classifying and creating "noise" instruments. >>championed a style of music labeled "Futurism" (also occasionally known as Bruitism and sometimes even, ironically Primitivism). They "appealed for songs of the factories, warships, motor cars, and aeroplanes …(of) machine guns, sirens, and steam whistles."

V. Last, but not least, the constant striving of Western composers, usually in collaboration with interested performers, to extend the know limits of a given instrument, or a given ensemble of instruments. Indeed…this idea has the longest history going all the way back to the late-1500s and early 1600s when composers first began specifying exact instrumentation and including "dynamic" (or loudness) markings in their scores. A particularly notable example from the 1600s involved and composer name Claudio Monteverdi who first called for his bowed string players to put down their bows and pluck their strings. This was such a novel idea back then that he felt compelled to write a little paragraph in the score explaining exactly how the player should perform the passage. Nowadays a composer need only put the abbreviation "PIZZ." in the score and performers instantly know what to do.

It would be difficult to present this phenomenon in detail today in class, since the changes often take place so slowly by gradual steps that considerable historical context is often required to fully appreciate it. Also … because of the gradual, evolving nature of this area of development, many steps along the way are not very dramatic when taken by themselves.

 > A few highlights along the way will suffice: After pizzicato was introduced into string literature, many other unusual techniques were developed to expand the sonic pallet of the string section: Harmonic (artificial and natural); col legno (hitting the string with the wood of the bow); hitting the body of the instrument as if it were a percussion instrument, etc.

Throughout the centuries, refinements in instrument design and virtuosos performance lead to ranges and and technical demands being extended — sometimes dramatically so. Single instruments were expanded to whole families of instruments by the 19th century [i.e. piccolo, flute, alto flute, bass flute ... E♭ soprano clarinet, B♭ clarinet, E♭ alto clarinet, B♭ bass clarinet, E♭ contrabass clarinet, etc.] In addition, a few new totally new instruments, such as the saxophone were added.

Early in the development of Western music there were a couple of specific types of composition which did the most to advance the ideas we are discussing today … all of these types fall under the general category of "Program Music" or "Programmatic Music."

I. The "Battle" composition *(the previously mentioned work by Monteverdi was one)*
II. The "Storm" composition
III. The "Tumultuous Celebration" composition

Listening List for Lecture 4: Origins of Electronic Music Esthetics

1. Haydn: *Symphony N° 100 in G* (i.e. "Military") — (1794)

   "Battle" composition — using so-called "Turkish" style, with percussion instruments borrowed from Turkish military bands. (Mozart and many other Classical composers through Beethoven borrowed from this "exotic style of music.

   …from a contemporary review printed in the *London Morning Chronicle* after the world premiere:

   > It is the advancing battle; and the march of men, the sounding of the charge, the thundering onset, the clash of arms, the groans of the wounded, and what may be called the hellish roar of war increase to climax of horrid sublimity! which , if other can conceive, (Haydn) alone can execute; at least he alone hitherto has effected these wonders.
   >
   > …absolute shouts of applause. Encore! encore! encore! resounded from every seat: the Ladies themselves could not forebear.

2. Beethoven: *Symphony N° 9 in d*, op. 125 — (1824)

   "Military" music — using so-called "Turkish" style

3. Beethoven: *Symphony N° 6 in F*, op. 68 — (1808)

   "Storm" composition >> using standard classical symphonic scoring and traditional music forms

4. Berlioz: *Symphonie Fantastique* — (1830)

   "Storm" composition >> using greatly expanded symphonic scoring and with increased emphasis on realism at the expense of traditional music forms

5. Tchaikovsky: *Overture, 1812* — (1880)

   "Tumultuous Celebration" composition >> written (some 60 years after the fact) to celebrate the retreat of Napoleon from Moscow

   >>note the use of noise makers themselves to add "realism" to the celebration. (cannons, church bells)

6. Charles Ives: Decoration Day (from *Symphony of Holidays*) — (1915)

   "Tumultuous Celebration" composition

7. Eric Satie: *Parade* — (1917) [theatre work; collaborators included Cocteau & Picasso]

   >>early example of "Futurism" uses mostly a traditional pit orchestra playing fairly conventional music, with a few notable exceptions (mostly from "noisemaker" additions to the percussion section):
   #1 typewriter & cap gun #2 dynamo #3 airplane propeller #4 boat whistle

   George Antheil: *Ballet Mechanique* — (1924) [written for film by Fernand Leger]
   more fully developed "Futurism" >> written for percussion ensemble plus (original version called for) >>10 grand pianos, sirens, airplane propellers, and wade range of more typical percussion

8. Edgard Varèse: *Intègrales* — (1924) [also wrote *Ionisation* for percussion ensemble]

   Fully developed "Futurism" — uses orchestra of winds, brass, & percussion. not only does he use percussion noisemakers, but virtually all of the instruments are treated in a similar manner. There is no real melody to speak of, no real harmonic progression, — just pure sound or timbre – for it own sake – animated by rhythm. In other words even the ordinary instruments have unusual functions to fulfill >>>> in effect, Varèse has invented "electronic music" before the technology has been developed.

9. Henry Cowell: *The Banshee* — (1924)

   >> unusual instrumental techniques<< "…listen now to Henry, himself, tell you about his piece"

10. John Cage: *Sonatas and Interludes* — (1948-49)

    "prepared piano" >> unusual instrumental technique <<
    >>>objects are placed so they come in contact with the string as the hammer is struck

    No. 5 & No. 2 >> from a live concert recording ...one pianist on one piano in real time!!!!!!!

Electric Sound, by Joel Chadabe — Selected Readings

Charles Ives, Satie, *(Parade)* Antheil *(Ballet Mechanique)*and John Cage *(Sonatas & Interludes)* — pp 22-26

Varèse — pp 58-59

Pierre Schaeffer & Pierre Henry — pp 26-28

Xenakis — pp 34-35

Luening & Ussachevsky — pp 44-45

Berio, Pousseur – pp 47-51

John Cage *(Williams Mix)* – pp 55-58

Varese and Xenakis (1958 Brussels World's Fair) pp 60-62

Le Caine pp 13-15 & p 64

Lucier – pp 75-76

Reich – p 77

Listing Guide: Exam 1

<u>Schrader*</u>

Pierre Schaeffer: *Études* [1948] [dur. 9'10"].. p. 10
- •composition accomplished by technological process
- •"unlimited" reproduction … exactly the same each time
- •replay not dependent on human performers
- •begins with real or "concrète" sounds
- •all *Études* use "direct to wax" technology [switched to tape (grudgingly!) in 1951]

Étude aux chemins de fer >><u>sound sources</u>: trains sounds [dur. 3']

Étude aux tourniquets >><u>sound sources</u>: whistling tops, xylophone, bells, kalimba [dur. 2']

Étude aux casseroles (pathetique) [dur. 4'10"]
>><u>sound sources</u>: spinning pan lids, harmonica, accordion, chanting Tibetan monks, Balinese, music, chugging canal boats, piano, human speech (French), coughing

Otto Luening: *Low Speed* [1952] [dur. 3'50"]..p. 45
>> <u>sound source</u>: flute sounds •low-speed playback; lots of tape feedback delay

Vladamir Ussachevsky: *Sonic Contours* [1952] [dur. 7'30"]... p. 32
>> <u>sound sources</u>: mostly piano sounds (w/keyboard), some human speech
- •significant use of direction change, often in combination with speed change; use of editing to control envelope of sound
- •also notable >>>tape feedback delay when the piano

Vladamir Ussachevsky: *Piece for Tape Recorder* [1956] [dur. 5'50"]............................ pp. 36-37
>> <u>sound sources</u>: 1 gong stroke, 1 kettle drum note, 1 jet plane sample, organ, piano (w/keyboard), 4 sine wave oscillators, tape recorder switch
- •lots of speed and direction change; some use of editing to control envelope of sound
- •also notable >>>prominent use of tape feedback delay ca. 1'40" — 2'00" with lively piano music

Hugh LeCaine: *Dripsody* [1955] [dur. 2'00"]... p. 32
>> <u>sound source</u>: ONE water drop recorded and duplicated <u>25</u> times! (Schrader is *wrong* here!)
- •uses a custom-made loop player (forerunner to Mellotron) w/ variable speed control

Iannis Xenakis: *Concrete PH-II* [1958] [dur. 2'50"]...p. 54
>> <u>sound source</u>: burning charcoal •high–speed playback
- •Philips Pavilion, 1958 Brussels World's Fair; 425 loudspeakers (avant garde "Muzak" ?!?)

Edgard Varèse: *Poem Electronique* [1958] [dur. 8'15"] ***Bonus Listening***.........................pp. 104-105
>> <u>sound sources</u>: bells, percussion, voices, sirens, jet planes, etc. (plus electronic sounds)
- •Philips Pavilion, 1958 Brussels World's Fair; 425 loudspeakers (avant garde "Muzak" ?!?)
- • see also Chadabe, pp. 58-62

Luciano Berio: *Thema Omaggio a Joyce* ("Homage to Joyce") [1958] [dur. 6'30"].................... p. 49
>> <u>sound source</u>: human speech [composer's wife noted soprano,Cathy Berberian] reading from James Joyce's Ulysses, beginning of chapter 11.
- •layers vocal tracks (mixed to stereo) — with speed change — to create "chorus" effects
- •<u>also</u>: bl•bl•bl•bl•blo•blo•blo•blo•bloo•bloo•bloo•bloo•bloo•bloom•bloom•bloom

---

***Schrader:** *Introduction to Electro-Acoustic Music,* Barry Schrader, Prentice-Hall, 1982.

This reading material is on reserve in two formats:

1) the entire book [call # ML1092 .S35]

2) a packet of exerpts containing just the pages listed here pertaining specifically to the listening (<u>note</u>: this packet has been formatted to make it easier to photocopy it for study purposes.)

Schrader

John Cage: *Fontana Mix* [1958] (excerpts) [edited dur. 5'20"]...p. 28

>See also discussion of *Williams Mix*, p. 25 [plus discussion in Chadabe pp. 54-58]
•vast array of sound sources utilized; tremendous number of edits (cut and splice)
•different shaped edits were used to create envelope (using full-track mono format)
•4 full-track mono machines used for mixing to 1 channel of stereo…(total: 8 tracks)
•chance operations derived from *I-Ching* (Chinese *Book of Changes*)…coin toss, etc.

> I wished, when I first used chance operations, to make a music in which I would not express my feelings, or my ideas, but in which the sounds themselves would change me…. So that rather than becoming a more refined musician, I would become more and more open to the various possibilities of sound. This has actually happened, so that my preference…is not any of my music, and not any of the music of other composers, but rather the sounds and noises of everyday life. — *J.C.*

Iannis Xenakis: *Orient-Occident III* [1959-60] [dur. 10'58"] ***Bonus Listening***.......................p. 53

>> sound sources: wide variety of sound sources: especially cello bow on gongs, metal rods, cardboard, etc. also: recordings of his orchestral music; signals from ionosphere converted to audio
•compendium of tape manipulation techniques: notable use of speed change; overlapping & cross-fading

Toru Takemitsu: *Water Music* [1960] [dur. 9'05"]...p. 47

>> sound source: water — lots of *different* water sounds; from individual droplets to large splashes
•recorded drops of water falling into/onto many surfaces; creates variety of timbres & envelope shapes
>> some surfaces (e.g. a bucket) create reverberation effects
•reversed playback; variable speed playback; tape loops
•very "oriental" sense of time & action -- long silent pauses [Japanese "No" drama]

Tod Dockstader: *Luna Park* [1961] [dur. 3'25"]..p. 45

>> sound sources: human laughter/speech (mostly); plus some electronic keyboard sounds
•high-speed playback; lots of tape feedback delay; tape loops

Steve Reich: *Come Out* [1966] (excerpts) [edited dur. 4'12"]..p. 19

•2 identical tape loops drifting out-of-phase (due to slight differences in transport speed)
•result recorded on a separate machine
•create 2 more identical loops and repeat phase drift recording (now 4 "images")
•create 2 more identical loops and repeat phase drift recording (now 8 "images")
>>sound source: human speech, by civil rights activist Daniel Hamm, victim of police beating (Harlem riots; 1964) explaining what he had to do in order to get medical treatment.

Alvin Lucier: *I am Sitting in a Room* [1980; orig. 1970] (excerpts) [edited dur. 7'54"] .... p. 54

>> sound source: human speech; the composer reads a speech into a microphone
•resonant frequency reinforcement (acoustic feedback)
tape loop of recorded speech is fed back into the room via loudspeakers and re-recorded on a second machine [15 generations in original 1970 version; 32(!) in 1980 version, a duration is 45'21"]. With each pass, successively more reinforcement of the room's natural resonant frequencies occurs until they completely obliterate the speech.
•no tape manipulation is used, except for loop splicing

Mark Phillips: *Samnyama* [1980] [dur. 3'55"]

>> sound sources: all sounds produced on an old upright piano (mostly played "inside") also use of "prepared" piano (layers of masking tape on strings)
•much use of tape speed manipulation [often to create "imitative counterpoint"]
•NO external reverb device used...all acoustic "reverb" sounds are from the piano itself.

John Lennon: *Revolution 9* [1968] (brief excerpt)

Its inclusion on the Beatles double album (i.e. *White Album)* made it perhaps the most widely known example of musique concrète.
>> wide variety of sound sources and manipulation techniques (seems most influenced by John Cage)
•human speech, piano, existing recording of music, prominent use of direction change

Music 413/513 Exam I Study Guide

Listening: (total of 40 points)

Be sure you can identify the following sound sources with the appropriate composition(s).
burning charcoal / one water drop / many water sounds / laughter /
human speech (English) / human speech (French) / coughing / text by James Joyce /
text by civil rights activist / Paris railyard sounds / spinning pot lids / flute / piano /
harmonica / chanting monks / extreme diversity of sound sources /
cello bow on various objects (gongs, metal rods, etc.) /

Be sure you can identify the following techniques with the appropriate composition(s).
tape loops / tape delay (feed back) / reversed playback direction
low-speed playback / high-speed playback / direct-to-wax recording /
tape loops drifting out-of-phase / extreme number of tape splices /
chance ordering of events (I Ching) / resonant frequency reinforcement
variable-speed multi-loop player

Bonus Listening:
Be able to distinguish between the following two works:
Poem Electronique (Edgard Varèse) and *Orient-Occident III* (Iannis Xenakis)

Terms to know for "matching" section: (ca. 20 points)

amplitude / frequency / waveform / harmonic / octave / Hertz / decibels /
combination tones / masking / beats / transducer / frequency response /
transient response / harmonic distortion (clipping) /
pre-amplifier / power amplifier / unity gain amplifier (attenuator)

Know the following for short answer section: (ca. 30 pts)

the four common waveforms (names and diagrams)
destructive and constructive interference / phase /
musique concrète (i.e. who, what, where, when, etc.)
the characteristics of sound waves and envelopes
the historical trends leading to the development of electronic music esthetics.

Know basics of Tape Echo Delay:
- the output of the tape recorder is routed directly back into the machine, creating a “feedback” loop
- the physical distance between the recorder’s record head and playback head causes the delay
- the speed of the tape (as well as the distance between heads) influences the amount of the time delay between echoes

Know basics about human hearing and perception of sound:
- Frequency Spectrum of human hearing 20 Hz to 20,000 Hz
- Between 1000 Hz and 4000 Hz the human ear is extremely sensitive to sound

Early History of Electro-Acoustic Instruments

Telharmonium: (also sometime referred to as the "Dynomophone")

- Thaddeus Cahill, American inventor [1902 v.1-1906 v.2] >>invented electric typewriter, too!
- very large and expensive "one-of-a-kind" instrument <u>cost</u>: over $200,000
  <u>size</u>: ca. 200 tons — moved from Holyoke, MA to NYC in 30 box cars [1906]
- used Edison "dynamos" [electric generators or alternators] spinning at audio-rate cycles/sec. to produce sine wave pitches [NYC unit used 145 dynamos]

>>not a <u>true</u> electronic instrument since physical sound waves were actually produced by the spinning dynamos

- used organ-like keyboards to control frequency
- used [limited] additive synthesis control changes in timbre
- "broadcast" music over NYC telephone lines to subscribers of the service (!?!?!) where it would be heard over large cone speakers (like those on early phonographs)
- not viable financially (perhaps too ambitious for its time). It closed down in 1909 after the novelty wore off.

> recently renewed interest in the project, for its historical significance has resulted in a brand new book being written, which is due out very soon.

Triode Vacuum Tube:

- Lee DeForest American inventor [1906]
- made amplification of electronic signals possible
- lead to experiments with electronic oscillators and <u>true</u> electronic instruments

Theremin:

- Leon Termen, Russian inventor [1919] [Clara Rockmore CD #1358]
- did not imitate any acoustic instrument, no "musically influenced" interface
- 2 "poles" used: 1 controls frequency, the other controls amplitude
  [determined by proximity: closer = higher/softer — further away = lower/louder
- produced 2 ultra-sonic sine waves oscillators — 1 fixed & 1 variable — <u>difference tone</u> is audible.
- great for wild vibrato, portamento, swooping glissando, difficult to play and articulate even simple melodies.
- oldest electronic instrument design still in use today [Bob Moog is currently marketing a kit!]
- Notable uses:
  >old horror and sci-fi films [e.g. Hitchcock's *Spellbound* and *The Day the Earth Stood Still* featuring performances by Samuel Hoffman.
  >Led Zepplin's concert tour movie; *The Song Remains The Same*
- Paul Tanner, the Electro-Theremin, and the Beach Boys (*Good Vibrations, Wild Honey, etc.)*

Trautonium:

- Friedrich Trautweine, Russian inventor [1928]
- monophonic keyboard interface; <u>subtractive</u> synthesis; used mostly in research
- some composers wrote for it — Paul Hindemith (German >> American)

Ondes Martenot:

- Maurice Martenot, French inventor [1928]
- monophonic keyboard interface plus "ribbon" controller; additive synthesis
- timbre and envelope were controllable, as was volume [by means of a special key or knee lever]
- quite popular from the late-1920's through the 1950s [especially in France] 100's of compositions!
- many composers: notably >>Oliver Messiaen, Darius Milhaud.

Hammond Organ:

- Laurens Hammond, American inventor [1929]
- develops the concepts of the Telharmonium, by with significant progress made in the miniaturization process, thus it is not a <u>true</u> electronic instrument since it to produces physical sound waves electromagnetically which are then amplified. [12 alternators used to produce sine wave harmonics 1,2,3,4,6,8,10,12]

******all these inventions are primarily "live performance" instruments*******

See also: < http://www.obsolete.com/120_years/ > A web page devoted to the tracing the historical evolution and development of "Electronic Musical Instruments 1870 - 1990" ... by decade. Great Site! Lots of information lots of links, plenty of pictures and details.

The Classical Electronic Music Studio

Review: Pierre Schaeffer …[1948] at Paris radio station RTF (Radiodiffusion-television Français) marks just the beginning of collaboration begin classical composers and state-run radio stations throughout the rest of the world. >> continued to be mostly involved in musique concrète.

Other Radio Stations based studios:

Cologne, Germany : [1951] NWDR (Nordwest Deutsche Rundfunk)

> primary composer: Karlheinz Stockhausen (two works on tape #2) [CD # 1343]

Typical equipment for early studios:

- several sine wave generators; calibrated (maybe other waveform generators, too)
- noise generators • filters • ring modulators • amplitude & envelope control
- maybe an early electronic keyboard
- several tape recorders, from full-track mono up to (eventually) 4-track; with variable speed
- tape loop players and editing equipment

Show–and–Tell: >>> antique Sine Wave Generator (?)
>>> also Additive Synthesis Demos (Max/MSP and Yamaha DX/TX)

Listening Example: *Study II*, by Stockhausen (1954

- discussion of work and process: additive synthesis (non harmonic); much tape editing!!!

In the early 1950s, Stockhausen led a heated war of esthetics over what was the "real" electronic music. The early opinion of Stockhausen and his followers was that only music in which the original sounds were generated electronically constituted the "real" electronic music, therefore musique concrète was invalid, in his opinion. By mid-1950's (with Stockhausen's *Gesang der Junglinge,* 1956) the issue had dissolved. Another major difference was the importance of abstract "rational" or mathematical organization of sound among Stockhausen and his followers versus the more intuitive approach for musique concrète.

Studios modeled on Cologne: mid-1950's

Tokyo, Japan: [1955] NHK (Radio Nippon Horo Kyokai)

> primary composers: Toshiro Mayuzumi and Toru Takemitsu (*Water Musi*c)

Milan, Italy: [1955] RAI (Radio Audizioni Italiane)

> primary composers: Luciano Berio (*Hommage to Joyce*) and Henri Pousseur (*Scambi*, on tape #2)

Warsaw, Poland: [1957] Studio of Experimental Music of the Polish Radio

---

Utrecht, Netherlands [1954] NRU (National Radio Union) — *but didn't blossom until university affiliation*

Philips Studio, Netherlands

> primary composer: Edgard Varèse (*Poèm Électronique* also for the Philips Pavillion at 1958 Brussels World Fair
> Utrecht University took over the Philips Studio [1960]

Meanwhile in the U.S. the early pioneering work in the early 1950s by such composers as Otto Luening, Vladimir Ussachevsky, Tod Dockstader, etc. was being done privately and independently; there were no real established centers for electro-acoustic music in the U.S. The "radio connection" between, so potent in Europe and Japan owed a great deal to the fact that radio there was non-commercial and state run. In the U.S. this model would not work. As is true of many areas of research and experiment as well as contemporary arts and culture, the establishment and growth of electronic music centers has occurred primarily in colleges and universities…oftern with assistance from private corporations.

University of Illinois Experimental Music Studio [1958]

> emphasis on research and computer music (home of the INIAC)
> primary composers: Lejaren Hiller and Kenneth Gaburo (on tape #1)

Columbia-Princeton Electronic Music Center [1959] (with $175,000 help from Rockefeller Foundation)

> became a major international center of activity
> from 1960-1970 over 200 compositions by more than 60 composers from 11 countries
> primary composers: Luening, Ussachevsky, Milton Babbit, Mario Davidovsky
> also home of the RCA Mark II Synthesizer (more later!)

University of Toronto [1959] (with considerable technical assistance from Hugh LeCaine)

Yale University [1960] director: Mel Powell — a more typical situation (beg, borrow, & build equipment)

The RCA Mark II Synthesizer

- large "one-of-a-kind" voltage-controlled synthesizer for the Columbia-Princeton Studios
- Mark I [1955] built by Harry Olsen & Herbert Blair while doing RCA sponsored research
- then Mark II [c. 1958] >> $250,000 • 1,700 vacuum tubes •20 ft. long 7 ft. high

Initially controlled by two alpha-numeric keyboards — storage for future performance required a binary code to be punched onto paper rolls (similar to a player piano) which are fed into the instrument and thereby determine all aspects of the sound. >>This process gives composer tremendous control as well as tremendous responsibility.

> mastery of intermediary language (binary code) between composer and music
> no physical tweaking or rearrangement of the instrument required during playback
> "real-time" playback of entire composition eliminates the need for tape manipulation

In the classical electronic studio, a sound might be produced directly using a basic patch. To change the pitch the composer could simply turn the frequency dial. To change the amplitude turn the amplifier dial; to change the timbre, adjust the filter cut-off; and so on. All could be accomplished while listening to the resultant sound. With the RCA synthesizer you had to make your decisions first — without audio feedback — then specify the values and punch the paper rolls and feed them into the synthesizer. If you didn't like the results, you simply had to get a new paper roll and start over. But if you got it right, you could easily duplicate results as long as the instrument was working properly [i.e. all 1700 vacuum tubes were functional, etc.] and the paper roll did not become damaged. In the classical studio, exact reproduction is a more difficult proposition, since the process would involve precise human action [e.g. adjusting the filter cut-off from 1650 Hz to 825 Hz over 1.5 seconds]. The RCA Mark II's intermediary binary language, with its precision and storage capability, makes it similar to computer music systems, but unlike computer music all of the the sound generation is pure analog. At the same time, the RCA Mark II shares with other voltage-controlled synthesizers, the integration of many classical electronic studio analog signal devices and and control gadgets into a single self-contained unit, but that same intermediary language differentiates it from all other voltage-controlled analog synthesizers. [The instrument came to an ignoble ending when thieves broke into the studio through the roof, stole everything stealable, and destroyed the wiring.]

Raymond Scott's Manhattan Research

Raymond Scott was on the frontier of electronic music, though his efforts only really began to be appreciated after his death. This is largely due to Scott's own efforts to keep his work secret. In addition to his career as an innovative electronic music inventor, he pursued a parallel career as musician, composer and band leader — composing for cartoons, TV, and film as well as performing and recording with his jazz group. Along the way he even mentored Robert Moog (he worked for Scott before starting his own company) and was director of electronic music research and development for Motown for a couple of years. He founded Manhattan Research, Inc. in 1946 which was combination recording facility/music studio/science lab. Among his significant musical inventions were the Clavivox, the Electronium, and the Circle Machine.

The Voltage-Controlled Synthesizer

Throughout 1960's much activity was directed towards the creation of smaller voltage-controlled synthesizers — notably by Americans, Robert Moog & Don Buchla, and by Italian Paolo Ketoff— each with his own distinct appraoch to instrument design: large composition oreinted studio system (Buchla); real-time performance instrument (Synket); and a combination of these approaches (Moog). Not too surprisingly, Moog's "best of both" approaches made sense to many. For a time, his models led the pack in numbers and influence. Other notable names in the further development of this type of synthesizer would include >> Arp, Korg, Oberheim, Roland, Emu, and Serge — for the most part, all familiar names and companies still around today in the world of digital synthesis.

> **Voltage-Controlled (...or V-C) Synthesizer**: An integrated, self-contained system of equipment (or hardware) containing a collection of v-c oscillators, v-c amplifiers, v-c filters, random noise generators and envelope generators — all designed specifically for the creation of electronic music.

The earliest units [appearing c. 1964] seemed to be attempts to assemble the primary components of the of the classical electronic music studio into one miniature, economical, and (for its time) musician-friendly package. The structure of most of these early units (with Syn-ket being an exception) was very "open" and modular, having no "presets." Indeed, there wasn't even a hardwire connection between the keyboard and the oscillator to control frequency or between the keyboard and the E.G. or the E.G. and the v-c amplifier to control volume. Heck ...in some cases (such as the original *Buchla*) there isn't even a keyboard!

Filters & Subtractive Synthesis

(opposite of additive synthesis — see diagrams on page 26-27)

Subtractive Synthesis requires filters, and assumes a more complex waveform signal than a sine wave. Several types of filters: Low Pass Filter (LPF); High Pass Filter (HPF); Band Pass Filter (BPF) Band Reject Filter (BRF)

Other Terms: • Cut-off Frequency • Resonance (…or E.Q. …or regeneration …or emphasis)
• Bandwidth • Center Frequency

Show–and–Tell: **E-A_MusicPrimer Demos: Filters&LFO_Modulation (>*Music 413.app*)**

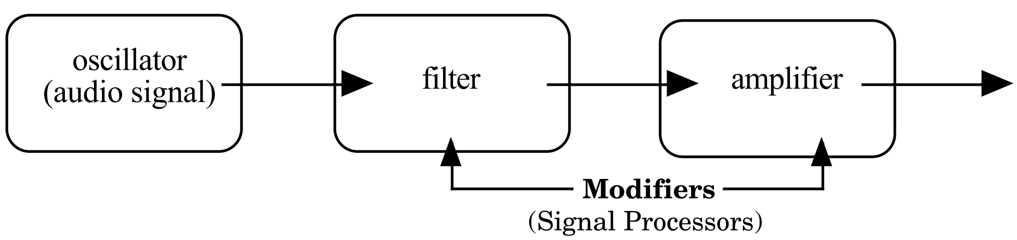
K2000 noise filtering demos >> #207-210 also >> #205-206 [Additive synthesis "proof"]
>> see Galaxy Editor Librarian Screen, p.30-33<<<

K2000 Demos

SineWaveDemo	220 BeatsDemo	240 SineFMDemo	260 SawAmpEG
201 TriangleDemo	221 MaskingDemo	241 SineAMDemo	261 Saw-LPFilter
202 RisingSawDemo	222 MaskingDemo2	242 SawAMDemo	262 SawFilterEG1
203 FallingSawDemo	223 LFO Sawtooth	243 SawTMDemo	263 AdditiveSine
204 SquareWavDemo	224 LFO Square	244 NoiseTMDemo1	264 AdditiveOrgan
205 Saw-LPFilter	225 LFO SineWave	245 NoiseTM-kbd	265 SawFreqEG
206 Squar-LPFilter	226 VHF Sine Wave	246	266 SawKybdFilter
207 Noise-LPFilter	227 VHF Sawtooth	247	267 NoizKybdFilter
208 Noise-HPFilter	228	248	268 SawKybd+Amp
209 Noise-BPFilter	229	249	269 SawKybd-Amp
210 Noise-BRFilter	230	250	270

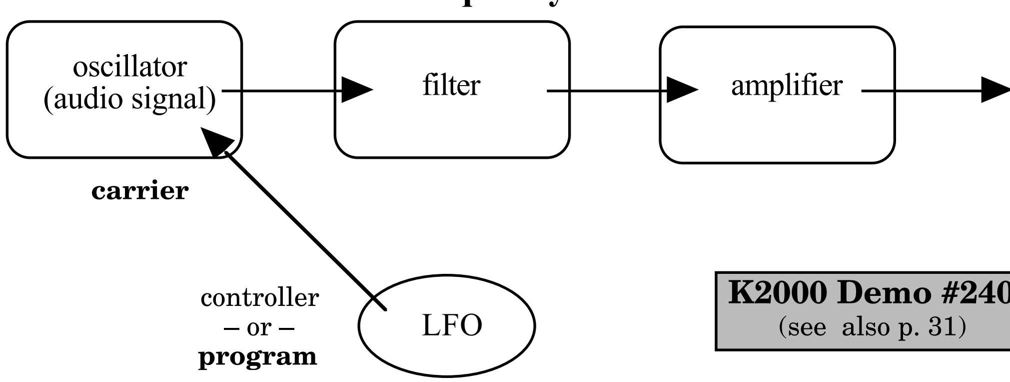
The Basic Patch

Show–and–Tell: >>>> Patch Cords (pass around the room) >>> also "Body Synthesis" examples!



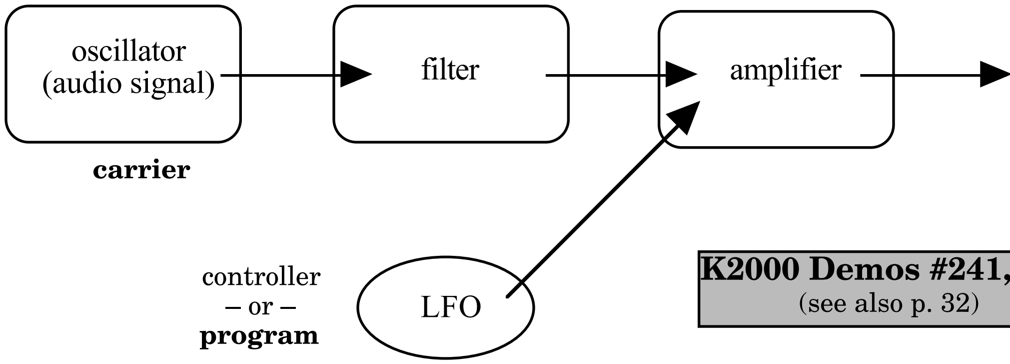
Also: sub-audio rate oscillator … or Low Frequency Oscillator = [LFO] >>> see #223-225

Sub-Audio Rate Frequency Modulation



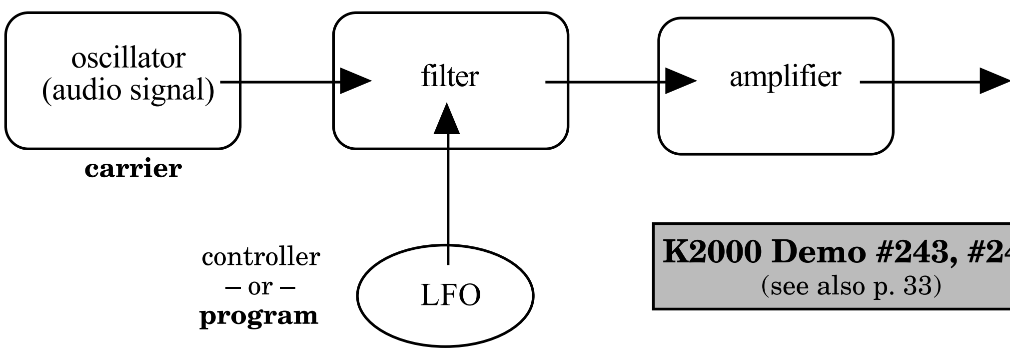
K2000 Demo #240
(see also p. 31)

Sub-Audio Rate Amplitude Modulation



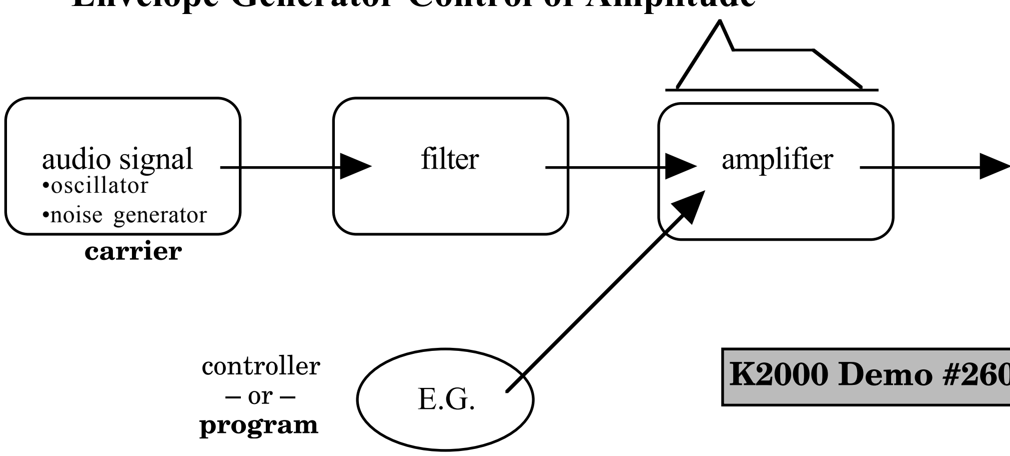
K2000 Demos #241, 242
(see also p. 32)

Sub-Audio Rate Timbre Modulation



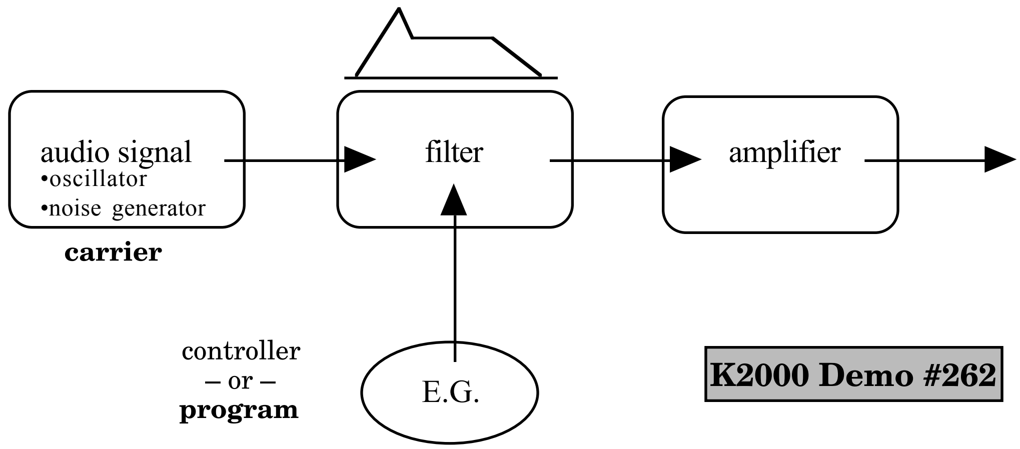
K2000 Demo #243, #244
(see also p. 33)

Envelope Generator Control of Amplitude

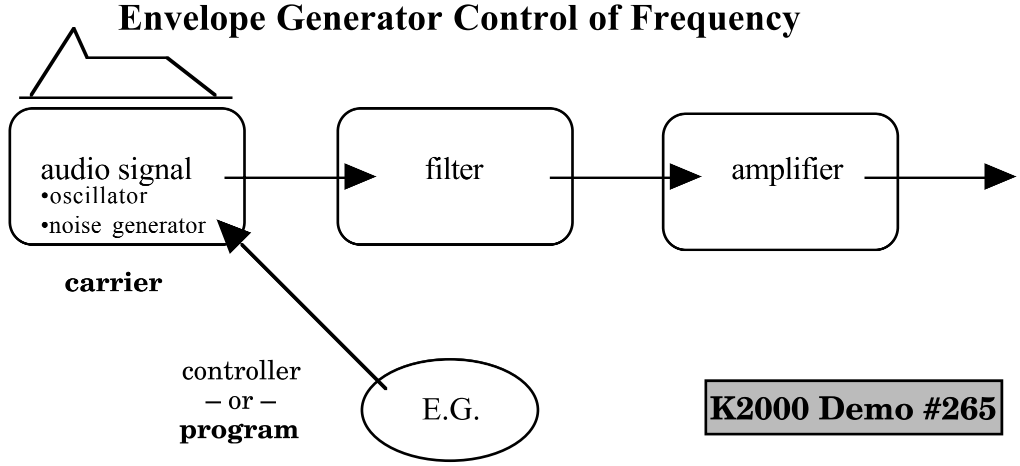


K2000 Demo #260

Envelope Generator Control of Timbre



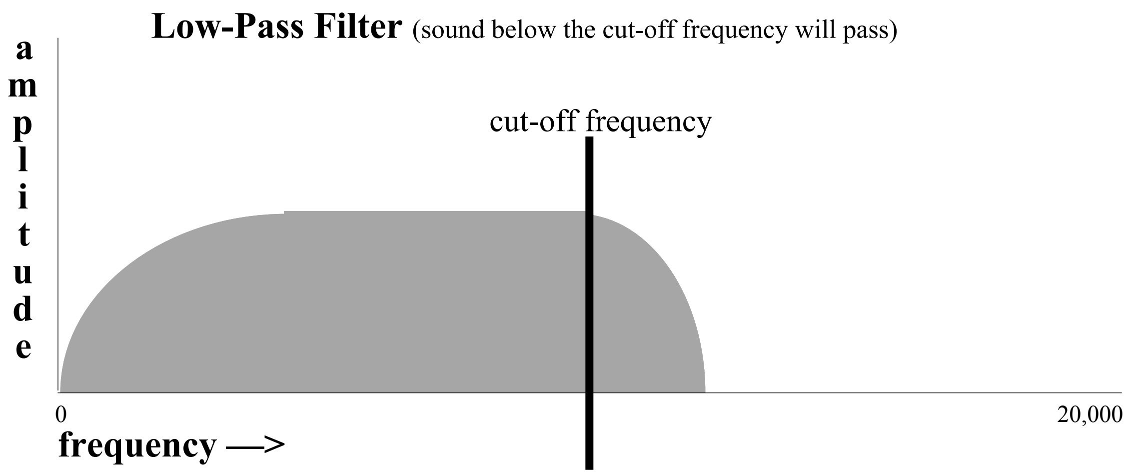
Envelope Generator Control of Frequency

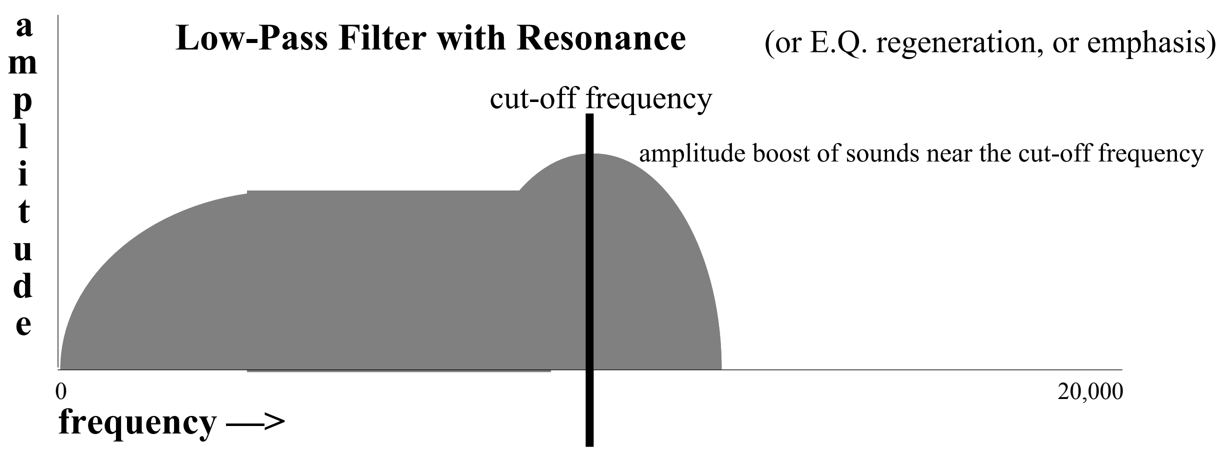


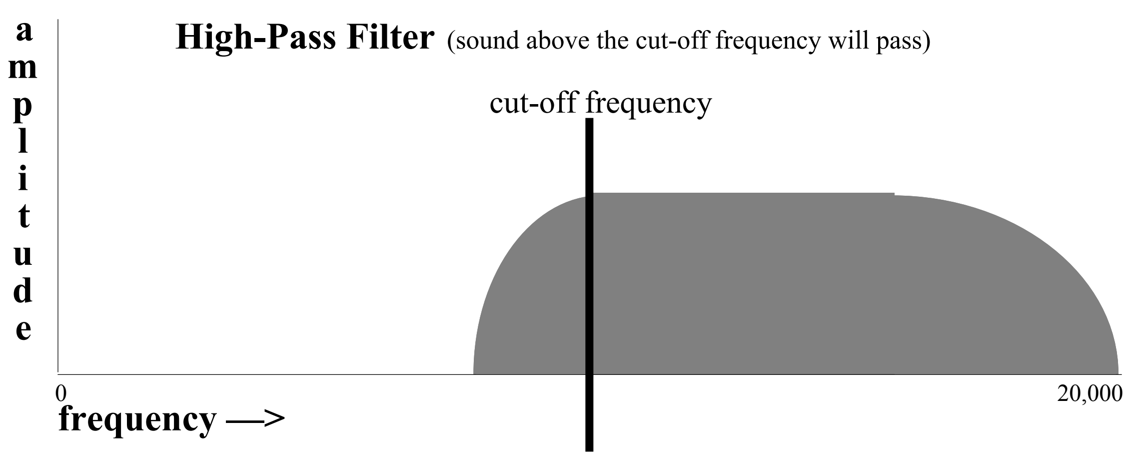
(See page 28 for a summary listing many common type of modules classified by their function.)

Basic Filter Diagrams

Shaded in areas indicate what frequencies will be heard (i.e. what frquencies are allowed to pass). Note that cutoff frequencies do not function like a brick wall. They tend to “roll off” more gradually.





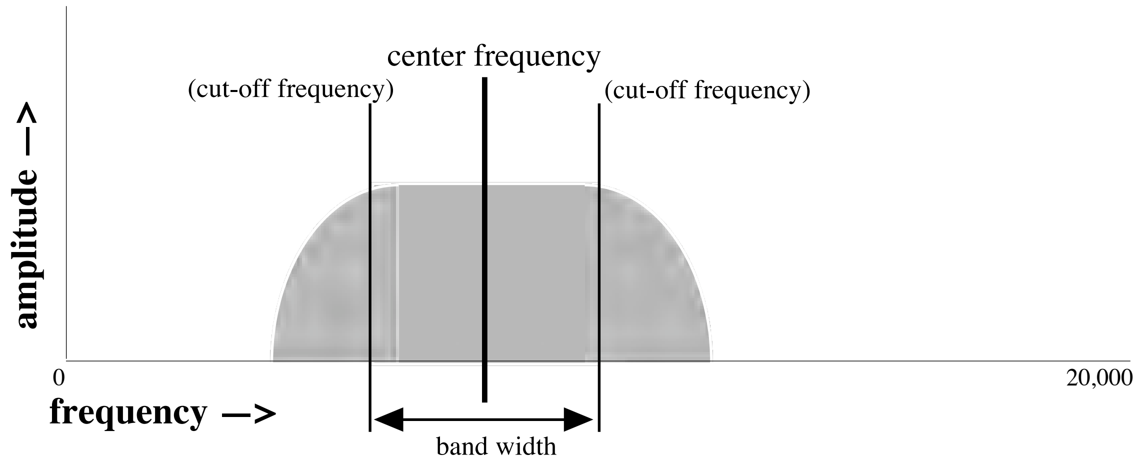


Band Filter Diagrams

Note that band filters have two cutoff frequencies. These are usually set indirectly by means of establishin
a center frequency and a frequency band width. Adjusting the band width moves the two cutoff frequencie
opposite directions away from or towards the center frequency. Adjusting the center frequency moves the two cutoff frequencies in parallel motion, sweeping through the frequency spectum.

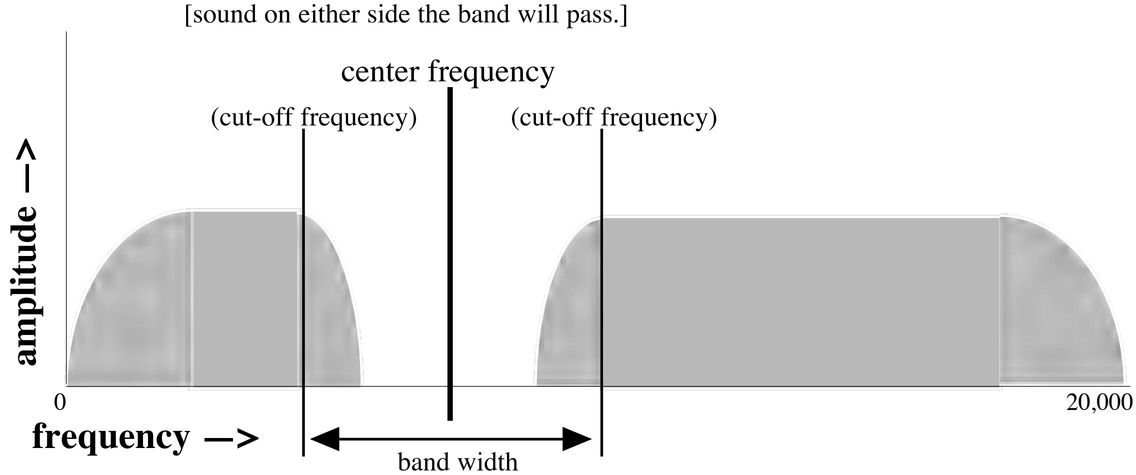
Band-Pass Filter

[sound between the cut-off frequencies (i.e. within the band) will pass.]



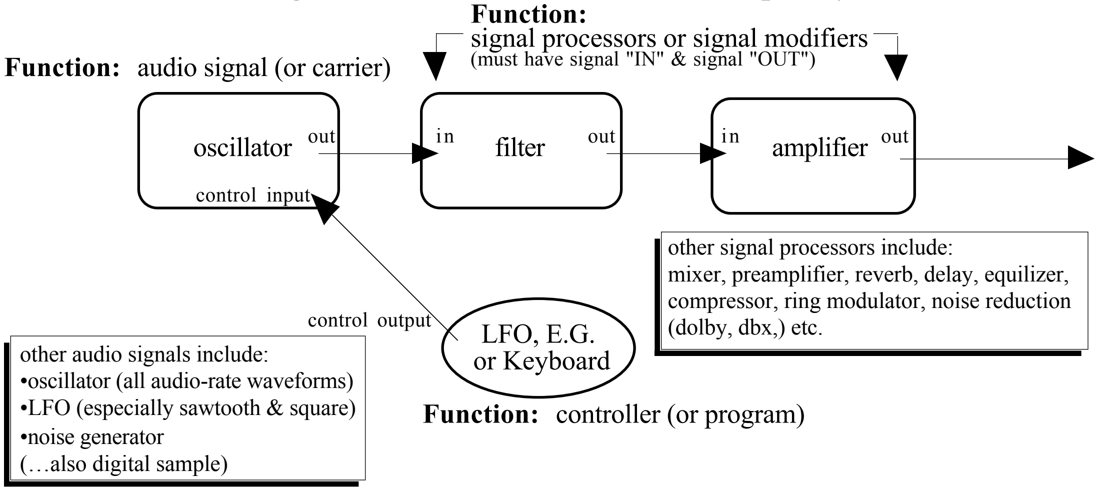
Band-Reject Filter

[sound between the cut-off frequencies (i.e. within the band) will be <u>rejected</u>.]
[sound on either side the band will pass.]

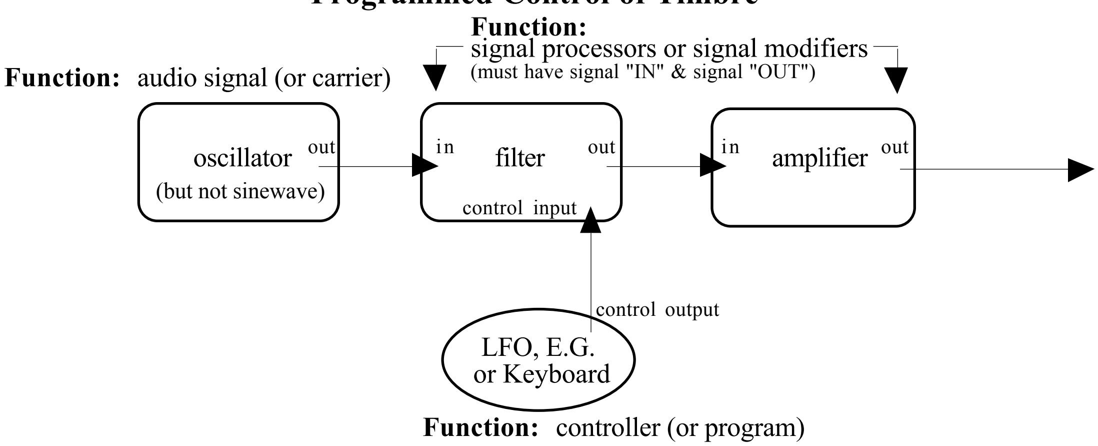


Basic Patch Diagrams

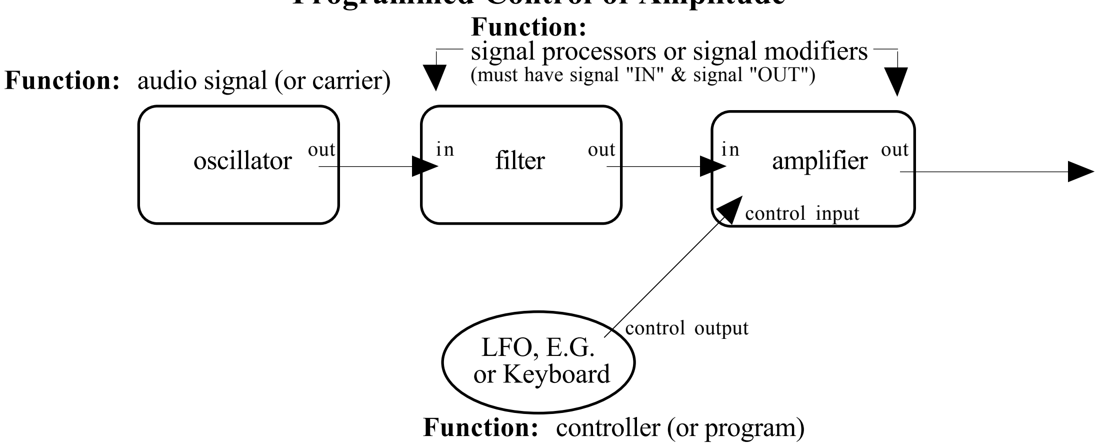
Programmed Control of Pitch or Frequency



Programmed Control of Timbre



Programmed Control of Amplitude



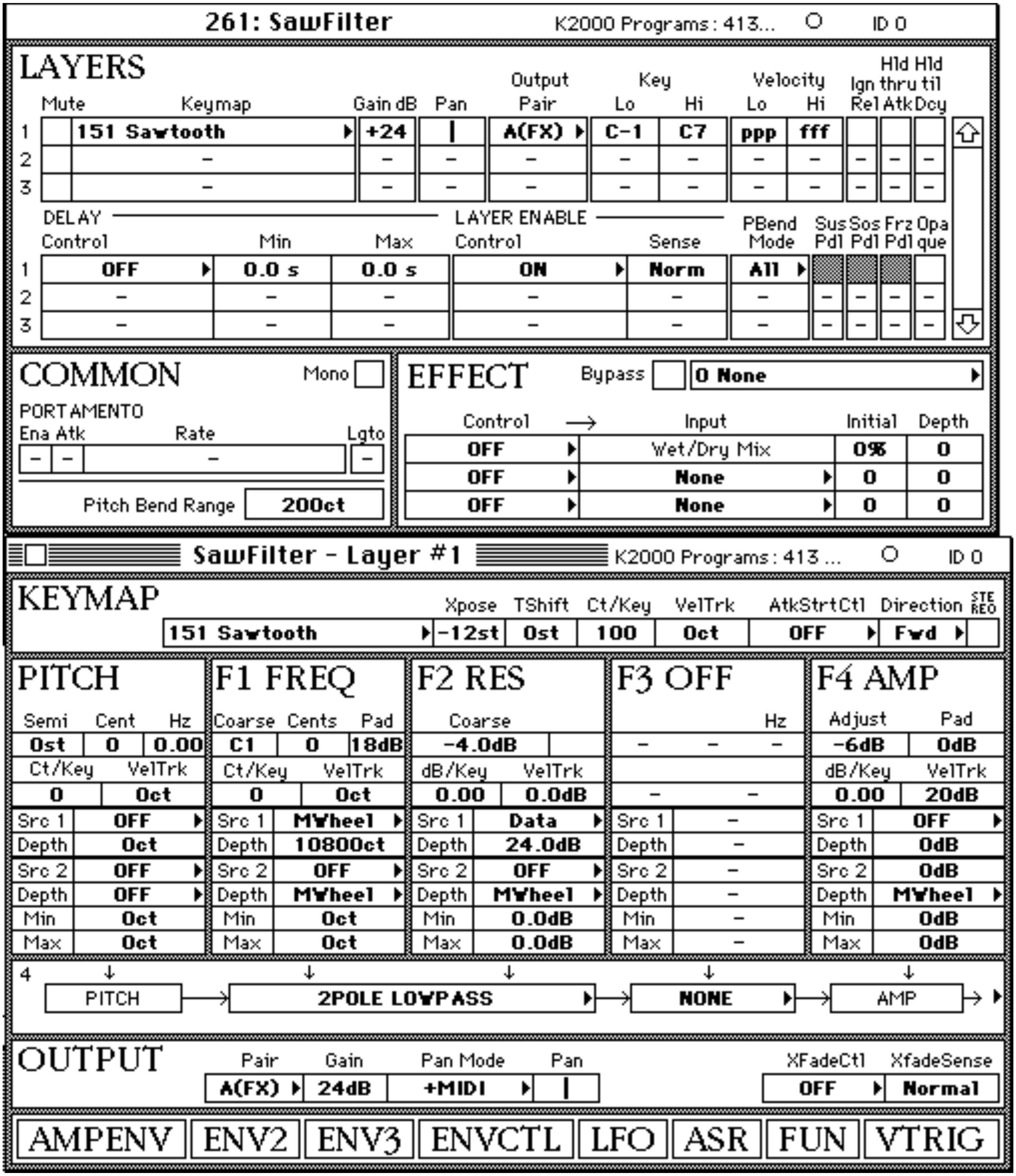
K2000 Demos

SineWaveDemo	220 BeatsDemo	240 SineFMDemo	260 SawAmpEG
201 TriangleDemo	221 MaskingDemo	241 SineAMDemo	261 Saw-LPFilter
202 RisingSawDemo	222 MaskingDemo2	242 SawAMDemo	262 SawFilterEG1
203 FallingSawDemo	223 LFO Sawtooth	243 SawTMDemo	263 AdditiveSine
204 SquareWavDemo	224 LFO Square	244 NoiseTMDemo1	264 AdditiveOrgan
205 Saw-LPFilter	225 LFO SineWave	245 NoiseTM-kbd	265 SawFreqEG
206 Squar-LPFilter	226 VHF Sine Wave	246	266 SawKybdFilter
207 Noise-LPFilter	227 VHF Sawtooth	247	267 NoizKybdFilter
208 Noise-HPFilter	228	248	268 SawKybd+Amp
209 Noise-BPFilter	229	249	269 SawKybd-Amp
210 Noise-BRFilter	230	250	270

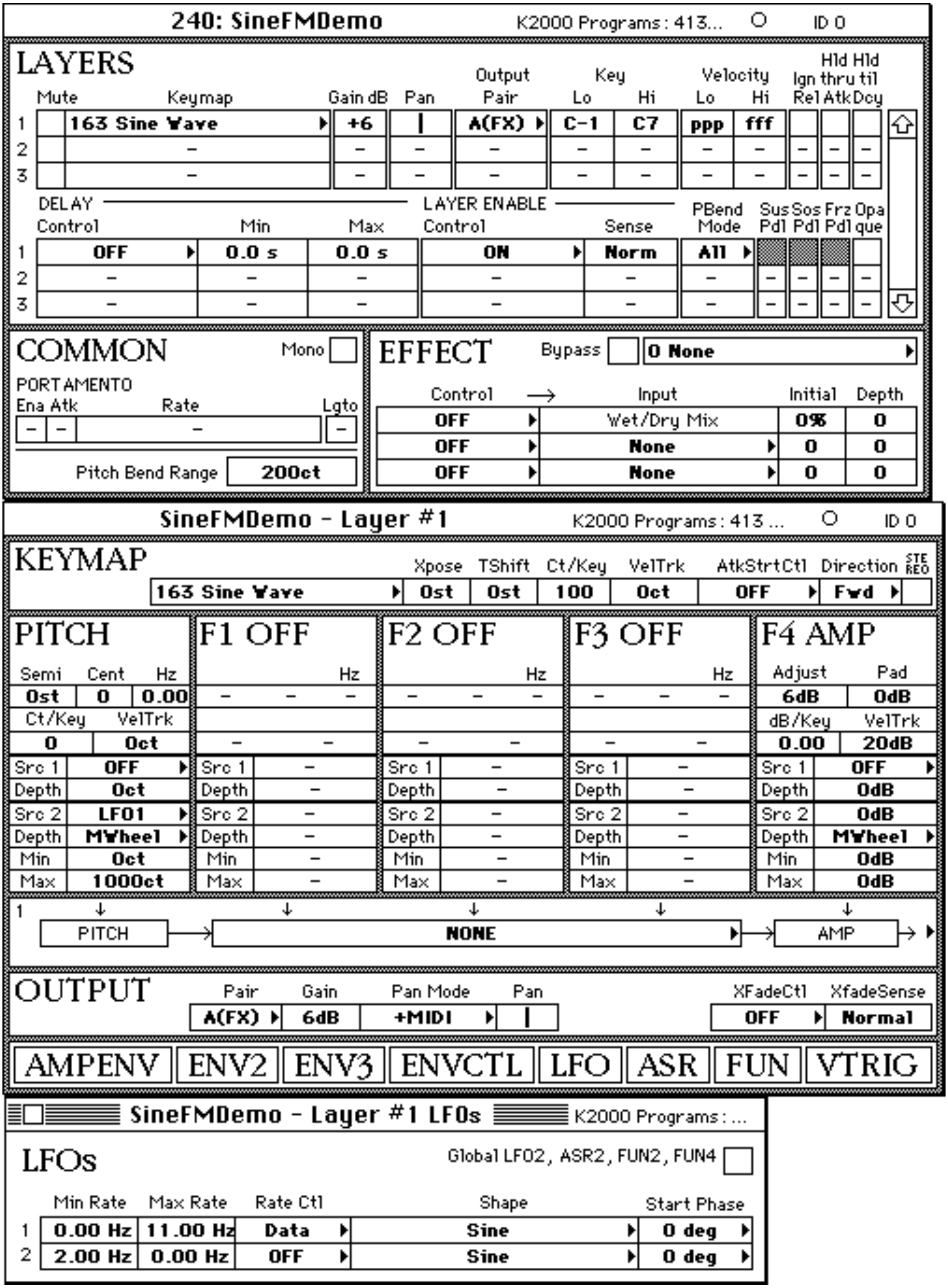
These are “patches” for the Kurzweil K2000 which demonstrate many basic aspects of sound synthesis. They are stored as an Opcode Galaxy file on the hard drive of the workstation attached to the K2000. Screen pictures from several of these “patches” are printed on the following pages to demonstrate how the K2000 would realize the basic concepts presented on the previous page of this Study Guide.

Opcode's Galaxy Editor/Librarian Screen for Kurzweil K2000

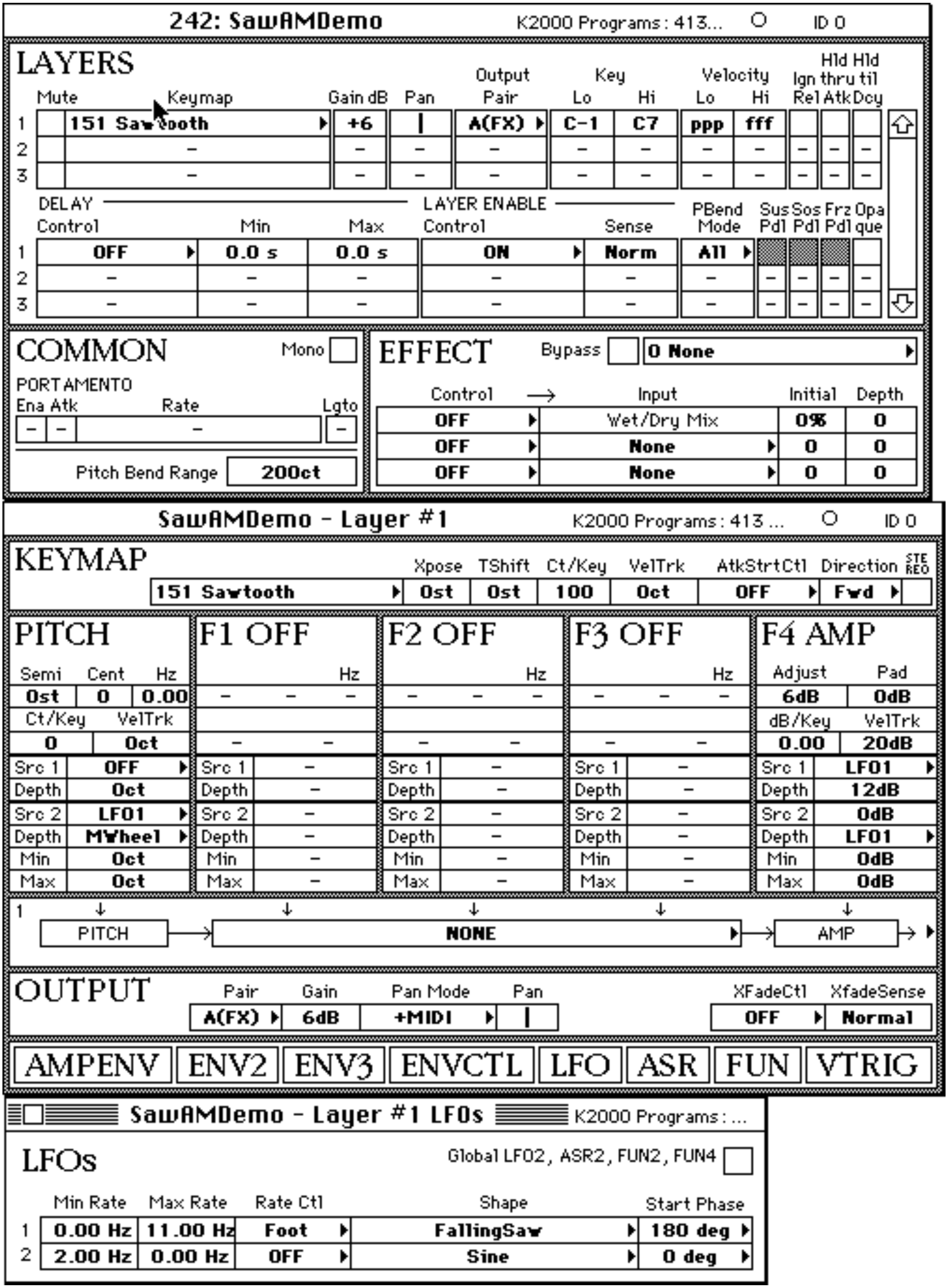
These screens give an overview of many of the parameters used to generate and control a sound on the Kurzweil K2000. This particular example shows a "basic patch" for subtractive synthesis using a sawtooth carrier patched first through a low-pass filter (w/ 2 "poles" — represented below by F1 & F2) then through an amplifier (F4). The Modulation Wheel controls the filter cutoff while the Data slider controls the filter resonance. Meanwhile, key velocity will control the amplitude. [Xpose = -12st (transposed down 2 octaves); Ct/Key = 100 (100 cents or one half-step per key — standard tuning info)]



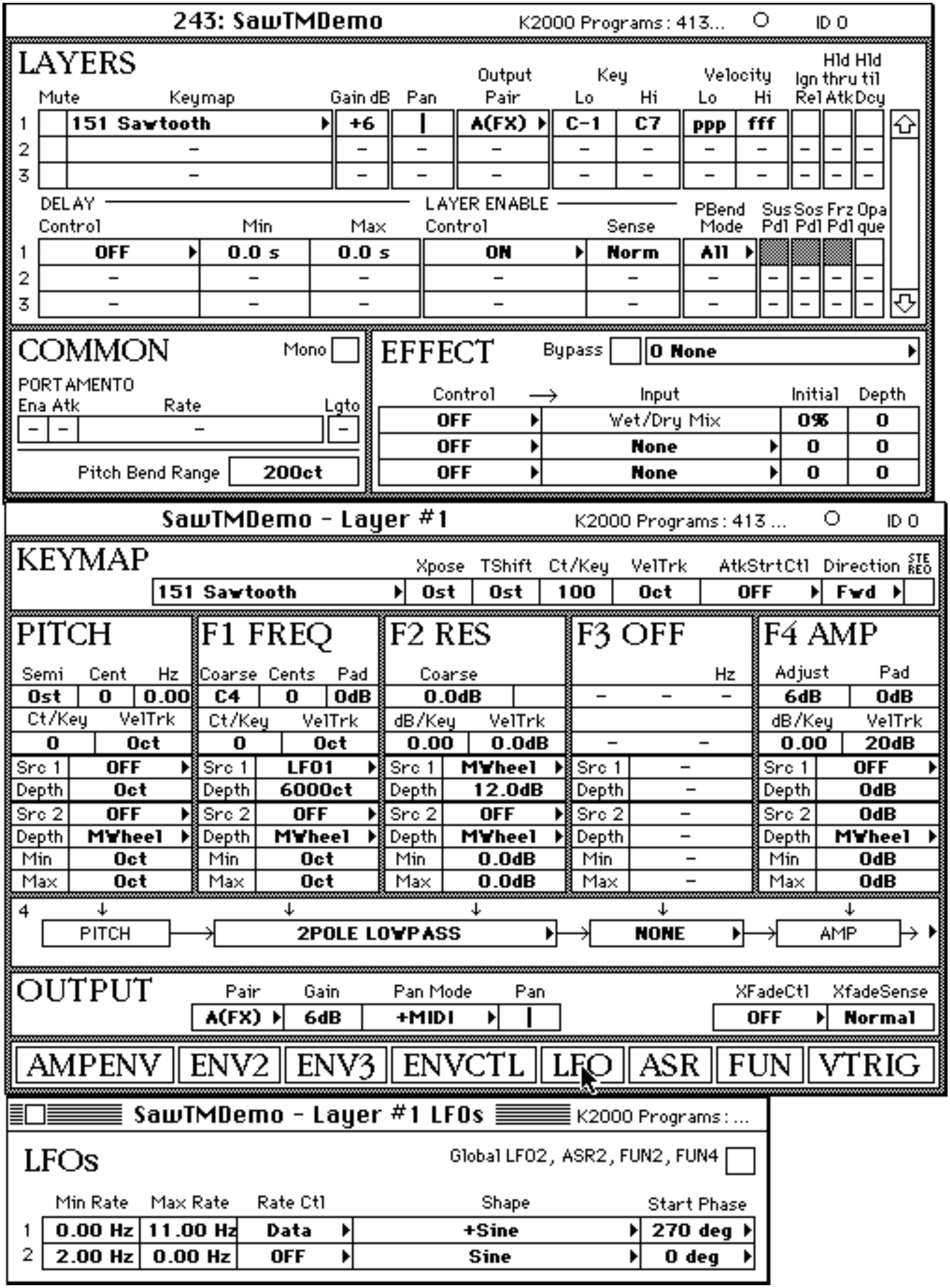
Sub-Audio Rate FM



Sub-Audio Rate AM



Sub-Audio Rate TM



Listening Guide: Exam 2 Schrader*

Darius Milhaud: *Suite for Ondes Martenot* [ca. 1930] (mvt. I).................................... pp. 66-7
"Real-time" performance on Ondes Martenot (also with piano accompaniment)

Camille Saint-Saëns: *The Swan* — **Clara Rockmore** performing on the Theremin (also with piano accomp.)

Karlheinz Stockhausen:
Study II [1954]... pp. 81-8
Additive Synthesis (exclusively); all sound created by combining 5 carefully tuned **sine wave generators**. Mathematics plays an important role — all computations derived from the number 5. A scale was designed using formula 25√5 resulting in an equal-tempered scale with with 81 frequencies from 100 Hz to 17,200 Hz; ca 10 notes/octave, though no in-tune octaves are produced; 5-note chords, grouped into 5 basic types.

Gesang der Jünglinge ("Song of Youth") [1955-56].. pp. 94-6
Title — nothing to do with idyllic impressions of youth, but rather — from the Bible's book of *Daniel* where King Nebuchadnezzar throws youths Shadrack, Meshack, and Abednego into a fiery furnace. (They miraculously survive!) As complex and complete a compendium of all the tricks available to early electronic musicians at the time as can be found. Uses **additive synthesis**, **subtractive synthesis**, and musique concrète [**boy soprano**].

Henri Pousseur: *Scambi* [1957-58] (excerpt) .... pp. 102-4
Uses purely **subtractive synthesis** with only **white noise** as a sound source. Pousseur established a series of procedures with intentionally random elements, thus the score can be viewed also as a "recipe" for a composition — there are in fact 5 known versions: including 2 by Pousseur & 2 by Berio.

Louis & Bebe Baron: soundtrack from *Forbidden Planet* [1956] (3 excerpts)................ pp. 78-9
1.) Robby Arranges Flowers & Zaps Monkey 2.) Love at the Swimming Hole 3.) Home Coming

"We began to explore the possibilities of simple circuits [that] had characteristic behavior patterns. The desiderata was that the behavior should not be regulated, but rather non-logical. The most interesting sounds often came from circuits that were not stable....We created individual cybernetic circuits for particular themes and *leit motifs*..."
— *L.B/B.B.*

As a result of this process and their earlier work with the tape manipulation techiniques of musique concrète (they were actually associated with John Cage in the early days of the Project for Music for Magnetic Tape in NYC) the music often consists of short ideas spliced together.

Raymond Scott: *Cindy Electronium* [1959]
Created using two of Raymond's analog synthesixers — the Clavivox, which included a keyboard (fyi: Bob Moog was his apprentice on this instrument), and the Electronium, which did not. It was this latter instrument that captured Berry Gordy's attention and led to his being hired as director of electronic research and development at Motown.

Milton Babbitt: *Ensembles for Synthesizers* [1964] (excerpt) .... pp. 123-7
Composed using the **RCA Mark II** Synthesizer. Highly complex, extremely detailed music derived from 12-tone serial processes. Makes use of the detail and precision required by the Mark II. Listen in the beginning of the piece to the dense clusters or chords that punctuate the music; each pitch has its own discrete **envelope generator**, so cluster "evaporates" one note at a time until there is a single pitch left "hanging." This gesture happens repeatedly throughout the opening of the piece. It would very difficult to duplicate, requiring extensive multi-track mixing, using any of the monophonic synthesizers available at the time — and for many years to come!

***Schrader:** *Introduction to Electro-Acoustic Music,* Barry Schrader, Prentice-Hall, 1982.

This reading material is on reserve in two formats:

1) the entire book [call # ML1092 .S35]
2) a packet of exerpts containing just the pages listed here pertaining specifically to the listening (note: this packet has been formatted to make it easier to photocopy it for study purposes.)

Wendy (Walter) Carlos: *Switched on Bach* [1967]

The music of J. S. Bach arranged and played on a **Moog 900 series synthesizer** (like OU's old Moog). This was a very commercially successful recording which brought wide recognition to the work Moog was doing in the field of v-c analog synthesizer design. The design differences between Buchla and Moog go hand-in-hand with the differences in the style of music represented on this listening list. The lack of an organ-like keyboard on the Buchla results in more abstract music less dependent on pitch than conventional music. "Switched On Bach" would be unthinkable on the original Buchla synthesizers, but the Moog, which had a monophonic organ-like keyboard, was a natural for this type of project. But since the keyboard was indeed monophonic (low-key priority), a multi-track tape recorder was required to realize the polyphonic textures of J. S. Bach. Carlos indicated in an interview that there was often a sly attempt at humor by highlighting — through tongue-in-cheek exaggeration — some common misguided approaches to which performers on conventional keyboards are prone (e.g., the cartoonishly fast tempo on the *Two-part Invention in F*).

• *Two-part Invention in F*

<u>NOTE</u>: Compare the envelope shapes used in the following two selections

• *Two-part Invention in d* … and … • *Jesu, Joy of Man's Desire*

• *Brandenburg Concerto No. 3 in G* — mvt. 2 (improvised*) & mvt. 3

Note: J. S. Bach had nothing to do with this improvised movement. His original score merely contains two chords for the keyboard player to improvise on between the two outer movements. Carlos's improvisation is testament to the power and flexibility of Moog's synthesizer. Not only could it handle "traditional" music well, but it could also create unique "synthetic" sonic events <u>not</u> duplicatable on traditional instruments.

Morton Subotnick: *Wild Bull* [1968].. pp. 130-1

Created using a **Buchla synthesizer**, a large modular, voltage-controlled synthesizer. The slow, amorphous opening of the work uses timbre for coherence – establishing a whole "family" of complex, yet related "Wild Bull" sounds. Later in the piece an important characteristic of Subotnick's music emerges; driving dance-like rhythms punctuated by unpredictable random bursts of activity. The driving rhythms are powered by an analog **sequencer**. This is a primitive storage device capable of storing a handful of set voltages. It has an adjustable clock-rate that determines the speed at which these stored voltages are triggered. Typical of the flexibility built into these large modular systems, these voltages could be used to control the pitch, timbre, amplitude (or any combination of these) of a carrier signal or a control signal. It could also be combined with any additional control voltages desired, which could lend tremendous variety to what might otherwise be a rather limited pattern produced by the limited storage capacity of these old sequencers.

Todd Rundgren: *Dog Fight Giggle* [1973]

Large modular analog synthesizer [Arp 2500 or 2600?] used to imitate musique concrète

Stevie Wonder: from side 2 of *Innervisions* [1973]

Uses **Arp & Moog analog synthesizer keyboards**. Stevie plays all instruments, but credits Robert Margouleff and Malcolm Cecil for programming synthesizers. Listen for extensive use of **E.G. control of low-pass filter** and for **TM** (LFO control of filter cut-off) in these songs.

• *Higher Ground* — & — • *Jesus Children of America*

Laurie Spiegel: *Appalachian Grove* [1974]

Spiegel was an early pioneer in computer music working at Bell Labs with Max Matthews, especially with the GROOVE system (Generated Realtime Operations On Voltage-Controlled Equipment), which is a hybrid of digital (or computer) control of analog audio equipment. All manual actions (pressing a key, tweaking a knob, etc.) were stored on a computer memory disk and could be selectively output to control the modular analog synthesizer through one of 14 independent control lines. This composition was composed after a visit to the Fiddlers' Grove Festival in NC ... and "in reaction to an overdose of heavy, sad, and introspective contemporary music. — *L.S.*"

Mark Phillips: *Lunar Illusions* — finale [1981]

Uses a large **Emu 2000** series modular analog synthesizer. Features extensive use of a new generation hybrid digital/analog sequencer. A sequence was created by repeatedly playing random 3-note "chord" groupings using only the first five notes of a major scale — but only the first two notes of each "chord" are allowed to trigger a sound. (Even trying hard, which I *didn't*, it is virtually impossible to play three notes at the exact same time. Though our ears may be fooled, a sensitive machine can detect the differences.) The resultant two-part sequence is stored in memory, then recorded to tape several times using different timbres, different sppeds, and different transpositions to create a multi-layered backdrop against which two new melodic sequences were composed — a bass line and a treble line, each derived from the pitch material of the original sequence. With sequencers handling all the notes during playback, both hands are free to manipulate the dials controlling filter cut-off, filter resonance, vibrato and tremolo speed, etc. A final element is the *very* **low-frequency (LFO) amplitude modulation** (AM) which causes a low-pitched drone (root & fifth) to periodically fade in and out.

Music 413/513 Study Guide for Exam II

Listening: (40 points) Know which examples use the following…

synthesizers or sound sources:

Moog / Buchla / RCA Mark II / Ondes Martenot / Emu2000 /
sine wave generators (ONLY) / noise generators (ONLY) / boy soprano / analog keyboard synthesizers /
ARP modular analog synthesizer / GROOVE system / custom-designed and -built circuits

techniques, control devices, processes, etc.

only additive synthesis / only subtractive synthesis / musique concrète techniques as well as both
additive & subtractive synthesis / analog sequencer / hybrid digital/analog sequencer /
many different envelopes generators triggered simultaneously /
very low-frequency AM / computer driven V-C analog synthesizer / E.G. control of filter cutoff /
analog imitation of musique concrète / custom-designed "unstable" electrical circuits

Questions 10A and 10B on Listen Exam will consist of modulation sounds produced on a Kurzweil K2000 synthesizer. For this question you should…

Be able to distinguish the following types of modulation:
sub-audio rate FM / sub-audio rate AM / sub-audio rate TM

Be able to recognize the following audio (carrier) signals:
audio or sub-audio rate sine / audio or sub-audio rate triangle / audio or sub-audio rate rectangle / audio or sub-audio rate sawtooth / noise generator

Be able to recognize the following LFO program waveforms:
Sine or Triangle / Sawtooth (positive going) / Sawtooth (negative going) / Pulse (or square)

Terms to know for matching: (ca. 30 points)

vibrato (sub-audio rate FM), tremolo (sub-audio rate AM), high-pass filter, low-pass filter, band (pass/reject) filter, filter resonance, carrier, program, subtractive synthesis, additive synthesis, basic patch, sequencer, envelope generator, voltage-controlled synthesizer, RCA Mark II, Telharmonium,

Short Answer:

3 early classical electronic music studios / common sources for audio signal or carrier /
common sources of program or control signals / commonly used signal processors /
3 primary functions used in synthesis (audio signal, signal processor, & control signal) /
be able to label a simple ADSR envelope diagram and indicate "key down" & "key up"
(See page 7 of your Study Guide)

Short Essays on: (ca. 10 points)

Early instruments: Telharmonium, Theremin, Trautonium, Odes Martenot, RCA Mark II,
Voltage-Contolled Synthesiser (Moog, Buchla, Syn-Ket)

Electric Sound, by Joel Chadabe — Selected Readings

>>Chapter 1: Early Instruments *and* >> Chapter 6: Synthesizers
Stockhausen (*Studies I & II* & *Gesang*)— pp 37-40
Pousseur – p 50
Subotnik – pp 86-91 &pp 146-149

Basic Computer Terminology

Bit: […from **B**inary **I**n**t**eger or **Bi**nary digi**t** ???] — a single binary digit or integer (i.e. a **1** or a **0**)

Byte: a unit of digital data consisting of 8 bits, sometimes divided into two so-called nybbles (or nibbles) of four bits each. Computer memory is usually measured in kilobytes (1,000 bytes or 1KB) or megabytes (1,000,000 bytes or 1MB). [Since computer memory is usually provided in units that are powers of two, a kilobyte is often used to represent 1,024 bytes instead of 1,000 bytes.]

Word: a group of bits that can be handled or "addressed" simultaneously (frequently these are powers of 2 — i.e. 8-bit, 16-bit, 32-bit, 64-bit, etc.). A 16-bit system, then, uses a word containing 2 bytes; 32-bit system uses a word containing 4 bytes, etc.

<u>NOTE</u>: Computers in lab are 64-bit machines; most of the newer lab synths are 16-bit (same as CD audio).

Common Types of Computer Memory

RAM: Random **A**ccess **M**emory. Any point or sequence of points can be accessed directly.

Both primary RAM and hard disks are random access, and both are user programmable or, in other words, both can be altered by the user. There are significant differences between these types of memory. In casual usage, RAM typically refers to the volatile primary RAM.

Primary RAM is typically made up of small integrated circuit chips, which are "volatile" — meaning all information is lost when power is removed. Standby battery power can be used to overcome this, as in most computers which use small batteries to "remember" the date and time when powered down. It is also common to use batteries in digital synthesizers to remember the changes you make to the factory presets.

Hard disks are used for longer term storage they are not "volatile" — meaning they retain information when powered down.

When you "open" or "read" a file from a disk you load it into primary RAM. The original is untouched. Any changes you make are in volatile RAM until you execute a "save" or "write" command which then saves the file along with your changes to the disk. Generally speaking "save" is an overwrite procedure meaning the original file is overwritten by the new version of the file. To preserve both versions of the file use a "save as..." or "rename" command and give the new file a different name.

<u>NOTE</u>: All the new lab computers have 1GB of RAM. All internal hard drives have 80-GB for storage.

ROM: Read **O**nly **M**emory. ROMs have permanent information placed in them during manufacture. Because a program written into a ROM cannot be changed, such programs are referred to as firmware.

ROMs are also random access devices, but when reference is made to the "RAM" of a computer, normally it is not the ROM, but only the writable RAM that is being discussed.

Other types of memory:
Programmable ROMs (PROMs) may be manufactured as standard units and then have information permanently implanted in them by specialized equipment. Erasable programmable ROMs (EPROMs) may have their programs erased and new information implanted from time to time. EPROMS that can be erased by ultraviolet light are used extensively in industry. Newer electrically erasable PROMS (EEPROMS) are now incorporated in some computers and synthesizers, to allow for firmware upgrading without replacing built-in circuitry.

More Computerese:
The iMacs in the lab have a G5 micro-processors operating at a minimum of 1.8 GHz. Each computer has 3 **USB ports** (<u>U</u>niversial <u>S</u>erial <u>B</u>us). These updated high-speed, serial-style ports are for devices that transfer data serially (i.e. in sequential "trains" of data). The computer's internal drive and the CD-ROM use a high-speed bus: ATA/IDE. [**ATA**chment/**I**ntegrated **D**evice **E**lectronics]. The G5's also have a "FireWire" bus (technical name: IEEE 1394a), which we are not currently using in this lab, so it is always available for users to plug in a firewire drive of their own. There is another older type of high-speed computer port which you may run into in audio system: **SCSI** [**S**mall **C**omputer **S**tandard **I**nterface — referred to verbally as "scuzzy"]. It was used to connect our hard drives and CD-Recorders to computers before firewire.

Digital Audio (…see also *Digital Audio Primer* excerpt from the Max/MSP manual — available on my web page)

Acronyms:
digital-to-analog converter (DAC or D-A converter) — analog-to-digital convert (ADC or A-D converter)

•**digital representation and storage of analog signal** — (see diagram later in study guide)

Nyquist's Theorem:

The sampling rate must be equal to or greater than two times the sampled frequency…or **S.Rate ≥ 2x Freq.** …otherwise aliasing (or low-frequency foldover) occurs. [The backwards revolving wheel seen occasionally in films is a good example of the concept in a visual medium.]

common sampling rates for high fidelity audio:

44.1 kHz (44,100 Hz) for CD format
48 kHz (48,000 Hz) for "pro audio" recording studios, as well as for DAT recorders, digital samplers, etc.
96 kHz — new DVD audio standard.

other digital audio "factoids":

number of "bits" per "byte" determines the accuracy of amplitude resolution
6 dB of resolution per bit … OR … 1 bit = 6 dB of amplitude capacity
SO …16-bit sampling (used in CD format) results in a theoretical 96 dB of amplitude capacity [reality ca. 90]
newer high-quality devices often use 20-bit and /or 24-bit resolution.

Even though most audio projects today are still destined for eventual CD distribution, there is still a significant advantage to formats in excess of 16-bit, especially in systems where processing (i.e. EQ, mixing, delays, reverb, etc.) is done in the digital domain, as is common these days. Consequently, many software-based digital audio systems will have higher-bit internal processing than their analog inputs and outputs.

Adding to the confusion, may devices use *low bit conversion* [especially 1-bit converters —analogies: number of engine cylinders and speed of firing] ALSO: *differential coding* based on measuring the difference between one instantaneous amplitude and the next. Low-bit converters are inherently MORE accurate than high-bit converters, but they required a much faster processing. Currently over-sampling (as high as 128x) is now commonly used to minimize the effects of brickwall filters.

Noise in a digital system is introduced as a result sampling errors (i.e. "**quantization errors**"). Unlike analog systems which have a fixed noise level, regardless of whether there is a signal, digital systems create noise only when a signal is present. In other words when there is no signal running through a digital system there can be no "quantizing errors" because there is no need for quantizing. If the the sampling rate is high enough (as it is in the CD format) and if the number of bits per byte is adequate (as in the 16-bit CD format) and the program material is reasonably strong enough, then the noise of the "quantizing errors" generally will be masked by the amplitude of the program.

"Headroom" is another area of significant difference between digital and analog recording. In an analog system you may be able to "peak" the meters for brief moments throughout the recording. In fact recording engineers used to "learn" their particular system so that they would know how much they could "push" the limits before getting into serious trouble. Some common timbres, especially those heard on many older pop records (i.e. sizzling cymbals, crackling snare drum, searing guitar solos, etc.), were in fact a result recording with the peaking meter warning lights flashing much of the time, indicating that a small bit of distortion was happening. (This minor distortion is heard as a slight, and sometimes desirable, change in timbre.) In a digital system there is no real headroom. The system will work flawlessly without any distortion or timbre change right up to the point where the limits are exceeded, at which time you will hear a serious glitch in the recording, rendering it unusable. Though it is true that noise from quantizing errors is more likely to be noticed at very soft amplitude levels, most analog recording systems would have even more noise problems handling the same low amplitudes. And since quantization errors in at soft amplitudes is still much less a problem than the clipping that occurs with over-recording, it is better to be conservative in setting record levels in a digital system

— Computer Music History —

just a very short list of some computer music pioneers and their accomplishments
(admittedly heavy on American innovators)

Max Mathews: computer music pioneer researcher at Bells Labs; developed Music V programming language [late 1950's through 1960's]. Written in FORTRAN, it was a portable to other computers (all you needed was about 3,500 properly punched computer cards!)

Jean-Claude Risset: computer music pioneer; worked with Max Matthews at Bells Labs [1960's]; established the computer center at IRCAM in Paris [1975], still today one of the leading centers for advanced computer music and research.

Barry Vercoe: first at Bell Labs/Princeton, then a long association with M.I.T. Developed a faster synthesis language (Music 360 written for the new IBM 360 in assembly language). Later (1985) created Csound, essentially a translation of Music V/360-style programs to run on personal computers.

>>>show and tell: Csound score example (see example in this study guide)

John Chowning: pioneered linear frequency modulation (which began as an experiment in "extreme" vibrato) as a means of controlling timbre at Stanford University [1970's]; revolutionized the synthesizer industry when he licensed his work to Yamaha, Synclavier, and other synthesizer manufacturers.

>>>show and tell: Basic FM synthesis on Yamaha DX7

Lejaren Hiller: University of Illinois; pioneered computer-generated (or computer-composed algorithmic) music, essentially using contained random number generators (using a generate-and-test principle). Originally computer-composed notes were simply transcribed by hand to standard music notation and performed by acoustic musicians. Notable collaboration with John Cage to produce *HPSCHD* (1969 premiere: 7 harpsichords, 51 audio tapes, 5000 slides of spaceship and rocket technology, abstract designs and films of space technology — "all process; no structured beginning, or end ... as Cage describes it "...like the weather."

Jon Appleton: pioneered the digital synthesizer; developed (with Sydney Alonso and Cameron Jones) the Dartmouth Digital Synthesizer [1972, the direct predecessor of the Synclavier [1976] which opened up vast frontiers in digital synthesis and digital audio to those who could afford it.

David Rosenboom and Larry Polansky: Music Mills College Center for Contemporary (CCM) — created HMSL (Hierarchical Music Specification Language), an object-oriented musical programming language.

Roger Dannenberg: Carnegie Mellon University, computer music pioneer who taught computers to "read" and follow musical scores by "listening" to performers so they could synchronize and participate as "just another performer."

Laurie Spiegel: after pioneering work with Max Matthews and the GROOVE system; Created *Music Mouse*

David Zicarelli: (IRCAM for a while) created numerous computer application which have had wide use, including one of the first synthesizer editor/librarian programs for the popular Yamaha DX7. also ... *M, Jam Factory*, expanded, extended, and commercialized *MAX* and later *Max/MSP*

Miller Puckette: wrote the original version of MAX while at IRCAM (c. 1987), later added digital audio extension (MSP)

Herbert Brun, Curtis Rhodes, and especially Barry Truax: were all early pioneers of "Xenakis-inspired" granular synthesis

Perry Cook: pioneer of computer-based physical modeling of acoustic instruments.

Karla Scarletti, Kurt Hebel, and Lippold Haken: designers and builders of Kyma/Capybara integrated software/hardware digital audio workstation, now its 5th version.

Emile Tobenfeld: "Dr. T" ... early pioneer of the commercial music software industry

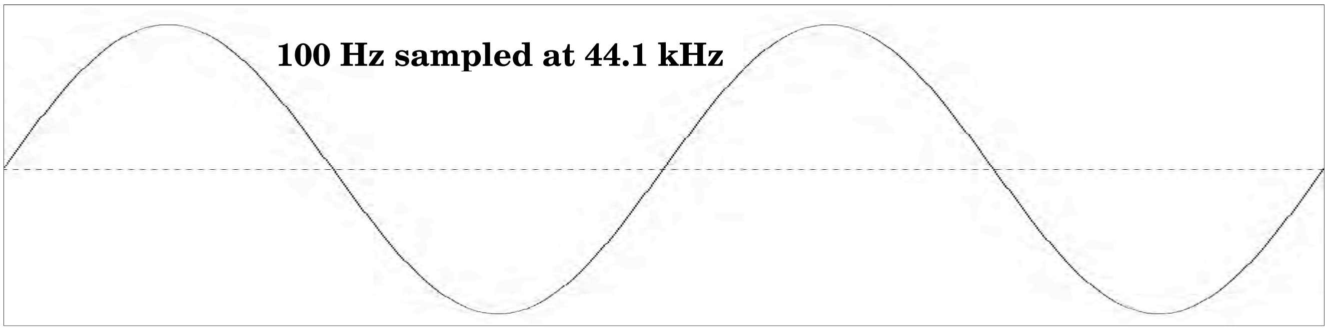
digital quantization & storage of an analog signal

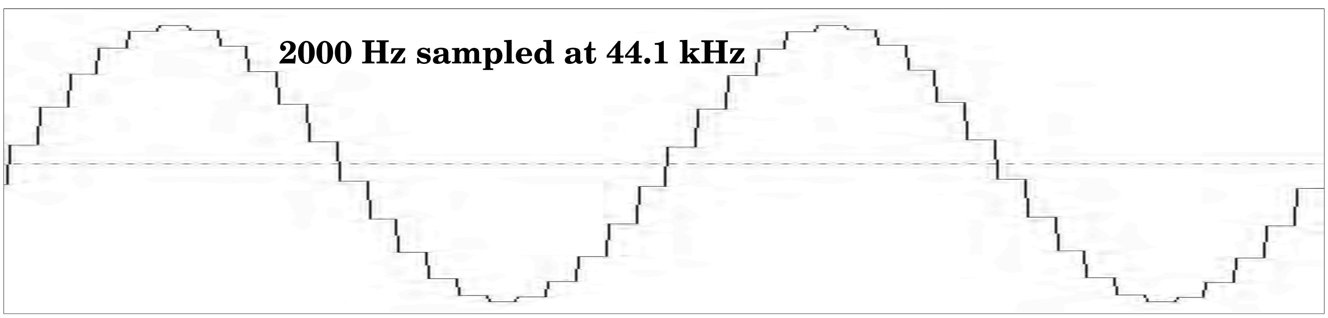
analog to digital conversion ("A" to "D" converstion)

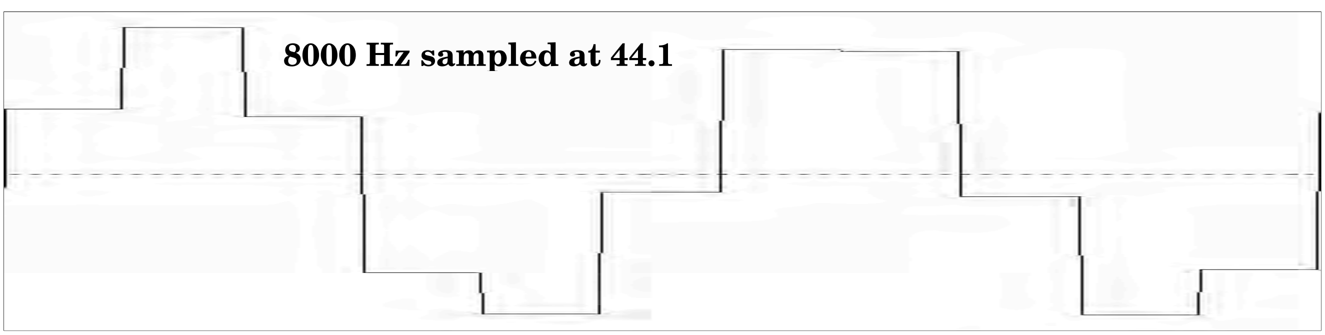
- sampling rate (horizontal axis resolution) determines accuracy of frequency and shape
- number of "bits" per "byte" (vertical axis resolution) determines accuracy of amplitude

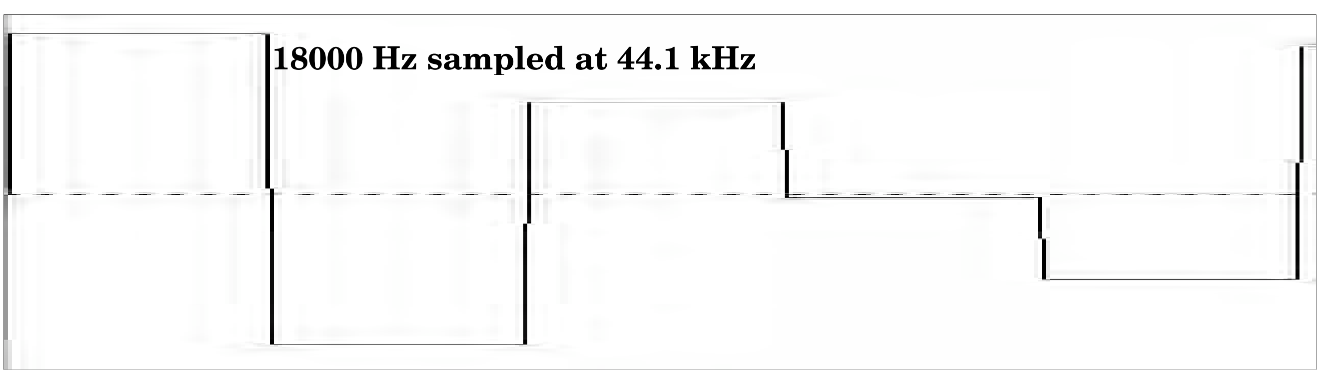
Various Frequencies of Sine Waves Sampled @ 44.1 kHz

Clearly, the higher the frequency more jagged and less accurately the waveform is represented.









Nyquist's Theorum

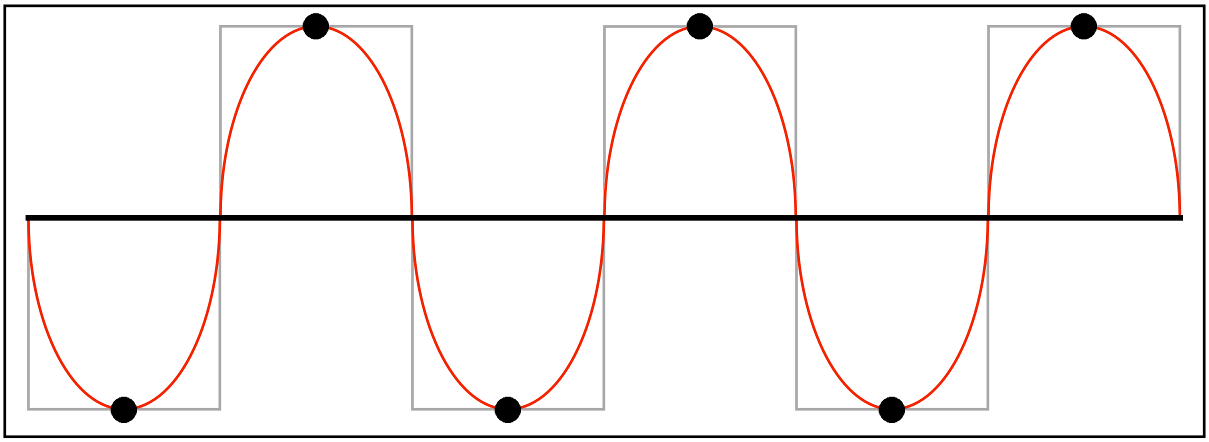
Nyquist's theorum states that the sampling rate must be at least two times greater than the freqency to be sampled. Or ...

$$\text{Rate}_{(\text{sampling})} \geq 2 \text{ x } \text{Freq}_{(\text{sampled})}$$

Example Diagrams:

If a 3000 Hz frequency is sampled at 6000 Hz, then ...

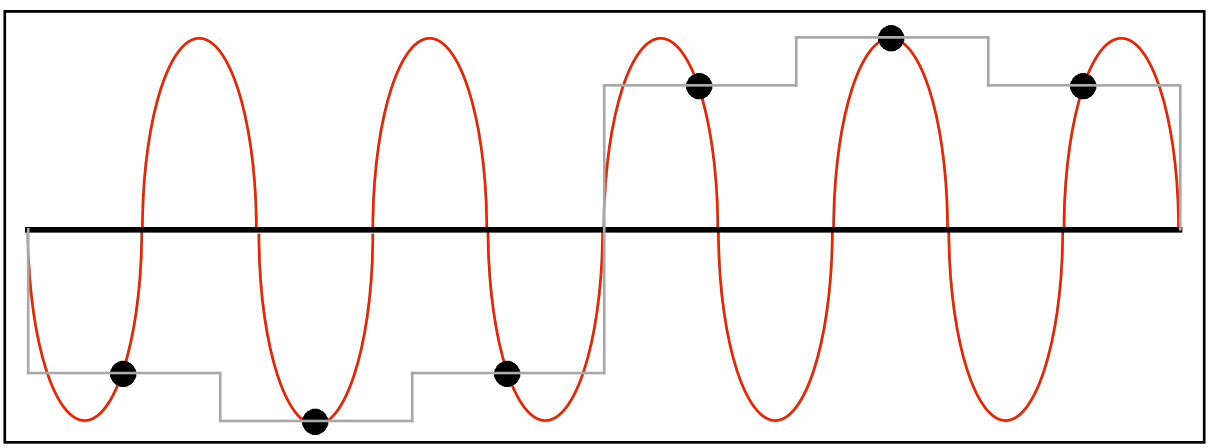
the frequency to be sampled = 1/2 the sampling frequency.



Note: While you lose any ability to track waveform shape, you still manage to capture the frequency.

If a 5000 Hz frequency is sampled at 6000 Hz, then ...

the frequency to be sampled = 0.833 x the sampling frequency.



Note: You have completely lost the ability to track the original frequency of the sinewave. Instead you have captured a frequency that is only 40% the original ... or 2000 Hz. This is sometimes called a "foldback" frequency.

A Simple Csound Example

This simple Csound example is presented to give you some idea of the processes involved in direct digital sound synthesis. You can easily glimpse both the strengths and weaknesses of the medium. Looking and the values for parameters (p1 – p7, etc.) you can see the tremendous precision and control that direct synthesis affords the composer. On the other hand, the language and syntax take considerable time and effort to master, with very little help provided by training and study in general musicianship.

---

a basic FM orchestra file

NOTE: all text after a semi-colon (;) is merely explanatory and is not part of the code.

```
sr = 44100 ;sets sample rate to 44.1kHz
kr = 4410 ;sets control rate to "1 every 10 samples"
ksmps = 10 ;sets control rate to "1 every 10 samples"   (i.e. 44100/4410 = 10)
nchnls = 1 ;sets the number of output channels to 1

;default parameters
;p1 = name
;p2 = start time
;p3 = duration
;p4 = amplitude
;p5 = frequency
;p6 = attack time
;p7 = release time


instr 1
k1   linen      p4,                          p6,   p3,     p7
a3   oscili     p4*.25,     p5,                            1
a2   oscili     p4*.5,      p5+a3,                         1
a1   oscili     k1,         p5+a2,                         1
out a1
endin
```

A simple score file

;uses **LINEN** for E.G. control on each note but manually sets note amplitude (i.e. p4)

```
f1     0    4096        10        1

;p1   p2        p3        p4        p5            p6            p7

i1     0        2       10000      440          .50          1.00
i1    .5        3        8000       880        1.00           2.00
i1     1        3       15000       220          .7           2.00
i1     2        4        6000      1320        1.25           2.50
i1     3        2       10000       660         .50           1.75
i1     4.5          5       20000       110          .75           4.00
e
```

NOTE: see folder named "Csound.OU" on lab computer hard drives

Listening Guide: Exam 3

Jean-Claude Risset: *Computer Suite for Little Boy* (1968)

Risset (b. 1938, France) is one of the early pioneers of computer music. He studied music as a pianist and composer; worked over three years with Max Matthews at Bell Laboratories on computer sound synthesis (1965-69); Pierre Boulez hired Risset to set up and head the Computer Department at IRCAM (1975-79).

This work was realized in 1968 at Bell Laboratories. It is one of the first substantial musical works entirely synthesized by computer: all the sounds used in the piece were produced with the Music V program (developed by Max Matthews, et al.). This *Computer Suite* is excerpted from music originally composed as incidental music for the play, *Little Boy* by Pierre Halet, which combined computer music with music for a chamber ensemble. The play is a fictionalized look at the recurrent nightmares, guilt, and mental anguish experienced by one of the reconnaissance pilots after the Hiroshima bombing. "Little Boy" refers to the A-bomb itself, with which the pilot personally identifies. Some of the play's action is "live" and some occurs only in the mind of the pilot. *Flight and Countdown* follows the pilot's reliving in dream the experience of the flight, ending with a (clearly audible) countdown to the release of the bomb. In addition to imitations of various instrumental timbres, using 1960's state-of-the-art digital synthesis which was a big part of the research he and Matthews were pursuing, you will hear many glissando (or sliding pitch) gestures used to simulate the flight of the plane.

Jean-Claude Risset: *Songes* (1979) — trans. "Dreams"

This work was realized in Paris at the Institut de Recherche et de Coordination Acoustique/ Musique (IRCAM) using a variant of the Music V program. The original program (Max Matthews, et al.) was augmented to enable the processing of digitized sounds, as well as digital sound synthesis. The work begins with these digitized sounds, extracted from recordings of material from an earlier instrumental composition by Risset. These sounds are digitally mixed and processed, often to produce spatial effects. A long-range organization of frequency can be heard in the work: at the beginning frequencies are generally near the middle of the audio spectrum; a gradual expansion of the frequency component takes place, filling the spectrum from low to high; after which the mid-range frequencies drop out, leaving only the lows and highs.

Charles Dodge: *Earth's Magnetic Field--Part I* [1970] (2 excerpts) .... (Scrhader p. 156)

This is an early example of computer music. The "score" for the work is derived from **Bartles' Diagrams**, which chart measurements recording the fluctuations of the earth's magnetic energy in a graphic representation not unlike traditional music notation. In fact they are sometimes referred to as *Bartels' musical diagrams.* Dodge translated the data for the year 1961 into music, where the vertical axis is converted to pitch and the horizontal axis to duration. In Part One, the pitch component is mapped to a diatonic major scale (7-notes per octave; do-re-mi-fa-so-la-ti-do, etc.) . It is monophonic (one note at a time) since Bartles' measurements have only 1 value at a time. Complexity is introduced later in the work using overlapping envelopes, complex timbres having audible high-pitched overtones, and by the addition of reverb.

John Chowning: *Stria* (1977) — def. "one of a series of parallel lines, bands, or grooves"

Chowning (b. 1934, New Jersey) studied composition with Nadia Boulanger in Paris and received a doctorate in composition from Stanford University. In 1964, while still a student, he set up the first on-line computer music system ever, with help from Max Matthews of Bell Laboratories and David Poole of Stanford's Artificial Intelligence Laboratory. In 1967 Chowning discovered the audio-rate linear frequency modulation algorithm. This breakthrough in synthesis allowed a very simple yet elegant way of creating and controlling complex time-varying timbres. His work revolutionized digital sound synthesis once Yamaha and Synclavier licensed the technology and packaged it for "non-computer jocks."

Stria was begun in late 1975 at the Stanford University Center for Computer Research in Music and Acoustics (CCRMA) as a commission for presentation at Luciano Berio's exhibit of electronic music at IRCAM in 1977. Chowning was the pioneer of linear frequency modulation (FM). This work, his third major work to use FM, makes consistent use of non-harmonic ratios and non-tonal divisions of the frequency spectrum. In other words, it does not attempt to synthesize traditional sounding instruments (though surely it was this aspect of FM's power that initially attracted Yamaha to license the technology).

Instead very gradually evolving clangorous bell-like sounds are used throughout the piece. Several levels of the piece, from "microscopic" frequency ratios to overall form, make use of the Golden Section ratio (y/x = x/(x+y) or roughly .618). Chowning has said, "*Stria* was the most fun piece I have ever composed. It was all with programming and that was incredibly enlightening to me." Unlike many of his works, *Stria* uses no random number generators. Instead, Chowning wrote a SAIL–program (Stanford Artificial Intelligence Language) which would generate note lists and such from very compact programming cells. Slight changes in a cell would result in different but deterministic results — meaning the results would be the same if the identical program were run again.

Jon Appleton: *Georganna's Farewell* (1975) .... (Scrhader p.141-147)

In 1972, Appleton (b. 1939, Los Angeles) began developing the Dartmouth Digital Synthesizer with engineers Sydney Alonso and Cameron Jones. This instrument was the direct predecessor or prototype of the Synclavier. The most striking difference between the two was the lack of an organ- or piano-like keyboard on the earlier instrument. See Barry Schrader's *Introduction to Electro-Acoustic Music*, pages 141 -147, for a discussion of this instrument and its close relatives as well as a detailed discussion of *Georganna's Farewell*.

Frank Zappa: *The Girl with the Magnesium Dress* (1984)

One of Frank's early compositions created on his Synclavier (liner notes credit The Barking Pumpkin Digital Gratification Consort), it was released on the album *Perfect Stranger* which also contained a number of Zappa's instrumental compositions conducted by Pierre Boulez, founder of IRCAM in Paris. Frank says it's about "a girl who hates men and kills them with her special dress. Its lightweight metal construction features a lethally pointed sort of micro-Wagnerian breastplate. When they die from dancing with her, she laughs at them and wipes it off." (?? OK Frank, whatever you say!) Meanwhile, notice that the work contains an interesting mix of realistic, sample-based instrument simulations, for which the later versions of the Synclavier were famous, along side of more artificial or synthetic sounds based on the original FM and additive synthesis of the older original instrument. Note also the super-human speed and improvisatory "keyboard" style of the material and its phrasing.

Patric Cohen: *Drum Machine* (1990)

Patric was an Illinois University doctoral student from Sweden when he composed this work. He writes the following:

> The task in *Drum Machine* was to drum the instruments's built -in expectations out of earshot and to machine the unexpected results into a piece of music. The work was created in the composer's home with a MIDI sequencer and a Roland D-10 synthesizer, part of which can be used as a drum machine.

Note: I have included this as an example, to demonstrate that even back in the 1990's , you didn't necessarily need a whole lot of fancy equipment to produce unusual and interesting work, just original ideas and a fresh approach to your instrument. Beginning with ideas extremely familiar to anyone who's ever heard a drum machine, Cohen has built a surprising composition by pushing these ideas beyond the artificially imposed limits of conventional usage: for example speeding up a "drum roll" until the frequency of "hits" enters the audio spectrum and is perceived as a pitch.

Paul Lansky: *Notjustmoreidlechatter* (1988)

Paul Lansky (b. 1944) went to High School of Music and Art, Queens College & Princeton University where he is currently Professor of Music. He formerly played horn in the Dorian Wind Quintet.

Notjustmoreidlechatter was realized in 1988 using the computer music facilities at Princeton University. It is the third in a series of chatter pieces in which thousands of unintelligible synthesized word segments are thrown (with relatively careful aim) into what at first appears to be a complicated mess, but adds up to a relatively simple and straight forward texture. There is a movement towards almost intelligible comprehension in the middle of the piece as the chattering comes "stage front." But then the illusion is lost and with it the hope of ever making any literal sense of it. As Lansky describes it, "All that is left is G-minor, which is not a bad consolation."

Wendy Carlos: *That's Just It* (1986)

That's Just It is from Carlos's *Beauty in the Beast* CD. The liner notes contain the following curious bit of "propaganda":

> All the music and sounds heard on this recording were directly digitally generated. This eliminates all the limitations of the microphone, the weak link necessary in nearly all other digital recording including those which include "sampling" technologies.

The pitch material from this composition comes from a "super-just intonation" scale which Carlos calls the *Harmonic Scale*. This is a just intonation scale having 144 notes per octave which allows for modulations to remain based on just intonation in the new temporary key, or in another words, it allows free transposition of a just scale to occur without retuning the synthesizer. The sounds are generated on her custom built Synergy digital synthesizers.

Live Acoustic Performance with Fixed Electronic Music Accompaniment

James Mobberley: *Caution to the Winds* (1990 — with Richard Cass, piano)

Mobberley (b. 1953) is Professor of Music at the University of Missouri-Kansas City and Composer-In-Residence of the Kansas City Symphony. He has received a Rome Prize and a Guggenheim Fellowship.

Caution to the Winds was commissioned by the Missouri Music Teachers Association. It is the fifth in a series of compositions for soloist and tape where only sounds derived from the solo instrument are used in the tape accompaniment. In this piece piano sounds were recorded, then digitally sampled and processed on a Fairlight Series IIX to complete the accompaniment. Serious memory limitations on the Fairlight imposed a strict limitation on the length of the samples available to the composer. To Mr. Mobberley's credit he solves the dilemma ingeniously with a powerful composition whose frenetic energy and momentum leave no room for long samples, which might only have impeded the flow of the energy anyway.

Mark Phillips: *Elegy and Honk* (2001 — with Donna Conaty, English horn)

The entire accompaniment was created using Symbolic Sound's Kyma system, which consists of a graphic software interface (running on a host computer) that communicates with a Capybara, which is essentially a box full of motorola DSP chips and sample RAM outfitted with input/output jacks for audio and data. MIDI (*Digital Performer*) was used extensively to control and synchronize the Kyma sounds. Refer to Study Guide Appendix for several Kyma screen images from this composition.

Not surprisingly given it's title, this work has two highly contrasting movements. The entire accompaniment to the *Elegy* is derived from a few English horn sounds — short notes in various registers, some "airy" key clicks, and a whoosh of air rushing through the instrument without the reed in place. The dense clouds of sound in the background come from a single very short English horn "blip" subjected to a process called granular synthesis, in which the original audio is exploded into tiny fragments that are subjected to a wide range of transpositions and recombined into overlapping layers . A MIDI breath controller is also used to shape and control some of the Kyma sounds.

There is a long historical tradition of lyrical and serene elegies, but throughout the early stages of my composition process, a brooding, edgy, ominous undercurrent kept "insisting" so persistently it belonged in this elegy that I finally gave in to it. While the work was premiered in August of 2001, the second performance of the Elegy took place a few weeks later on September 13 at an outdoor ceremony held in response to the events of September 11. For both the performer and myself, as well as many in the audience, the music was permanently redefined. The intensity and dark mood of the work seem absolutely "right" for an elegy.

A long transition, serving as an introduction to the second movement, gradually and slyly alters the mood of the work completely. This raucous movement expands on the palette of English horn source material to include a menagerie of geese and duck sounds, joined by an old-fashioned bicycle horn that I analyzed and resynthesized in Kyma, using sine wave additive synthesis ,so that it can be transposed over a wide range and still fit the driving rhythm of the movement. Granular synthesis also plays a significant role in the "jam session" between the two main sections of the movement.

Live Performance of Electro–Acoustic Music Without Tape or CD Playback

Gary Nelson: *Variations of a Theme (and a Process) of Frederic Rzewsky* (1987)

Gary Nelson is the founder and director of the TIMARA center (Technology in Music and Related Arts) at Oberlin College, where he is a professor. He is a former tuba player and has long been a pioneer in algorithmic composition, computer interactivity, and artificial intelligence. The excerpts on this tape are from a live concert performance given at the Ohio University (9/28/89). Nelson performs on his MIDI Horn, an instrument he designed and built which combines elements of a brass (valve) instrument with a typewriter into a wind controller for MIDI synthesizers. The instrument has a trumpet mouthpiece with an air pressure sensor; keys which allow 12 chromatic alterations of the pitch in a pattern derived from a 4--valve fingering chart (like many tubas) to be played by one hand; keys operated by the other hand for determining which octave the instrument will sound in (again using a fingering pattern derived from brass instrument fingerings); programmable joysticks to be operated by the thumbs; and a "shift" key which turns all the other keys into a hexadecimal numeric key pad for sending non-musical data to his computer (a Macintosh) or to his bank of MIDI synthesizers (various Yamaha FM tone generators derived from DX7 technology).

Nelson's music is based on live, real-time interactivity with MIDI computer programs he has designed. No tape is used; all of the sounds are under the control of the soloist (Nelson) and are generated in real time by various tone generators. The nature and degree of control over his computer-generated "accompaniments" varies from one piece to the next. Though it is not always the case in Nelson's music, it is relatively easy in this piece to distinguish what music is being played in "real time" by the soloist (i.e. the melodic theme itself and its variations) and what music is generated by the computer. The computer-generated accompaniment is generally busier and more percussive than the theme. The pitches of the melodic theme (unaccompanied at the beginning) are actually programmed into the computer, so the performer doesn't need to execute the fingerings for each new note of the theme; a simple articulation of the tongue tells the computer you want to play the next note of the theme (pretty cool, huh!). Variations include transposition, inversion, retrograde, and retrograde inversion, and are all initiated by various key combinations on the MIDI Horn. In addition, the performer can, at will, directly influence many aspects of the accompaniment, such as volume and timbre, as well as subject it to the same sorts of variations as the theme. The general character of the accompaniment — i.e. tempo, melodic intervals, number of "percussion players" available — may not be freely altered during the course of the performance. Such changes would generally require modifying the specific program for the piece beforehand.

Russell Pinkston: *Lizamander* (2004) *for flute and Max/MSP* — with Elizabeth McNutt, flute

Russell Pinkston (b. 1948) is a composer, author/researcher/teacher in the field of electro-acoustic music, and a professor of Music at the University of Texas. *Lizamander* (composed for the soloist on this recording) is the second in a series of works for solo acoustic instruments and Max/MSP. The computer captures material played by the soloist during the performances and uses that material (as well as some prerecorded sounds) to generate a rhythmic accompaniment. The computer track the pitch of the flute throughout, and uses this information to follow a preset list of cues, which trigger a variety of real-time processing techniques, including time stretching, pitch shifting, and harmonization. Refer to Study Guide Appendix for Max/MSP screen images.

Jomenico: *Iteration 31* (2004) (Jomenico = Margaret Schedel, John P. Young, and Nick Collins)

"We are sitting in a room, indifferent to everyone but the now. We are distorting the sound of Alvin Lucier's voice, and we are going to fray and hack into the gloom again and again, until the recontextualised frequencies can bloom and be a force themselves ... so that any semblance of his piece, with perhaps the deception of schism, is enjoyed. What you will hear then, are the supernatural resonant sequences of the tune articulated by beats. We regard this activity not so much as a demonstration of a creative act, but more as a to groove out any regularities we geeks might have."

The trio performed this work on live on stage with three laptop computers at the SEAMUS 2004 National Conference.

Exam III Study Guide: Part A — Listening: (ca. 39 points on Exam)

Learn which composition on your Tape #3 uses the following: (one answer for each)
(i.e. each of the following is distinctive to only one composition)

- computer: with early version of Music V program
- Music V — enhanced to process digital audio
- Bartels diagrams, which measure the Earth's magnitic field (EMF).
- computer: with FM synthesis and SAIL program
- Dartmouth Digital Synthesizer (an early digital synthesizer <u>without</u> a music keyboard)
- Synclavier digital synthesizer/sampler
- MIDI–controlled Roland D–10 synthesizer
- computer synthesized speech fragments
- custom designed digital synthesizers with extensive micro-tuning capabilities
- Fairlight digital synthesizer/sampler
- MIDI–controlled Symbolic Sound Kyma system
- realtime audio pitch-tracking by computer running Max/MSP software.
- live performance on electronic instrument: a MIDI wind controller
- live performance on laptop computer(s)

Part B — More on Listening Examples: (ca. 20 points on Exam)

Learn which composition(s) on your Exam #3 list use the following: (several answers for each)
(i.e. many of the compositions share characteristics with each other)

- synthesizer or controller having a music keyboard … *(look for ca. 3 answers)*
- MIDI — synthesizers and/or sequencers … *(look for ca. 3 answers)*
- a performance requires a live acoustic instrumentalist … *(look for ca. 3 answers)*
- digital samples of acoustic instruments or other real world sounds … *(look for ca. 7 answers)*
- direct digital synthesis using computer equipped w/ digital-to-analog converters … *(look for ca. 5 answers)*
  > all on the first page of listen guide, plus Paul Lansky.
- computer as "performer" in a live performance setting … *(look for ca. 3 answers)*

MIDI (an acronym standing for **<u>M</u>**usical **<u>I</u>**nstrument **<u>D</u>**igital **<u>I</u>**nterface)
MIDI was developed in the early 1980's and establishes a common language which synthesizer manufacturers and software designers can use to communicate with each other's products and with computers. It is a digital (or binary) language which uses a serial format, meaning that messages are sent one "bit" at a time. By using a high enough transmission speed (31.25 Kbaud or 31,250 bits per second or over 3,000 MIDI events per second) most timing problems are minimized. (Note: a serial format was chosen to keep MIDI more economical and to simplify the connection between MIDI devices.) A standard MIDI cable is capable of handling 16 individual channels of MIDI data.

Common MIDI messages: note on/off on/off velocity pitch bend patch changes after touch (or pressure) MIDI controller data system exclusive messages

Common MIDI Controllers: Mod Wheel (#1); Foot (#4); Volume (#7); Pan (#10); Sustain (#64)

General MIDI:
The General MIDI standard is a subset of current MIDI standards. By insuring that certain minimum standards are met (see below) and by imposing limitations on timbral diversity it increases compatibility between synthesizers.

- How many different instrument sounds ("patches") must be stored in a General MIDI synthesizer? (≥128)
- How many simultaneous "patches" are required to comply with General MIDI standard? (16)
- How many polyphonic voices (or notes) are required to comply with General MIDI standard? (28)
- Which MIDI channel is reserved for the drum tracks on a General MIDI synthesizer? (MIDI channel 10)

Yamaha DX7 and FM Synthesis

How many sine wave oscillators or operators are there in a DX7 or DX7II algorithm? (6)
Which of the following sound parameters can be affected by the DX7II modulation wheel?

• LFO Modulation (both AM & FM) • Amplitude (E.G. Bias control of Carrier) • Timbre (E.G. Bias control of Modulator)

In the DX7's FM synthesis, what specifically substitutes for changes in filter cut-off frequencies used to control timbre in subtractive synthesis. (amplitude changes in program oscillator)

What is the difference between a MIDI channel and a MIDI sequencer track?

- A sequencer track is the information while a MIDI channel is the conduit through which the information passes from sequencer to synthesizer.

Electric Sound, by Joel Chadabe — Selected Readings

>>Chapter 5: Computer Music *and* >> Chapter 7: The MIDI World

Risset *(Suite)* — pp 127-128

Chowning *(Stria)* — p 127

Lansky *(Notjustmoreidlechatter)* – pp 133-134

Gary Nelson — pp 225-226

Introduction to MIDI

an article for *Mixdown* magazine, Jan. 1999

by Mark Phillips

Some History:
MIDI is an acronym for Musical Instrument Digital Interface. Essentially it is a digital language, optimized to control electronic musical instruments, with standardized hardware connections and transmission speeds designed to allow devices by different manufacturers to communicate effectively with each other. MIDI was the brainchild of Roland's Ikutaro Kakehashi. He initially collaborated with Sequential Circuits's Dave Smith and Oberheim Electronics's Tom Oberheim in the early 1980s. Very soon a number of other Japanese companies — notably Yamaha and Korg — had joined the effort. Every aspect of the development of MIDI from the specs to its name was the result of much back and forth discussion and compromise. Differences were generally resolved in favor of keeping MIDI simpler in its design, more efficient in its use of hardware resources, and cheaper to produce. This was in keeping with Kakehashi's vision of MIDI as serving the consumer market — and in contrast to one of the more noticeable industry trends of that time toward ever bigger, more integrated and powerful, and more expensive products for well-financed industry professionals (e.g., Fairlight, Synclavier, GDS) At the January 1983 NAMM show, two off-the-shelf units — a Roland JP-6 and a Prophet 600 — successfully communicated with each other when simply connected together by MIDI cables. Later that year Yamaha released the DX7, which became the first big hit of the MIDI era, selling over 200,000 units. The MIDI revolution, aided significantly by the concurrent explosion in the personal computer industry, was in full swing. Indeed the real power and flexibility of MIDI is best realized in a system that connects several MIDI devices to a personal computer. In the ensuing years, adventurous programmers, engineers, and musicians have extended MIDI applications far beyond their original intent, but the basic protocols and limitations are still intact. By the end of the 1980s, the industry had designed a subset of MIDI features called General MIDI for the purpose of further standardization of several aspects of synthesizer design. General MIDI increases compatibility between synthesizers in several ways: by ensuring certain minimum standards (128 different preset sounds in ROM; 16 different simultaneous preset sounds, or "patches"; at least 28 simultaneous or overlapping notes); by imposing limits on timbral diversity (preset patch #1 will always be a piano; preset patch #127 will always be a gunshot, etc.); and by standardizing most non-pitched percussion (drums and percussion will always occupy track 10; each instrument is assigned as specific note on the keyboard — e.g., the bass drum will always be triggered by note #36, the lowest C on a standard 60-note keyboard).

Some MIDI Basics:
MIDI can be used to send and receive a variety of messages between MIDI devices. Note-on and note-off messages are perhaps the most basic, but there are many more. Key velocity (how fast or hard the key was struck); pitch bend; key and channel pressure (how much downward force is maintained on the key after the initial attack); patch change messages (which call up different stored presets from RAM or ROM); data from modulation wheels, sliders, foot pedals and switches; are all typical types of events handled by MIDI. An important distinction to be understood is that MIDI does NOT generally deal directly with audio. (Years after its introduction, when MIDI samplers began to be popular, a method of sending and receiving samples over a MIDI network was developed — the MIDI Sample Dump Standard— but it is nowhere near fast enough to actually listen to the results as audio.) In order to actually hear the results of MIDI transmission, you need to send the MIDI data to a synthesizer, which then creates sounds based on the incoming MIDI messages.

Some Technical Stuff:
MIDI employs a serial interface for communication — meaning events are handled one at a time sequentially (imagine a single train traveling in one direction on a track). This was, in fact, one of the biggest compromises in establishing a consumer oriented standard. Parallel communication (imagine a large multi-lane freeway with cars whizzing in many lanes and both directions) was favored by some of the developers because of its potential for handling far more events and data in a given time, but the cost difference would have been significant. Parallel communication would also seem the logical choice if one thinks about all the events in music that seem to happen simultaneously to our ears. However, by using a serial transmission with a high enough speed (31.25 Kbaud or 31,250 bits per second) it is possible to send over 3,000 MIDI events per second. This is more than enough speed to fool most ears into accepting the illusion of simultaneity and to avoid timing glitches for most casual users. More advanced and demanding users may notice problems, however, if they begin to approach that maximum density of data, or if they require a large number of events to be executed simultaneously.

MIDI devices are equipped with both a transmitter and a receiver that make use of a computer industry-standard connector— the 5-DIN pin —though only two of the pins are actually used by MIDI. Again the adoption of this connecting system with 3 "wasted" connectors was controversial, but at least the use of a computer industry standard connector instead of a simple stereo connector helped to strengthen the distinction between the data flowing through a MIDI system and the analog audio and control signals to which electronic musicians where accustomed. It also virtually eliminated the possibility of delicate computer components being subjected to these other signals. Connecting the MIDI OUT jack of one unit to the MIDI IN jack of a second unit allows the first to send MIDI messages to the second. Assuming both units are keyboard synthesizers, you could now access and play the sounds of the second keyboard using the keys buttons, wheels, and knobs of the first. If you also connect the second keyboard's MIDI OUT to the first keyboard's MIDI IN, you could then control either keyboard from the other. To connect a third synthesizer to the system, you could connect the MIDI THRU jack of either the first or second keyboards to the MIDI IN of the third synthesizer. This new addition to the set-up would then be controllable by whichever keyboard did not have the MIDI THRU jack in use. In other words, the MIDI THRU jack sends out data received by its own MIDI IN jack, not messages generated from within itself.

To connect these standard MIDI devices to a computer generally requires a computer-specific MIDI interface — with the standard MIDI IN and MIDI OUT jacks (at least one of each) — which is connected to one of the computer's serial ports. This could be accomplished with an internal card or with a stand-alone unit. Depending on your choice of MIDI devices, you may be able to skip the dedicated unit since many keyboard synthesizers and tone generators now have built-in switchable interfaces for PC and Macintosh computers. However, as a MIDI system becomes larger and more complex, the increased convenience, flexibility, power, and features of a stand-alone unit become more valuable.

Some Really Technical Stuff

MIDI is an 8-bit binary language with two main types of bytes: status bytes and data bytes. The status byte uses the first 4 bits to announce what type of data byte(s) will follow and uses the second 4 bits to indicate which MIDI channel is affected by those data byte(s) it precedes. With four bits it is possible to count from 0 to 15 — so this limits MIDI to a maximum of 16 different channels for the data bytes to use on any single MIDI cable. Other bytes of MIDI data make use of the first bit to indicate whether what follows is a status byte or a data byte (1=status byte; 0=data byte). This leaves 7 bits for the actual message, so the range of expressible numbers runs from 0-127 (or 128 different values). This limitation affects a broad range of MIDI messages. Possible notes numbers run from 0-127, with middle-C being note #60. Key velocities (both down and up), Modulation wheels, keyboard sliders, foot controllers, patch change messages (which change the synthesizer from one preset to another), channel pressure (how hard you continue pressing down on the key after it is sounded), etc. all have a maximum range of 0-127. Even the number of possible MIDI controllers total 128. Many of the more common controllers have assigned numbers (Modulation = 1; MIDI Volume = 7; Pan = 10, Sustain Pedal = 64, etc.), while others are left undefined for future expansion. Pitch wheel messages often use two data bytes in order to enhance the resolution of continuous pitch changes and smooth out potential bumps. MIDI also uses System Messages which address devices irrespective of MIDI channel. System Common Messages are primarily used for timing messages of the sort required by devices with built-in sequences (more on sequencers later). Another powerful type of message, System Exclusive Messages (or SysEx Messages) allow for communication with a specific instrument in the system and are used for extending the real-time programmability of a device. It is this type of MIDI message that is at the heart of all external or computer based synthesizer editor/librarian packages.

This binary number crunching goes on in the background and is largely unobserved by the average MIDI user. Certainly it is not essential to become fluent in 8-bit binary conversions in order to use make good use of MIDI. Most synthesizer interfaces and certainly all modern computer-based sequencing software bury most of this number crunching behind layers of sophisticated, user-friendly graphics. But every once in a while, when trouble-shooting or looking for new ideas, it can be helpful to understand a little about what's really going on in the background. For example, it is useful to understand that sending a MIDI note-on message for note #60 does not ensure that you will get an in-tune middle-C. The receiving synthesizer may be set up to transpose it up or down anywhere from a few cents to an octave or more. This is in fact one way to exceed the 128-note range of MIDI. Also a given pitch bend message will have a different result when sent to a synthesizer set up with a pitch bend range of +/- 2 half-steps versus one set up with a range of +/- 1 octave.

Some Common Uses For MIDI:

Aside from simply connecting one MIDI synthesizer to another and playing two at once, one of the main things MIDI is used for is sequencing. This can be accomplished with a built-in or on-board sequencer in a keyboard or drum machine, or a dedicated stand-alone device, or a computer-based software sequencer. The built-in devices offer the convenience of easy portability (and perhaps to a certain extent stability).

Computer-based sequencers offer a host of user-friendly editing features, virtually unlimited storage, ability to print sheet music, etc., and at the high end the ability to combine MIDI with digital audio recording, which greatly expands the realm of possibilities. All MIDI sequencers accomplish essentially the same two tasks. They record and store MIDI data such as note-on, note-off, pitch wheel changes, modulation wheel changes, etc. — either in realtime or otherwise. They also handle playback of these stored events in realtime. The non-realtime aspect of MIDI recording is a powerful difference between recording MIDI and recording audio. The duration and timing of note messages can be preset and then notes may be entered at any speed irrespective of the final desired result. An hour's worth of long, slow drones could be entered in seconds or minutes; a half-second flurry of notes could be entered with a coffee break taken in the middle. Inaccuracies can easily be edited or fixed without the necessity of rerecording a passage. Playback can be sped up or slowed down without affecting the pitch. Transpositions (large-scale or small-scale) can be executed quickly without a change in playback speed. (While modern digital audio workstations may be able to perform these editing tasks that require the ability to handle pitch and speed independently, the process is much slower, more limited in scope, and often results in degraded audio quality.)

In addition to sequencing, some other notable uses for MIDI include the editing and storage of libraries and banks of sounds for synthesizers. Excluding some very low-end models, most synthesizers have some programmable memory for users to create and store their own sounds. Computer-based editor/librarian programs utilize MIDI's System Exclusive features to allow unlimited storage and easier editing of these sounds.

A third common use of MIDI technology is as an aid in preparing printed music. As mentioned before, some computer-based sequencing programs have music printing ability. Likewise many software applications primarily dedicated to music printing use MIDI to speed up the process of getting note information into the computer and for "proofing" the music for wrong notes.

In all of these previous uses, MIDI data is created by the user one event or parameter change at a time and stored for future use. There is another class of MIDI application which involves the use of a computer program to generate MIDI events in realtime according to certain formulas or algorithms. In some cases the user gives the program some input to start the process and then the formula takes over, outputting additional MIDI data. A simple example of this would include a MIDI-based arpeggiator such as can be found in some synthesizers and software sequencers. Holding down a collection of notes generates a series of patterned repetitions and/or transpositions of the held notes. A computer program such as Band-In-A-Box generates MIDI data based on chord and style information entered into the program by the user. Select a different style from the menu and a new algorithm outputs a completely different drum pattern, different bass line, different keyboard chord voicings, and so on. At least one application, Opcode's MAX, is designed specifically to enable users to design their own algorithms to generate MIDI data. Users can create their own mathematics formulas, convert one type of MIDI data into another (key velocity into pitch bend, modulation wheel into note number, etc.), or use various MIDI events to trigger processes.

The future of MIDI:
At roughly a quarter-century old in the rapidly evolving field of technology, MIDI is decidedly "old" and in some ways virtually primitive. (What other area still relies heavily on 8-bit technology nowadays?) However, the tremendous advances in MIDI synthesizer design combined with the growing sophistication and power of MIDI software applications running on ever faster and more powerful computers all bode well for the future of MIDI. These newer computers are capable of processing massive amounts of MIDI data while simultaneously handling complex digital audio. Most high-end computer-based MIDI sequencers handle digital audio to some extent. A new software package, MSP (for the Macintosh), adds very powerful user programmable digital audio processing capabilities to MAX. Most serious digital audio software packages from ProTools to Csound to Kyma incorporate MIDI capabilities. The synergy of these two distinct technologies, MIDI and digital audio, seems likely to extend the functional life of MIDI far beyond the expectations of its founders, while at the same time greatly expanding the flexibility of digital audio applications.

Music Instrument Digital Interface (MIDI)

by Jeffrey Hass, Center for Electronic and Computer Music, Indiana University

Index

Overview

The development of the MIDI system has been a major catalyst in the recent unprecedented explosion of music technology. MIDI has put powerful computer instrument networks and software in the hands of less technically versed musicians and amateurs and has provided new and time-saving tools for computer musicians. The system first appeared in 1982 following an agreement among manufacturers and developers of electronic musical instruments to include a common set of hardware connectors and digital codes in their instrument design. The original goal was to interface instruments of different manufacture to control common functions, such as note events, timing events, pitch bends, pedal information, etc. Though several classes of codes have been added to the MIDI 1.0 Specification (International MIDI Association, 1989) and MIDI applications have grown far beyond the original intent, the basic protocol has remained unchanged. MIDI is a system very much like a player piano roll in that it is used to specify the actions of a synthesizer or other electronic devices, while the tone or effect is generated by the instrument itself.

Summary of the MIDI Hardware Specification

The MIDI specification begins with a set of common hardware requirements. Each MIDI-capable instrument is equipped with a transmitter and receiver, though certain peripheral devices, such as signal processors, may have only a receiver or a transmitter. The interface operates at a 31.25 Kbaud asynchronous (serial) transmission rate. Each byte consists of a start bit, eight data bits, and a stop bit, for a total duration of 320 ms. per byte. While this was adequate for the applications originally envisioned, it has proven to be one of the major stumbling blocks for generating more complex data streams in time-dependent situations. Though several manufacturers originally included alternative parallel interfaces on their instruments for linking their own peripheral devices, the overwhelming success of MIDI has seen the abandonment of almost all other interconnecting formats.

MIDI cables connect instruments by means of MIDI IN, MIDI OUT, and MIDI THRU jacks. The MIDI THRU jack provides a direct copy of data coming into the MIDI IN jack, providing the ability to "daisy chain" several instruments and devices together. Any action on an instrument that corresponds to a particular MIDI code (such as a key depressed, or a program button changed) will normally transmit that action's code to the MIDI OUT, but not to the MIDI THRU. The major potential of MIDI is realized when a network of instruments and other musical devices are connected to a computer by means of a MIDI interface. The primary function of the interface is to match clock speeds between the specified rate from MIDI devices and the computer. An interface's connections may be as simple as one MIDI IN and one MIDI OUT jack with information distributed by means of the daisy chain mentioned above.

In recent years, interfaces have become more elaborate and now providing more MIDI INs and OUTs for more complex and flexible networks, merging facilities to handle several input devices simultaneously, coded routing of data to specified cables, and generation and/or conversion of timing synchronization codes for use with video and multitrack audio equipment. Some interfaces are also capable of selective filtering and/or remapping of MIDI data.

MIDI Data Format

The majority of MIDI communication consists of multi-byte packets beginning with a "status byte" followed by one or two "data bytes". Messages fall into the following five formats:

1.) Channel Voice Messages   2.) Channel Mode

3.) System Common   4.) System Real Time   5.) System Exclusive

Channel Messages

Almost all MIDI devices are equipped to receive MIDI messages on one or more of 16 selectable MIDI channel numbers. A device will respond to messages sent on the channel it is tuned to and ignore all other channel messages, analogous to a television set receiving only the station it is tuned to. The exception to this is OMNI mode. An instrument set to receive in OMNI mode will accept and respond to all channel messages, regardless of the channel number. The most common MIDI message is the Channel Voice Message which conveys information about whether to turn a note on or off, how much key pressure to exert (called aftertouch), etc. (See chart at end of article.) A sample message for turning on a note (middle C) on MIDI channel #5 very loudly (with a velocity or force of 127, the maximum) is shown below in binary.

10010100	00111100	1111 1111
status byte	data byte	data byte

The first four bits of the status byte (1001) tell MIDI that the following message is a note-on command, while the last four bits tell MIDI what MIDI channel the message is for (0000=MIDI channel #1, 1111=MIDI channel #16). The first data byte tells MIDI what note to play (60='middle C), while the second data byte tells MIDI how loud to play the note, in this case the maximum velocity of 127 is sent. The note will sound until a message to turn off the same note number is received.

Simultaneous events in MIDI must be sent as a string of serial commands. A 3-note chord, for example, will be transmitted as three separate note #-velocity pairs. Because of the 31 Kbaud transmission speed, this is normally perceived as a simultaneity. However, as polyphonic instruments (those capable of playing more than one note simultaneously) have increased their number of voices and more MIDI set-ups have networks of ever-increasing numbers of instruments and tone modules, the speed of both the interface's processor and sheer volume of serial data make large simultaneous events susceptible to glitches, undesired arpeggiations, and data errors.

Many instruments transmit and respond to key velocity, the speed at which a key is depressed. Most simply allow dynamic range to be controlled, while others have the capability to alter timbre or spatial location through velocity. Recent instruments often have the capacity to crossfade or switch between two different sounds, based on the intensity (speed) of a keystroke.

System Messages

System Message codes are addressed to all instruments and devices regardless of their channel assignments. System Common messages primarily deal with instruments that contain their own sequencers (more on sequencing below), and allows external control of a sequence starting point. By far the most innovative applications of MIDI have been in the realm of System Exclusive (SysEx) codes.

SysEx codes allow storage of large amounts of patch data, the standard recallable unit by which synthesizers store their sounds. SysEx codes have enabled a class software called editor/librarians to extend the storage, cataloging, and real-time patch editing feature of synthesizers, by incorporating the resources of the microcomputer. In many cases SysEx software has added extremely useful graphic representations of patch parameters not available from the limited LCD's on most instruments. It has also allowed the computer to generate its own patches, randomly or algorithmically.

MIDI Time Code (MTC) is a further example of SysEx implementation . The MTC protocol attempts to bridge the gap between MIDI Clock timing, which uses the relative time measurement of note value and tempo, and SMPTE, which uses absolute values of hour::minute::second::frame. MTC enables a sequencer, dealing in relative musical values, to synchronize with videotape or multi-track audio tape with recorded SMPTE information. A relatively large part of film post-production is now done utilizing some type of timecode synchronization. For example, a sound editor, with sequencing software that is locked to the video via MIDI Time Code, may stop a video clip at frame 00::10::22::10 where a gun muzzle flashes and enter a MIDI Note On command to play a gunshot sample on that specific frame.

MIDI Controllers

Because MIDI instruments were made available in both keyboard and rack-mount module versions, most studios and individuals opted for one keyboard synthesizer controlling an array of tone modules. Many synthesizer keyboards are limited to 61 notes and lack a piano-like feel. To improve upon this, 88-key wooden weighted keyboard controllers, generating MIDI codes but no sound themselves, were introduced as input devices. A prototype keyboard was introduced by Robert Moog, creator of the many generations of Moog synthesizers, which increases the performer's ability to shape notes by sliding fingers forward or backwards and side-to-side on each key, in addition to expressive use of key pressure and velocity. The assignment of controller and/or SysEx codes to these axes is controlled by a programmable PC interface to the keyboard.

Alternative controllers to the MIDI keyboard have developed over the last several years. Gary Nelson, both an electronic composer and tubist developed a controller with a tuba mouthpiece and brass fingerings. Both Yamaha Corporation and Casio have marketed wind controllers with saxophone and/or clarinet fingerings. In both of these instruments, wind pressure generates a Note On command and continues to control the amplitude by converting the pressure into controller information. These controllers require an intervening microprocessor to convert the various fingering combinations into note numbers. MIDI guitars, violins, and drums have become commonplace in the world of commercial music. There is, in fact, a MIDI body controller, a series of programmable pads sewn into a leather jumpsuit which the performer plays on, which has done away with the need for an instrument at all.

The desire to track acoustic instruments or the human voice has led to the development of pitch-to-MIDI converters, such as the IVL Pitchrider . Either a contact mike or standard "air" mike is connected to the converter, which then outputs MIDI note numbers and other programmed parameters, such as volume and pitch bend. Certain problem are inherent in such devices. For example, the converter will often fail to track the fundamental pitch of instruments or voices with strong or complex overtones. In addition, they often have difficulty tracking rapid note patterns. However, despite their current limitations, these converters have opened up a new area of interactive performance between instrumentalist and computer in conjunction with some of the software mentioned below.

MIDI Sequencers

The most common type of software used to link MIDI instruments and microcomputers is a vast array of sequencing programs. The term sequencer was carried over from voltage-controlled analog synthesizers, in which sequencing modules could be set to step through 8 or 16 stages of voltage-controlled notes. The limited number of sequenced notes was increased when small microprocessors, capable of recording and playing back several hundred notes, were added to digitally-controlled analog synthesizers. Current sequencing software allows recording and playback of notes, patch changes, controller information, and even SysEx codes to the limits of the program's memory allocation, usually hundreds of thousand of events. Sequencers normally organize their information along the lines of multitrack audio recorders. Information is recorded and edited on individual tracks which are assignable to one or more MIDI channels. Previously recorded tracks can be played back while new ones are recorded. Current software accommodates up to 200 tracks of information. Rhythm is usually organized along traditional musical divisions. Note events are assigned measures, beats, and beat subdivisions (down to resolutions of 480 per beat). Unlike audio tape, sequenced performances can be recorded at a slow tempo and sped up for playback with no change in pitch. As mentioned above (see section 3.2.2), the inaccuracies of real-time performance can be quantized to the nearest selected note values. Notes can be entered in a precise rhythmic fashion by step-recording them -- that is, selecting a rhythmic value, usually from a computer keyboard, then playing the pitch or chord on a MIDI keyboard and proceeding to the next note. While this facility has led to a plethora of unimaginative, "switched-on Bach"-style musical transcriptions, the computational facilities that usually accompany step-recording features can be of great service to serious composers in need of complex rhythmic precision.

Sequencer tracks can be edited in a wide variety of manners. For example, one may select a recorded area and transpose it, shorten its rhythmic durations, and have a smooth increment of velocity for a crescendo. Since MIDI's numerical structure makes rapid musical operations quite simple, many compositional techniques, such as inversion, retrograde, rhythmic diminution or augmentation, are often included as editing features. Tracks or portions of tracks also may be looped to create continuous ostinato patterns.

Recently, the capability to edit metronomic tempi has been added to many sequencers. Composers can now specify smooth accelerandi or ritards from one measure to another. Sequencing software often provides composers with both musical time (measure, beat, and subdivision) and real time (minutes, seconds, hundredths of seconds) by computing all the metronomic changes in a particular sequence. The computational capability of the sequencer enables film composers, who previously used cues sheets and stop watches, to request the rate of acceleration from measure x to arrive at measure y by a particular SMPTE frame number.

Interactive Performance via MIDI

One of the most innovative applications of MIDI to a real-time performance situation is MAX, a graphic programming environment developed at IRCAM, a center for electronic music research in Paris, by Miller Puckette. MAX, named after Max Mathews, an innovator in computer music research, was originally developed for a Macintosh, but was ported to the IRCAM NeXT-based workstation, and recently re-released commercially for the Macintosh. MAX is a collection of graphic objects, written in C, that a composer can connect or "patch" together onscreen.

MAX patches usually take real-time MIDI input, process the information according to the composer's specifications and output what could be a very sophisticated set of responses to MIDI instruments. MAX objects simplify the programming tasks of parsing MIDI data and timing events or performing musical and temporal calculations. The user can, if he chooses, also program objects to be added to an already extensive library. Many interesting pieces, such as Bestor's Soliloquies (1991) for oboe and Macintosh, involve the instrumentalist playing through a pitch-tracking device interfaced to a MAX patch. The patch tests for certain pitches, phrases, density of note-events, or even dynamic ranges which then trigger a variety of MIDI events on a bank of synthesizers and/or effects devices. A more recent work by Bestor, Cycles (1992), is a sculpture, sound, and lighting installation in which MAX interactively controls both sound and lighting. One of the most recent developments is the extension of MAX from simply a MIDI, or control, processor, to an acoustic processor as well, by the addition of DSP (digital signal processing) objects (Puckette, 1991). Other interactive software applications, such as Intelligent Music's M, allow the performer to program certain boundaries or parameters of response by the computer. MIDI input then triggers an algorithmic set of MIDI-out responses. For example, from the input of a short group of
notes, M might repeat the phrase several times with reordered notes, transpositions, rhythms, or dynamic levels. It was envisioned that performers would develop a knowledge of the program's tendencies and a sense of control in shaping the interactions during performance.

MIDI has been the most readily available source of output for algorithmic composers, such as Gary Nelson. One of Nelson's aims is to tailor extensive computer algorithms to make many of the compositional decisions in real time that he himself would intuitively make and output them to MIDI devices in a performance situation. In compositions such as Fractal Mountains (1989), he improvises with a MIDI wind controller over an algorithmically generated background. In each performance, the computer may made an entirely different set of controlled compositional decisions.

As early as 1970, Mathews and Moore implemented a microcomputer-based GROOVE system to control real-time performance on analog synthesizers (Lincoln, 1972). The computer would record a series of analog voltages from a small keyboard, joystick, and various knobs. The performance playback, through 14 D/A converters, could be altered by editing the disk file. With the advent of MIDI and digital instrument technology, Mathews developed a software/ hardware system that combines the resources of a sequencer and MIDI controller to form the Mathews Radio Baton, also called the Radio Drum. In Mathew's system a score file of MIDI note data can be read and output by his Conductor program, while expressive musical parameters such as tempo, dynamics, timbre, or any of the MIDI controller values can be controlled in real time by the Radio Drum. The controller consists of a flat surface with a complex array of receiving antennae and small radio transmitters placed in the head of drumsticks. By analyzing the signal strengths, the position of each stick in relation to the drum can be determined in three axes. One axis is usually used to "beat" the tempo of playback, while the other five variables can be assigned to key velocity, timbre control, or any other MIDI value in the Conductor program, giving the "conductor" a wide range of expression during performance. Several composers, such as Boulanger, have composed works that involve more than one performer/conductor, each controlling a different bank of instruments.

References and Bibliography

International MIDI Association (1989). "MIDI 1.0 Detailed Specification, Document Version 4.1." IMA, Los Angeles.

Lincoln, H. B. (1972). Uses of the Computer in Music Composition and Research. In "Advances in Computers 12 " (M. Rubinoff, ed.), pp. 73-114. Academic Press, New York.

Puckette, M. (1991a). Combining Event and Signal Processing in the MAX Graphical Programming Environment. Computer Music Journal 15 (3), 68-77.

Chart of MIDI Channel Voice Messages

Note On (includes channel n | note # | velocity #)	Control Change (includes channel n | contoller # | value #)
Note Off (includes channel n | note # | velocity #)	Pitch Bend (…or Change) (includes channel n | pitch bend #)
Channel Pressure (…or Aftertouch) (includes channel n | pressure #)	Program Select (includes channel n | program #)

n = MIDI Channel 1–16 | # = possible values 0–127

NOTES: Even at this point very few synthesizer implement "Polyphonic Key Pressure" (Synclavier and Ensonic are two companies that DO) and even fewer implement "Note Off" velocity. Many controller numbers have become "standard" meaning most MIDI gear will use the same number if they decide to implement the feature, but the most gear implements only a fraction of the 128 possible controllers. Even if it is implemented, it doesn't mean that there will always be 0–127 useful setting values. Several common controllers, such as Sustain (#64) are switches which accept only two values: 0 for "off" and 127 for "on" — other numbers are ignored. Some continuous controllers, such as MIDI Pan (#10) are virtually always shortchanged in their implementation of possible value settings, with perhaps as few as 2 or as many as 24 possible values spread evenly across the to 0–127 spectrum.

Chart of MIDI Channel Mode Messages

Local/Remote Keyboard Control (includes channel n | controller 122 | on/off)	Omni/Poly/Mono Select (includes channel n | contoller xxx | 0000*
All Notes Off (includes channel n | controller 123 | 0000)	

n = MIDI Channel 1–16 | 0000 = "dummy" byte
xxx = possible values 124–127
[Omni Off = 124; Omni On = 125; Mono On = 126; Poly On = 127]
* for Mono On (controller 126) last byte indicates the number of MIDI channels affected

NOTES: By far the most important and commonly used Channel Mode Message is the "Local/Remote Keyboard Control." This message allows greatly expands the flexibility multi-timbral MIDI controllers in a MIDI sequencing environment as well as in large live-performance set-ups. With Local Control set to off, the controller's keyboard is disconnected from the units internal tone generators, while all appropriate MIDI messages are still sent out the MIDI out port, where it could be routed to another remote keyboard, or a tone generator, or a sequencer, or even back to itself via its MIDI in port — to any combination of the above.

Chart of MIDI System Common Messages (All Channels)

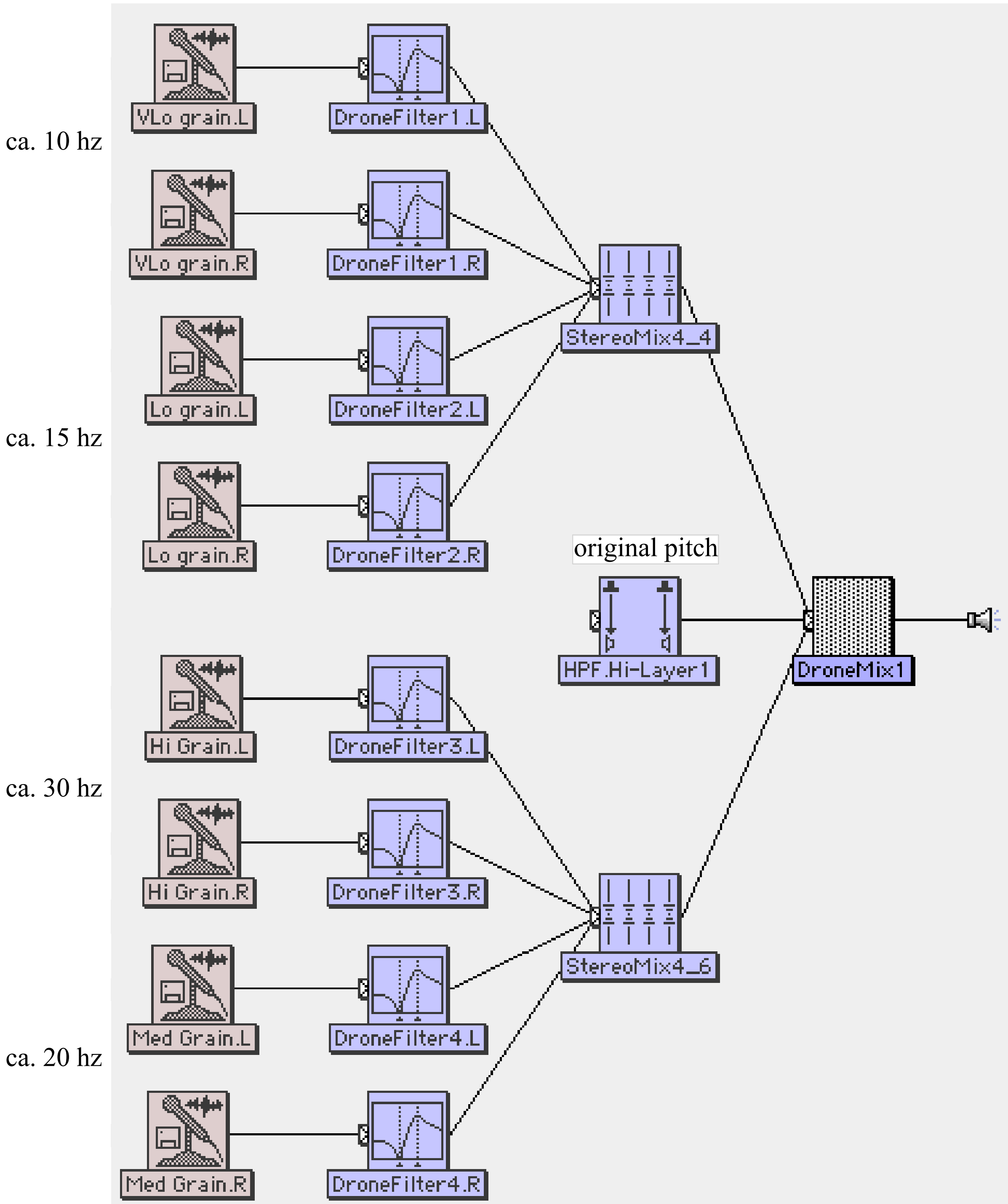
System Exclusive	Tune Request
Song Pointer	Song Select

NOTES: By far the most important of the MIDI System Common Messages is the System Exclusive Message (often abbreviated SysEx). This is the message type that allows for editor/librarian software, real-time patch editing using a computer software (such as a sequencer or *MAX*), MIDI sample dumps, and even MIDI Time Code (MTC) which allows MIDI to bridge the gap between MIDI Time Clock and SMPTE, timecode used in the motion picture and video (and increasingly in the music) industry.

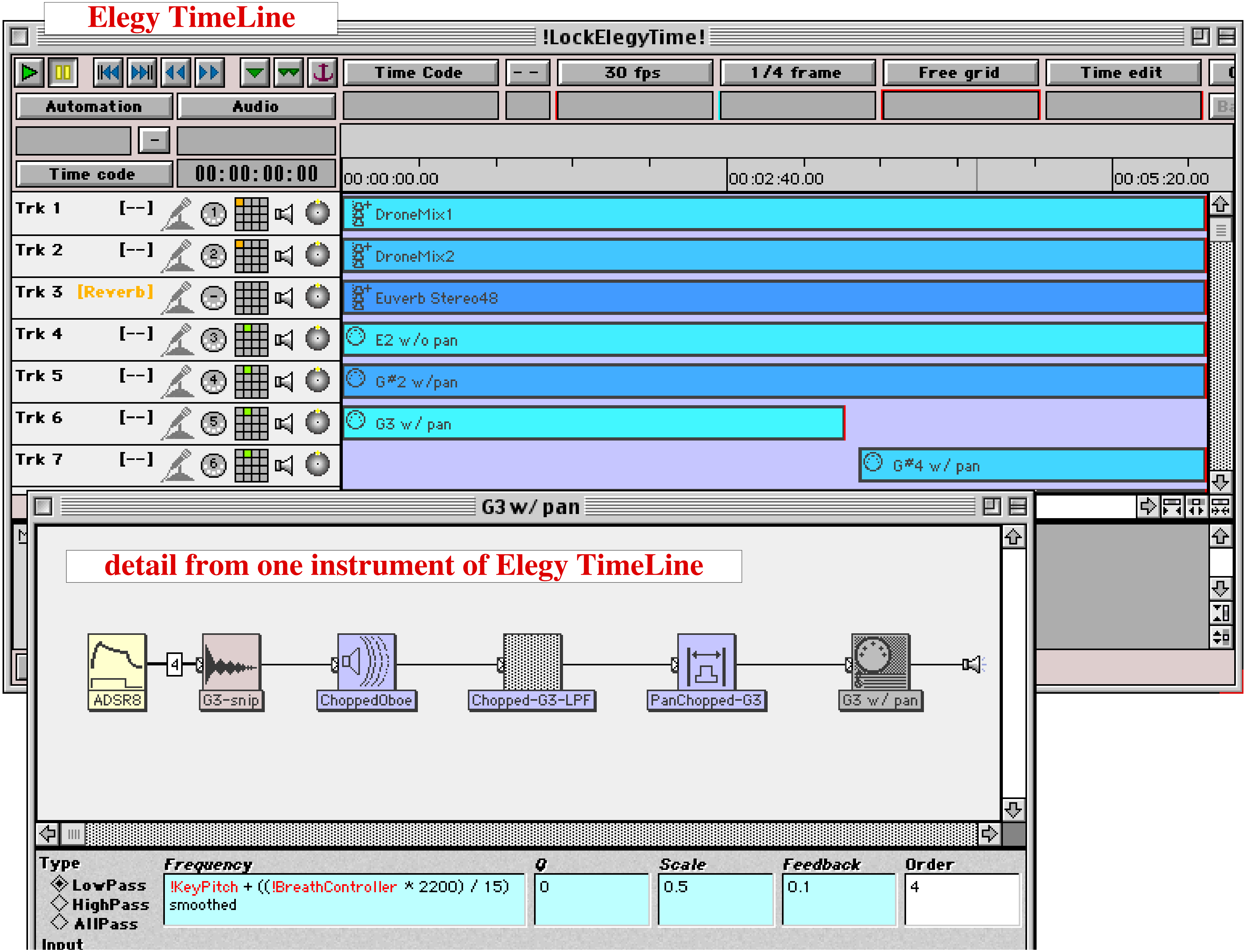
Chart of MIDI System Real-Time Messages (All Channels)

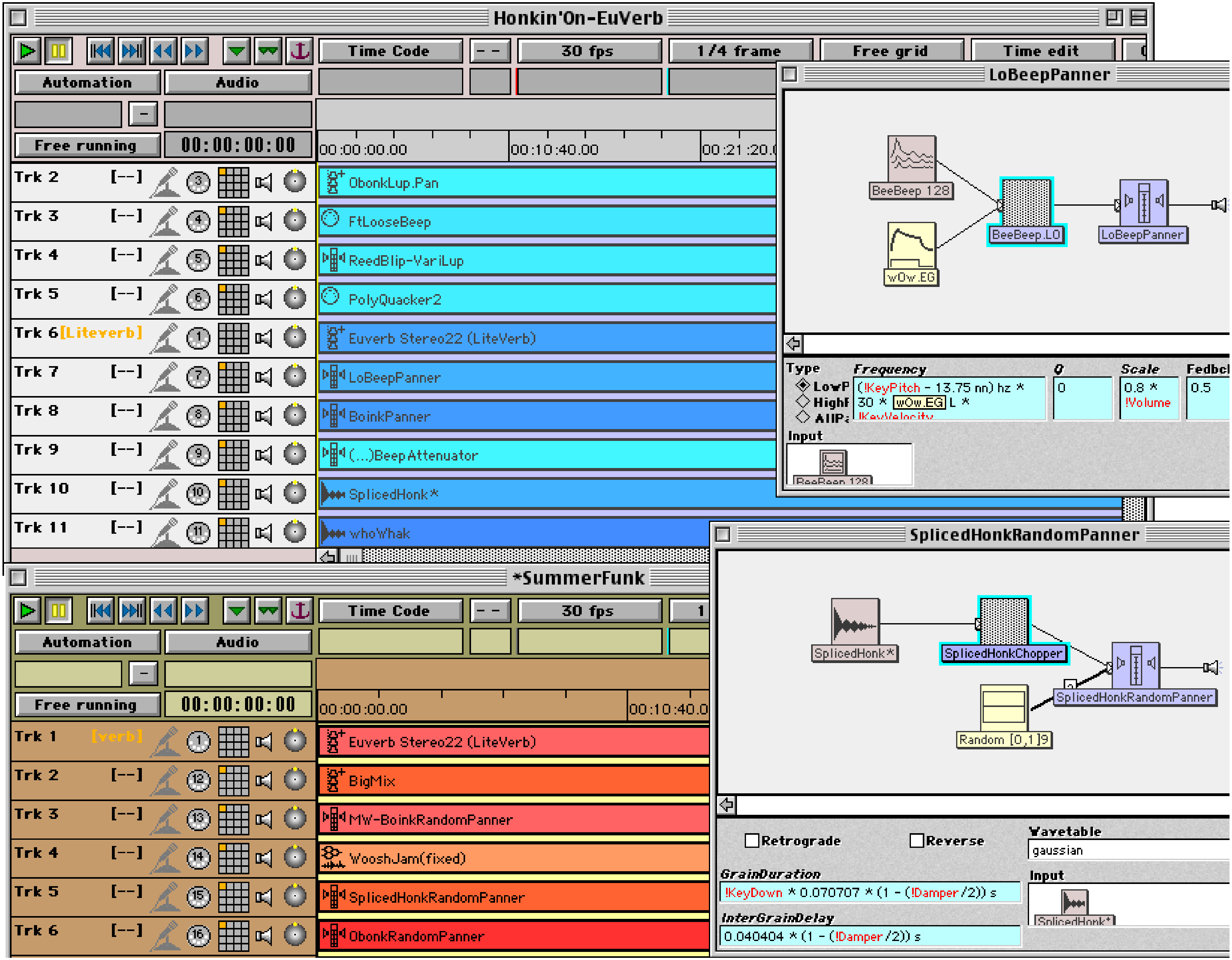
System Reset	Start From First Measure
Timing Clock	Stop
Active Sensing	Continue Start

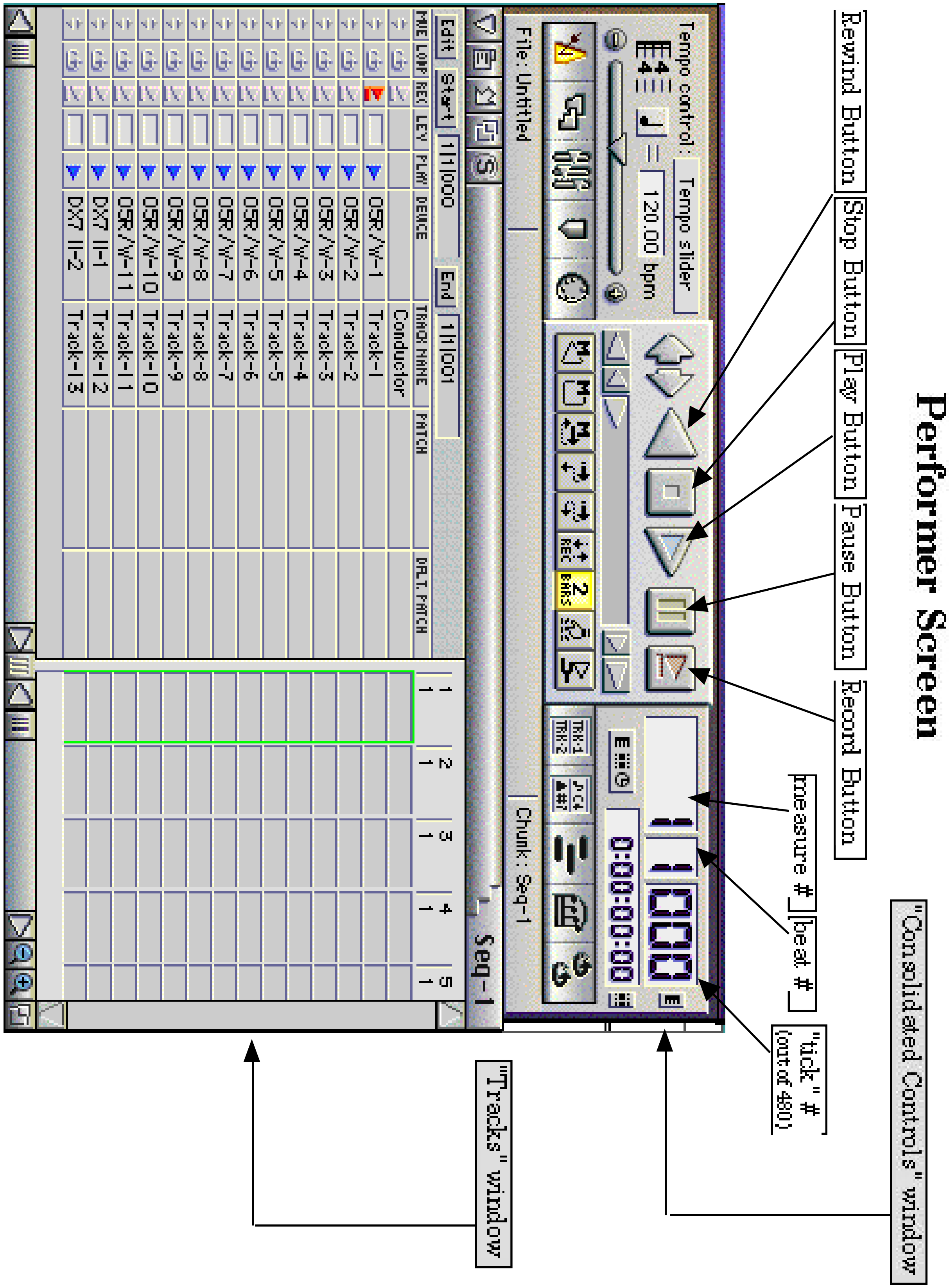
Kyma DroneMix1 from *Elegy*



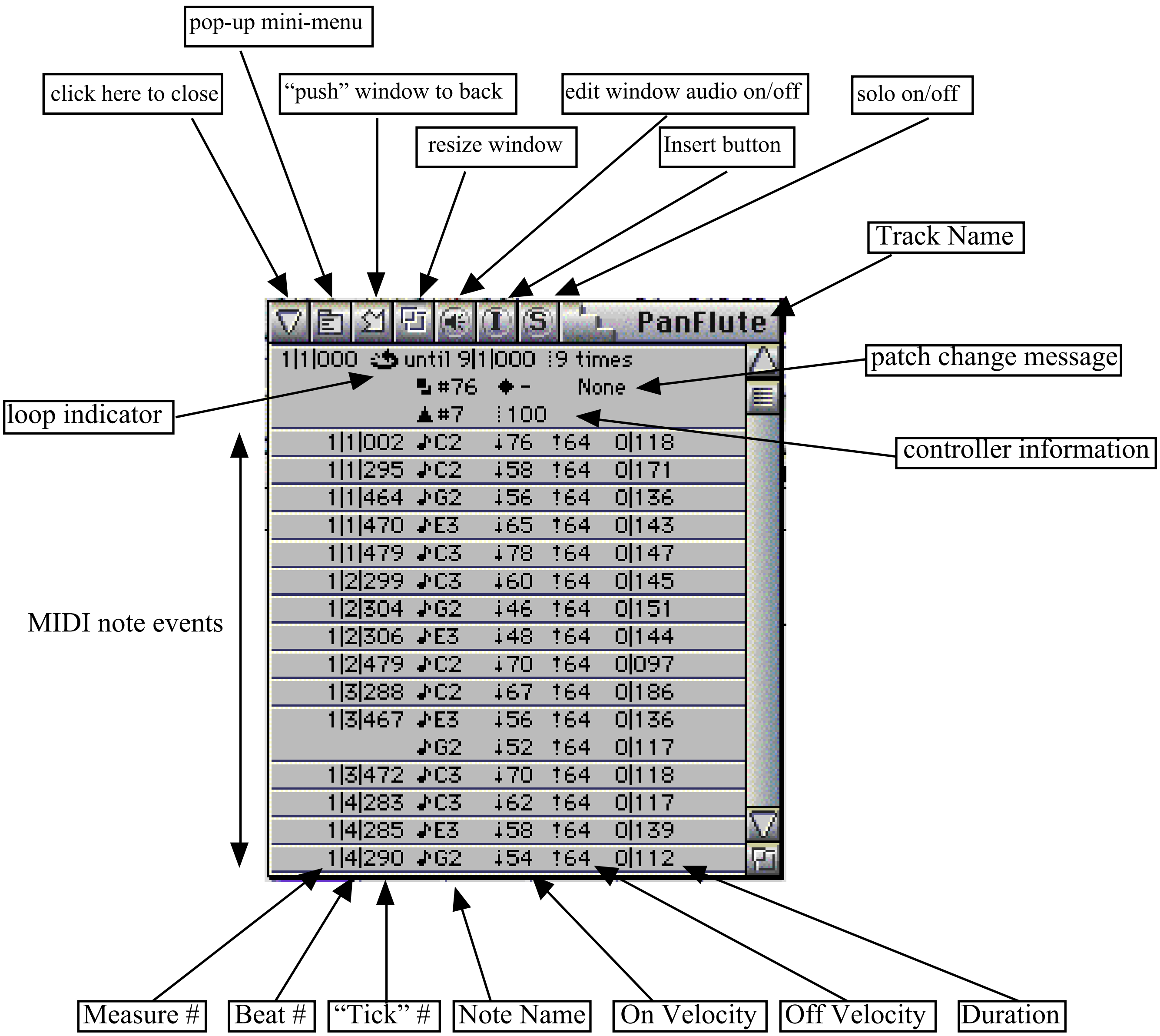
Kyma screen shots



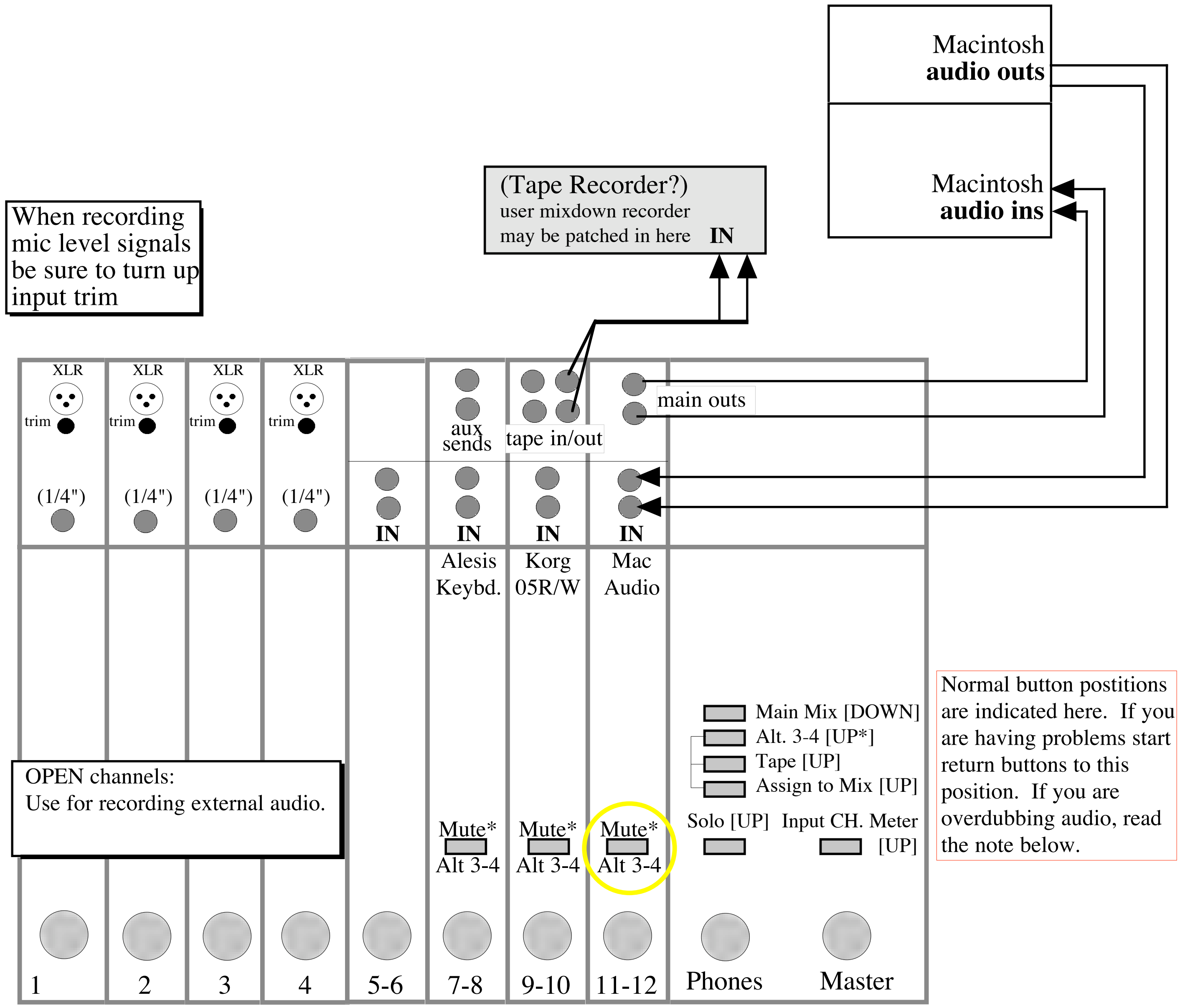

Honkin'On-EuVerb
Time Code
30 fps
1/4 frame
Free grid
Time edit
Automation
Audio
Free running
00:00:00:00
00:00:00.00
00:10:40.00
Trk 2
Trk 3
Trk 4
Trk 5
Trk 6 [Literverb]
Trk 7
Trk 8
Trk 9
Trk 10
Trk 11
ObonkLup.Pan
FtLooseBeep
ReedBlip-VariLup
PolyQuacker2
Euverb Stereo22 (LiteVerb)
LoBeepPanner
BoinkPanner
(...)BeepAttenuator
SplicedHonk*
whoWhak
LoBeepPanner
BeeBeep 128
wOw.EG
BeeBeep.LO
Type
Frequency
Q
Scale
LowP
HighP
(!KeyPitch - 13.75 nn) hz *
30 * wOw.EG L *
0
0.8 *
!Volume
0.5
Input
*SummerFunk
Time Code
30 fps
Automation
Audio
Free running
00:00:00:00
00:00:00.00
Trk 1 [verb]
Trk 2
Trk 3
Trk 4
Trk 5
Trk 6
Euverb Stereo22 (LiteVerb)
BigMix
MW-BoinkRandomPanner
WooshJam(fixed)
SplicedHonkRandomPanner
ObonkRandomPanner
SplicedHonkRandomPanner
SplicedHonk*
SplicedHonkChopper
Random [0,1]9
Retrograde
Reverse
Wavetable
gaussian
GrainDuration
!KeyDown * 0.070707 * (1 - (!Damper/2)) s
InterGrainDelay
0.040404 * (1 - (!Damper/2)) s
Input


Performer Screen
Rewind Button
Stop Button
Play Button
Pause Button
Record Button
measure #
beat #
"tick" # (out of 480)
"Consolidated Controls" window
"Tracks" window
Tempo control:
Tempo slider
120.00 bpm
File: Untitled
Chunk: Seq-1
Seq-1
0:00:00:00
Edit
Start
1|1|000
End
1|1|001
MVE LOOP REC LEV PLAY DEVICE
TRACK NAME
PATCH
DFLT. PATCH
Conductor
Track-1
Track-2
Track-3
Track-4
Track-5
Track-6
Track-7
Track-8
Track-9
Track-10
Track-11
Track-12
Track-13

Performer Event Edit Window



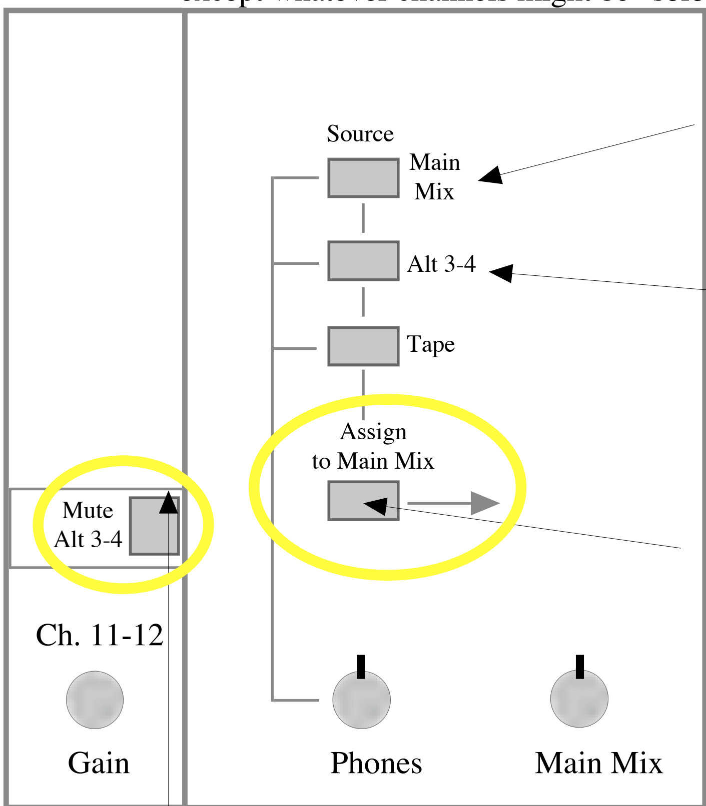
Wiring Diagram for Mackie 1202 VLZ



*__Note:__ MUTE/Alt 3-4 button on Mackie Mixer track 11-12 can be used to facilitate overdubbing when using *Digital Performer*. After pushing this button, you can then push down the Alt 3-4 button above the phones dial. This will allow you to hear all your audio tracks in the phones without causing the feedback you'd get if you didn't mute track 11-12 of the Mackie inputs.

Output module of 1202-VLZ Mackie Mixers

Source Selection Buttons:
The three source selection buttons are used to determine what signal(s) are actually going to be heard in the headphones. If no button is pressed down, you won't hear anything, except whatever channels might be "solo-ed."



General Use:
The most common source selection is MAIN MIX. You will rarely need to select any other source.

The one exception (discussed below) involves using the ALT 3-4 button in combination with the channel 11-12 Mute button to allow overdubbing using *Deck* or *ProTools* software.

NOTE: When listening to the Main Mix, the volume level in the headphones is a composite of the levels settings for the Phones knob AND the Main Mix knob.

Assign to Main Mix:
This button should *always* be UP!!!!!

I can't think of a single reason you would ever want this assignment switch button pushed down on the 4 workstations that have a Mackie 1202-VLZ mixer.

MAIN MIX: Basic operations
In normal operation, ALL inputs to the mixer (excluding RCA style tape input jacks) are fed to directly to the main mix. This signal is always sent to the audio input of the Macintosh computer and the RCA tape output jacks. It is also sent to the headphone jack …
> IF (and only if) the Main Mix button under Selection is pushed down.

Tips for Overdubbing Audio Tracks in *Digital Performer*:

1. Push down the MUTE/Alt 3-4 button for Channel 11-12.

   This will remove the Macintosh Audio signal from the Main Mix so …
   >>you won't have to worry about FEEDBACK
   >>you won't re-record your entire session mix everytime you want to record a new track.

2. Push down the ALT 3-4 selection button under Source in the output module of the mixer.

   This will allow you to hear the muted channel from the Macintosh even though they are not going to the Main Mix.

   NOTE: If you want to hear what you are recording in your headphone mix leave down the Main Mix button under Source other wise pop it up.

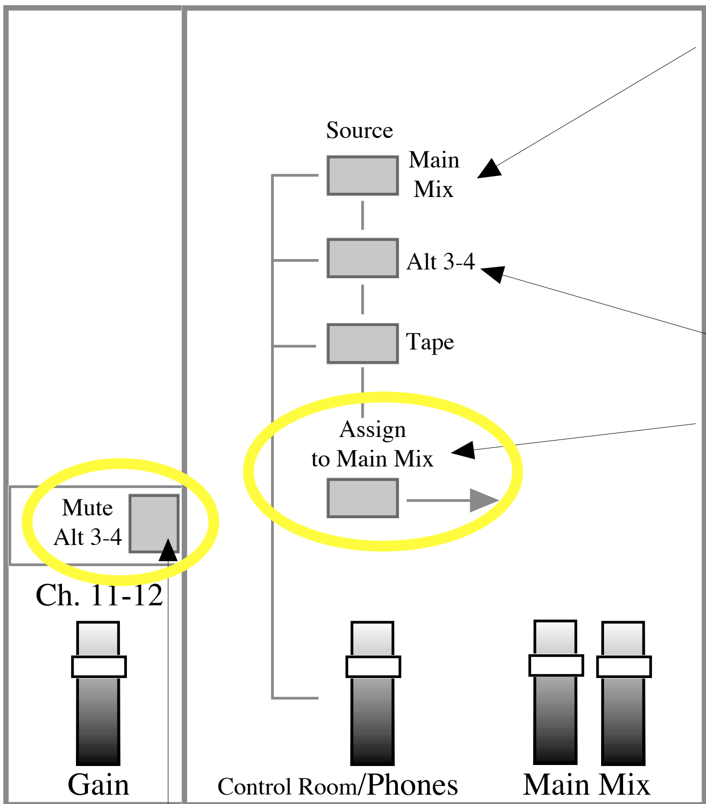
PLEASE!
At the end of your session, return the board to the normal set-up.

>>Channel 11-12 <u>unmuted</u>

>><u>Main Mix</u> selected under source

Output module of 1402-VLZ Mackie Mixer

Source Selection Buttons:
The three source selection buttons are used to determine what signal(s) are actually going to be heard in the headphones and also in the loudspeakers. If no button is pressed down, you won't hear anything, except whatever channels might be "solo-ed."



Common Uses:
The most common source selection is MAIN MIX. This allows you to hear everything coming into the channel inputs of the mixer and to hear exactly what is going to be recorded.

See discussion below for use of ALT 3-4 buton).

Assign to Main Mix:
This button should almost *always* be UP!!!!!

The only reason to push it down is if your want to record FROM cassette TO the DAT or the Computer. Make certain the cassette is NOT in input mode or you will have FEEDBACK! Also be aware that when this button is down the Main Mix source circuit is automatically cut to avoid feedback, so you will NOT be able to hear anything from the Main Mix.
>>When finished recording PLEASE pop this button back UP!!!

NOTE: When your source is the Main Mix, the volume level in the headphones or the loudspeakers is a composite of the levels settings for the Phones knob AND the Main Mix knob. The Phones knob has <u>NO</u> impact on the levels sent to any of the recording devices.

MAIN MIX: Basic operations
In normal operation, ALL inputs to the mixer (except from casstte deck) are fed to directly to the Main Mix. This signal is always sent out to ALL of the recording devices: the DAT player, the cassette player, and the audio input of the Audiomedia card in the computer. It is also sent to the headphone jack …
> IF (and only if) the Main Mix button under Selection is pushed down.

Tips for Overdubbing Audio Tracks in *Digital Performer*:

1. Push down the MUTE/Alt 3-4 button for Channels 3 & 4.

   This will remove both channels of the Audiomedia II signal from the Main Mix so …
   >>you won't have to worry about FEEDBACK
   >>you won't re-record your entire session mix everytime you want to record a new track.

2. Push down the ALT 3-4 selection button under Source in the output module of the mixer.

   This will allow you to hear the muted channel from the Macintosh even though they are not going to the Main Mix.

   NOTE: If you want to hear what you are recording in your headphone mix leave down the Main Mix button under Source other wise pop it up.

PLEASE!
At the end of your session, return the board to the normal set-up.
>>All Channels <u>unmuted</u>
>><u>Main Mix</u> selected under source

Wiring Diagram for older Mackie 1202

NOTE: The *older* Mackie 1202 mixer is not flexible enough to allow easy over-dubbing while listening to your other tracks. For this task you should use a station which has the newer Mackie 1202 vlz mixer which features a "Channel Mute/Alt 3 & 4" button that will allow you to listen to material that isn't also being sent to the record input of your software. In a pinch you can do limited overdubbing, but you can not monitor (or listen) to any previously recorded audio tracks while recording new ones.

If the computer will not be generating the new audio you want to record, you simply turn down channels 11-12 on the mixer and record using another mixer channel. If the computer *IS* producing the new audio, you will have to find a way of muting playback of any other tracks in the software you are using.