DTP2_PROJ_6

School of Engineering

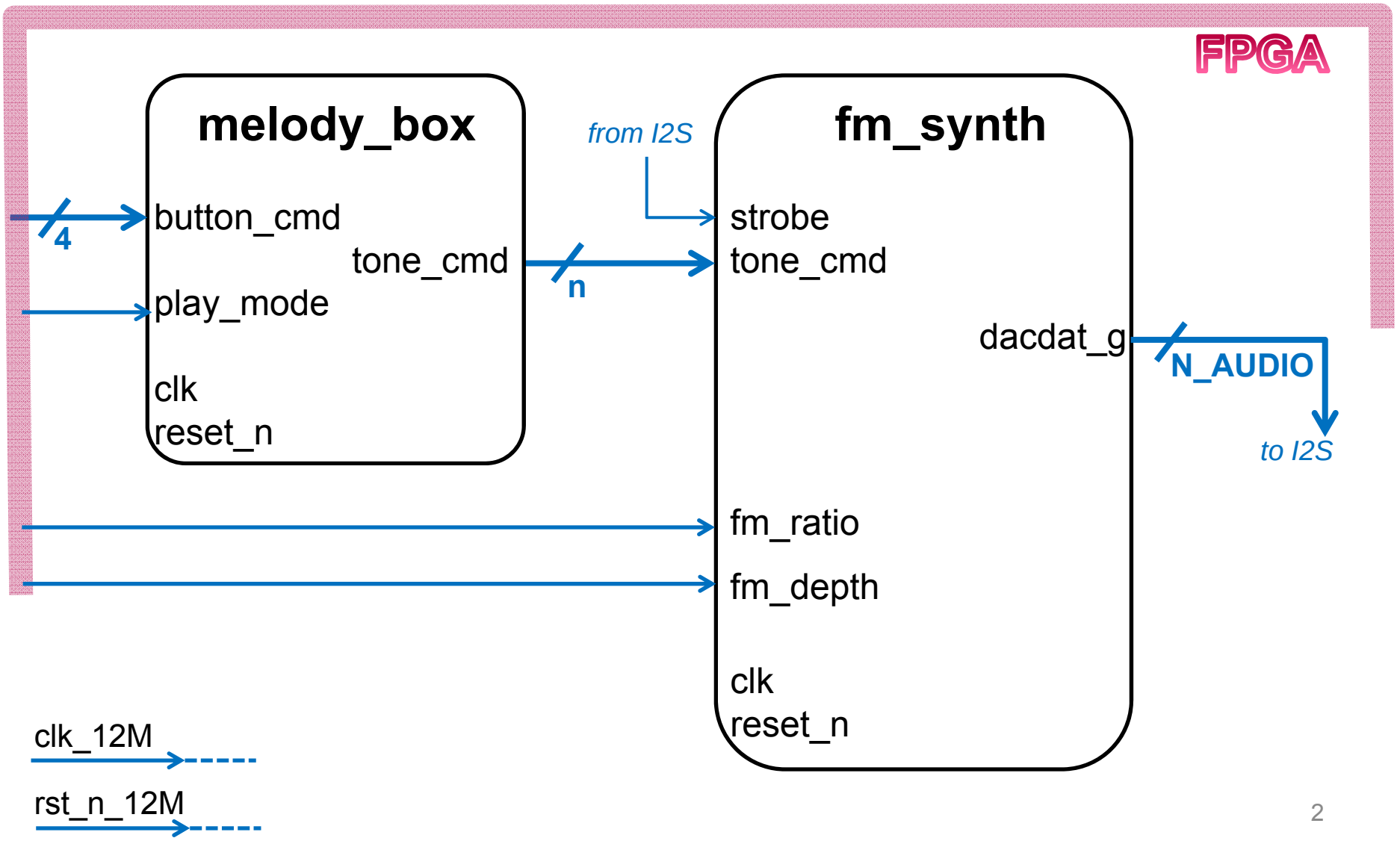
Some audio extra features

Melody-Box

FM-Synthesizer

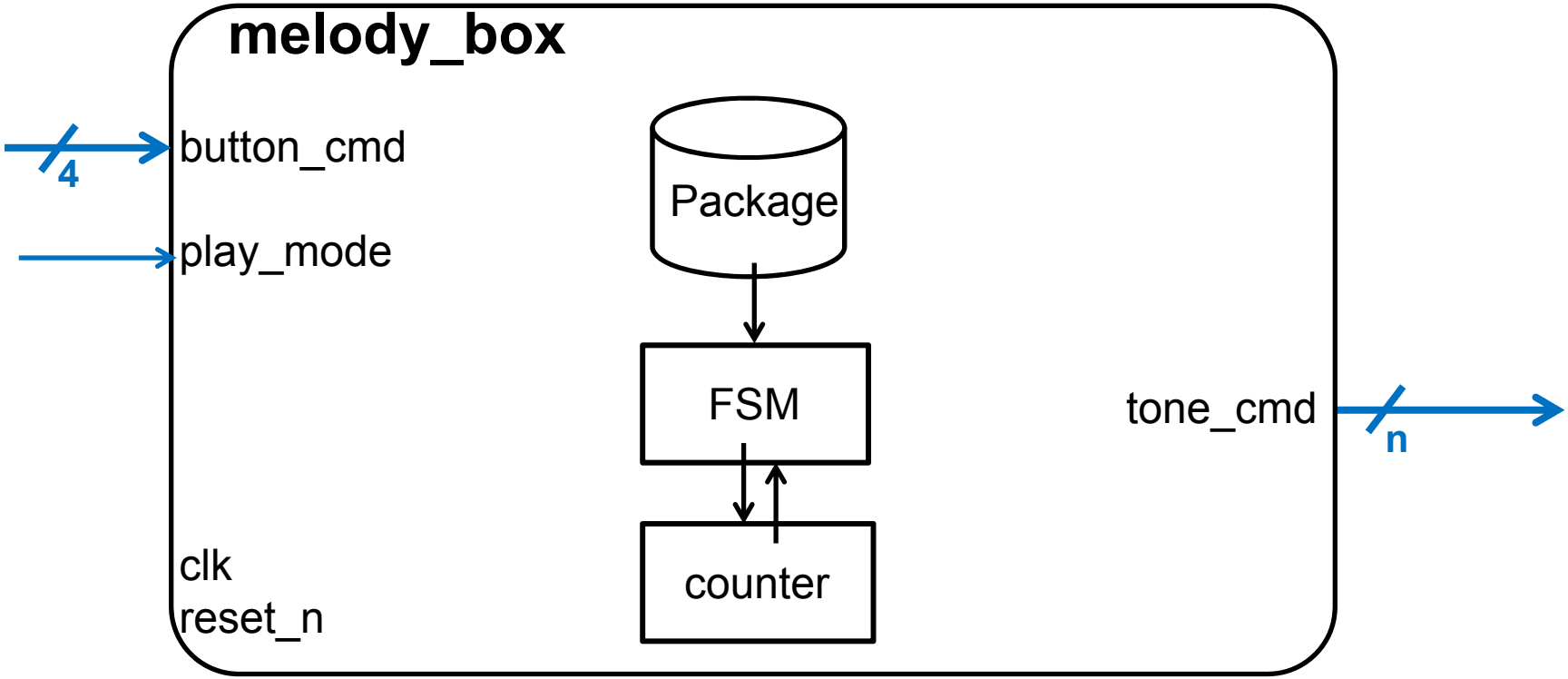
Some Audio Extra Features

School of Engineering



Melody-Box

School of Engineering

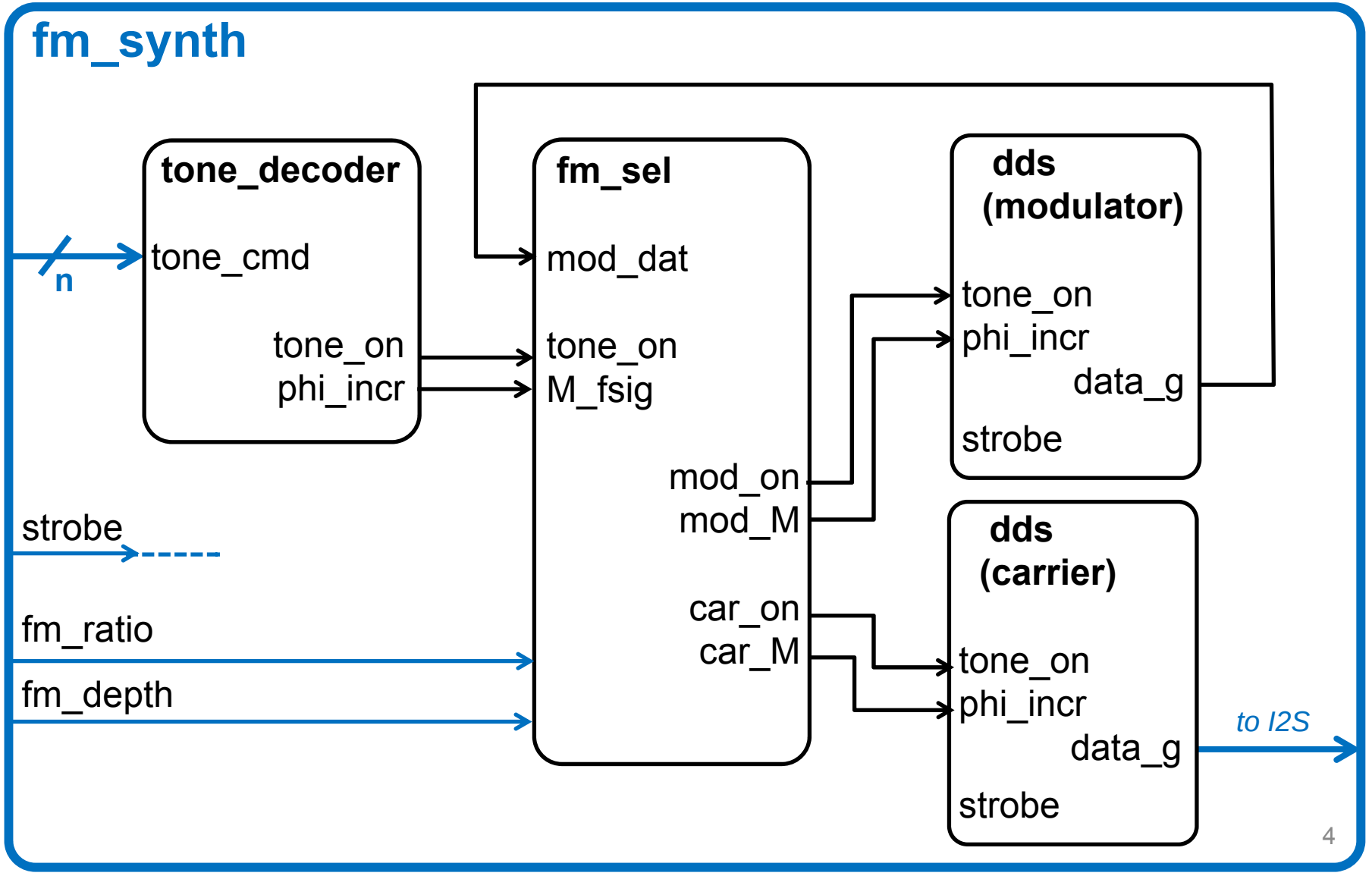


Comments

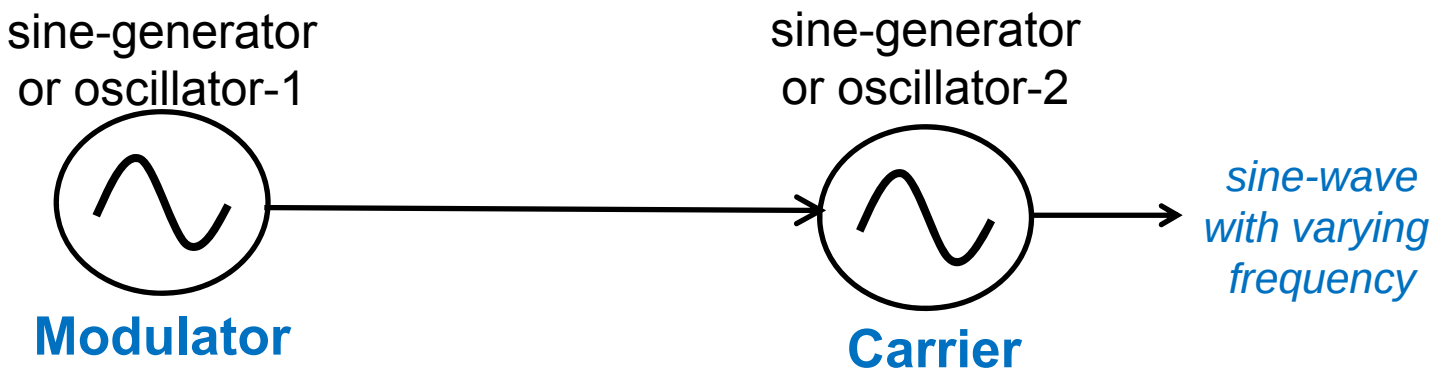
- *The package contains arrays describing the songs*
- *The songs can be described as a sequence of tones, and eventually have extra parameters like beat-speed (how long each tone lasts)*
- *The input play-mode selects either piano-mode (button_cmd are then the piano-keys) or song-mode (button_cmd are then the song-selector)*

FM-Synthesizer

School of Engineering



Basic Idea



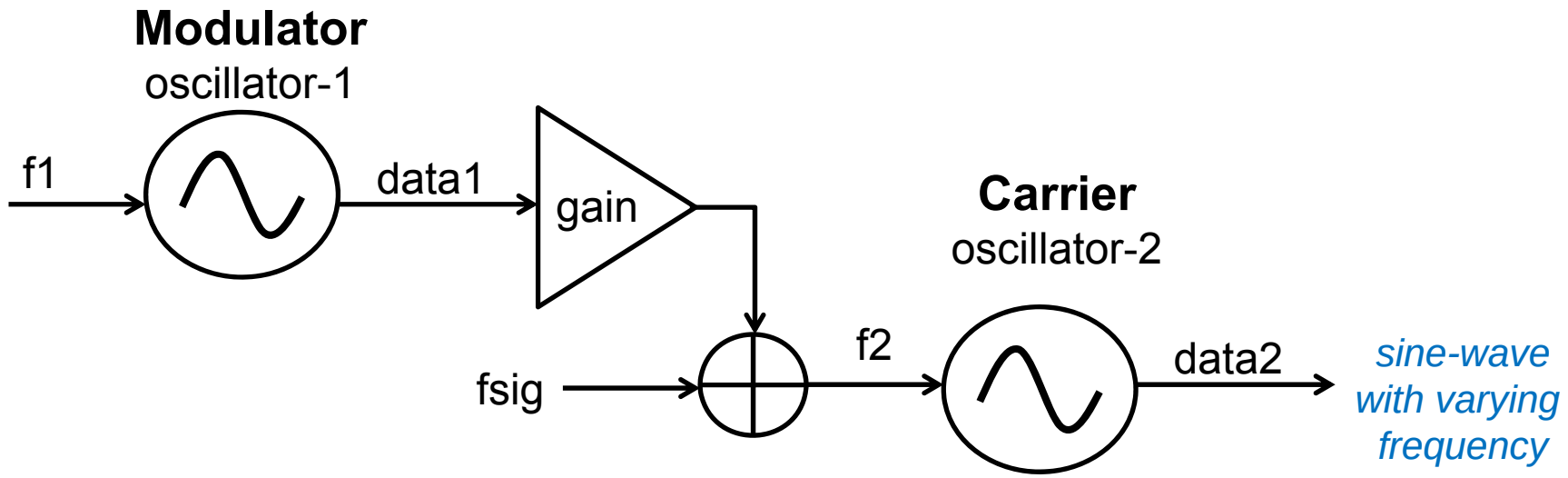
The frequency of the carrier sine-wave varies around a central value (fsig). This frequency variation generates several harmonics and give the sound a characteristic tone colour or timbre («Klangfarbe»).

The frequency variation is given by the output of the modulator sine-wave.

FM-Synthesizer

School of Engineering

Parameters



How are the values of f1 and gain selected? Using the parameters:

- *Carrier-to-modulator Ratio (c:m)* => *fsig / f1*
- *Modulation Depth* => *max{data1}.gain / fsig*

Example_ carrier to modulator ratio

Use input fm_ratio to select different (c:m) values.

For instance with c:m = 2:1

$$\begin{cases} \dfrac{c}{m} = \dfrac{f_{sig}}{f_1} = 2 \\ \Rightarrow \quad f_1 = \dfrac{f_{sig}}{2} \end{cases}$$

Tendency
(c:m) < 1 : generates round / full tone colour
(c:m) > 1 : generates shrill / metallic tone colour

Some common used values
(c:m) = (1:0) : no modulation, pure carrier
(c:m) = (4:1) ; (2:1) ; (3:2) ; (1:1) ; (2:3) ; (1:2) ; (1:4) ; …

FM-Synthesizer

School of Engineering

Example_ modulation depth

Use input fm_depth to change the modulation depth,
by selecting the gain value.

Let us call:
A1 : *amplitude of modulator sine-wave*
g : *variable gain factor*
fsig : *central frequency of carrier sine-wave*

$$\begin{cases} 0 \le A_1 \cdot g \le f_{sig} \\ \Rightarrow \quad 0 \le \dfrac{A_1 \cdot g}{f_{sig}} \le 1 \end{cases}$$

But our input controlling fsig is phi_incr (or M) which is actually proportional to the phase (2π.fsig.Ts). Therefore it gets a bit tricky to precisely calculate the modulation depth (we miss some theory you will learn in following semesters…).
Order of magnitude: phi_incr_mod about 2^1 till 2^7 times smaller than phi_incr_car

Proposal: try out some values for gain g and set a range you find effective.
Tendency: lower modulation-depth values allows to better notice the difference of tone colour depending on the (c:m) ratio.

Tutorial Reference (video)

Simon Cann's Synthesizer Boot Camp #5

(Synthesis Modulation Synthesis – part 1 of 2)
http://www.youtube.com/watch?v=h3yrd2YvkUo&list=PL916F525835A30EBF

Observation:
The other parts of the Simon Cann's Synthesizer Boot Camp (#1-#4 and #6) present tone-shaping methods which are also very interesting, but they give more work to be implemented in VHDL and the basic Frequency-Modulation Synthesizer of video #5 is rather easy to implement with DDS, and very effective (to cause changes of timbre).