A Hand Controlled Digital Audio Synthesizer

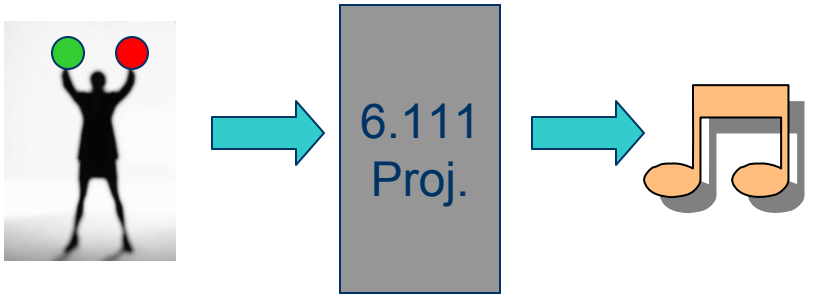
6.111

Behram Mistree

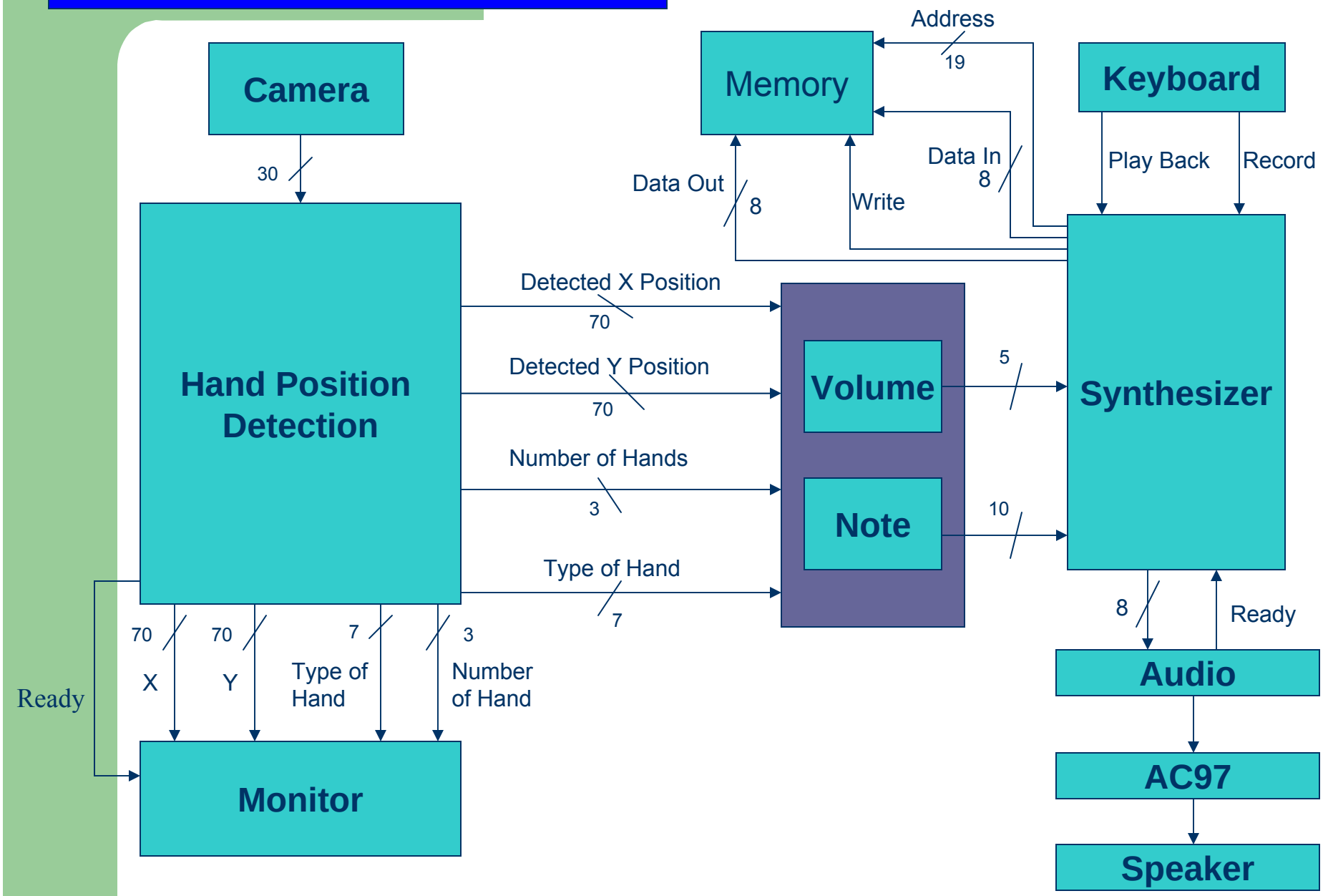
Alexander Sanchez

Overview of Functionality

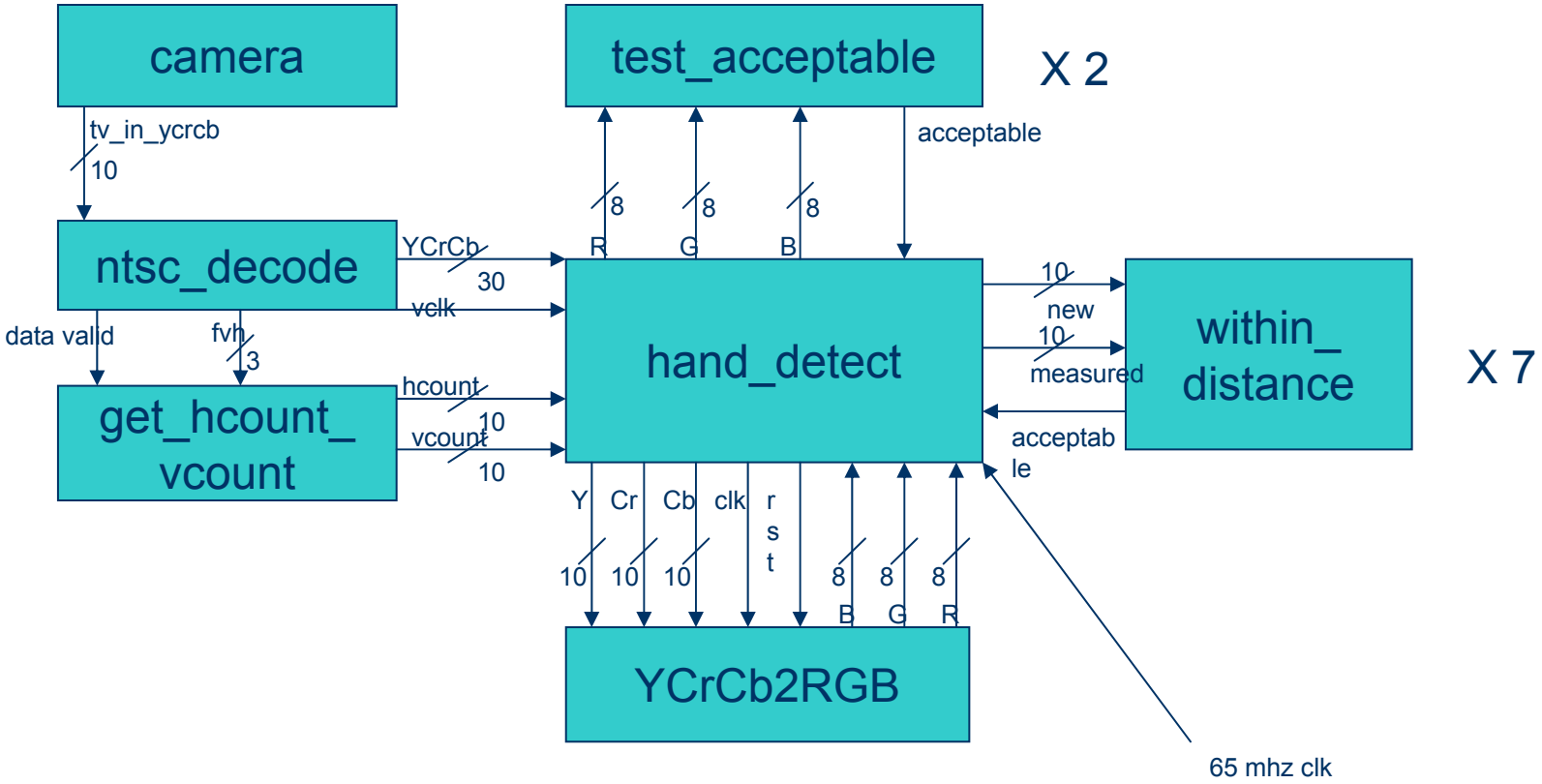
- Hand Controlled
  - LEDs detected by camera
  - LED's on hands correspond to musical notes and volumes
- Synthesizer
  - 2 Proposed Methods
  - Looping Capability
  - Single Instrument



Hand Controlled Audio Synthesizer



Hand Detection Block Diagram

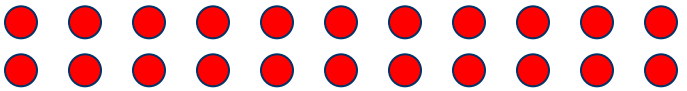


Hand Detection Essentials

- Checks each pixel for coloring (test_acceptable).

- If vector of last 10 pixel colors all contain correct color value, then identified as LED (hand_detect).

- Differentiates between distinct LED's by making sure sequences are far enough apart (within_distance).



Hand Detection Status

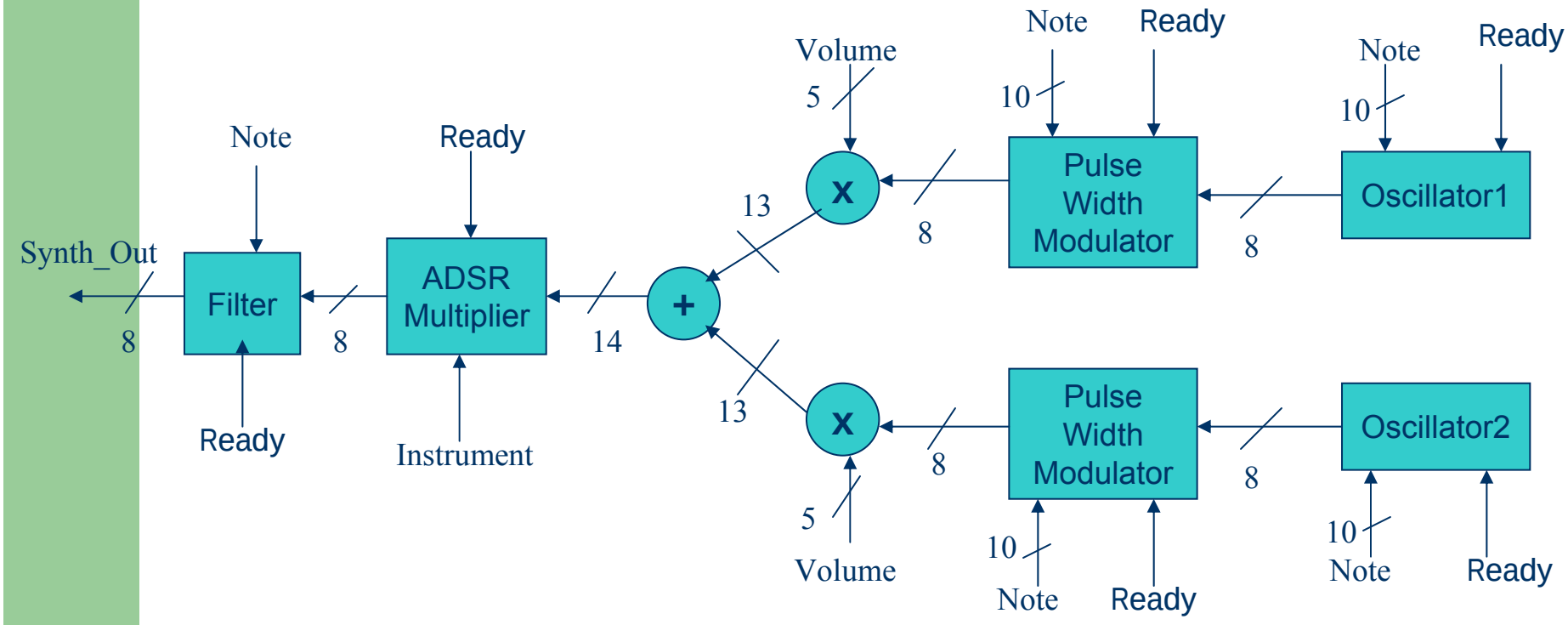
- Currently, recognizing horizontal motion of red LED.
- Need to test and retest thresholds.
- Better interface to monitor module.

Plan A for the Synthesizer…

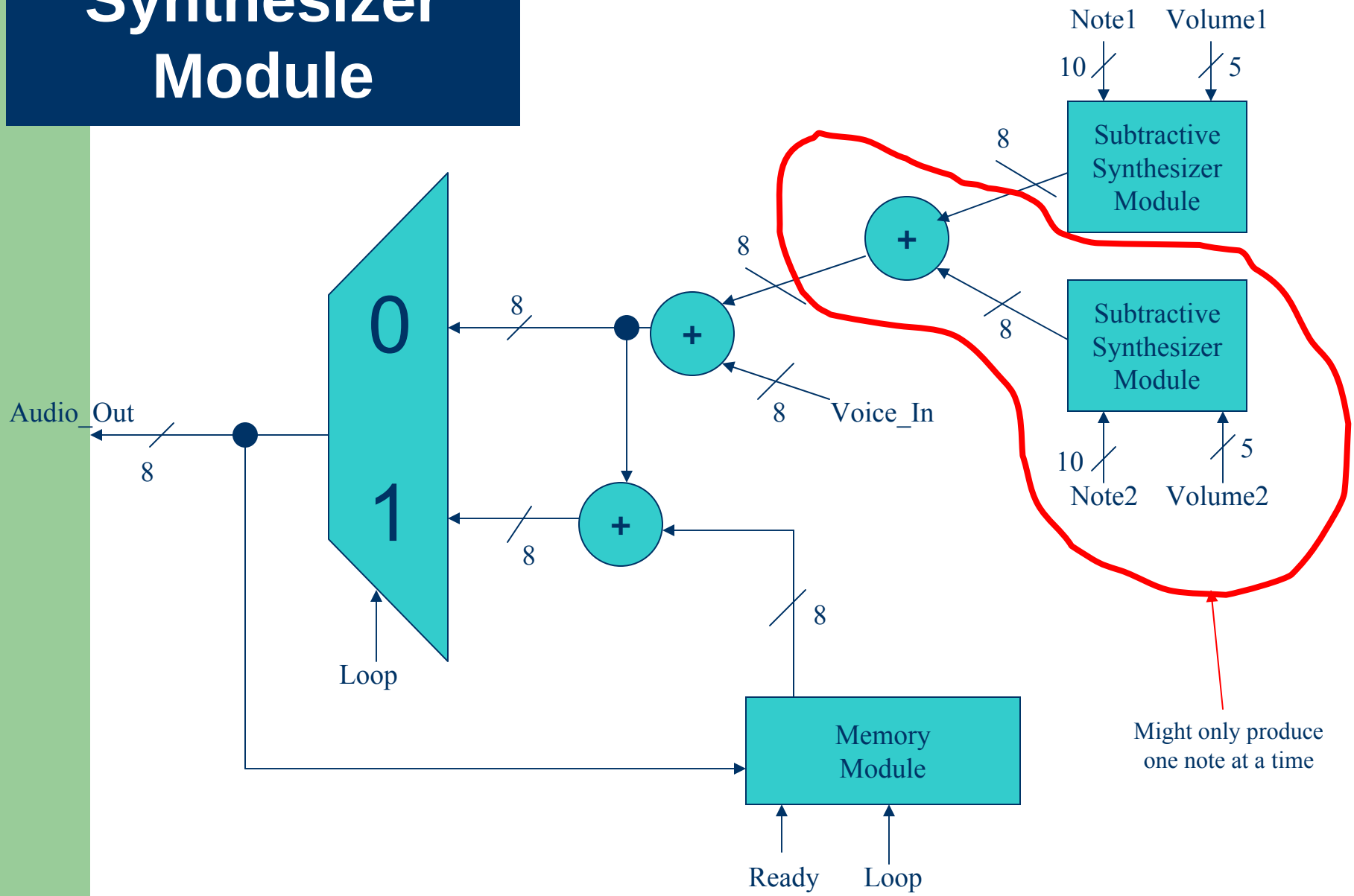
Method of Synthesizing

- Subtractive Synthesize
  - Create Oscillatory signals rich in harmonics
    - Sines, Square waves, Saw Tooth
  - Pulse Wave Modulation
  - Combine signals
  - Apply an Attack, Decay, Sustain, Release (ADSR) envelope
  - Low Pass Filter

Subtractive Synthesizer Module

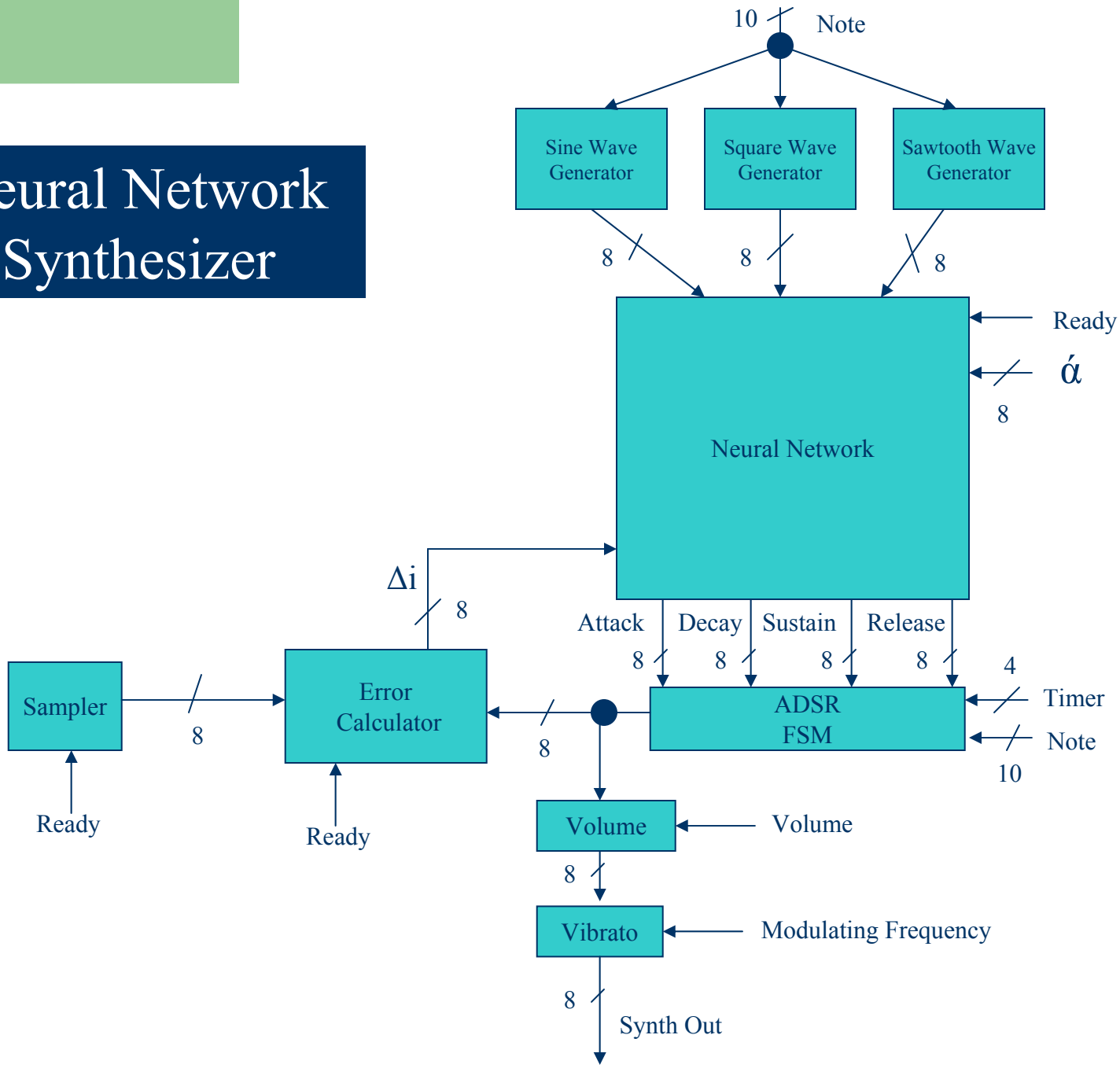


Synthesizer Module



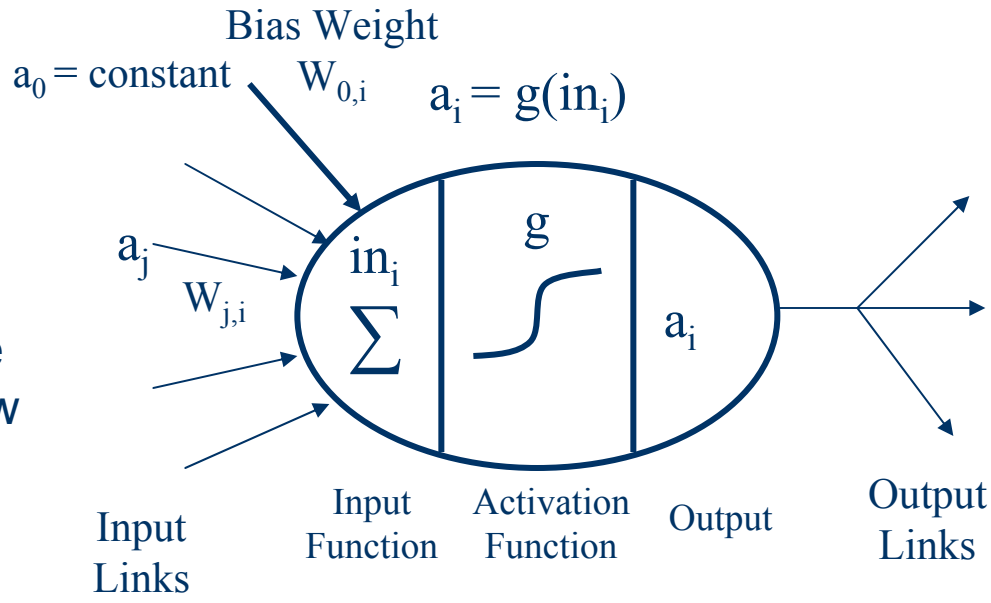
And Plan B for the Synthesizer…

Neural Network Synthesizer



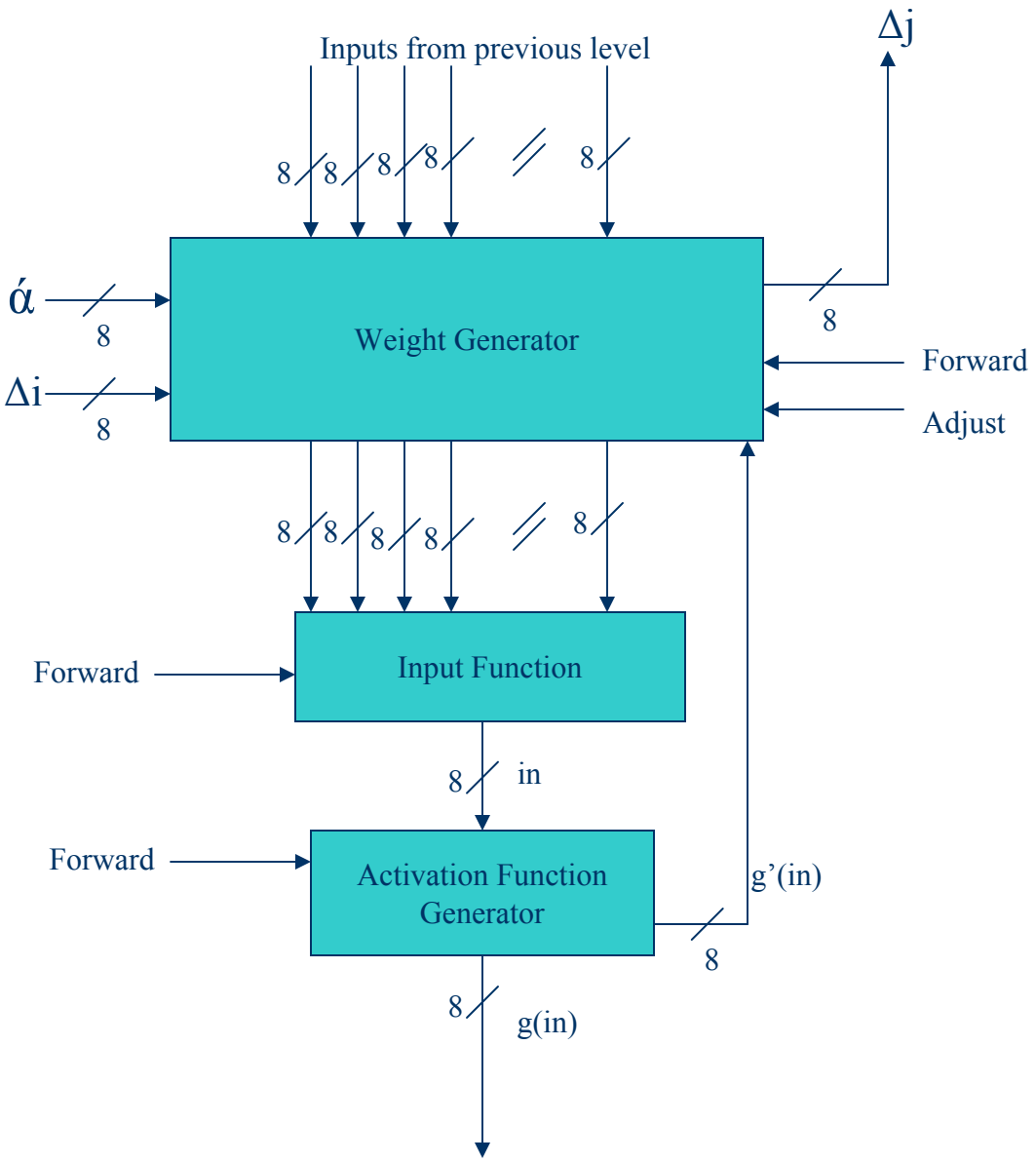
Building Up the Network: The Neuron

- Neuron
  - Takes weighted sum of inputs
  - Applies the Sigmoid function
    - High if in surpasses the threshold, otherwise low
  - Outputs a single value that can drive multiple neurons



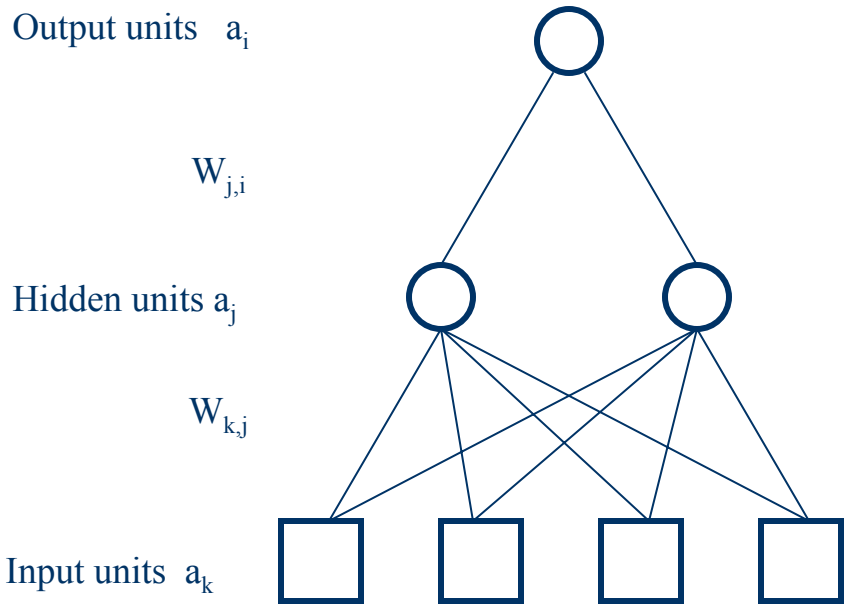
A Mathematical Model for a Neuron

Neuron


Inputs from previous level
Δj
8
8
8
8
8
α
8
Δi
8
Weight Generator
8
Forward
Adjust
8
8
8
8
8
Forward
Input Function
8
in
Forward
Activation Function Generator
g'(in)
8
8
g(in)

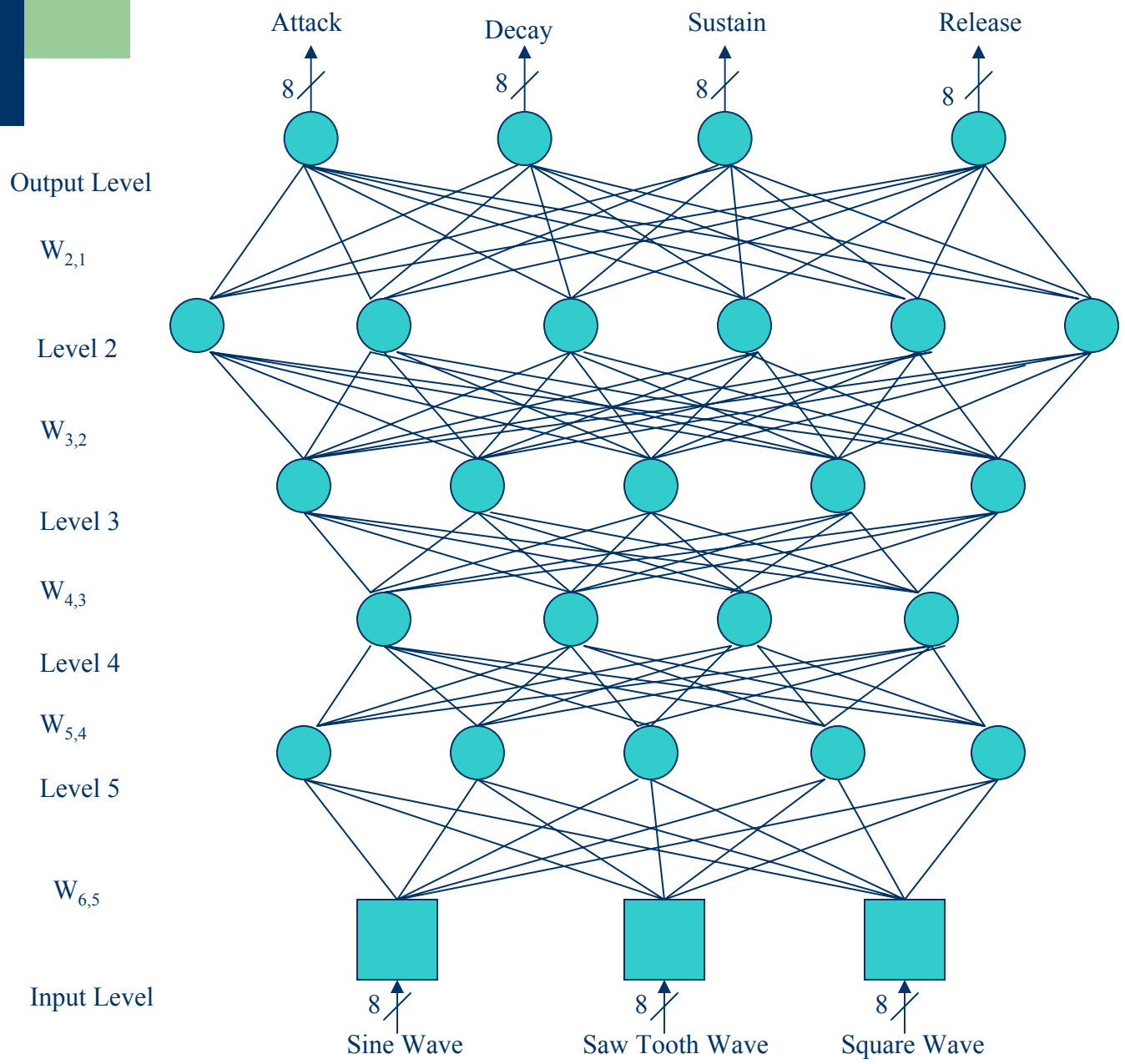
Building Up the Network: Making Connections

- Cascaded layers of neurons
  - Outputs of one layer drive inputs of next layer
    - Weights applied to inputs at each level
- Forward Propagation



A Simple Multilayer Neural Networks

Neural Network



Some Formulas

- Compare output of network to a sampled audio file of instrument to get squared error vector
  - **E** = 1/2 $\sum(y_i - a_i)^2$, where **y** is desired output vector
- E_i = ith component of error vector
- Modified Error $\Delta_i = E_i \times g'(in_i)$
- Weight update rule for output neurons in output layers:
  - $W_{j,I} \leftarrow W_{j,i} + \acute{\alpha} \times a_j \times \Delta_i$
- Propagation rule for Δ values:
  - $\Delta_j = g'(in_j) \sum W_{j,i} \Delta_i$
- Weight update rule for neurons in hidden layers:
  - $W_{k,j} \leftarrow W_{k,j} + \acute{\alpha} \times a_k \times \Delta_j$

Backward Propagation

- Find error vector
- Find Δ values for output units
- Start at output layer and repeat for all layers
  - Propagate Δ values to previous layer
  - Update weights between layers
- Training
  - Continue to adjust weights via backward propagation until network produces satisfactory output

Schedule

- **11.3.2006**: Several LED hand devices built.
- **11.11.2006**: MONITOR module completed (x, y coordinates can be displayed to monitor).
- **11.15.2006**: HAND POSITION_DETECTION module completed (Camera capable of extracting xy coordinates of LED's).
- **11.15.2006**: Code for KEYBOARD module (found online) is incorporated into project. Code for interfacing to AC 97 codec (also found online) is incorporated into project.
- **11.15.2006**: Decision on methodology for synthesizing notes finalized.
- **11.17.2006**: NOTE and VOLUME modules completed and tested.
- **11.22.2006**: Interface to ZBT for playback and recording completed and tested.
- **11.29.2006**: SYNTHESIZER module completed and tested.
- ****Debug, Debug, Debug****
- **12.7.2006**: Work on possible extensions. (See final section of proposal.)
- **12.11.2006**: Report completely written.

Possible Added Features

- Sound Effects:
  - In researching how to synthesize sounds, we observed that there were several other effects that we could incorporate into our project. Examples include an echo and slowing or speeding up the sound during playback.
- LED Control Tower:
  - Our synthesizer can be controlled by any LED movement. Therefore, if we build a device that turns sequences of LED's on and off, the synthesizer should play notes that correspond to those LED sequences. If we make this LED device controllable by circuitry, we should be able to play complicated music precisely.
- Voice In
  - Our synthesizer allows users to make music using body movements. It would be nice to add a voice element
- More Instruments
  - Our lab report calls for only synthesizing a piano. If our efforts are successful, we may be able to add on other instruments such as guitar, saxophone, and bass guitar.
- Display
  - Any musical instrument is difficult to learn. If our synthesizer proves particularly nonintuitive, we may make a more sophisticated display that makes our synthesizer easier to play.

References

- Norvig, Peter; Stuart, Russel. Artificial Intelligence: A Modern Approach. Second Edition. 2003. Pages 736-748.