Gesture-Based Frequency Controller Utilizing Direct Digital Synthesis and Computer Vision

Harshvardhan Singh(22BEC120) and Tanishq Kumar(22BEC131)
Guide: Dr. Viranchi Pandya

NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED 'A+' GRADE

Direct Digital Synthesis

Direct digital synthesis (DDS) is a digital approach to generating precise analog waveforms by leveraging a lookup table, phase accumulator, and tuning word to convert digital phase information into an analog signal. In this implementation, a precomputed 256-sample sine wave lookup table provides the amplitude values corresponding to a complete sine cycle. A 32-bit phase accumulator is incremented by a tuning word—calculated based on the desired frequency—at a fixed sampling rate, with its most significant 8 bits serving as an index into the lookup table. The selected amplitude value is then output using pulse-width modulation (PWM) on the Arduino, which effectively simulates an analog sine wave. This process mirrors data acquisition methods by converting digital computations into continuous waveforms, offering high precision and stability in frequency control.

.

Block Diagram

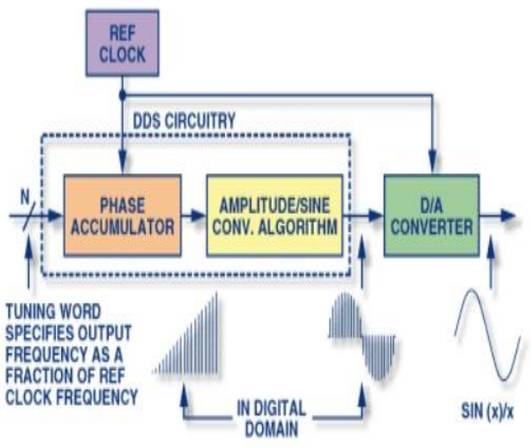


Figure 5. Signal flow through the DDS architecture.

Components

Arduino Uno :

The **Arduino Uno** serves as the core microcontroller that generates a PWM signal based on Direct Digital Synthesis (DDS) principles.**DAQ Device:**

RC Filter:

The RC low-pass filter, consisting of a resistor and capacitor, is connected to the PWM output to smooth the digital pulses into an analog-like sine waveform.

DSO:

DSO (Digital Storage Oscilloscope) is used to visualize and analyze the output waveform, allowing verification of the frequency, shape, and stability of the generated signal.

representation of an analog signal at an instant in time. In practice, analog signals continuously vary over time and an ADC takes periodic "samples" of the signal at a predefined rate. These samples are transferred to a computer over a computer bus where the original signal is reconstructed from the samples in software.

How it works?

In this system, the **Arduino Uno** is used to generate a sine wave using Direct Digital Synthesis (DDS) by outputting a PWM signal whose duty cycle follows a sine pattern stored in a lookup table. This PWM signal is passed through an **RC low-pass filter** (a resistor and capacitor) to smooth it into an analog-like sine wave. The filtered signal is then observed using a **Digital Storage Oscilloscope (DSO)**, which allows visualization of the waveform's frequency, shape, and stability. This setup effectively demonstrates digital waveform generation and analog reconstruction.

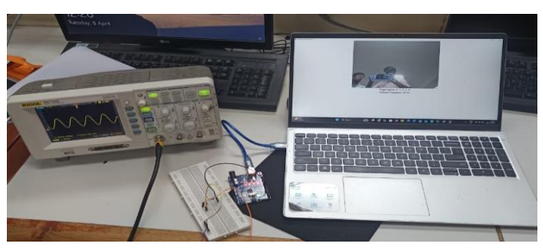
How It All Comes Together:

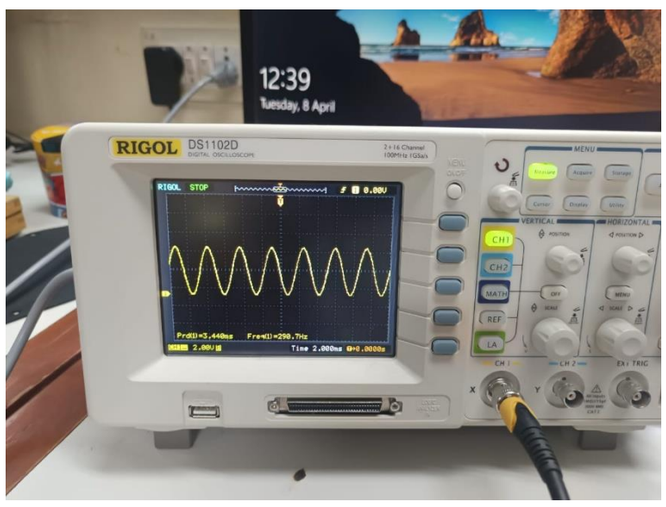
1. **Gesture detection** on the PC calculates a frequency.
2. The frequency is sent to the **Arduino via Serial**.
3. Arduino updates the **tuning word**.
4. Timer interrupts drive the DDS mechanism.
5. PWM signal is output on Pin 9.
6. The RC filter converts PWM to analog.
7. DSO displays the final sine waveform.

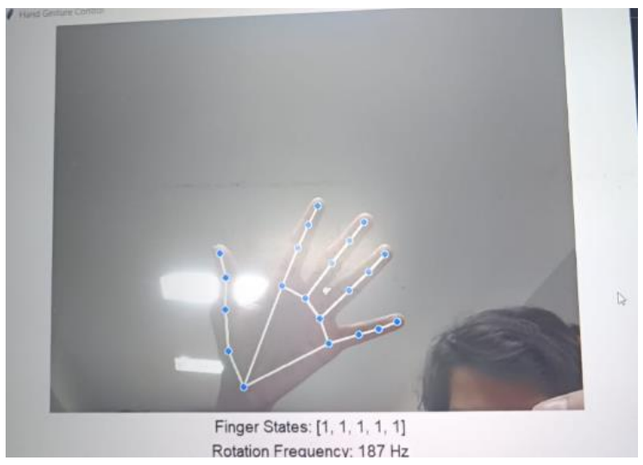
Functionality

. The system leverages hand gesture detection, digital synthesis, analog filtering, and real-time verification to produce a controlled sine waveform. First, Python uses OpenCV and MediaPipe to capture a live video stream and analyze hand landmarks from which it extracts finger states and wrist rotation. These gestures are translated into frequency values within a range of approximately **100 Hz to 1000 Hz** and sent to the Arduino over a serial connection. The Arduino, utilizing Direct Digital Synthesis (DDS), generates a sine wave by incrementing a **32-bit phase accumulator** with a tuning word derived from the desired frequency; the upper bits of this accumulator index into a precomputed **256-sample sine lookup table** to define the waveform's amplitude. The resulting PWM signal is then passed through an RC low-pass filter, which smooths out the rapid transitions to create a continuous analog sine wave. Finally, a Digital Storage Oscilloscope (DSO) is used to observe the analog output, verifying its frequency, amplitude, and shape, and ensuring accurate performance of the overall system.

Hardware







Conclusion

By leveraging Python with OpenCV and MediaPipe for real-time hand gesture detection, the system effectively maps human movements to a dynamic frequency range. The Arduino's implementation of Direct Digital Synthesis (DDS) using a sine wave lookup table and phase accumulator provides precise control over PWM signal generation. This digitally produced waveform is then transformed into a smooth analog signal through an RC low-pass filter, with its fidelity and characteristics confirmed via a Digital Storage Oscilloscope. Overall, the project highlights the powerful synergy between digital computation, analog circuitry, and real-time processing in creating interactive and adaptive electronic systems.