

AnemoSynth

Sintetizzatore ambientale generativo basato su condizioni meteorologiche

Gruppo di lavoro - CMLS 2025

1 Descrizione del progetto

AnemoSynth è uno strumento per la produzione di musica ambientale, che trasforma i dati ambientali quali temperatura, umidità, luminosità, in suoni dinamici e immersivi.

2 Componenti e tecnologie

Hardware

- Arduino MKR 1010 WiFi (preferito per capacità embedded e connettività)
- Sensori utilizzabili:
 - DHT11/DHT22 – Temperatura e Umidità
 - LDR (fotoresistore) – Luminosità ambiente
 - BMP180/BMP280 – Pressione atmosferica (opzionale)
 - Anemometro – Vento (costoso, può essere omesso o simulato)

Software

- Arduino IDE – Acquisizione dati
- SuperCollider – Sintesi sonora e generazione dei pattern
- JUCE – Elaborazione effetti (riverberi, filtri)
- Processing – Interfaccia grafica e feedback visivo

3 Architettura del sistema

- Input: Dati da sensori o API meteo
- Processo:
 - Arduino raccoglie i dati e li invia (via seriale o WiFi)
 - SuperCollider genera i suoni
 - JUCE applica effetti dinamici
 - Processing visualizza i parametri in tempo reale
- Output: Paesaggio sonoro ambient + visual feedback

4 Modalità di generazione sonora

Il sistema può funzionare sia in modo interattivo sulla base dell'utente, sia in modo generativo sulla base di numeri irrazionali.

Approccio diretto/interattivo

I dati misurati influenzano direttamente le forme d'onda generate. Esempio in tabella:

Dato Ambientale	Parametro Audio
Temperatura	Frequenza fondamentale
Umidità	Forma d'onda o timbro
Luminosità	Volume o densità del suono
Vento	Modulazioni/spazialità
Pressione	Intervallo temporale tra eventi

Approccio generativo astratto

I dati ambientali influenzano parametri legati a numeri irrazionali (π , e , ϕ , ecc.), modificando:

- Quantità di cifre significative
- Velocità di scorrimento
- Frequenza di generazione eventi
- Ordine di estrazione

- Quantità di numeri usati
- eccetera...

5 Comunicazione tra moduli

- Arduino - SC: via seriale, OSC, o WiFi (HTTP)
- SC - JUCE - Processing: via Open Sound Control (OSC)
- API Meteo: (es. OpenWeatherMap) per dati reali

6 Sensori problematici

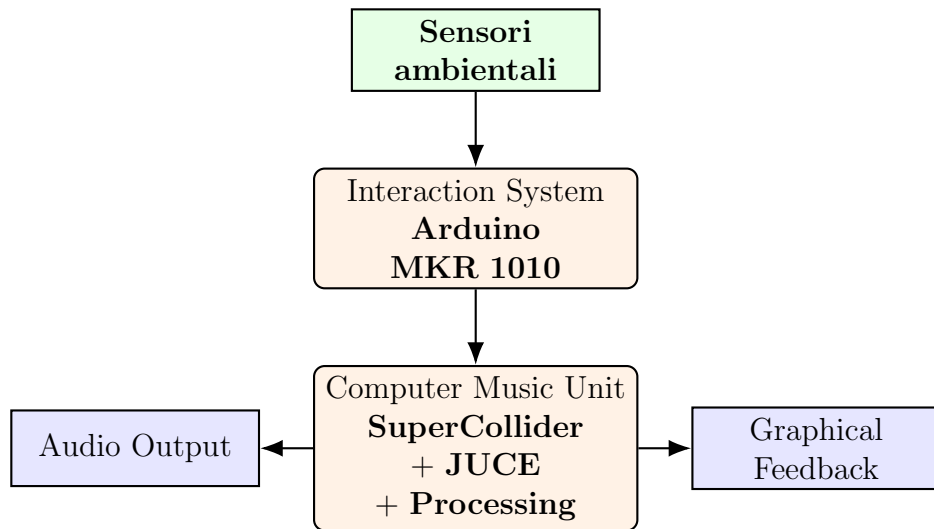
Come detto il sensore di pressione atmosferica e l'anemometro sono costosi hanno caratteristiche poco agevoli per il progetto. Le possibili soluzioni sono o eliminarli (soprattutto la pressione che tanto cambia poco) o sostituirli con dati da stazioni meteo online (API). Questo sblocca la possibilità di sfruttare al meglio Arduino MKR 1010 WiFi perché lo facciamo collegare a stazioni meteo esistenti sulle quali basare la generazione del suono. A questo punto si potrebbe far interagire l'utente basando la variazione dei parametri dei sintetizzatori tramite le differenze tra i dati misurati:

|Dati stazioni nel mondo – Dati nostro sistema|

7 Obiettivi del progetto

- Accessibilità: usabile senza conoscenze musicali
- Estetica: contemplativa, ambient, evocativa
- Creatività: struttura aperta, esplorabile
- Tecnica: pipeline coerente e robusta SC–JUCE–Processing
- Usabilità: grafica chiara, informativa e poetica

8 Diagramma a blocchi



9 Cosa manca?

- Capire come far comunicare i sistemi e i linguaggi al meglio,
- Come rendere il sistema il più intuitivo possibile per l'utente,
- Cosa far vedere all'utente.