HUPUF2X8A

Technical Documentation

Hutech - PulseForge Series

May 16, 2025

Contents

1
Introduction

1.1 Purpose and Scope

The HUPUF2X8A (Hutech PulseForge Stereo 2x8b Analog) is an audio synthesis chip designed for embedded systems requiring flexible and efficient sound generation. This technical documentation provides comprehensive information on the architecture, configuration, and usage of the HUPUF2X8A APU. It is intended for hardware developers, embedded software engineers, and audio system integrators who aim to incorporate the APU into audio processing applications.

1.2 Features

The HUPUF2X8A offers the following key features:

- Dual-channel stereo output with 8-bit resolution per channel.
- Support for up to 8 audio channels:
  - 6 Segmented Channels (2 High-Quality, 2 Medium-Quality, 2 Low-Quality).
  - 2 Sampled Channels for raw audio playback.
- Integrated waveform generation, including square, triangle, pulse, sawtooth, and noise.
- Modulation capabilities: Amplitude, Delta, Noise, and Period modulation.
- Dynamic mixing with individual channel volume control and stereo balancing.
- Flexible configuration via parallel register interface.
- Efficient handling of segmented waveforms, enabling the creation of complex audio patterns.
- Compact DIP-20 package with low power consumption.

1.3 Scope of the Document

This document covers the following aspects:

- APU Architecture: Detailed block diagrams and functional module descriptions.
- Register Map: Comprehensive guide to configuring the APU.
- Performance Characteristics: Channel capacity, latency, and power consumption.
- Integration Guidelines: Recommendations for audio output circuitry and signal conditioning.
- Usage and Programming: Initialization routines and real-time control methods.
- Debugging and Testing: Diagnostic features and troubleshooting tips.

1.4 Target Audience

This document is designed for:

- Audio hardware designers integrating the APU into embedded systems.
- Firmware developers writing control code for audio synthesis.
- System integrators looking to optimize audio performance.
- Quality assurance teams validating audio output quality and stability.

1.5 Packaging and Pinout

The HUPUF2X8A APU is available in a DIP-20 package. The table below describes the pin functions and directions:

Pin(s)	**Name**	**Direction**	**Description**	**Usage**
RA0-RA3	Register Address	In	Register Access Address	Exposed Registers
RD0-RD7	Register Data	In/Out	Register Access Data	Exposed Registers
RRW	Register R/W	In	Register Access Read/Write Control	Exposed Registers
RAE	Register Enable	In	Register Access Enable Signal	Exposed Registers
L	Left	Out	Sound Left Analog Signal	Sound Output
R	Right	Out	Sound Right Analog Signal	Sound Output
CLK	Master Clock	In	Master clock input	Clock
RST	Reset	In	Reset signal	Boot
VSS	Ground	-	Ground connection	Power supply
VDD	Power	-	Positive power supply	Power supply

Table 1: Pinout of HUPUF2X8A - DIP-20 Package

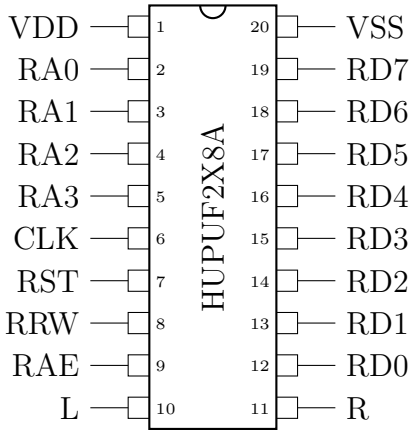

Figure 1: Pinout of HUPUF2X8A - DIP-20 Package

2

Architecture Overview

2.1 High-Level Description

The HUPUF2X8A Audio Processing Unit (APU) is a compact and versatile chip designed for generating high-quality stereo audio in embedded systems. The architecture leverages a modular design to accommodate a wide range of audio synthesis requirements while maintaining low power consumption and efficient processing.

The APU supports up to 8 audio channels:

- 6 Segmented Channels (2 High-Quality, 2 Medium-Quality, 2 Low-Quality)
- 2 Sampled Channels (configurable for raw audio playback and modulation)

The core of the APU is a pipeline-based audio engine that synthesizes sound by processing discrete segments or samples through specialized channels. Each channel can be independently configured for waveform generation, modulation, and mixing, allowing for complex audio patterns and dynamic effects.

The APU directly outputs analog signals through two dedicated stereo pins (L and R), providing high-quality sound at 8-bit resolution per channel.

2.2 Block Diagram

The following diagram provides an overview of the internal architecture of the HUPUF2X8A APU:

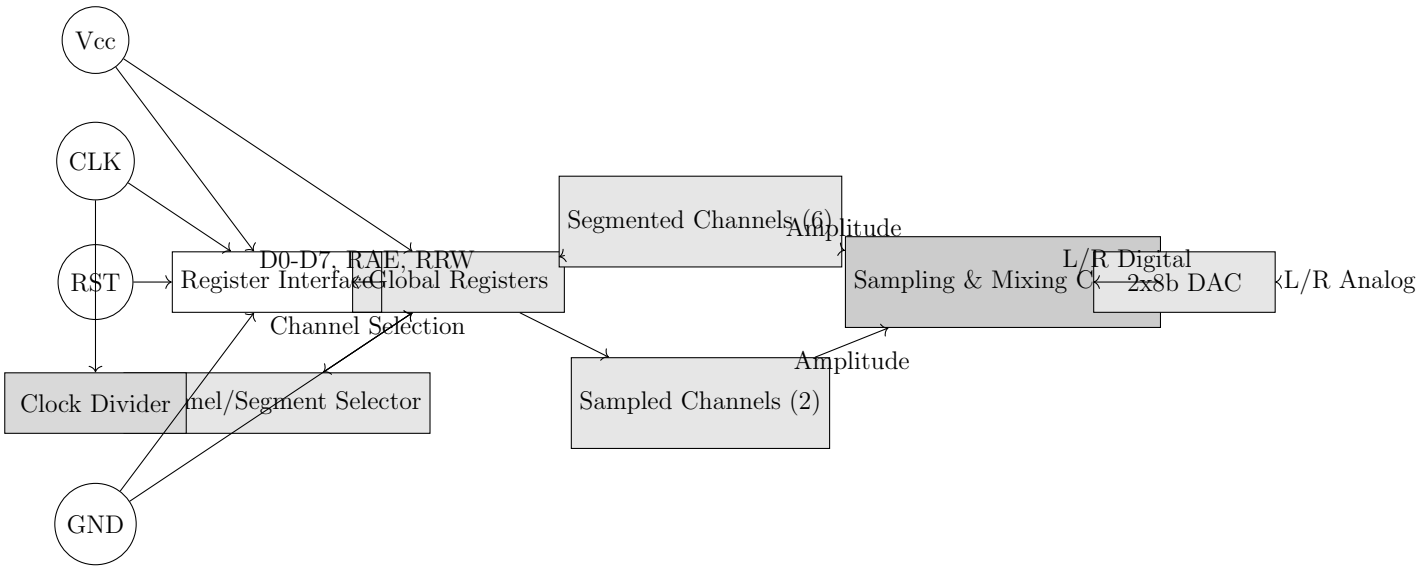

Figure 2: Block Diagram of HUPUF2X8A - APU

2.3 Signal Flow

The APU operates through a sequence of audio processing steps as follows:

1. **Clock and Reset Management:** The system clock synchronizes all internal operations, while the reset line initializes the APU state.
2. **Channel Configuration:** Each of the 8 channels can be independently configured through dedicated registers. Segmented channels generate complex waveforms, while sampled channels handle raw audio data.

3. **Waveform Generation:** The segmented channels utilize a series of amplitude steps to create dynamic audio patterns. Sampled channels output audio directly from stored data.

4. **Mixing and Output:** The outputs of all active channels are summed and balanced by the mixer, which can independently control the volume and stereo positioning of each channel.

5. **Digital-to-Analog Conversion:** The mixed digital audio signals are converted to analog output via a dual-channel DAC, feeding the left and right audio lines.

2.4 Audio Processing Pipeline

The audio processing pipeline is designed to handle multiple channels concurrently, optimizing resource usage while maintaining audio fidelity. The pipeline consists of:

- **Waveform Generation Stage:** Produces digital audio signals based on the selected waveform configuration.
- **Modulation and Mixing Stage:** Combines audio signals with volume scaling and stereo distribution.
- **Digital-to-Analog Conversion Stage:** Outputs the final mixed signal through the DAC.

The modular nature of the pipeline allows for fine-tuned control of each channel, including volume adjustment, waveform modulation, and stereo balancing, making it suitable for a wide range of audio applications.

3
Functional Modules

3.1 Overview

The HUPUF2X8A APU is designed around a modular architecture, with each functional block responsible for a specific aspect of audio processing. The main functional modules are:

- Register Access Module
- Segmented Channels (6)
- Sampled Channels (2)
- Sampling & Mixing Circuit
- DAC Output Stage

Each module is configured through a set of registers. The complete list of registers, along with detailed descriptions and usage, is provided in the **Register Map and Configuration** section.

Figure 3: Block Diagram of Functional Modules

3.2 Register Access Module

The Register Access Module acts as the interface between external control signals and the APU's internal configuration registers. It centralizes all read/write operations and manages data distribution to the appropriate submodules.

Role:

- Manages register read/write operations via dedicated I/O pins.
- Handles address decoding and data transfer.
- Interfaces with both global and per-channel registers.

Components:

- **Demultiplexer:** Routes data to the correct internal register based on the address (RA0-RA3).
- **Read/Write Control:** Uses RRW (Read/Write) and RAE (Access Enable) to manage data flow.
- **Data Bus Interface:** Handles data transfer (D0-D7) between external signals and internal registers.
- **Global Register Storage:** Stores configuration and status information shared across the APU.
- **Channel Register Access:** Manages access to per-channel configuration registers.

Figure 4: Register Access Module Structure

Usage: The Register Access Module handles all I/O-based control of the APU. It decodes the register address and either writes or reads data through the external pins. Detailed information about the available registers is provided in the **Register Map and Configuration** section.

3.3 Segmented Channels (6)

Segmented channels are designed for generating complex waveforms by chaining multiple amplitude segments. Each segment represents a unidirectional amplitude change (rising, falling, or steady), implemented as a series of discrete steps.

Features:

- Up to 16 segments per channel.
- Configurable step length, height, and count.
- Flexible waveform generation, including pulse, triangle, and sawtooth patterns.
- Independent configuration per channel.

Channel Configuration: Segmented channels use a series of configuration registers to control waveform properties. Detailed configuration options for segmented channels can be found in the **Register Map and Configuration** section.

3.4 Sampled Channels (2)

Sampled channels are designed to play back raw audio data from a buffer. These channels are ideal for reproducing recorded audio or generating complex synthesized sounds.

Features:

- Supports various modulation types: amplitude, delta, period, noise.
- Configurable buffer size and offset for efficient data management.
- High flexibility for real-time audio playback.

Channel Configuration: Each sampled channel has dedicated configuration registers for buffer management and modulation settings. The details of these registers are presented in the **Register Map and Configuration** section.

3.5 Sampling & Mixing Circuit

The Sampling & Mixing Circuit combines the outputs of all active channels to produce a stereo audio signal. It operates at a fixed sampling rate derived from the master clock.

Functionality:

- Samples the output from each active channel.
- Applies volume scaling and stereo configuration.
- Combines the signals into left and right audio streams.
- Sends the mixed signal to the DAC for conversion.

Figure 5: Sampling and Mixing Circuit

Sampling Process: The circuit periodically samples each active channel, applies the configured volume and stereo balance, and mixes the results into a composite audio signal.

3.6 DAC Output Stage

The DAC output stage converts the mixed digital audio signals into analog waveforms, which are then output on the L and R pins.

Output Characteristics:

- Dual 8-bit resolution (Left and Right channels).
- Output voltage range: 0 to 3V.
- Compatible with line-level audio inputs.

Parameter	Value	Unit
Output Voltage Range	0-3	V
Output Impedance	600	
Recommended Load	10k	

Table 2: DAC Output Characteristics

Signal Flow: The final digital audio samples from the Sampling & Mixing Circuit are converted to analog signals using two DACs, one for the left channel and one for the right channel. These signals are suitable for direct output to audio devices or amplifiers.

3.7 Integration and Usage

For detailed programming and usage instructions, including how to configure each channel and control the output, please refer to the **Integration and Usage** section.

4

Register Map and Configuration

4.1 Overview

The HUPUF2X8A APU is controlled via a set of memory-mapped registers, accessed through the **Register Access Module**. These registers are divided into:

- **Global Registers:** Control the overall behavior and status of the APU.
- **Segmented Channel Registers:** Configure waveform generation for segmented channels.
- **Sampled Channel Registers:** Manage audio playback and modulation for sampled channels.

Registers are accessed via the **RA0-RA3** (address) and **RD0-RD7** (data) pins, with **RRW** controlling read/write operations and **RAE** enabling the access.

Bus Access Control: The **RAE (Register Access Enable)** pin controls the access to the data bus:

- When **RAE** is *not asserted* (low):
  - The **D0-D7** bus is set to **high impedance (Z)**, allowing other devices to use the data lines without interference.
  - The APU remains idle and will not respond to any read or write requests.
- When **RAE** is *asserted* (high):
  - The APU takes control of the data bus, allowing read or write operations.
  - The data pins (D0-D7) become active, either driving the data for a read operation or accepting input for a write.

4.2 Global Registers

Global Registers store configuration and status information for the entire APU. These registers are fixed and are not affected by channel selection.

Address	Name	Access	Description
0x0000	Status	R	Current state of the APU, including error flags.
0x0001	Active Channels	R/W	Bitmask of active audio channels (8 bits).
0x0010	Global Flags	R/W	Configuration flags (MixingAvg, etc.).
0x0011	Mixing Volume A	R/W	Volume for channels 0-3 (4x2b).
0x0100	Mixing Volume B	R/W	Volume for channels 4-7 (4x2b).
0x0101	Mixing Stereo L	R/W	Left channel enable (8x1b).
0x0110	Mixing Stereo R	R/W	Right channel enable (8x1b).
0x0111	Select Exposed	R/W	Channel (4b) and Segment (4b) selection.

Table 3: Global Registers

Usage: Global registers manage the overall audio configuration, including:

- Enabling or disabling channels.
- Setting global audio parameters.
- Configuring stereo output (Left/Right).
- Adjusting mixing volume for each channel.

4.3 Segmented Channel Registers

Segmented channel registers control waveform generation by managing individual segments. These channels are capable of producing complex waveforms by chaining amplitude segments. Each channel has a specific quality level, affecting the number of segments and the precision of the waveform.

All channels have the same properties, but the number of segments and the size of each segment property may vary based on the channel's quality.

The common properties for all channels are stored in the two **Channel Conf** registers. The **Segment Conf** registers are used to configure the segments of the channel.

Address	Name	Access	Description
0x1000	Channel Conf0	R/W	SegmentIdShift (4b) + Flags (4b).
0x1001	Channel Conf1	R/W	SegmentActiveId (4b) + SegmentMaxId (4b).

Table 4: Segmented Channel Registers (Common)

High-Quality Channels (0 and 1): The high-quality channels (0 and 1) can have up to 16 segments, with each segment having a maximum of 256 steps. The step precision is 9 bits (signed), allowing for precise control of the amplitude modulation (between -255 and +255 per step). The step length is 15 bits with a scale of +1, allowing for a maximum of 2^{15+1} cycles per step (1 to 65536).

The total size of one segment configuration is 4 bytes.

Address	Name	Access	Description
0x1010	Segment Conf0	R/W	StepSign (1b) + StepLength (7b hi).
0x1011	Segment Conf1	R/W	StepLength (8b lo).
0x1100	Segment Conf2	R/W	StepHeight (8b unsigned).
0x1101	Segment Conf3	R/W	StepCount (8b).

Table 5: Segmented Channel Registers (High Quality)

Medium-Quality Channels (2 and 3): Medium-quality channels (2 and 3) provide a balance between configurability and resource usage. Each channel supports up to 8 segments, with each segment having up to 128 steps. The step precision is 9 bits (signed), allowing for precise control of the amplitude modulation (between -255 and +255 per step). The step length is 8 bits with a scale of +4, allowing for a maximum of 2^{8+4} cycles per step (1 to 4096).

The segment configuration is more compact, using 3 bytes per segment.

Address	Name	Access	Description
0x1010	Segment Conf0	R/W	StepSign (1b) + StepCount (7b).
0x1011	Segment Conf1	R/W	StepLength (8b).
0x1100	Segment Conf2	R/W	StepHeight (8b unsigned).
0x1101	Unused	-	Unused.

Table 6: Segmented Channel Registers (Medium Quality)

Low-Quality Channels (4 and 5): Low-quality channels (4 and 5) are optimized for minimal configuration and fast updates, exposing their two segments per channel directly. The step precision is 5 bits (signed), multiplied by 17 allowing for approximate control of the amplitude

modulation on the [-255,+255] range. The step length is 7 bits with a scale of +5, allowing for a maximum of 2^{8+5} cycles per step (1 to 4096).

Each segment uses a compact 2-byte configuration, suitable for simple waveforms or rapid modulation.

Address	Name	Access	Description
0x1010	Segment0 Conf0	R/W	`SLLLLLLL`: StepSign (1b), StepLength (7b).
0x1011	Segment0 Conf1	R/W	`CCCCHHHH`: StepCount (4b), StepHeight (4b unsigned).
0x1100	Segment1 Conf0	R/W	`SLLLLLLL`: StepSign (1b), StepLength (7b).
0x1101	Segment1 Conf1	R/W	`CCCCHHHH`: StepCount (4b), StepHeight (4b unsigned).

Table 7: Segmented Channel Registers (Low Quality)

4.4 Sampled Channel Registers

Sampled channels are designed to play back raw audio data from a buffer and support various modulation techniques. These channels are suitable for producing complex audio patterns and recorded sound playback.

4.4.1 Common Configuration Registers

The sampled channels share a set of common configuration registers, which define the basic properties of the channel and its buffer.

Note that the buffers are always exposed (whatever channel/segment is selected) on ports 0x1110 and 0x1111, allowing for continuous streaming of audio data without needing to switch the selected channel.

Address	Name	Access	Description
0x1000	Channel Conf0	R/W	Modulation Mode (3b) + Flags (5b).
0x1001	Channel Conf1	R/W	BufferOffset (4b) + BufferMaxOffset (4b).
0x1010	Channel Conf2	R/W	BufferShift (4b) + BufferWriteOffset (4b).
0x1011	Mode Conf0	R/W	Mode-specific configuration.
0x1100	Mode Conf1	R/W	Mode-specific configuration.
0x1101	Mode Conf2	R/W	Mode-specific configuration.
0x1110	Sample Buffer A	W	Write to buffer of the first sampled channel.
0x1111	Sample Buffer B	W	Write to buffer of the second sampled channel.

Table 8: Sampled Channel Registers

4.4.2 Sampling Modes

The sampled channels support multiple modulation types, allowing for diverse sound synthesis:

- **AmpMode (Amplitude Modulation):** Directly sets amplitude from sample data.
- **SignMode (Delta Sign Modulation):** Uses a single bit to toggle the sign of the delta.
- **PeriodMode (Period Modulation):** Alternates between MinAmp and MaxAmp.
- **NoiseMode (Random Noise Modulation):** Uses pseudo-random bits as the sample stream.
- **DeltaMode (Delta Modulation):** The sample provides a signed delta to apply.

Buffer Writing: The sampled channels use two dedicated registers to write audio data. These registers are always exposed, allowing continuous audio data streaming without switching the selected channel.

4.4.3 AmpMode (Amplitude Modulation)

Description: In AmpMode, each sample directly sets the amplitude. This mode is ideal for playing back audio with precise amplitude control.

Address	Name	Access	Description
0x1011	PeriodLength_Lo	R/W	Duration of each sample in SamplingCycles (low byte).
0x1100	PeriodLength_Hi	R/W	Duration of each sample (high byte).
0x1101	BitsPerSample	R/W	Number of bits per sample (1-8).

Table 9: AmpMode Registers

4.4.4 SignMode (Delta Sign Modulation)

Description: In SignMode, each sample bit toggles the sign of a fixed amplitude delta. This mode produces waveforms with rapid sign changes.

Address	Name	Access	Description
0x1011	PeriodLength_Lo	R/W	Duration of each sample in SamplingCycles (low byte).
0x1100	PeriodLength_Hi	R/W	Duration of each sample (high byte).
0x1101	DeltaAmplitude	R/W	Amplitude change when sign toggles.

Table 10: SignMode Registers

4.4.5 PeriodMode (Period Modulation)

Description: Alternates between MinAmp and MaxAmp based on the sample value, creating a periodic waveform.

Address	Name	Access	Description
0x1011	AAABBBB0	R/W	A=BitsPerSample (3b), B=PeriodE2Scale (4b).
0x1100	AmpMin	R/W	Minimum amplitude during the period.
0x1101	AmpMax	R/W	Maximum amplitude during the period.

Table 11: PeriodMode Registers

4.4.6 NoiseMode (Random Noise Modulation)

Description: Uses a pseudo-random bit stream to generate noise. The configuration allows control over the noise frequency and characteristics.

Address	Name	Access	Description
0x1011	PeriodLength	R/W	Duration of each noise sample in SamplingCycles.
0x1100	XorMask	R/W	Mask for noise generation.
0x1101	RandomConfig	R/W	Configuration for pseudo-random number generation.

Table 12: NoiseMode Registers

4.4.7 DeltaMode (Delta Modulation)

Description: Uses the sample as a signed delta value, modifying the current amplitude directly.

Address	Name	Access	Description
0x1011	PeriodLength_Lo	R/W	Duration of each sample in SamplingCycles (low byte).
0x1100	PeriodLength_Hi	R/W	Duration of each sample (high byte).
0x1101	BitsPerSample	R/W	Number of bits per sample (1-8).

Table 13: DeltaMode Registers

Usage Note: Choosing the appropriate mode is critical for achieving the desired audio effect. Switching between modes dynamically can result in unexpected audio artifacts, so it is recommended to stop playback when changing modes.

4.5 Register Access Protocol

To perform a read or write operation:

1. Set the address on **RA0-RA3**.
2. Set **RRW** (1 for read, 0 for write).
3. Place data on **RD0-RD7** (for write).
4. Assert **RAE** (high) to enable access.
5. Pulse **CLK** to latch the data.
6. Deassert **RAE** (low) to release the bus.

5

Clock and Timing

5.1 Master Clock

Description of clock source and synchronization.

5.2 Sampling Rate

Relation between clock speed and audio output.

5.3 Latency and Constraints

Expected processing delays.

6
Performance Characteristics

6.1 Channel Capacity

Max simultaneous channels and tone generation.

6.2 Latency

Measured delays from input to output.

6.3 Power Consumption

Energy efficiency and expected consumption.

7

Integration and Usage

7.1 Wiring Diagrams

Example wiring for headphones and RCA output.

7.2 Usage Scenarios

Step-by-step setup for basic sound generation.

7.3 Programming Interface

Methods to program and interact with the APU.

8

Debugging and Testing

8.1 Built-in Diagnostics

Test modes and validation procedures.

8.2 Signal Validation

Test signals and output expectations.

8.3 Troubleshooting

Common issues and their resolutions.

A
Electrical Characteristics

Power and signal specifications.

B

Example Code

Sample scripts and configurations.

C
Revision History

Track changes and updates to the document.