

VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY  
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY  
FACULTY OF ELECTRICAL AND ELECTRONICS ENGINEERING

—o0o—



---

LABORATORY REPORT

**Waveform generator**

---

SUPERVISOR: Nguyễn Tuấn Hùng

SUBJECT: Digital signal processing on FPGA

GROUP: 08

List of Members

STT	MSSV	Họ Và Tên	Lớp
1	2213874	Nguyễn Thanh Tùng	L01
2	2210780	Nguyễn Đại Đồng	L01
3	2213496	Nguyễn Quốc Tín	L01

Ho Chi Minh, ../../20..

# Mục lục

<b>1</b>	<b>Theoretical Background</b>	<b>4</b>
1.1	Phase accumulator . . . . .	4
1.2	Ý tưởng tạo các loại sóng . . . . .	5
1.2.1	Sóng Sin . . . . .	5
1.2.2	Sóng ECG . . . . .	7
1.2.3	Sóng vuông . . . . .	7
1.2.4	Sóng răng cưa . . . . .	8
1.2.5	Sóng tam giác . . . . .	9
1.3	Giao thức $I_2C$ . . . . .	9
1.3.1	Cấu hình thanh ghi cho WM8731 . . . . .	9
<b>2</b>	<b>Using SystemVerilog</b>	<b>12</b>
2.1	Topmodule Waveform generator . . . . .	12

## Danh sách hình vẽ

1.1	Bánh xe pha. . . . .	4
1.2	Sóng Sin khi đổ KIT. . . . .	6
1.3	Sóng ECG khi đổ KIT. . . . .	7
1.4	Sóng Răng cưa. . . . .	8
1.5	Sóng tam giác. . . . .	9
2.1	Các chức năng thực hiện bởi người dùng. . . . .	12
2.2	Sơ đồ khối của bộ điều khiển module Waveform Generator. . . . .	13

Danh sách bảng

1.1    Cấu hình thanh ghi cho WM8731    . . . . . 10

## 1 Theoretical Background

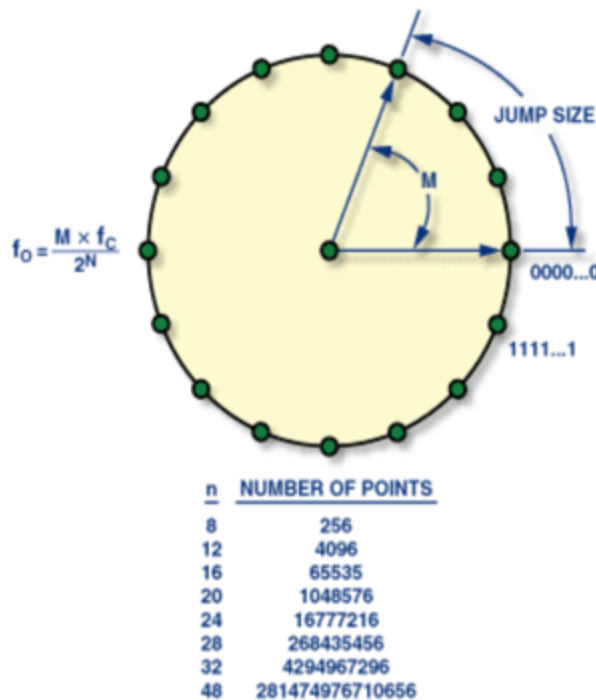
### 1.1 Phase accumulator

Phase Accumulator là một thanh ghi (register)  $n$ -bit chạy modulo  $2^n$ , mỗi chu kỳ xung clock nó sẽ cộng thêm một giá trị cố định  $M$  (Frequency Control Word: gọi tắt là FCW hay tuning word) là một số (thường là số nguyên) được đưa vào bộ tích lũy pha để điều khiển tốc độ tăng pha, từ đó quyết định tần số sóng đầu ra.

$FCW = M$  là giá trị “*nhảy pha*” (phase step) mỗi chu kỳ. Nếu FCW lớn, pha tăng nhanh, tần số sóng cao. Nếu FCW nhỏ, pha tăng chậm, tần số thấp.

**Cách hoạt động:** Mỗi khi có một xung clock (nhịp đồng hồ), bộ tích lũy pha sẽ cộng thêm một giá trị cố định vào giá trị hiện tại của nó

⇐ Kết quả tích lũy chính là “*pha*” (phase) hiện tại, tăng đều đặn từ  $0 \leftarrow 2^n - 1$  rồi quay vòng về 0, xây dựng nên một “*bánh xe pha*” (phase wheel) tuyến tính như hình 1.1.



Hình 1.1: Bánh xe pha.

Giá trị này tăng dần, giống như một đường dốc (ramp). Khi nó đạt đến giá trị tối đa thì

nó sẽ "tràn"(overflow) và quay lại 0 giống như kim đồng hồ quay hết một vòng rồi trở lại vị trí ban đầu.

⇐ Ta còn gọi đây là hiện tượng wrap-around.

Tốc độ tăng của nó quyết định tần số sóng đầu ra theo công thức:

$$f_{out} = \frac{M}{2^n} f_{clk}$$

Trong đó,

- $FCW = M$  là giá trị “*nhảy pha*” (phase step) mỗi chu kỳ. Nếu FCW lớn, pha tăng nhanh, tần số sóng cao. Nếu FCW nhỏ, pha tăng chậm, tần số thấp.
- $n$ : Số bit của bộ tích lũy pha.
- $f_{clk}$ : Tần số điều khiển phase accumulator.
- $f_{out}$ : Tần số đầu ra mong muốn.

### 1.2 Ý tưởng tạo các loại sóng

Trước tiên, đối với các sóng dạng phức tạp như sin hay ECG thì ta tạo sóng bằng cách cộng dồn pha và ánh xạ pha này sang biên độ sóng thông qua **bảng tra (LUT)**.

#### 1.2.1 Sóng Sin

Ý tưởng thực hiện: Tổng có 1024 mẫu tức là phải lưu đầy đủ từ  $0^\circ \leftarrow 360^\circ$  cho cả 1 chu kỳ sóng sin. Nhưng làm vậy sẽ tốn gấp 4 lần dung lượng ROM ← Nếu chỉ cần  $\frac{1}{4}$  chu kỳ, nghĩa là lưu từ  $0^\circ \leftarrow 90^\circ$ .

Đầu tiên, ta dùng bit MSB để phân biệt 2 vùng sin dương ( $0^\circ \leftarrow 180^\circ$ ) ở 2 phần tư đầu của chu kỳ và vùng sin âm ( $180^\circ \leftarrow 360^\circ$ ) ở 2 phần tư còn lại. Tiếp theo, tại mỗi vùng, ta dùng bit[MSB-1] làm mốc đối xứng để cho ta biết khi nào cần thực hiện đảo địa chỉ và thu được nửa sóng sin hoàn chỉnh.

Với nửa chu kỳ dương đầu tiên của sóng sin, ta chọn phase step tùy thuộc vào tần số đầu ra mong muốn. Bộ Phase accumulator sẽ tích lũy và tăng dần địa chỉ ở phần tư đầu tiên theo  $\frac{1}{4}$  mẫu đã lưu khi bit[MSB-1] = 0. Nếu bit[MSB-1]=1 thì thực hiện đảo (mirror) bằng cách lấy

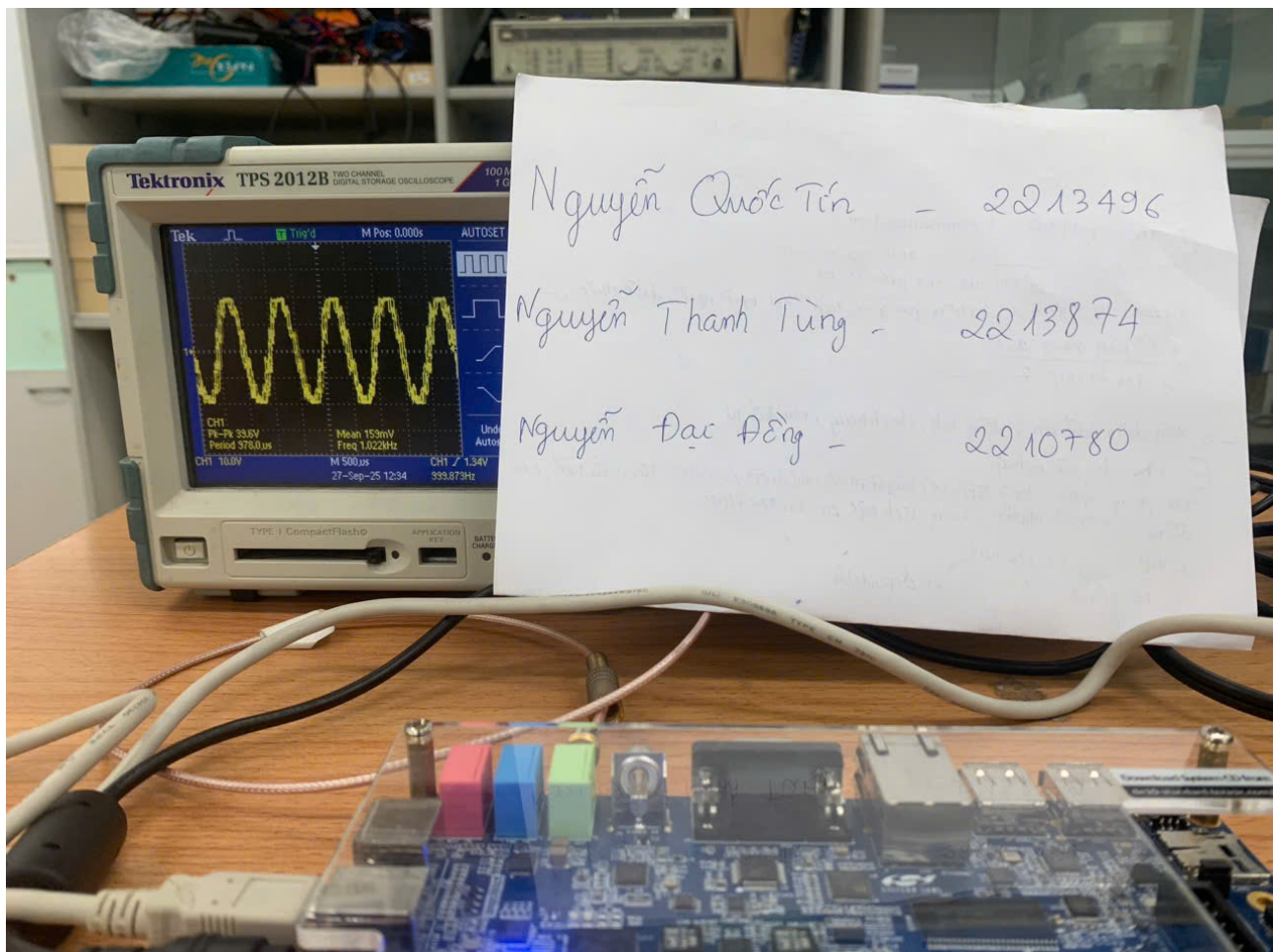
## 1 THEORETICAL BACKGROUND

giá trị Maxium - địa chỉ của phần tử đầu tiên hoặc ta thực hiện phép NOT địa chỉ của phần tử đầu tiên. Ta thu được phần thứ hai và hoàn chỉnh được nửa chu kỳ dương của sóng sin.

Với nửa chu kỳ âm, ta cần thực hiện lấy bù 2 để chuyển vùng từ dương sang âm dựa vào bit  $MSB = 1$ . Sau đó, mirror để thu được phần tử thứ ba của sóng sin. Cuối cùng, phần tử cuối sẽ mirror lại phần tử thứ ba.

Tóm lại, bit  $MSB$  để quyết định vùng dương hoặc âm, bit  $[MSB-1]$  để quyết định lấy đảo (mirror) hay không.

Kết quả thu được:

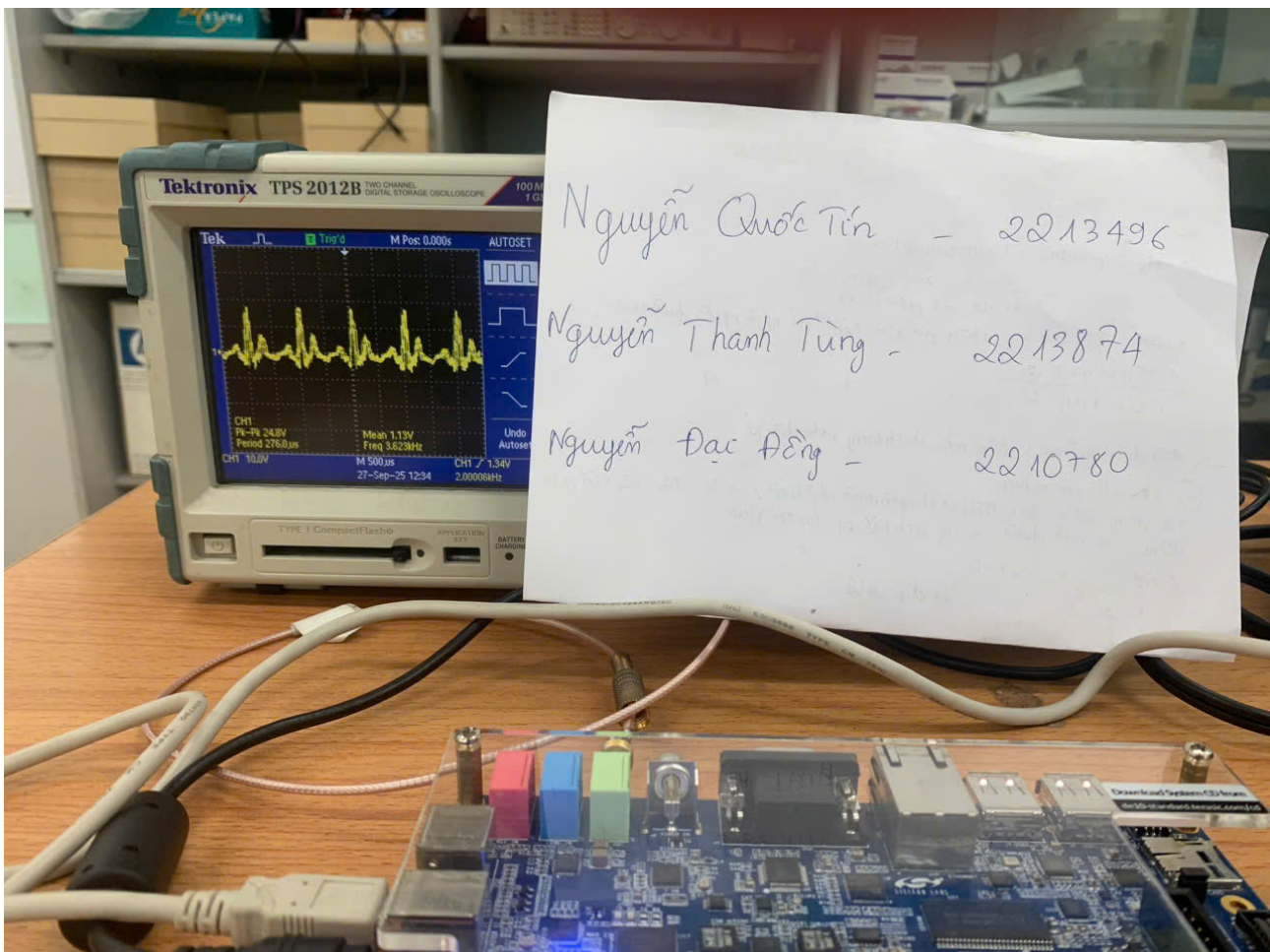


Hình 1.2: Sóng Sin khi đổ KIT.

## 1 THEORETICAL BACKGROUND

### 1.2.2 Sóng ECG

Với sóng ECG khá phức tạp, nên nhóm sẽ sử dụng file HEX chức năng là làm LUT với 1024 mẫu. Tương tự, Phase accumulator sẽ tích lũy pha và tăng dần địa chỉ với mỗi địa chỉ chứa một giá trị biên độ của ECG. Nhờ đó, ROM lần lượt xuất ra từng mẫu ECG. Thực hiện ghép các mẫu liên tiếp theo thời gian sẽ thành sóng ECG hoàn chỉnh. Kết quả thu được:



Hình 1.3: Sóng ECG khi đổ KIT.

### 1.2.3 Sóng vuông

Đối với sóng vuông đơn giản, ta thực hiện nó bằng logic bằng không cần dùng tới LUT. Cụ thể:

Bộ Phase accumulator đóng vai trò làm “thước đo” thời gian trong một chu kỳ sóng. Một chu kỳ sóng vuông có hai mức: HIGH (24'h0FFF00) và LOW (24'h0). Sóng được xác định bằng



## 1 THEORETICAL BACKGROUND

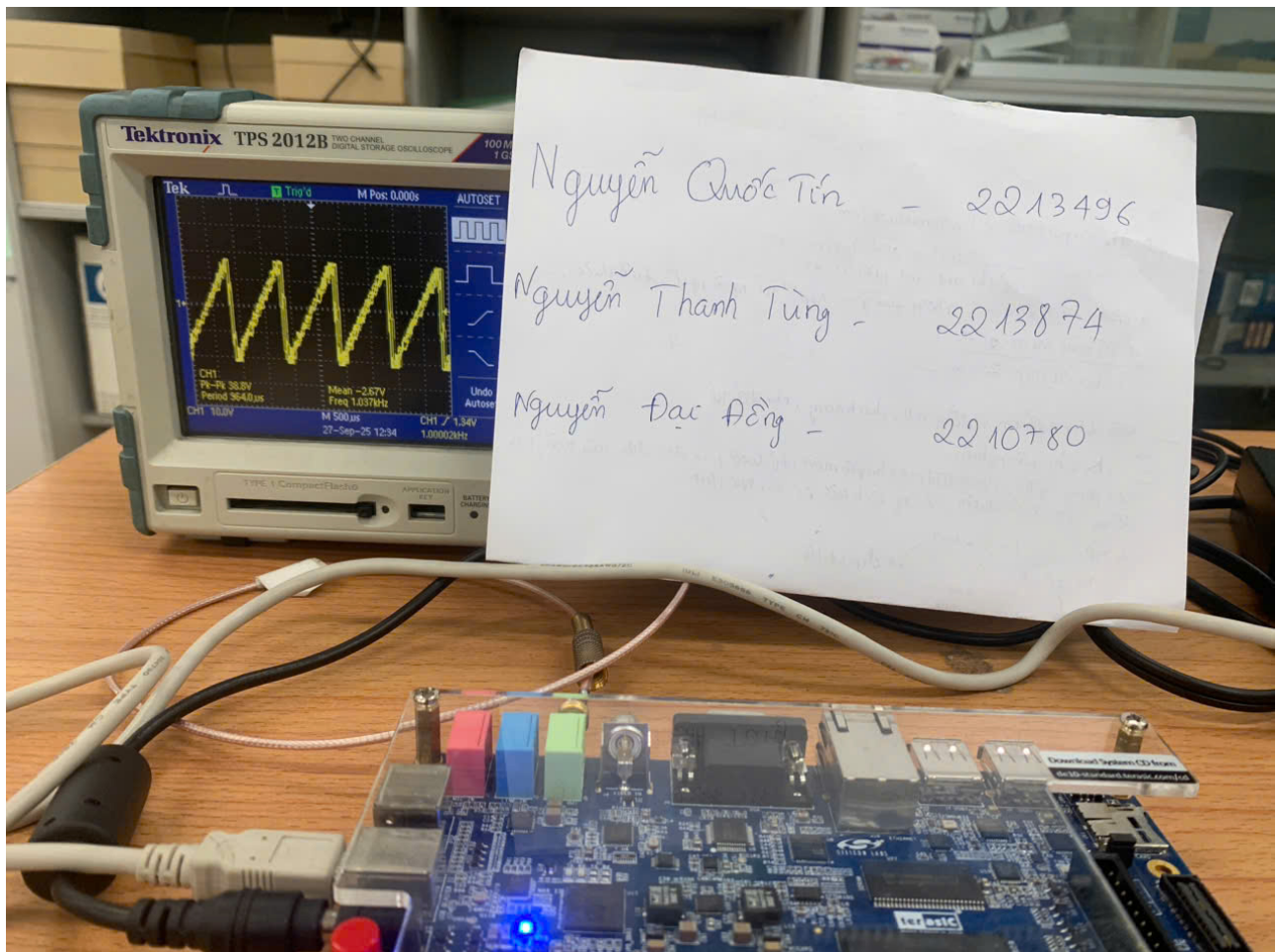
cách so sánh pha đã tích lũy được với các ngưỡng (threshold).

Xác định những giá trị duty cycle mong muốn là bao nhiêu. Từ đó, xác định giá trị ngưỡng để xác định HIGH và LOW. Ví dụ: duty cycle = 90% của 1024 giá trị thì cần đếm 10'đ102 để thực hiện mức LOW trước và ta chọn điểm này làm ngưỡng. Khi vượt ngưỡng thì đảo lại HIGH. Tương tự cho các giá trị duty cycle khác.

Một vài duty cycle đặc biệt như 50% thì ta có thể lợi dụng tính đối xứng dựa vào bit MSB. Đếm đủ một nửa MSB = 0 thì thực hiện đảo ở MSB = 1. Hay duty cycle = 75%, 25% thì ta thực hiện OR 2 bit MSB.

### 1.2.4 Sóng răng cưa

Ta thực hiện tích lũy pha như mọi sóng khác, lợi dụng hiện tượng wrap-around để tích lũy pha dần (đường ramp tuyến tính) và khi tràn sẽ lập tức quay về 0. Kết quả thu được:



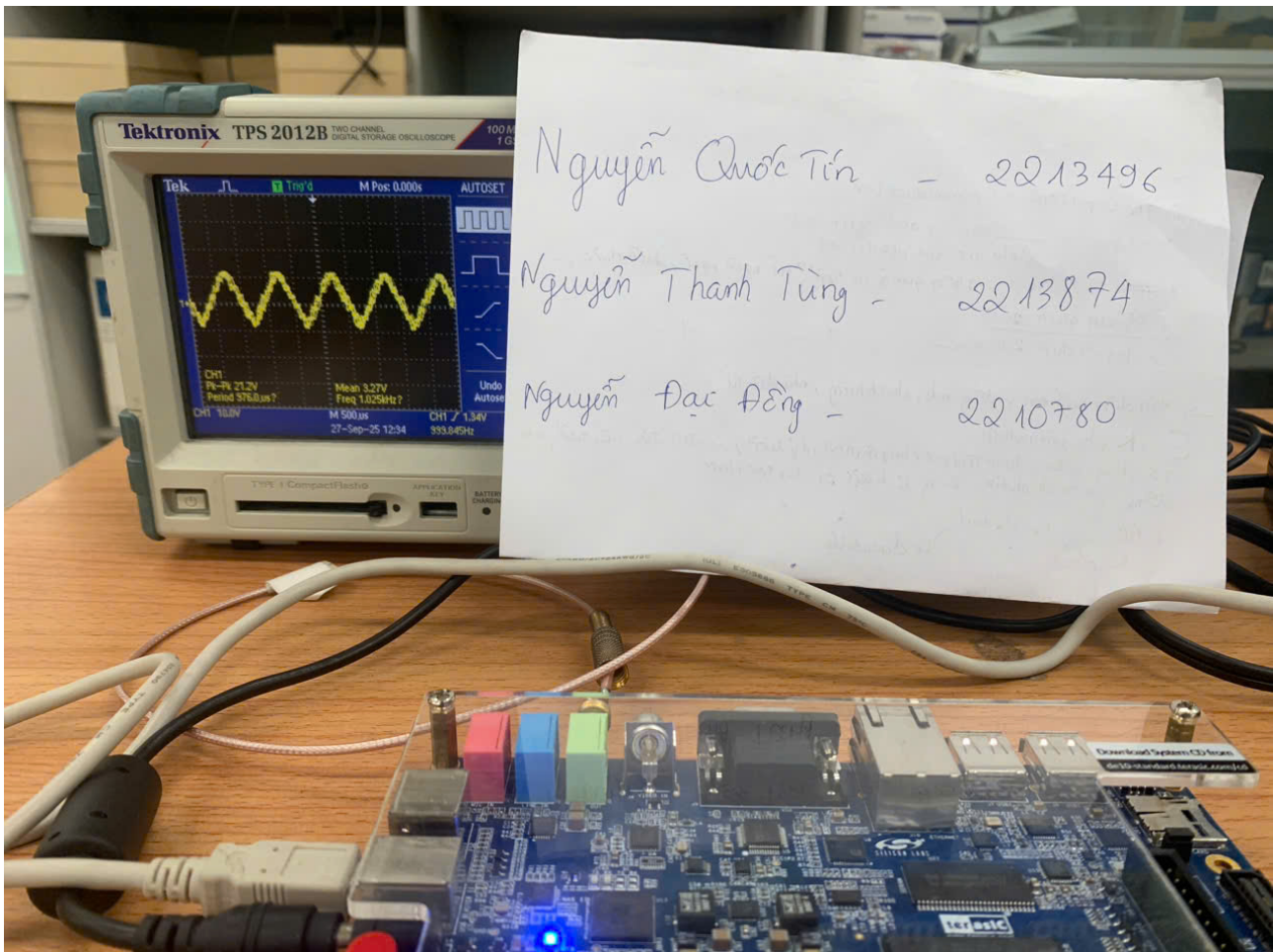
Hình 1.4: Sóng Răng cưa.



## 1 THEORETICAL BACKGROUND

### 1.2.5 Sóng tam giác

Được tạo dựa vào sóng răng cưa với 2 đường ramp tăng tuyến tính ở nửa đầu và giảm tuyến tính ở nửa sau. Thực hiện bằng cách tích lũy pha tạo đường ramp tuyến tính như sóng răng cưa. Sau đó, lợi dụng tính đối xứng dựa vào bit MSB để đảo đường ramp ở nửa phần sau hoặc lấy giá trị Maximum – giá trị tích lũy dần ở nửa đường ramp thứ nhất. Kết quả thu được:



Hình 1.5: Sóng tam giác.

## 1.3 Giao thức $I_2C$

### 1.3.1 Cấu hình thanh ghi cho WM8731

Ta thực hiện cấu hình cho chip WM8731 để chip hoạt động theo yêu cầu của Lab và các thanh ghi cấu hình này sẽ được truyền đi thông qua giao thức  $I_2C$ .

Bảng 1.1: Cấu hình thanh ghi cho WM8731

Index	Address	Data	Thanh ghi	Chức năng
0	0x0F	0x00	Reset Register	Reset toàn bộ codec về trạng thái mặc định
1	0x06	0x10	Power Down Control	Tắt MIC input, bật LINEOUT, LINEIN, ADC, DAC
2	0x02	0x79	Left Headphone Out	Volume tai nghe trái: 0 dB, bật Zero-Cross detect
3	0x03	0x79	Right Headphone Out	Volume tai nghe phải: 0 dB, bật Zero-Cross detect
4	0x00	0x17	Left Line In	Volume line-in trái: 0 dB, không mute
5	0x01	0x17	Right Line In	Volume line-in phải: 0 dB, không mute
6	0x04	0x10	Analog Audio Path Control	Chọn LINEIN → ADC, tắt MIC, tắt bypass, bật DAC
7	0x05	0x00	Digital Audio Path Control	Tắt high-pass filter, tắt de-emphasis, tắt soft-mute DAC
8	0x07	0x01	Digital Audio Interface Format	Chọn mode Left-justified, 24-bit, MSB first
9	0x08	0x01	Sampling Control	USB mode, MCLK=12 MHz, Fs=48 kHz
10	0x09	0x01	Active Control	Kích hoạt codec (ACTIVE=1)

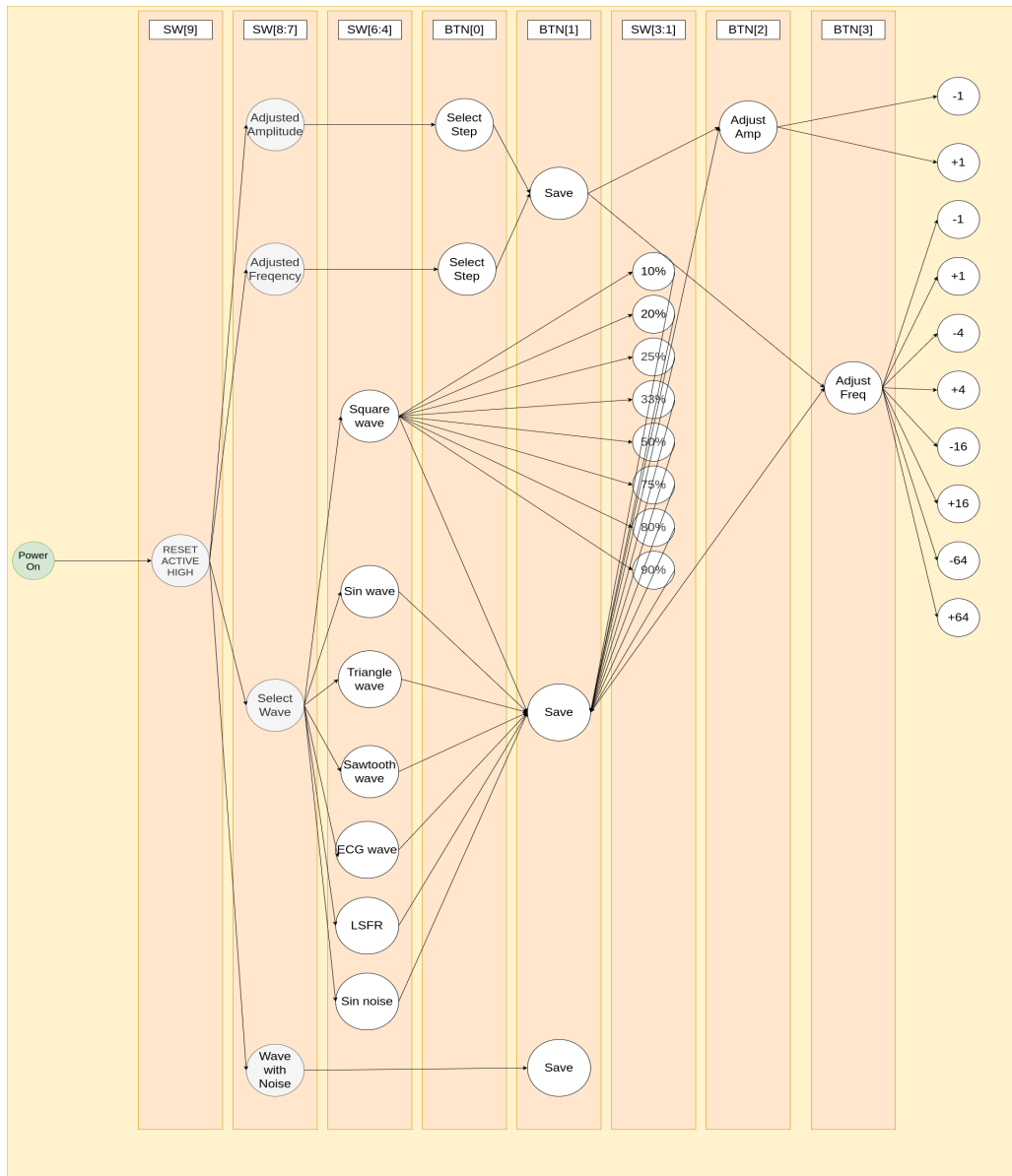
Next page...

**Bảng 1.1 Cấu hình thanh ghi cho WM8731**

<b>Index</b>	<b>Address</b>	<b>Data</b>	<b>Thanh ghi</b>	<b>Chức năng</b>
11	0x06	0x02	Power Down Control	Giữ ADC, DAC, LINEIN, LINEOUT hoạt động; chỉ PD Microphone

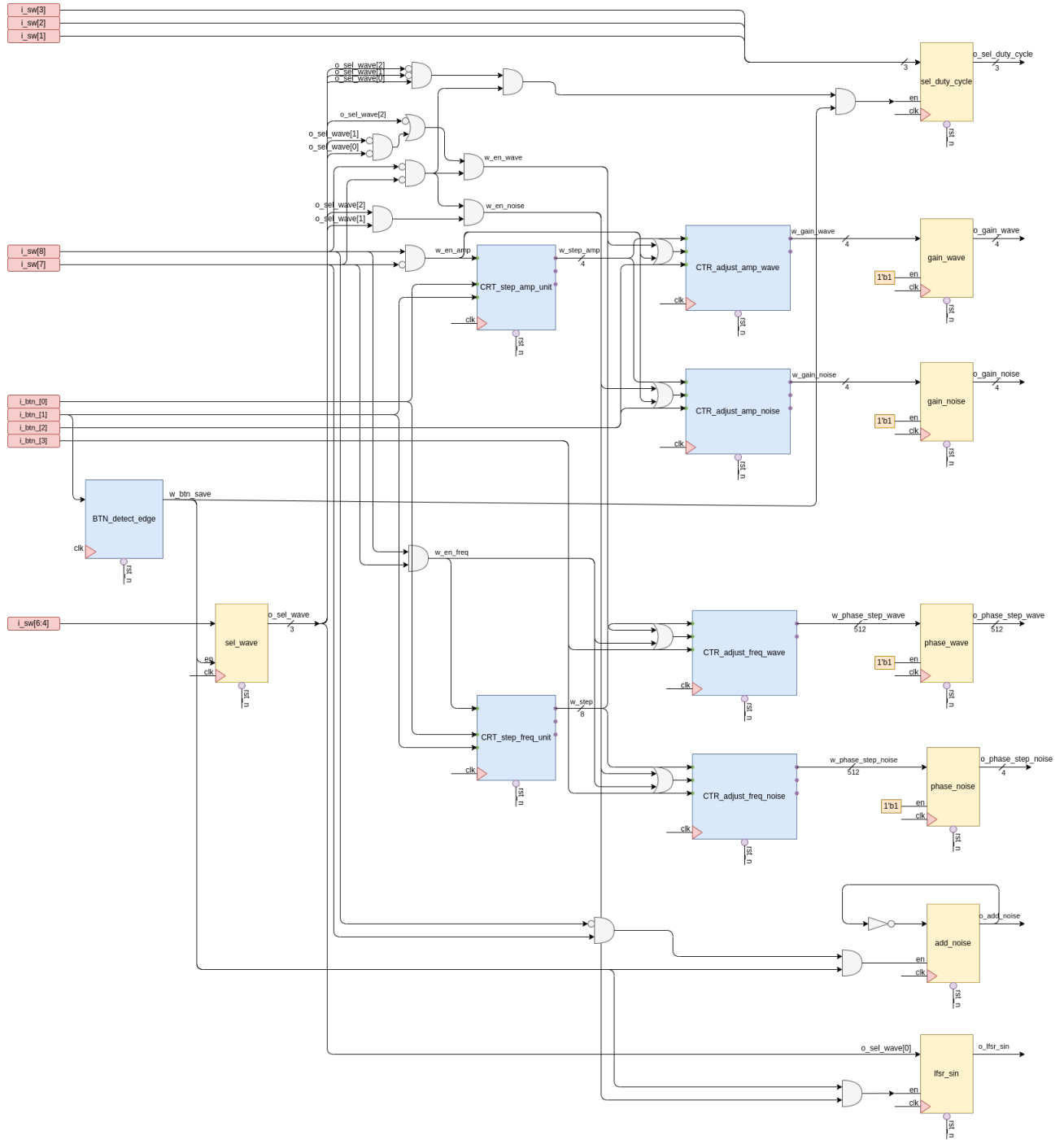
## 2 Using SystemVerilog

### 2.1 Topmodule Waveform generator



Hình 2.1: Các chức năng thực hiện bởi người dùng.

## 2 USING SYSTEMVERILOG



Hình 2.2: Sơ đồ khối của bộ điều khiển module Waveform Generator.