The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

# A lightweight sensing method of tooth-touch sound for disabled person using remote controller

Akira Yamawaki[1] and Seiichi Serikawa[1]

[1] Kyushu Institute of Technology
1-1, Sensui, Tobata, Kitakyushu, Fukuoka 804-8550, Japan
yama@ecs.kyutech.ac.jp

**Abstract:** To support disabled people using remote controllers, biological signals have been applied to. We propose a lightweight sensing method extracting the tooth-touch without a sophisticated signal processing to eliminate the normal audio sound. Proposal uses a shock wave (i.e. ultrasonic wave) which is generated when the upper and lower tooth hit each other, instead of the sound wave of tooth-touch. By our method, the signal processing has only to perform a high-pass filter eliminating lower frequency domain than the ultrasonic domain. Through the preliminary experiment with a conventional microphone, we show that the tooth-touch has the larger power than the voice sounds in the ultrasonic region. Then, we design the filtering hardware to implement a small and cheap SoC. Through the implementation to the FPGA, and the simulation, we show that our hardware is small and has the enough performance for a real-time operation.

**Keywords:** Disable people, Supporting system, Remote control, Tooth-touch, Ultrasonic

## 1 INTRODUCTION

Since the remote controllers for the house appliances are becoming complex, a method supporting disabled people to use them is very important for providing more efficient social life. To tackle such problem, the biological signals such as voice, eye blinks, chin operated control sticks, mouth sticks [1], and brain computer interfaces [2, 3] have been applied to the supporting systems.

The tooth-touch sound is one of the simplest biological signals, i.e. the simplest human actions. It can be easily sensed by the trivial sensors such as electret condenser microphone. In general, to realize the remote controller by the tooth-touch sound, the sequences of the tooth-touch sounds are converted to the operation code sequences [4]. However, since the tooth-touch sound is mixed with the normal audio sound, the sophisticated signal processing to extract only the tooth-touch sound is needed [4]. In order to spread the supporting system using tooth-touch sound to many disabled people, the system must be very cheap, small and low-power consumption for the battery. That is, the signal processing has to be simple and mature enough to realize a low-cost hardware module which can be mounted to a cheap system-on-chip (SoC).

This paper proposes a lightweight sensing method extracting the "**tooth-touch**" that does not need a sophisticated signal processing to eliminate the normal audio sound. This method uses an shock wave (i.e. ultrasonic wave) which is generated when the upper tooth and the lower tooth hit each other, instead of the sound wave of tooth-touch. Conventional microphones such as electret condenser microphone have a sensitivity to the ultrasonic wave. By our proposed method, the the signal processing extracting the tooth-touch has only to perform a well-known filter eliminating lower frequency domain than the ultrasonic domain.

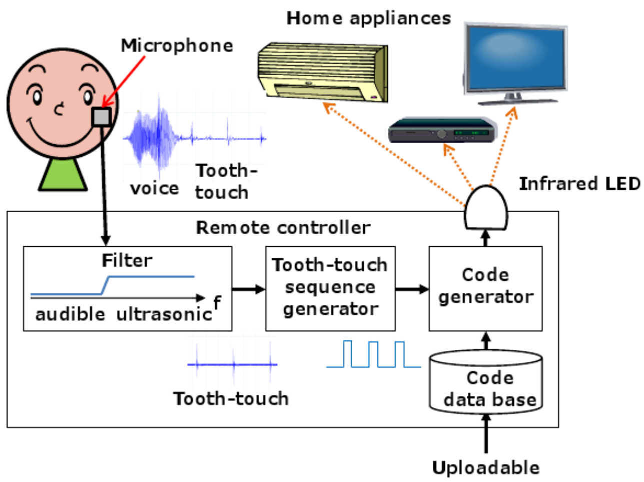


Fig. 1. System Overview.

The rest of this paper is organized as follows. Section 2 describes the system overview which employs our proposed method. Section 3 confirms that the ultrasonic wave can be detected by the conventional microphone, while hitting the upper tooth and the lower tooth. Section 4 designs the filtering hardware based on the above analysis by the conventional FFT and IFFT pair [5]. Then, Section 5 shows the performance and the hardware size of our filter hardware. Finally, Section 6 concludes this paper.

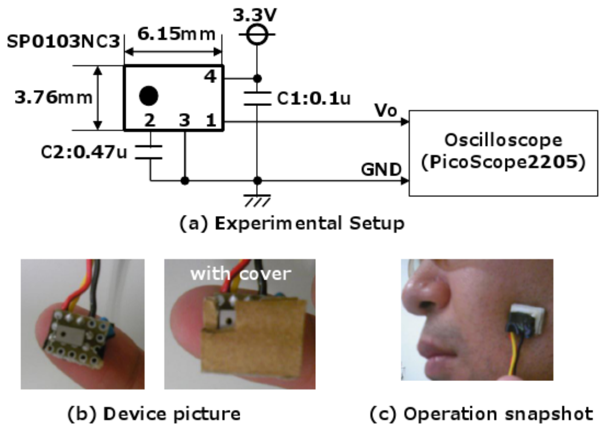


Fig. 2. Prototype Hardware.

## 2 SYSTEM OVERVIEW

Fig. 1 shows a system overview employing our proposed method. A microphone which has sensitivity to the ultrasonic region is attached to the cheek of the user. This is because if the microphone resides in front of the mouth, it may disturb the user's talking, eating, drinking and so on.

The shock wave is generated when one material strikes another material [8]. That is, when the upper tooth strikes the lower tooth, the ultrasonic wave that is equal to the shock wave may also be generated. However, this case would need to strike them rather strongly compared with the normal chewing. In general, many cheap and trivial microphones can generate some output to the ultrasonic wave [6, 7]. Thus, a special and expensive microphone is not needed to sense the ultrasonic wave.

The waveform from the microphone includes the voice and the tooth-touch. To leave only the ultrasonic wave of the tooth-touch from the waveform, it is input to a highpass filter with the cutoff frequency near the entrance of the ultrasonic region.

To make the command sequence from the tooth-touch sequence, the waveform including only the ultrasonic wave is rectified to the pulse wave. Considering of the number, the time duration and the time interval of pulses, the command sequence is generated [4]. Finally, the remote controlling code is generated by matching the generated command sequence with the registered it in the data base.

The data base can be updated arbitrarily. By using the data base, the differences among the makers and the products of the same maker can be hidden.

## 3 WAVEFORM ANALYSIS

### 3.1 Experimental Setup

To confirm that the ultrasonic wave can be detected by the conventional microphone, while hitting the upper tooth and the lower tooth, we have developed a prototype hardware shown in Fig. 2.

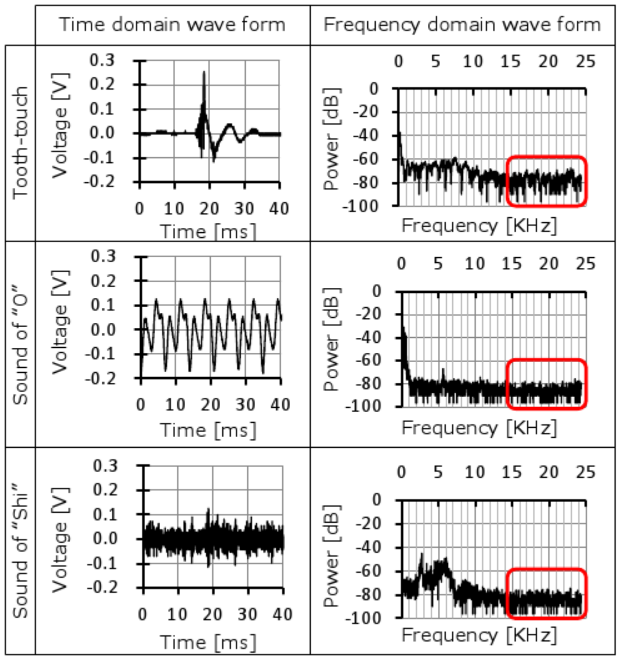


Fig. 3. Waveform Analysis.

The microphone used is the SP0103NC3 of Knowles Acoustics co. which has the wide sensitivity ranging from 100Hz to 10KHz or more [6]. The microphone is given 3.3V power. The capacitor of C1 in Fig. 2 (a) is a bipass condenser. This microphone includes an amplifier. The capacitor C2 is used to make its gain 10 times. Fig. 2 (b) is the pictures of the device we have developed. We also have made a cover of the corrugated cardboard so that the microphone does not contact directly to the cheek. When measuring, the microphone with the cover is attached to the user's cheek as shown in Fig. 2 (c).

We have probed the output voltage of the microphone by the oscilloscope, PicoScope2205 [9]. Via this oscilloscope, the waveform is acquired into the personal computer. The acquired waveforms are analyzed by the FFT we have developed by C language at the frequency domain. The rectangular window is used in this analysis. The time frame is about 42 ms and the number of data points is 2048. Thus, the sampling period is 20.5 us and the sampling frequency is 48 KHz. Consequently, the FFT can analyze the frequency ranging from 0 Hz to 24 KHz, with the resolution of about 23 Hz.

### 3.2 Result and Discussion

The waveforms of the time and frequency domains are shown in Fig. 3. As shown in Fig. 3, the tooth-touch has the larger power than that of the voice sounds, in the ultrasonic region from 15 KHz to 24 KHz. The difference between them reaches about 10 dB. This fact indicates that a conventional microphone can actually sense the shock wave generated by

 

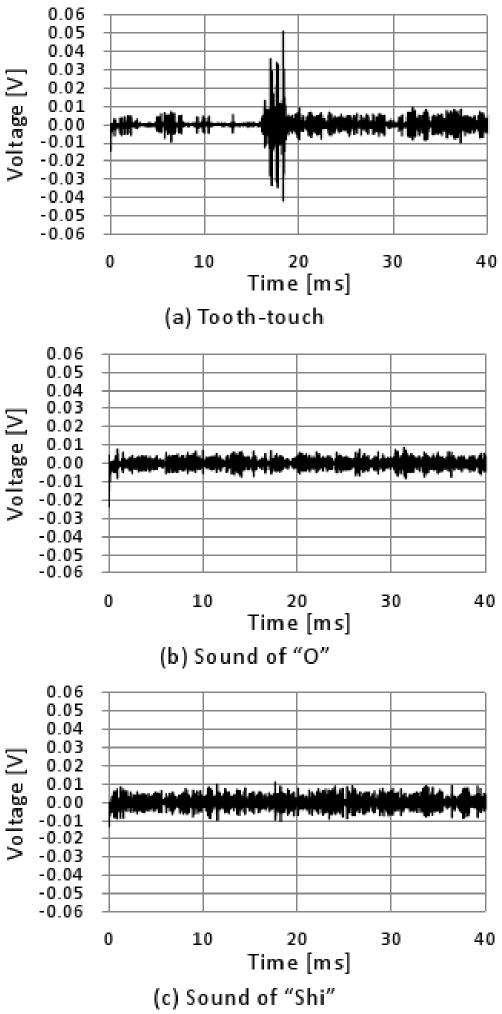


Fig. 4. Filtered Waveform by FFT–IFFT Pair.

the tooth-touch that the voice sounds never include.

Through the analysis mentioned above, the highpass filter with the cutoff frequency of 15 KHz is built by using FFT–IFFT pair [5]. This filter pads 0 from DC to 15 KHz on the waveform of the frequency domain generated by the FFT. Then, the filtered wavform of the time domain is regenerated by IFFT to this padded waveform of the frequency domain. This result is shown in Fig. 4. The waveform of the tooth-touch shows obviously larger peaks than those of the voice sounds. Thus, by the simple method setting the appropriate voltage threshold, the tooth-touch can be detected except for the voice sounds.

## 4 HARDWARE DESIGN

Fig. 5 is an FFT filter hardware we have designed. It has 6 pipelined stages. This hardware employs the decimation-in-frequency (DIF) radix-2 FFT. The width of the real number and the imaginary number is 16 respectively. That is, the value is the fixed-point number with 16bit width. The decimal point is set to the 1bit lower point from MSB. Thus, the input data must be normalized to [-1,1).

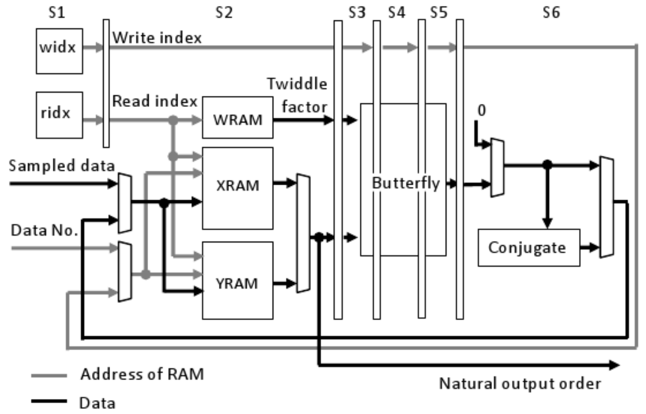


Fig. 5. Pipelined FFT Filter Hardware.

The sampled data comes from an AD converter to the data RAM (initially XRAM) according to the sampling rate used. Once the data RAM is filled with the sampled data, this hardware starts the pipelined execution. The FFT is constructed by log2(number of data points) stages. Our hardware executes the pipelined execution at each stage in the FFT. Thus, the continuous stages in the FFT are sequentially executed.

In each FFT stage, this hardware outputs the intermediate results calculated at the current stage to the temporary RAM (initially YRAM). Until the forward FFT finishes, this hardware uses the data RAM and temporary RAM, flipping their role per stage. In the final stage of the forward FFT, this hardware conjugates the complex number and stores the temporary RAM in order to execute the inverse FFT (IFFT) next. At the same time, the result with the index less than the index corresponding to the cutoff frequency is set to 0 in order to play the role of the highpass filter.

Once the forward FFT finishes, the IFFT starts immediately using the processed data in the internal RAM. Since all input data of IFFT has already been conjugated, the IFFT has only to perform the same execution as the FFT. After the IFFT, the filtered data in the data RAM are outputted sequentially to a DA converter according to the sampling rate.

This hardware consumes the $(6 + N)$ clocks in each FFT stage. The 6 clocks are for filling the pipeline with the data. The $N$ is the number of data points. Thus, the forward and inverse FFT takes the $(2 \times log2(N) \times (6 + N))$ clocks for the completion of the FFT filter.

## 5 HARDWARE EVALUATION

### 5.1 Size and Clock Speed

To evaluate the amount of hardware and the achievable clock speed, we have implemented our hardware and the conventional FFT hardware to FPGA. The used tool is ISE13.2 and the used FPGA is spartan 6. The conventional hardware is generated by the Xilinx IP generation tool (Core-

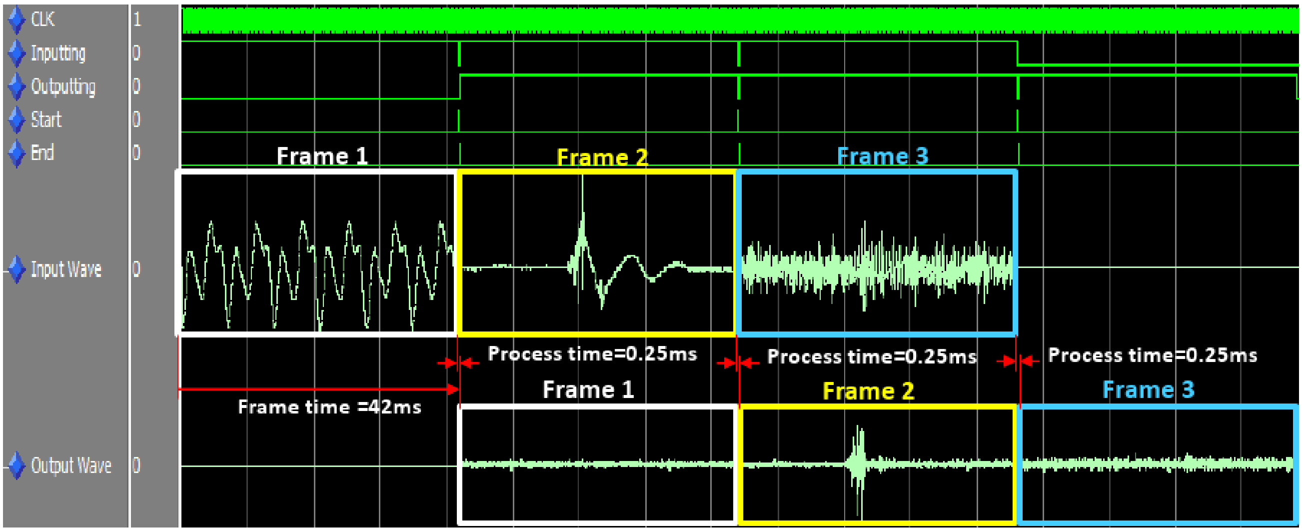


Fig. 8. Execution Snapshot.

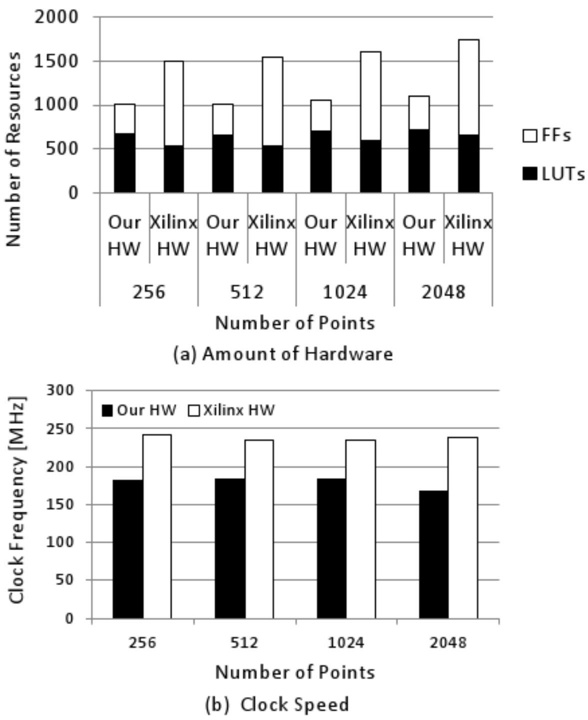


Fig. 6. Result of Hardware Evaluation.

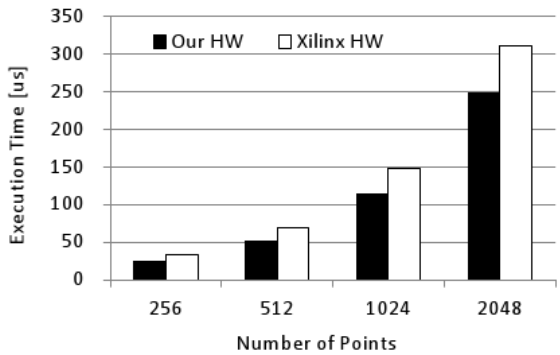


Fig. 7. Performance Evaluation.

Gen) [10]. The Xilinx FFT can switch the forward and reverse operations by the input signal. The parameters of the conventional FFT hardware generated by the CoreGen have been chosen where the hardware architecture of each other becomes equivalent.

Fig. 6 shows the result of implementation. Our hardware is smaller than the Xilinx FFT. However, the achievable clock speed of our hardware is less than the Xilinx FFT. This is because the Xilinx FFT may have deeper pipeline of the butterfly calculation than our hardware. That is, the Xilinx FFT consumes more flip-flops (FFs) than us to improve the clock speed. The hardware generated by the CoreGen is not the hardware description language (HDL) program but the netlist highly optimized to the target device. Thus, the Xilinx FFT cannot be ported to other FPGAs. In contrast, our hardware can be ported without any modification since it is described only by the generic features of HDL. In addition, the our hardware has room to optimize the pipeline structure to improve clock speed more.

### 5.2 Performance

Fig. 7 shows the result of the performance evaluation. We have used the ModelSim SE 10.0b as a logic simulator. The clock period is set to 10 ns; the clock frequency is 100MHz. Our hardware can improve the performance compared with the Xilixn FFT. In the Xilinx FFT, all data processed by the FFT have to be loaded from the internal output memory before IFFT. Then, the loaded data is re-inputted into the internal input memory for the Xilinx FFT setting the functionality to reverse operation. In contrast, our hardware immediately executes the IFFT using the processed data in the internal memory once the FFT finishes. The execution time that is less than 250us may be neglectable to the frame time of 42 ms as mentioned in Section 3.

Fig. 8 depicts the snapshot that our hardware actually operates in the environment that is same as Section 3. This result indicates that the execution time of our hardware is actually neglectable to the frame time. That is, our hardware has enough performance such as not to affect the realtime operation.

## 6 CONCLUSION

This paper has proposed a lightweight sensing method extracting the tooth-touch that does not need a sophisticated signal processing to eliminate the normal audio sound. This method uses an shock wave (i.e. ultrasonic wave) which is generated when the upper tooth and the lower tooth hit each other, instead of the sound wave of tooth-touch. By our proposed method, the the signal processing extracting the tooth-touch has only to perform a well-known filter eliminating lower frequency domain than the ultrasonic domain.

Through the preliminary experiment by using the prototype hardware with a conventional microphone, we have confirmed that the tooth-touch has the larger power than that of the voice sounds, in the ultrasonic region from 15 KHz to 24 KHz. Based on the above analysis, we have designed the filtering hardware by the conventional FFT and IFFT pair. Through the implementation to the FPGA, and the simulation near the actual environment, it has been confirmed that our hardware is small and has the enough performance for a realtime operation.

As future work, we will develop the total system as shown in Fig. 1. Then, we will perform the operation test by some testees.

## REFERENCES

[1] Barnes MP (1994), Switching devices and independence of disabled people, British Medical Journal 309:1181-1182.

[2] Crag A, MaIsaac P et.al. (1997), Mind over matter: Brain signals and control of electrical devices, Today's Life Science 9:12-15.

[3] Kirkup L, Searle A, Craig A et. al. (1997), EEG system for rapid on-off switching without prior learning, Medical and Biological Engineering and Computing 35:504-509.

[4] Kuzume K, Morimoto T (2006), Hands-free man-machine interface device using tooth-touch sound for disabled persons, Proceedings of the 2010 ACM Symposium on Applied Computing:1159-1164.

[5] Press WH, Teukolsky SA, Vetterling WT, Flannery BP (2007), Numerical recipes third edition:667-668.

[6] Knowles Acoustics (2003), Product Specification: SP0103 Series with Integrated Amplifier.

[7] Panasonic (2004), Omnidirectional Back Electret Condenser Microphone Cartridge.

[8] Leighton TG (2007), What is ultrasound?, Progress in Biophysics and Molecular Biology 93:3-83.

[9] Pico Technology (2011), http://www.picotech.com.

[10] Xilinx (2011), LogiCORE IP Fast Fourier Transform v8.0 Product Specification DS808.