Symmetric Signal Reconstruction and Frequency-Division Differential Driving for High Rate Touch Screen Sensing

GyeongSeop Choi, M.G.A. Mohamed, and HyungWon Kim

***Abstract*—As touch screens become larger, touch screen sensing techniques are facing growing demands for higher resolution and speed. While a fast touch screen sensing technique, frequency division concurrent sensing (FDCS) can provide substantial speed enhancement, its scan speed is still limited by the frequency characteristics of the touch screen and driving signals. We propose an enhanced FDCS technique to further improve the scan rate up to four times. The proposed technique drives only the first half-period of driving signals, while reconstructing the remaining half of the signals at the new fast Fourier transform (FFT) called a reconstruction FFT. This way, we can double the scan rate of the FDCS's maximum rate. In addition, we introduce a differential driving technique, which halves the number of required frequencies. This consequently allows us to increase the driving signal frequencies by two times. We also introduce a frequency shuffling technique to eliminate the multitouch cancellation problem, a common problem that makes it difficult to use differential signals for multitouch detection. Simulation experiments show that the proposed method can achieve four times higher scan rate at the cost of little overhead and SNR loss.**

***Index Terms*—Differential sine waves, fast Fourier transform (FFT), frequency division concurrent sensing (FDCS), frequency shuffling, mutual-capacitance touch screen.**

I. Introduction

TOUCH screen techniques are widely used for mobile devices and monitor devices with various ways. In particular, capacitive touch screen panels (TSP) have been adopted for not only small size devices, but also large size screens [1], [2], [3], [16], [17]. Capacitive touch screens are grouped into two types; self-capacitance TSPs and mutual capacitance TSPs. Self-capacitance touch screens with a row-column construction tends to have ghost point problem, which may not detect multi-touch correctly. While Self-capacitance touch screens with a multi-pad construction can detect all multi-touches, all pad need to be individually addressed and so these methods are limited only to small screens. On the other hand, mutual capacitance touch screens do not suffer from the ghost touch problem even with row-column constructions. The mutual-capacitance TSPs are, therefore, increasingly adopted by appliances with large touch screens such as PC monitors, electronic white boards, and digital signage screens. The increased demands for faster multi-touch detection and higher accuracy are pushing their performance requirements to the level unachievable with previous multi-touch detection methods [4], [5], [6], [18], [19].

Among the recent techniques developed for large TSPs, the frequency division concurrent sensing (FDCS) technique [3], [5], [9], [15] was reported to achieve substantial performance gain. While most conventional methods use pulse driving signals to detect the variation of mutual capacitance [1], [8], [14], [16], [21], this technique uses concurrent sine waves of orthogonal frequencies. It enhances the frame scan rate and signal to noise ratio (SNR), and is suitable for scan large TSPs. In the FDCS technique, sine waves are applied concurrently to all driving (TX) lines, while each sensing (RX) line is sensed sequentially and processed by a Fast Fourier Transform (FFT).

Most conventional techniques try to improve the frame scan rate by duplicating sensing circuits to sense multiple RX lines in parallel [8], [16]–[21]. Such methods, however, often incur excessive chip areas and increased power consumption. Thus the FDCS technique can significantly increase the scan rate without the excessive hardware overhead like the conventional methods.

FDCS, however, has limitation on further speed enhancement, since the driving signal period is relatively long, while the signal frequency is limited by the touch screen's characteristics [11]. This paper proposes an enhanced FDCS technique to further improve the scan rate up to 4 times higher than the basic FDCS technique. The proposed method consists of two techniques: a reconstruction FFT for symmetric signals and a differential driving technique. The two proposed techniques provide 4 times increase in the scan rate over the basic FDCS. The differential driving technique also resolves a multi-touch cancellation problem by employing a frequency shuffling method.

This paper is organized as follows. In Section II, we describe the frequency division concurrent sensing (FDCS) method and compare its SNR to conventional method. We then introduce the proposed method auto-differential using opposite signals without differential sensing circuit. In Sections III and IV, we present the proposed sensing circuit with new FFT design and how to detect multi touch point in the special cases and solve the problem. Section V presents experimental results and compares the SNR measurements with previous schemes and area. Finally, Section VI draws the Conclusions.

Manuscript received June 17, 2016; accepted August 6, 2016. Date of publication August 24, 2016; date of current version October 11, 2016. This work was supported by the Center for Integrated Smart Sensors funded by the Ministry of Science of Korean Government, ICT & Future Planning as Global Frontier Project (CISS-2016). *(Corresponding author: Prof. H. Kim.)*

The authors are with the Department of Electronics Engineering, Chungbuk National University, Cheongju 28644, South Korea (e-mail: gschoi@cbnu.ac.kr; hwkim@cbnu.ac.kr).

Color versions of one or more of the figures are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JDT.2016.2602244

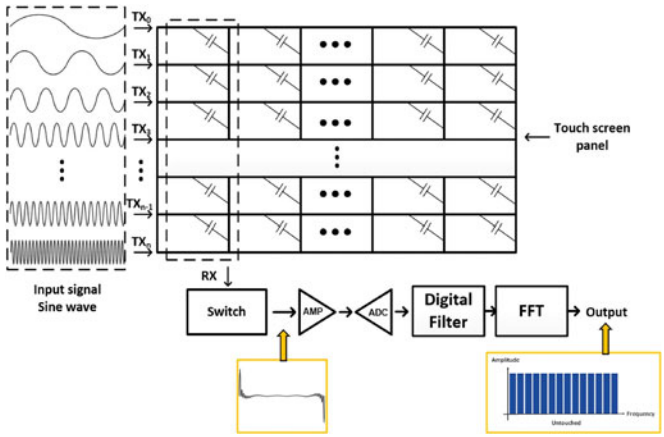

Fig. 1. Overall structure of FDCS.

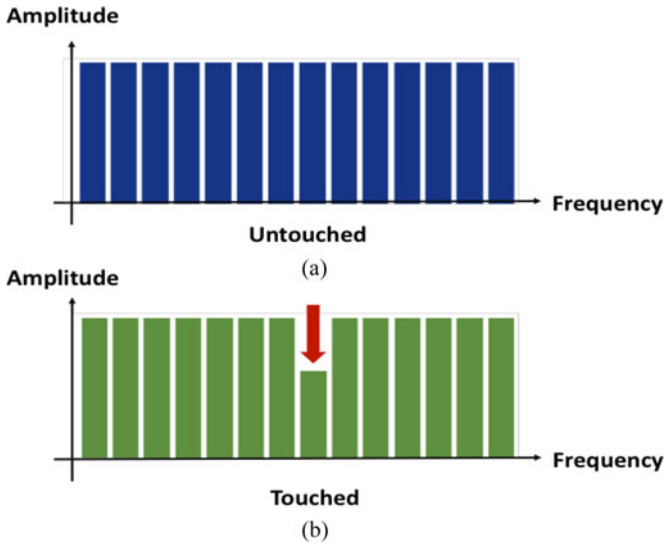

Fig. 2. Touch detection with FFT results: (a) When no touch occurs, (b) when one touch occurs.

II. Frequency Division Concurrent Sensing (FDCS)

The frequency Division Concurrent Sensing (FDCS) technique has been reported as a method to enhance the detection speed of mutual capacitance TSPs. In the FDCS technique, the TSP controller drives multiple sine wave signals with different frequencies to the TX lines of the TSP. The TSP controller then sequentially senses Rx lines to detect the changes in the mutual capacitance. The FDCS's controller employs an FFT to convert the RX output signals to the frequency domain or spectral density. It detects the amplitude changes at the frequencies of all sine waves applied to all TX lines. So all the touch points on each RX line can be discovered concurrently.

Fig. 1 shows an FDCS touch screen detection system. For example, consider a commercial 23-inches TSP that is chosen for the experiments in this paper. Its mutual capacitance sensing points (or cells) are composed by 44 TX lines and 78 RX lines. The FDCS system applies 44 sine waves of orthogonal frequencies to drive all TX lines concurrently, and measures all the 44 TX sine waves simultaneously from each RX line.

Eq. (1) expresses the TX driving signal on the *n*-th TX line for FDCS.

For all *i* with $0 \leq i < N_{\mathrm{TX}}$,

$$\boldsymbol{TX_i} = \boldsymbol{A} \times \mathrm{Sin}\left[2\pi \boldsymbol{f}\left(\boldsymbol{i}\right)\boldsymbol{t}\right] \tag{1}$$

$$\boldsymbol{f}\left(\boldsymbol{i}\right) = \boldsymbol{f_b} + \boldsymbol{i} \times \boldsymbol{f_c} \tag{2}$$

$$\boldsymbol{f_b} = \boldsymbol{k} \times \boldsymbol{f_c},\ \boldsymbol{k} \geq 1 \text{ (k is an integer)}$$

$$\boldsymbol{RX_m} = \sum_{i=0}^{N} \boldsymbol{TX_i} = \boldsymbol{TX_0} + \boldsymbol{TX_1} + \cdots + \boldsymbol{TX_{N-1}} + \boldsymbol{TX_N} \tag{3}$$

In Eq. (1), *A* is the amplitude of TX sine wave f_b is the base frequency and f_c is the interval of frequencies. Eq. (2) restricts the base frequency f_b as an integer multiple of the interval frequency f_c. Eq. (1) and (2) select the frequencies that satisfy the orthogonal frequency rules of Orthogonal Frequency Division Multiplexing (OFDM) [3], [5], [15]. Eq. (3) represents the RX signal appearing on the *m*-th RX line is. N_{TX} is the number of TX lines and N_{RX} is the number of RX lines in the TSP. In other words, in each RX line, an RX signal is formed by the sum of TX sine waves $\boldsymbol{TX_k}$ from all the TX lines. The controller can sense the changes in any mutual capacitance by measuring the changes in the spectral density at the frequencies of the applied sine waves. For instance, as illustrated by Fig. 2, when a cell (a sensing point) is touched, the spectral density decreases. FDCS improves frame scan rate better than conventional touch detection techniques. Conventional methods usually drive pulse signals and sense each RX line. Thus their frame scan rate can be expressed by Eq. (4).

$$\text{Frame scan rate} = \frac{1}{\boldsymbol{T}_{\mathrm{Period}} \times \boldsymbol{N}_{\mathrm{Integration}} \times \boldsymbol{N}_{\mathrm{TX}} \times \boldsymbol{N}_{\mathrm{RX}}} \tag{4}$$

Here T_{period} is the period of RX output signal, and N_{TX} N_{RX} is the number of TX lines, N_{RX} is the number of RX lines. $N_{\mathrm{Integration}}$ is the number of integrations conducted by the integrator circuit. If the touch screen size is increases, the number of the TX lines and RX lines tend to increase accordingly. As a result, the frame scan rate decreases rapidly according to Eq. (4). On the other hand, the frame scan rate of FDCS technique is given by Eq. (5) below.

$$\text{Frame scan rate} = \frac{1}{\boldsymbol{T}_{\mathrm{Period}} \times \boldsymbol{N}_{\mathrm{RX}}} \tag{5}$$

Here the frame scan rate is determined only by the number N_{RX} of RX lines and the period T_{period} of the minimum-frequency of sine waves applied. This follows from the fact that FDCS drives all TX signals concurrently and so it is independent of N_{TX}, while it doesn't need any integrator and so is independent of $N_{\mathrm{Integration}}$. Due to this advantage, FDCS can improve the scan rate by up to 44 times for the 23-inch TSP as reported in [5]. So this technique has efficiency and good performance.

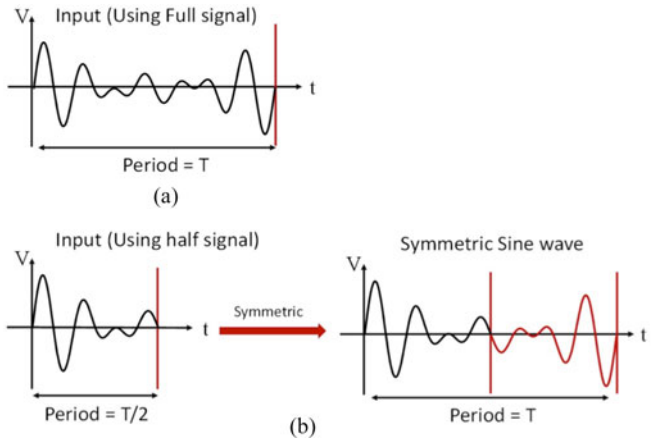

Fig. 3. Reconstruction of symmetric signal: (a) A full one period of symmetric RX signal, (b) a half period of the signal, and the original RX signal recovered by reading the first half-period signal and reversing it to reconstruct the second-half period.

III. Speed Up With Symmetric Signal Reconstruction

A. Input Frequency Range of TSPs

In the FDCS technique, the frequency range of input signal is limited by the frequency response (transfer function) of TSP. Thus the frame scan rate is also limited. Input sine waves of frequencies beyond the TSP's transfer function cannot be selected. Acting as a band pass filter, the TSP would significantly attenuate the higher frequency signals. The first step to increase the scan rate is to raise the frequencies of all sine waves as much as TSP's frequency response accepts. A naive way for increasing the frame scan rate may be using multiple sensing circuits including RX signal amplifiers, analog-to-digital (ADC) converters, and FFTs except canceling sine waves method. This method, however, tends to incur a high area overhead.

In FDCS, the FFT takes an RX signal from each RX line. One RX signal is the sum of all sine waves that are applied to all TX lines, pass through the TSP, and arrive at the corresponding RX line. To produce correct frequency analysis of such an RX signal, an FFT must receive the RX signal including all sine waves with at least their one full period. This restriction limits the maximum achievable scan rate of FDCS.

B. Reconstruction FFT Using Half Period Signals

To overcome the above restriction of the full period, we propose a new FFT architecture, which needs the RX signal with only half period of the sine waves and still guarantees accurate frequency analysis. The proposed technique reconstructs the full period of the signal from only the half period of the signal using the symmetry property of the sinusoids. Fig. 3 shows the symmetry of RX output signal and how we can generate the full period signal from the half-period signal by flipping over the first half. Driving only half-period of RX signals therefore can double the scan rate of FDCS.

The proposed FFT uses a pipelined architecture similar to conventional pipelined FFTs with a single delay path. Unlike the conventional FFTs, however, the proposed FFT employs a dual buffer structure in the 1st stage of the pipeline. The dual buffer stores 2 consecutive half-period signals from 2 RX lines. One buffer stores the signals from even Rx line (ex. Rx0, Rx2, . . .), and the other stores the signals from odd Rx line (ex. Rx1, Rx3, . . .).

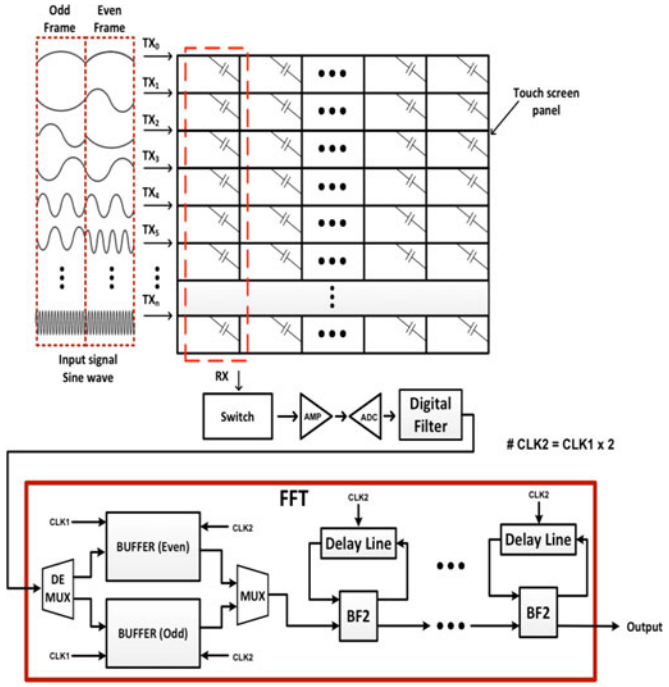

Fig. 4. Structure of the reconstruction FFT and its input signals of only the half period.

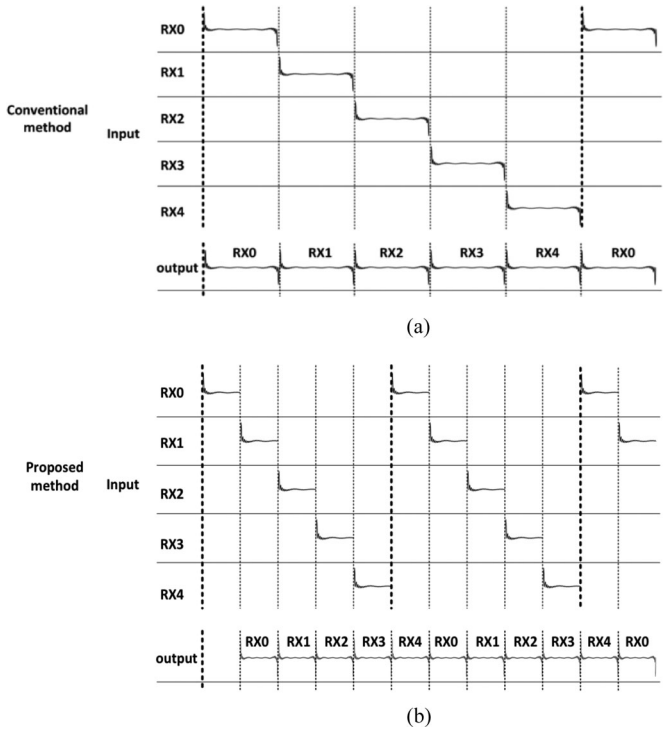

Fig. 5. Timing diagram of RX signals: (a) conventional FDCS, (b) reconstruction FFT.

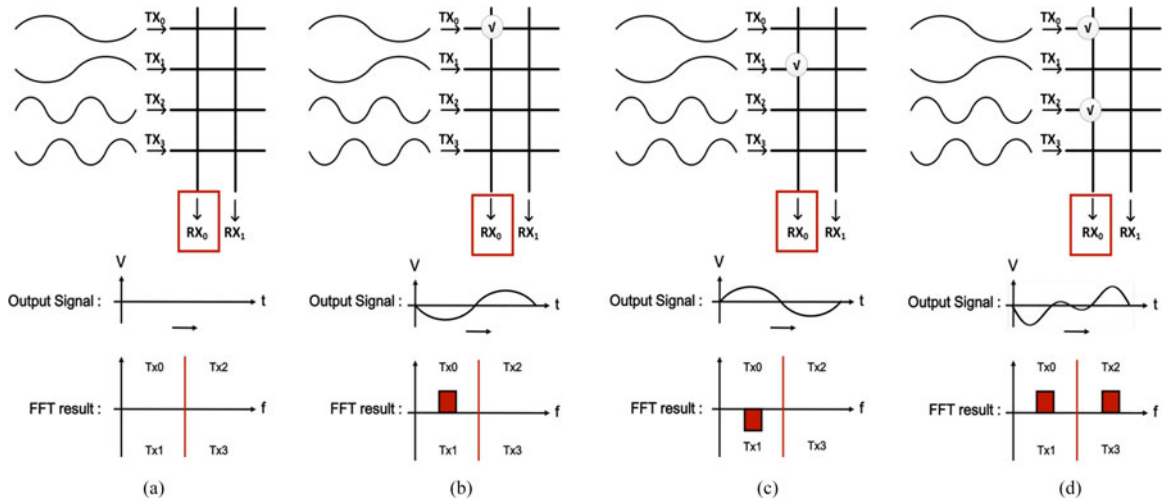

Fig. 6. The detection process FDDD: (a) When no touch occurs, (b) when TX0 is touched, (c) when TX1 is touched, (d) when multi-touches at TX1 and TX2 occur.

The RX signal's unknown half period is reconstructed by reading out the buffer twice: the first read gives the received samples, while the second read with sign inversion produces the unknown samples. Two different pointers are used for read and write operations as shown in Fig. 4. Reading and writing operations of the dual buffer are conducted at two different speeds. The read operations are two times faster than the write operations to complete the reconstruction of one full period of signals, while receiving and storing one half-period signal from the next RX line. Except for the 1st pipeline stage, the remaining stages use the faster speed for reading and writing operations.

This reconstruction operation can recover any half-period signals that are odd symmetric such as sine waves. For even-symmetric signals such as cosine signals, the reconstruction operation does not need the sign inversion process.

While the frame scan rate for the basic FDCS is given by Eq. (4), the scan rate improves to Eq. (6) with the reconstruction FFT.

$$\text{Scan rate} = \frac{1}{(T_{\text{period}} \,/\, 2) \times N_{\text{RX}}} = \frac{2}{T_{\text{period}} \times N_{\text{RX}}} \tag{6}$$

The detailed hardware structure and operations of the reconstruction FFT are described in [15]. Even if the frequencies of the sine waves can no further increase due to the TSP's limitation, the reconstruction FFT can further increase the scan rate by 2 times. While the half period signals can make sharp transitions at the frame boundaries as shown in Fig. 5, their high frequency components experience very small attenuation, and so have negligible impact on the SNR.

IV. Speed Up With Frequency Division Differential Driving

A. *Differential Sine Wave Driving Method*

In this section, we introduce a new approach called Frequency Division Differential Driving (FDDD) method. FDDD further increases the scan rate of FDCS in additional to the reconstruction FFT described above. It reduces the number of required

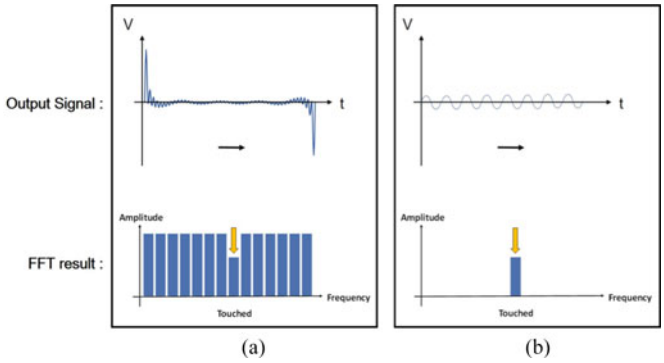

Fig. 7. FFT results: (a) FDCS method (b) FDDD method.

frequencies of the sine waves to a half of the basic FDCS case. Unlike FDCS which generates one sine wave per one frequency, FDDD generates 2 sine waves for each frequency: a positive sine wave and its negative sine wave with 180° phase shift. As a result, FDDD can boost the scan rate by an additional 2 times.

Fig. 6 shows examples of differential sine waves applied to 4 TX lines of a TSP. Fig. 6(a) illustrates a case with no touch. Here the differential sine waves would experience the same amplitude reduction since all the mutual capacitors remain unchanged. Thus the differential sine waves with the opposite polarity cancel each other, and so result in RX signals with zero values. Fig. 6(b) depicts a case with a touch event, where the sine wave on the touched TX line TX0 experiences more reduction in its amplitude than the differential counterpart. The amplitude difference would appear as a sine wave at the RX lines. Similarly the touch even in Fig. 6(c) would result in a sine wave at the RX lines but in the opposite polarity than Fig. 6(b).

In Fig. 6, the output signals illustrate the sensing signals at the RX lines. We can detect the change of mutual capacitance values by using the FFT step of FDCS. The FFT calculates the amplitude and phase of each frequency component. For example, in the FFT result in Fig. 6(b), the amplitude indicates TX0 or TX1 has a touch, while the phase tells that TX0 has a touch. Fig. 6(d) shows a multi-touch case, where the FFT result

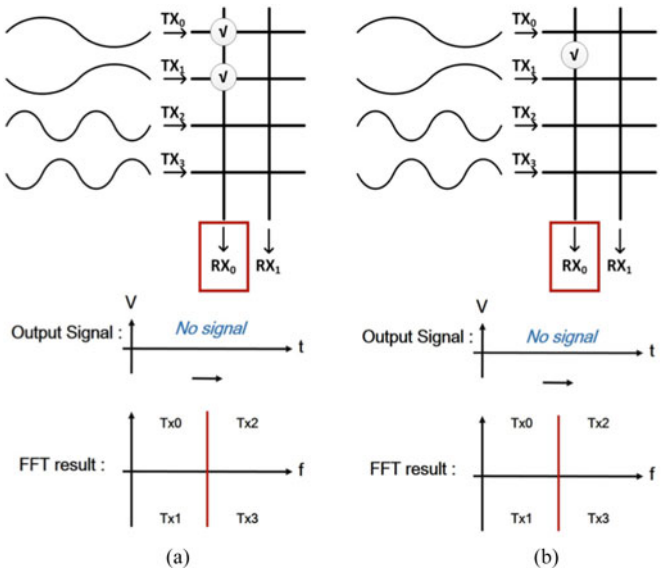

Fig. 8. Touch cancellation problems: (a) Multi-touch on the same frequency lines (Touches on TX0 and TX1), (b) middle-of-lines touch (Touch between TX0 and TX1).

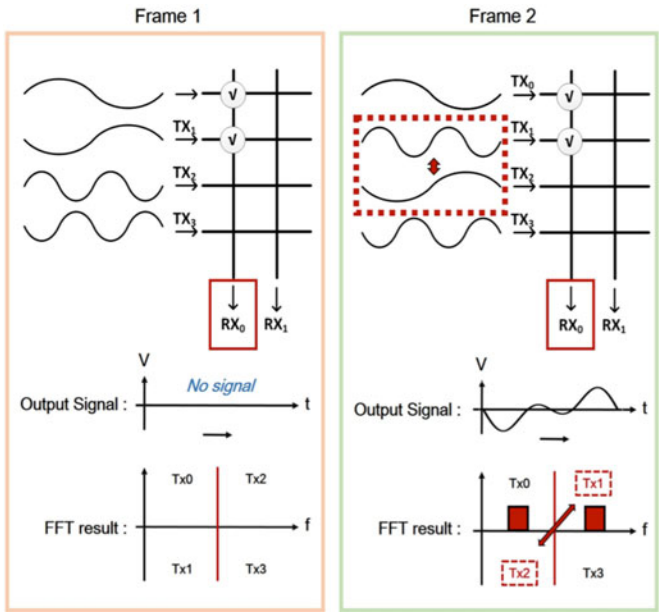

Fig. 9. Shuffling sine waves to avoid multi-touch cancellation problems.

indicates that both TX0 and TX1 are touched. Fig. 7 compares the RX signals of the FDCS (a) and the FDDD (b) in time domain and frequency domain. While FDCS shows a reduced spectral power at the frequency of the touched TX line, FDDD shows high spectral power at the touched TX frequencies and zero spectral power at the untouched TX frequencies. Since FDDD passes only the touch signal to the RX line, we can increase the amplitude of the TX sine waves and also the gain of the sensing circuit. This consequently can lead to a higher SNR than FDCS.

The following formula describe how the TX signals are generated for FDDD:

For all i with $0 \leq i < N_{\mathrm{TX}}$,

$$\boldsymbol{TX_i} = \boldsymbol{A} \times \mathrm{Sin}\ [2\pi \boldsymbol{f}(\boldsymbol{i})\, \boldsymbol{t}]\,, \text{ if i is even} \tag{7}$$

$$\boldsymbol{TX_i} = -\boldsymbol{A} \times \mathrm{Sin}\ [2\pi \boldsymbol{f}(\boldsymbol{i})\, \boldsymbol{t}]\,, \text{ if i is odd} \tag{8}$$

$$\boldsymbol{f}(\boldsymbol{i}) = \boldsymbol{f_b} + \boldsymbol{i} \times \boldsymbol{f_c}/2 \tag{9}$$

$$\boldsymbol{f_b} = \boldsymbol{k} \times \boldsymbol{f_c},\ \ \boldsymbol{k} \geq 1 \text{ (k is an integer)}$$

$$\begin{aligned} \boldsymbol{RX_m} &= \sum_{\boldsymbol{i}=0}^{\boldsymbol{N}} \boldsymbol{TX_i} \\ &= \boldsymbol{TX_0} + \boldsymbol{TX_1} + \cdots + \boldsymbol{TX_{N-1}} \\ &\quad + \boldsymbol{TX_N} \end{aligned} \tag{10}$$

The following example describes how FDDD can double the scan rate. Consider the 23-inch TSP that was used for experiments. It consists of 44 TX lines and 78 RX lines, and has an input frequency range from 1 KHz to 450 KHz. The scan rate is determined by the lowest-frequency sine wave, and so we should maximize the lowest frequency. The basic FDCS generates 44 sine waves with 44 frequencies. For example, it may select the frequencies from 10 kHz to 440 kHz with the interval of 10 kHz. FDDD, on the other hand, use generates 44 sine waves with only 22 frequencies. It, therefore, can select frequencies from 20 kHz to 440 kHz with an interval of 20 kHz. Thus FDDD increased the scan rate 2 times by reducing T_{period} by a half.

B. Frequency Shuffling to Prevent Touch Cancellation Problems

While FDDD can efficiently increase the scan rate, it may experience touch cancellation problems, which commonly appear in most of differential sensing methods. For example, in Fig. 8(a) the two touches on TX0 and TX1 are cancelled, as the differential sine waves would experience the same amplitude reduction, and thus the corresponding output signals become zero at all the RX lines.

Another problem is the middle-of-lines touch cancellation. For example, in Fig. 8(b) if a touch is positioned between TX0 and TX1, the output signals become zero, since the two sine waves at TX0 and TX1 experience the same amplitude reduction, and cancel each other. Therefore, the multi-touch and middle-of-lines touch cancellation problems may significantly degrade the detection accuracy.

As an efficient solution to the touch cancellation problems, we propose a frequency shuffling technique. It generates driving signals in a sequence of two frames. In the 1st frame, it generates differential sine waves based on FDDD as described above. In the 2nd frame, it shuffles the frequencies of the 1st frame's sine waves such that any touch event can be detected by either the 1st or the 2nd frame.

For example, Fig. 9 shows the same example as Fig. 8(a), where Frame1 uses the same frequency assignment as Fig. 8(a). On the other hand, Frame2 shuffles the frequency assignment by swapping the frequencies between TX1 and TX2. Although Frame1 experiences the multi-touch cancellation, Frame2 can identify the two touches since the two touches are now driven by different frequencies. The example of frequency shuffling in

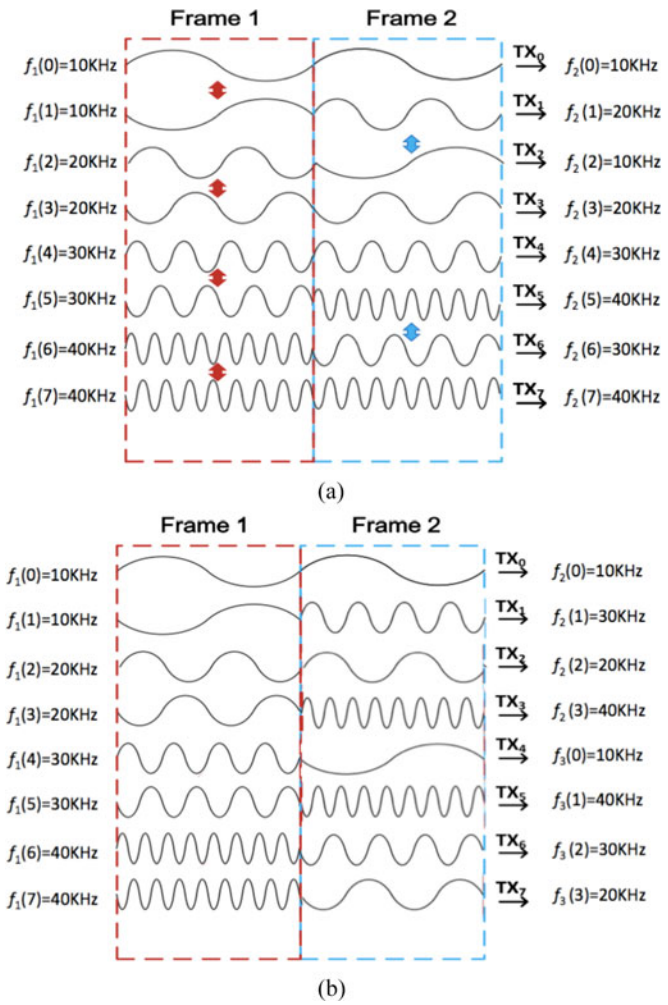

Fig. 10. Sine waves in Frame1 and 2 generated by the frequency shuffling method.

Fig. 10(a), however, may still lead to touch cancellation, if 4 or more consecutive TX lines are touched simultaneously.

We introduce a better frequency shuffling algorithm, which shuffles positive sine waves and negative sine waves in separate groups. Fig. 10(b) illustrates an example of such frequency shuffling. It shuffles all positive sine waves in the group of the 1st half TX lines, while shuffling all negative sine waves in the group of the 2nd half TX lines. The advantage of this shuffling algorithm is that it avoids touch cancellation for any multi-touch cases covering any consecutive TX lines; See Fig. 10(b). This frequency shuffling algorithm is expressed by the following formula. Eq. (11)–(13) describe the sine waves for Frame1, which give the same differential sine waves as in Eq. (7)–(9). Eq. (9)–(13) represent the shuffled sine waves for Frame2.

For all i with $0 \leq i < N_{\mathrm{TX}}$,

$$\boldsymbol{TX}\,{}^{1}_{i} = \boldsymbol{A} \times \mathrm{Sin}\,[2\pi \boldsymbol{f}_1(\boldsymbol{i})\,\boldsymbol{t}], \quad \text{if i is even} \tag{11}$$

$$\boldsymbol{TX}\,{}^{1}_{i} = -\boldsymbol{A} \times \mathrm{Sin}\,[2\pi \boldsymbol{f}_1(\boldsymbol{i})\,\boldsymbol{t}], \quad \text{if i is odd} \tag{12}$$

$$\boldsymbol{f}_1(\boldsymbol{i}) = \boldsymbol{f_b} + \boldsymbol{i} \times \boldsymbol{f_c}/2 \tag{13}$$

$$\boldsymbol{f_b} = \boldsymbol{k} \times \boldsymbol{f_c}, \quad \boldsymbol{k} \geq 1 \text{ (k is an integer)}$$

For all j with $0 \leq j < N_{\mathrm{TX}}$,

$$\boldsymbol{TX}\,{}^{2}_{j} = \boldsymbol{A} \times \mathrm{Sin}\,[2\pi \boldsymbol{f}_2(\boldsymbol{j})\,\boldsymbol{t}], \quad \boldsymbol{if\ j} < \frac{\boldsymbol{N}_{\mathrm{TX}}}{2} \tag{14}$$

$$\boldsymbol{f}_2(\boldsymbol{j}) = \boldsymbol{f_b} + ((2\boldsymbol{j}+1) \bmod \boldsymbol{N}_{\mathrm{TX}}) \times \boldsymbol{f_c}/2 \tag{15}$$

$$\boldsymbol{TX}\,{}^{2}_{j} = -\boldsymbol{A} \times \mathrm{Sin}\,[2\pi \boldsymbol{f}_3(\boldsymbol{j})\,\boldsymbol{t}], \quad \boldsymbol{if\ j} \geq \frac{\boldsymbol{N}_{\mathrm{TX}}}{2} \tag{16}$$

$$\boldsymbol{f}_3(\boldsymbol{j}) = \boldsymbol{f_b} + ((3\boldsymbol{j}+1) \bmod \boldsymbol{N}_{\mathrm{TX}}) \times \boldsymbol{f_c}/2 \tag{17}$$

Here, N_{TX} is the number of TX lines. The integer numbers, i and j, represent the TX line indices of Frame2 and Frame1, respectively. $TX\,{}^{1}_{i}$ is the sine wave for the i-th TX line in Frame1, while $TX\,{}^{2}_{j}$ is the sine wave for the j-th TX line in Frame2. $f_1(i)$ defines the frequencies assigned to the sine waves for Frame1, while $f_2(j)$ and $f_3(j)$ define the frequencies for Frame2. f_b is the starting frequency and f_c is the frequency spacing as described by Eq. (2) in the previous section.

It can be observed that the sine waves in Frame1 and Frame2 generated by Eq. (11)–(17) are guaranteed to distinguish any single or multi-touch events. Therefore, we can eliminate the touch cancellation problems. Fig. 10 shows an example sine waves in Frame1 and Frame2 generated by the frequency shuffling algorithm of Eq. (11)–(17). There are other shuffling algorithms that can avoid touch cancellation problems such as a pseudo-random sequence for example. We chose the proposed shuffling algorithm, however, since it is simple to implement either in hardware or software.

Fig. 11 shows three examples of multi-touch signal detection. For example, in Fig. 11(a), if TX0 is touched, the corresponding output becomes non-zero, and so the touch can be detected either by Frame1 or Frame2. In Fig. 11(b), on the other hand, the two touches on TX0 and TX1 are cancelled in Frame1. The touch can, however, be detected by Frame2, where the frequencies are shuffled by the proposed algorithm of Eq. (11)–(17). Fig. 11(c) illustrates that the proposed frequency shuffling can detect a touch of middle-of-lines using Frame2.

The proposed frequency shuffling method may seem to increase the latency of touch detection due to the need for the 2nd Frame. In most of touch events, however, checking only one of the two Frames is sufficient to identify all touches. Only in rare events, the detection process needs the FFT results from both Frames. From the calculation of all possible touch combinations, the probability of such rare touch events occurring is as low as 0.09%. Fig. 12 shows the probability of touch cancellation occurrence in various touch cases. While the proposed method checks both Frame1 and Frame2 in an alternating fashion, the majority of touch events (>99%) are correctly identified in the first Frame. In other words, if a touch occurred at frame1, the shuffling algorithm will detect the touch at frame1. If a touch occurred at frame2, it will detect the touch at frame2. Therefore, the scan rate would not decrease as far as the processor detects the touch within the frame where the touch event occurred.

In the case of touch cancellation, however, multi-touches are not detected in frame1 but detected in frame2, or vice versa. The shuffling algorithm is guaranteed to detect such

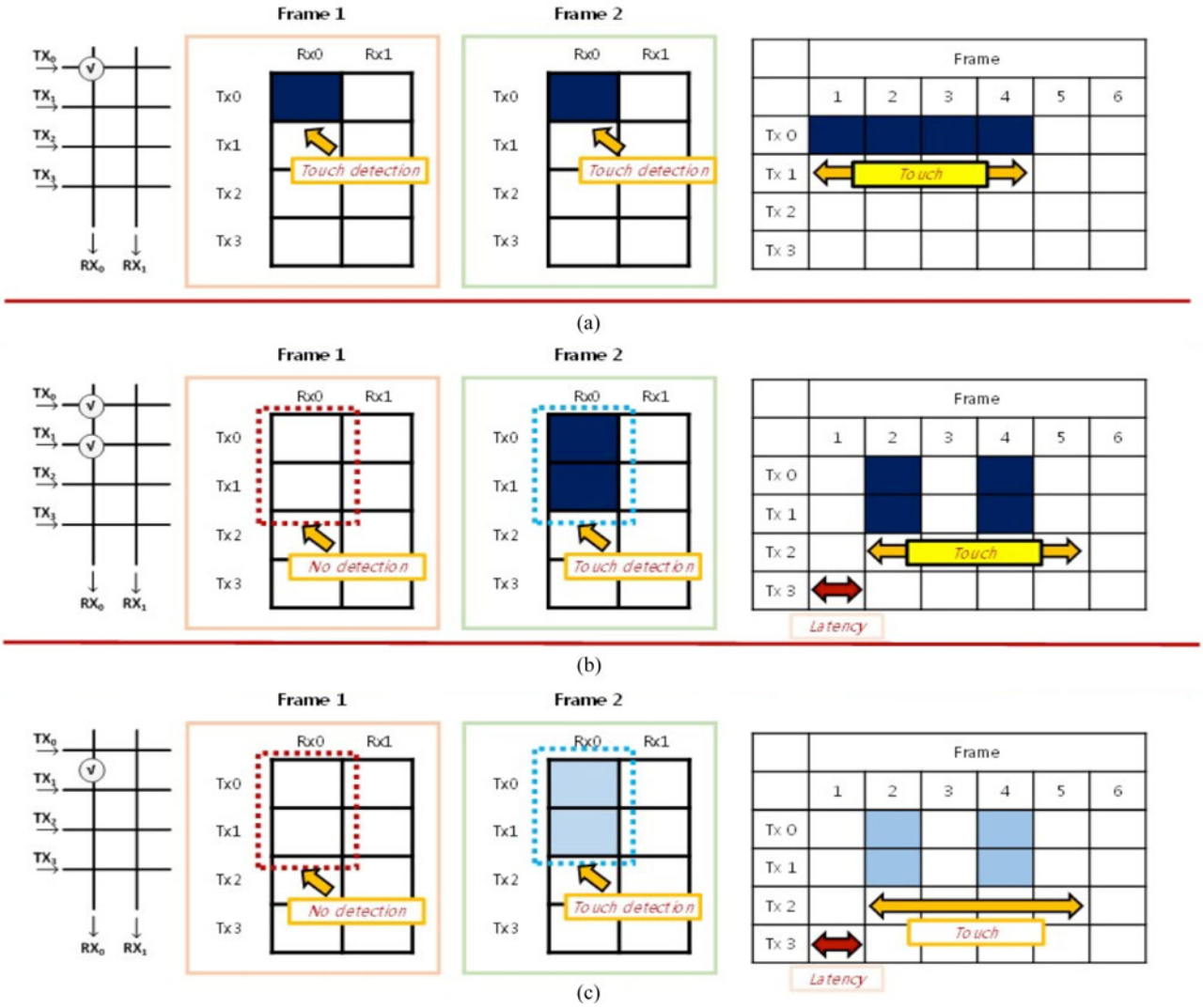

Fig. 11. Process of Multi-touch signal recovery (a) single touch (b) multi-touch (c) middle-of-lines touch.

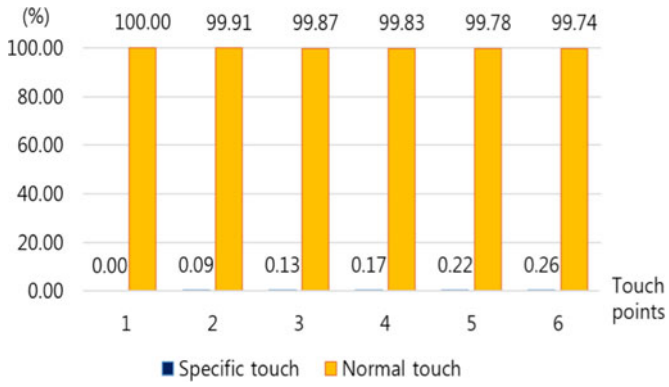

Fig. 12. Probability of touch cancellation occurrence that requires the 2nd frame with frequency shuffling.

multi-touches as far as the finger remains on the same position for the 2 frames, while the conventional method fails to detect. For these rare cases, the latency of detection increases to 2 frames, while the scan rate remains unchanged. The frequency-shuffling algorithm, therefore, can increase the scan rate 2 times in most of touch cases.

V. Experimental Results

To evaluate the performance of the proposed method, we designed a touch screen controller chip using CMOS 65 nm process. The chip includes the design of FDCS, Reconstruction FFT, and FDDD with frequency shuffling techniques. In this chip, we implemented the FDCS technique using both a conventional FFT and the reconstruction FFT. Since the fabrication of the chip takes a long time, we evaluated the performance of the proposed sensing techniques using simulations with an accurate TSP model. We also used a test platform consisting of an FPGA, analog front-end board, and a commercial 23-inch touch screen. Fig. 13 shows this test platform.

The higher the frequencies for the sine waves are selected, the higher the scan rate would be obtained. The maximum frequency of TX sine waves, however, is limited by the frequency characteristic of the TSP. Sine waves beyond this limit degrade

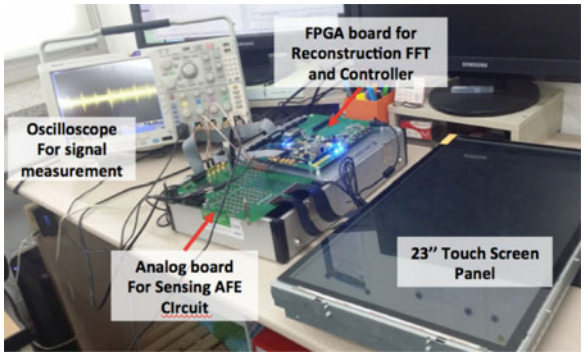

Fig. 13. TSP Sensing Test Platform with FPGA and AFE Board implementing the proposed technique.

TABLE I
COMPARISON OF PERFORMANCE AND COST FOR FOUR SENSING CIRCUITS

Method	*Frame scan rate* (Hz)	SNR gain (dB)	Area of sensing circuit (μm^2)	Number of logic gates
FDCS with Conventional FFT	128.20	52.30	127046.1	66170
FDCS with Reconstruction FFT	256.40	50.12	130159.1	68260
FDDD with Conventional FFT	248.40	52.84	127046.1	66170
FDDD with Reconstruction FFT	512.80	51.51	130159.1	68260

the SNR. To obtain the highest scan rate, therefore, we first select the highest frequencies within the allowed limit for the given TSP. We then further increase the scan rate twice by using the reconstruction FFT, as demonstrated by the following experiments.

We tested the FDCS and FDDD sensing circuits using a realistic simulation model for a 23-inch commercial TSP with 44 TX lines and 78 RX lines. The band-pass frequency characteristic of this TSP allows the maximum TX signal frequency to be 440 KHz. For FDCS, we need to generate 44 orthogonal frequencies for the 44 TX lines. We, therefore, applied 44 sine waves of the frequency range from 10 KHz to 440 KHz with an orthogonal frequency space of 10 KHz. In the case of the FDCS with the conventional FFT, these sine waves can only provide a scan rate of 128.2 Hz (10 KHz / 78 RX lines = 128.2 frames per second). See the 2nd column of Table I for the comparison of the maximum scan rate.

In the case of the FDCS with the reconstruction FFT, on the other hand, we need only a half the period of the above sine waves. This doubles the scan rate to 248.4 Hz (2 × 10 KHz / 78 RX lines = 256.4 frames per second) as shown in Table I. In the case of the FDDD with conventional FFT, we need only half the number of frequencies, which are 22 frequencies of a range from 20 KHz to 440 KHz with an orthogonal frequency space of 20 KHz.

Fig. 14 illustrates sensing signals measured at one RX line for the FDDD with conventional FFT and frequency shuffling. It shows 3 different touch cases under ambient noise, which are detected by Frame1 or Frame2. Fig. 14(a) shows a sensing signal for FDDD without frequency shuffling, where only TX_0 is touched. Fig. 14(b) and (c) show sensing signals for FDDD with frequency shuffling. Fig. 14(b) is the case when only TX_1 is touched. Since only one TX line is touched, it can be detected by Frame1 or Frame2. We can observe a large sensing signal in both Frame1 and Frame2. Fig. 14(c) is the case when 2 TX lines, TX_0 and TX_1, are touched. In Frame1, the sensing signal disappears in the RX line, since the sine waves for TX_0 and TX_1 cancel each other due to their same frequency but opposite polarity. In Frame2, however, the sensing signals with shuffled frequencies now appear in the RX line, and so can be detected via FFT. Hence the FDDD with frequency shuffling can be a viable solution to the touch cancellation problems.

We then evaluate the FDDD with the reconstruction FFT, which needs only a half the number of orthogonal frequencies in additional to the advantage of reducing the period of TX signals to a half. We tested it by generating differential sine waves of a frequency range from 20 KHz to 440 KHz with an orthogonal frequency space of 20 KHz. This provides a scan rate of 512.80 Hz (2 × 20 KHz / 78 RX lines = 512.8 frames per second), which is 4 times higher than the FDCS with a conventional FFT. This test result is summarized in the last row of Table I.

In order to evaluate the sensing performance of the above thee sensing techniques, we calculated the signal to noise ratio (SNR) using the following method.

$$\text{TouchStrength} = \text{Signal}_{\text{Avg}}^{\text{Touched}} - \text{Signal}_{\text{Avg}}^{\text{Untouched}} \tag{18}$$

$$\text{Noise}_{\text{RMS}}^{\text{Touched}} = \sqrt{\frac{\sum_{N_{\text{Samples}}} \left(\text{Signal}\,[n] - \text{Signal}_{\text{Avg}}^{\text{Touched}}\right)^2}{N_{\text{Samples}}}} \tag{19}$$

$$\text{SNR}\,(dB) = 20\text{Log}\frac{\text{TouchStrength}}{\text{Noise}_{\text{Avg}}^{\text{Touched}}} \tag{20}$$

$$\text{SNR Gain}\,(dB) = (\text{SNR of Sensing Circuit Output}) - (\text{SNR of TSP RX Output}) \tag{21}$$

Eq. (18) defines Touch Strength, which is the difference of average signal (voltage) between the touched and untouched cases. Eq. (19)–(21) give the Noise RMS (root mean square) value, SNR, and SNR gain respectively. This SNR calculation method is widely used by the touch screen industry and research community [10]. In order to make fair comparison among various sensing techniques under different conditions such as different TSP, driving signals, and noise signals, we adopt a notion of SNR gain instead of SNR. The SNR gain defined by Eq. (21) is a relative SNR improvement obtained by the sensing technique over the reference SNR of TSP's RX output. Thus the SNR gain gives a constant value independent of the driving signals, noise

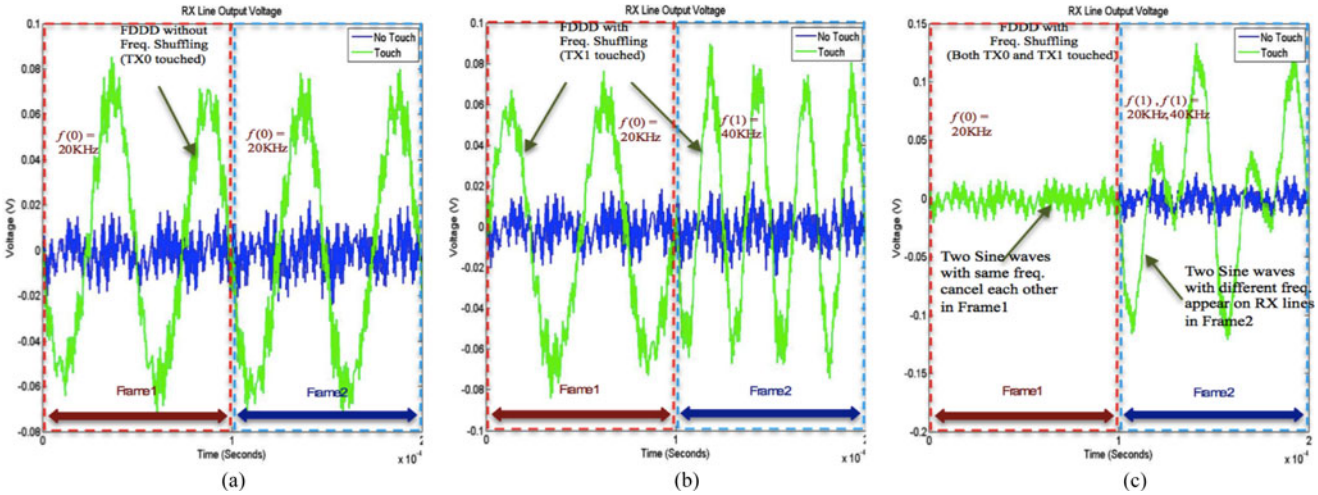

Fig. 14. Sensing signals measured at RX lines with various touch cases under ambient noise: (a) FDCS with TX0 touched; (b) FDDD with TX1 touched; Symmetric Signal Reconstruction and Frequency-Division Differential Driving (c) FDDD with both TX0 and TX1 touched.

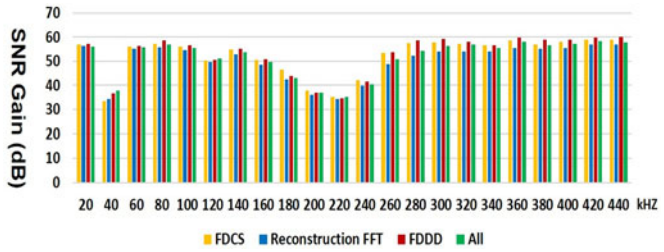

Fig. 15. SNR gain at the all TX frequencies for the 23-inch TSP for the 4 sensing techniques: conventional FDCS, FDCS with reconstruction FFT, FDDD with frequency shuffling, and FDDD with reconstruction FFT.

signals, and the TSP's properties, while the SNR varies along with such factors.

To evaluate the SNR gain under realistic environment, we applied real noise signals measured from the commercial 23-inch TSP. With realistic simulation experiments for the above four sensing circuits, we calculated the average SNR gain values, which are reported in the 3rd column of Table I. The FDCS with conventional FFT obtained an SNR gain of 52.30 dB, while the FDCS with reconstruction FFT produced an SNR gain of 50.11 dB. Here the reconstruction FFT incurs a small SNR loss of about 2∼3 dB compared with the conventional FFT. This is an expected behavior, since it uses only half the period of RX signals leading to a reduced noise cancellation effect. We obtained an SNR gain of 51.51dB for the FDDD with reconstruction FFT. The improvement of 1.4dB over the FDCS with reconstruction FFT can be explained by the following analysis. FDDD cancels all common mode TX signals except only for the touched TX lines leading to smaller amplitude in the RX signals. Hence it allows us to increase the gain of the sensing circuit, and so results in a higher SNR. Fig. 15 compares SNR gain results of the 4 different sensing techniques. It illustrates SNR gain values at the 22 TX frequencies chosen from 20 KHz to 440 KHz.

Although the reconstruction FFT is operated with a clock 2 times faster than the conventional FFT, it finishes each FFT calculation in a half time due to the half period signals. Thus its higher clock frequency does not incur overhead on the computation time nor the power consumption. As reported in Table I, the area overhead of the reconstruction FFT is about 2.4% calculated by (130,159 – 127,046)/127,046 = 2.4%. In general, the power consumption of CMOS circuits is proportional to the chip area. The reconstruction FFT can, thus, incur about 2∼3% power consumption overhead compared with the conventional FFT. On the other hand, the FDDD with frequency shuffling doubles the frame scan rate again at no additional hardware overhead, since its frequency shuffling operation is done in software. In summary, the proposed FDDD technique with reconstruction FFT can provide 4 times enhancement in the frame scan rate at a negligible hardware overhead. On the other hand, the conventional methods would need 4 duplicate RX sensing circuits incurring large overhead.

VI. Conclusion

This paper presents a cost-efficient touch screen sensing technique aimed to enhance the scan rate and resolve the touch cancellation problem. We employed a symmetric signal reconstruction FFT, which reduces the period of TX signals to a half, and so doubles the scan rate of the FDCS method. In addition, we proposed a frequency division differential driving (FDDD) method, which halves the number of frequencies required, and thus doubling the scan rate again. The FDDD with reconstruction FFT, therefore, can provide a scan rate enhancement of 4 times. Differential driving methods, however, suffer from multi-touch cancellation or middle-of-lines touch cancellation problem. As a solution to this problem, we presented the FDDD with frequency shuffling algorithm. We have implemented the proposed sensing circuits in a touch screen controller chip using a 65 nm CMOS process. Simulation experiments with real noise signals showed that the proposed technique provides a scan rate enhancement of 4 times at the cost of as little as 2.4% increase in the FFT size and only 1.4dB loss in the SNR. The proposed sensing technique,

therefore, is well suited to a large capacitive touch screens that need increasingly higher speed and resolution.

ACKNOWLEDGMENT

The authors would like to thank the valuable contributions of MSIS Lab members in Chungbuk National University, especially the help from M.G.A. Mohamed.

REFERENCE

[1] M. G. A. Mohamed, H. W. Kim, and T. W. Cho, "Efficient multi-touch detection algorithm for large touch screen panels," *IEIE Trans. Smart Process. Comput.*, vol. 3, no. 4, pp. 246–250, Aug. 2014.

[2] S. Ko, H. Shin, H. Jang, I. Yun, and K. Lee, "A 70 dB SNR capacitive touch screen panel readout IC using capacitorless trans-impedance amplifier and coded orthogonal frequency division multiple sensing scheme," in *IEEE Symp. VLSI Circuits*, Jun. 2013, pp. C216–C217.

[3] I. Seo, T.-W. Cho, H. Won Kim, H.-G. Jang, and S.-W. Lee, "Frequency domain concurrent sensing technique for large touch screen panels," in *Proc. IEEK Fall Conf. 2013*, Nov. 2013, pp. 55–58.

[4] M. G. A. Mohamed *et al.*, "Efficient algorithm for accurate touch detection of large touch screen panels," in *IEEE Int. Symp. Consum. Electron.*, Jun. 2014, pp. 243–244.

[5] U. Y. Jang, H. Kim, and T. W. Cho, "Frequency division concurrent sensing method for high-speed detection of large touch screens," *J. Korea Inst. Inf. Commun. Eng.*, vol. 19, no. 4, pp. 895–902, Apr. 2015.

[6] I. C. Seo, and H. Kim, "Dual sensing with voltage shifting scheme for high sensitivity touch screen detection," *J. Inst. Electron. Inf. Eng.*, vol. 52, pp. 71–79, Apr. 2015.

[7] G. Choi, M. G. A. Mohamed, and H. W. Kim, "Distributed architecture of touch screen controller SoC for large touch screen panels," in *Proc. IEEE Int. SoC Design Conf.*, South Korea, Nov. 2014, pp. 22–23.

[8] S. Ko, H. Shin, H. Jang, I. Yun, and K. Lee, "A 55 dB SNR with 240 Hz frame scan rate mutual capacitor 30×24 touch-screen panel read-out IC using code-division multiple sensing technique," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2013, pp. 388–389.

[9] Jiho Kim, M. G. A Mohamed, and H. W. Kim, "Design of a frequency division concurrent sine wave generator for an efficient touch screen controller SoC," in *IEEE Int. Symp. Consum. Electron.*, Spain, Jun. 2015, pp. 1–2.

[10] ATMEL Corporation, "Touch Sensors Design Guide, 10620D-AT42," Sep. 2004.

[11] M. G. A. Mohamed and H. W. Kim, "Concurrent driving method with fast scan rate for large mutual capacitance touch screens," *J. Sens.*, vol. 2015, Jul. 2015, Art. no. 768293.

[12] G.-S. Choi, M. G. A. Mohamed, and H.-W. Kim, "Distributed architecture of touch screen controller SoC for large touch screen panels," in *IEEE Int. SoC Design Conf.*, Nov. 2014, pp. 22–23.

[13] I. Seo, U. Jang, M. Mohamed, and H. W. Kim, "Voltage shifting double integration circuit for high sensing resolution of large capacitive touch screen panels," in *18th IEEE Int. Symp. Consum. Electron.*, 2014, pp. 1–2.

[14] A. Ragheb and H. W. Kim, "Differentiator based sensing circuit for efficient noise suppression of projected mutual-capacitance touch screens," in *IEEE Int. Conf. Electron., Inf., Commun.*, 2016, No. 1570221701.

[15] G.-S. Choi and H. W. Kim, "New FFT design with enhanced scan rate for frequency division concurrent sensing of mutual-capacitance touch screens," in *IEEE Int. Conf. Electron., Inf., Commun.*, 2016, No. 1570222094.

[16] K. Lim, K.-S. Jung, C.-S. Jang, J.-S. Baek, and I.-B. Kang, "A fast and energy efficient single-chip touch controller for tablet touch applications," *J. Display Technol.*, vol. 9, no. 7, pp. 520–526, Jul. 2013.

[17] S. Gao, J. Lai, C. Micou, and A. Nathan, "Reduction of Common mode noise and global multi-valued offset in touch screen systems by correlated double sampling," *J. Display Technol.*, vol. 12, no. 6, pp. 639–645, Jan. 2016.

[18] S.-M. Kim, H. Cho, M. Nam, S.-G. Choi, and K. Cho, "Low-power touch-sensing circuit with reduced scanning algorithm for touch screen panels on mobile devices," *J. Display Technol.*, vol. 11, no. 1, pp. 36–43, Jan. 2015.

[19] S.-Y. Liu, W.-H. Li, Y.-J. Wang, J.-G. Lu, and H.-P. D. Shieh, "One glass single ITO layer solution for large size projected-capacitive touch panels," *J. Display Technol.*, vol. 11, no. 9, pp. 725–729, Sep. 2015.

[20] B. Li *et al.*, "A touch prediction and window sensing strategy for low power and low cost capacitive multi-touch screen systems," *J. Display Technol.*, vol. 12, no. 6, pp. 646–657, Jan. 2016.

[21] Y.-H. Tai, H.-L. Chiu, and L.-S. Chou, "Large-area capacitive active touch panel using the method of pulse overlapping detection," *J. Display Technol.*, vol. 9, no. 3, pp. 170–175, Mar. 2013.

Gyeongseop Choi received the B.S. degree in electronics engineering from ChungNam National University, Daejeon, South Korea, in 2012, and the M.S. degree in electrical engineering from Chungbuk National University, Cheongju, South Korea, in 2016.

He was a Member of Mixed Signal Integrated System Laboratory, Chungbuk National University. He is currently with TLI Corporation, a Korea-based display chip maker as a Digital Circuit Design Engineer. His research areas include low-power SoCs, embedded software for systems, digital SoC design, and touch screen controller SoCs.

Mohamed G. A. Mohamed received the M.Sc. and B.Sc. degrees in electronics and communication engineering from Minia University, Minya, Egypt, in 2006 and 2009 and the Ph.D. degree in electronics engineering from Chungbuk National University, South Korea, in 2016.

He is currently working as a Senior Research Engineer at Silicon Works Co., Ltd., South Korea, where he is developing touch display driver ICs for mobile and automobile solutions.He did research in various areas during his work in academia. His current research covers the areas of mixed signal SoC design, data converters, low power circuits, computer human interaction, digital signal processing, and embedded systems.

HyungWon Kim received the B.S. and M.S. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 1991 and 1993, respectively, and the Ph.D. degree in electrical engineering and computer science from the University of Michigan, Ann Arbor, MI, USA, in 1999.

In 1999, he joined Synopsys, Inc., Mountain View, CA, USA, where he developed electronic design automation software. In 2001, he joined Broadcom Corporation, San Jose, CA, where he developed various network chips including a WiFi gateway router chip, a network processor for 3G communication, and 10-Gb Ethernet chips. In 2005, he founded Xronet Corporation, a Korea-based wireless chip maker, whereas as CTO and CEO, he managed the company to successfully develop and commercialize wireless baseband and RF chips and software including WiMAX chips supporting IEEE802.16e and WiFi chips supporting IEEE802.11a/b/g/n. Since 2013, he has been an Assistant Professor in the Department of Electronics Engineering, Chungbuk National University, Cheongju, South Korea. His current research interests includethe areas of wireless sensor networks, wireless vehicular communications, mixed signal SoC designs for low-power sensors, and biomedical sensors.