AN ARBITRARY WAVEFORM SYNTHESIS STRUCTURE WITH HIGH SAMPLING RATE AND LOW SPURIOUS

Wenhao Zhao, Shulin Tian, Guangkun Guo, Jiajing You, Qiong Wu, Ke Liu

University of Electronic Science and Technology of China, School of Automation Engineering, Chengdu 611731, China
(vinhou@163.com, shulin@uestc.edu.cn, gack16@163.com, 18801277096@163.com, sherry31597@icloud.com, liuke@uestc.edu.cn, +86 6183 1321)

Abstract

Arbitrary waveform generator is characterised by its flexible signal generation, high frequency resolution and rapid frequency switching speed and is wildly used in fields like communication, radar system, quantum control, aeronautics and biomedicine. With the continuous development of technology, higher requirements are put forward to arbitrary waveform generator. Sampling rate determines the bandwidth of the output signal, spurious-free dynamic range determines the quality of generated signal. Due to above, these two indicators’ improvement is vital. However, the existing waveform generation methods can’t generate signals with good enough quality due to their technical defect, and in order to realize high system sampling rate, to accomplish waveform generation process in FPGA, multipath parallel structure is needed. Therefore, we proposed a parallel waveform synthesis structure based on digital resampling, which fixed the problems existing in the current methods effectively and achieved high sampling rate, high quality arbitrary waveform synthesis. And build up an experimental test bench to validate this proposed structure.

Keywords: Arbitrary waveform generator, high sampling rate, low spurious, digital resampling, parallel structure.

1. Introduction

Arbitrary waveform generator (AWG) is a type of signal source that can flexibly generate various kinds of waveform signals according to user’s need, it plays an important role in multiple fields like radar test, communication, quantum control, aeronautics, biomedicine and so on. For example, in the fifth generation of wireless communications (5G), Veyrac et al. applied the arbitrary waveform synthesis to the 5G handset transmitter [1]. In a radar test system, it can generate noise waveform for reference, signal detect and analog process [2]. In quantum control, Bowler et al. designed a multi-channel arbitrary waveform generator to process ion-based quantum information [3]. As for aeronautics, it can be applied to chirp-transform spectrometer for planetary system research [4]. In biomedical field, the waveform signals generated by AWG can be used for ion isolation, excitation and ejection in mass spectrometers for biomolecular analysis [5].

However, with the continuous improvement of electronic technology, higher requirements are put forward for AWG such as high bandwidth and high spurious-free dynamic range (SFDR) [6-8]. The only way to increase the bandwidth of AWG is to synthesize waveforms in parallel through multiple DACs and now there are two structures available: band interleaving digital-to-analog converter (BI-DAC) and time interleaving digital-to-analog converter (TI-DAC). The two structures can be used to break through DAC’s performance limitation and increase the bandwidth of output signal [9-10]. In recent years, there have been a large number
of researches on BI-DAC and TI-DAC, and progress has been made in increasing the bandwidth of AWG [9, 11-17].

SFDR is affected by the performance of DAC and the synthesis method. At present, the SFDR of DAC with high speed (1GSPS) can be as high as 80dBc [18]. However, the current digital waveform synthesis method brings redundant spectrum components due to technical defects, which reduces the SFDR of output. In direct digital frequency synthesis (DDFS), due to the fixation of the system sampling clock, in order to gain higher frequency resolution that we need to add the bits of the phase accumulator and this unfortunately brings in phase truncation error. Cordesses proposed some methods to reduce the influence of the phase truncation error on SFDR, including increasing the capacity of waveform memory, the odd-number approach, the phase-dithering approach, and the noise-shaping approach [19-20]. Although these methods expand SFDR to a certain extent, they still have limitations. The first method is limited by the performance of the memory, the odd-number approach can only improve 3 dB of SFDR, the phase-dithering approach and the noise-shaping approach will increase noise floor. In [21], Zhang et al. proposed a design and FPGA implementation of DDS based on waveform compression and Taylor series, the SFDR can reach 114 dB. DDFS has a characteristic that it needs to reading the samples from the memory at equal intervals according to the frequency control word. It makes DDFS impossible to use high-speed synchronous dynamic random access memory (SDRAM) which limits the realization of high sampling rate.

Direct digital waveform synthesis (DDWS) changes the frequency of output waveform by changing the sampling rate of DAC, so the samples are read from the memory point by point. It makes the DDWS be the best choice for high sampling rate arbitrary waveform synthesis. Meanwhile, the variable sampling rate also makes the image components hard to restrain as they move around when the sampling clock changes. Therefore, in order to achieve an arbitrary waveform synthesis method with low spurious, we propose a parallel structure based on digital resampling. We separate the waveform synthesis process into digital waveform synthesis and digital-to-analog conversion. In digital waveform synthesis part, we use variable sampling clock, choose suitable sampling clock according to the output signal’s feature and avoid phase truncation. In the digital-to-analog conversion part, in order to restrain the range of the image components, DAC’s sampling rate will be fixed. To ensure the match of the two parts’ rate, we use digital resampling module to realize sampling clock conversion. As the rate inside FPGA can only be hundreds MHz, we need to process sample points via parallel method to increase the rate, but the sample rate conversion module requires irregular interpolation of the input sample points and isn’t able to be achieved by traditional multipath parallel method. As for that we propose a structure for implementing digital resampling in multiplexed parallelism within FPGA, provide high sampling rate for the system and ensure a low spurious in the mean time. In section II, we introduce the limitations of DDWS. In section III, we introduce our new method based on digital resampling and its implementation in parallel structure. Finally, we give experimental results, proving that the digital waveform synthesis method based on digital resampling can effectively generate waveform with low spurious.

2. Current arbitrary waveform synthesis methods

For complex high-speed arbitrary waveforms, on-site sampled waveforms, and occasional waveforms, DDWS is more suitable. Because this type of waveform usually requires the synthesis system to be able to output waveform samples point by point, and the system’s rate is variable. The structure diagram of DDWS is shown in Fig. 1. It consists of a variable clock generator, an address generator, a waveform lookup table, a DAC, and a low-pass filter. Under the control of the sampling clock, the address in the address generator is incremented one by one. Then the waveform lookup table exports the stored waveform data to the DAC in sequence.
In DDWS, the waveform sample is sequentially in the order of storage, so there is no phase truncation error. The output waveform frequency is determined by the sampling clock frequency and the waveform period length:

$$f_o = \frac{f_s}{N_{\text{length}}}.$$  

where, $N_{\text{length}}$ is the number of samples of a complete waveform.

However, the variable sampling rate of DDWS also leads to a problem that the image components is difficult to suppress. In digital sampling process, the image components exist at $f_s - f_o$ [22], so it needs a low-pass filter with a cut-off frequency of $f_s/2$ to filter out the image components, as shown in Fig. 2(a).

However, as the sampling frequency decreases, the range of the image components will fall within $f_s/2$ when $f_o' - f_o < f_s/2$.

It results in the fact that the image components cannot be completely filtered out, as shown in Fig. 2(c). Also, as the sampling rate increases, the effective components of the output signal will be in the cut-off band when $f_o > f_s/2$, causing the effective components being filtered out, as shown in Fig. 2(d). Therefore, in DDWS, a large number of low-pass filters with different cut-off frequencies are needed to recover the waveform signal from the sampled signal better, which is difficult to achieve in hardware.

![Fig. 2. Spectrum of DDWS if the low-pass filter with a fixed cut-off frequency.](image-url)
3. Waveform synthesis method based on digital resampling

3.1. The waveform synthesis based on digital resampling

According to the discussion in the second section, the variable sampling rate makes the image component difficult to suppress in DDWS. The error of DDWS occurs in the digital-to-analog conversion part. Therefore, fixing the sampling rate of DAC can effectively reduce the spurious of the output waveforms. As shown in Fig. 3, in the digital-to-analog conversion part, we use a variable clock and output waveform samples point by point to avoid phase truncation errors; in the digital-to-analog conversion part, use a fixed sampling clock to limit the image components to a fixed range. In order to match the data rate of the two parts, a digital resampling module is added to realize the conversion of the sampling rate.

Firstly, in the variable clock domain, we get the sampling rate $f_s$ and the samples $x[i]$ according to the required waveform. Then the digital resampling module resamples samples $x[i]$ to get samples $y[j]$. As shown in Fig. 4, the resampled waveform samples $y[j]$ are determined by the original waveform samples $x[i]$ and the time interval $u[j]$. $x[i]$ and $y[j]$ are not in a one-to-one correspondence. For example, $y[j+3]$ and $y[j+4]$ correspond to $x[i]$, but the time interval $u[j+3]$ and $u[j+4]$ are different. We use the $J$ to mark the position of the previous input sample closest to the current output sample $y[jT_{DAC}]$. The resampled waveform samples $y[j]$ and the original waveform samples $x[i]$ have the relationship as below:

$$y[jT_{DAC}] = y[(J + u[j])T_s] = \sum_{i=0}^{\infty} x[(J - i)T_s] \cdot h[(u[j] + i)T_s]$$

(2)

where

$$J = \left\lfloor jT_{DAC} / T_s \right\rfloor, u[j] = jT_{DAC} / T_s - J.$$

Therefore, it is necessary to adjust the system function $h(t)$ in real time according to the change of $u[j]$. The variable delay FIR filter based on the Farrow Structure can meet the needs of real-time variable delay. And the transfer function of the filter can be expressed as:

$$H(z, \mu) = \sum_{n=-N}^{N} h(n, \mu) \cdot z^{-n} = \sum_{n=-N}^{N} \sum_{m=0}^{M} a(n, m) \cdot u^m \cdot z^{-n}.$$  

(3)

where $a(n, m)$ is the coefficient of FIR filter bank, $N$ is the filter’s order, $M$ is the number of FIR filters, $u$ is the time interval. The filter coefficients can be obtained by referring to the
design method based on the second-order cone programming proposed in the [23], which can minimize the peak error of the variable-frequency response and yields a true minimax design. Therefore, the equation (2) can be written as:

$$y[jT_{DAC}] = \sum_{n=-N}^{N-1} \sum_{m=0}^{M-1} a(n,m) \cdot u[j]^m \cdot x[(J-n)T_s]$$

(4)

3.2. Parallel implementation

For design flexibility, digital waveform sample processing is usually implemented in FPGA. However, the maximum work speed of FPGA can only be up to several hundred MHz, which cannot meet the high sampling rate requirement of the waveform synthesis system. Therefore, the processing of waveform sample in FPGA is in parallel structure. In [24], they proposed a parallel resampling structure to realize sampling rate conversion effectively. However, it can only extract samples and cannot insert them, resulting in the fact that it can only realize the conversion from high sampling rate to low sampling rate. In AWG, the sampling rate determines the bandwidth of the output waveform, so the sampling rate should be set to maximum, usually the maximum sampling rate of the DAC. As a result, we need a parallel resampling structure that can realize the conversion from low sampling to high sampling rate.

We proposed a parallel resampling structure as shown in Fig. 5, including waveform samples First Input First Output (FIFO), waveform samples registers, phase controller and k-path FIR filter bank. Among them, the waveform registers are used to provide the sample sequence for the filter; the phase controller provides the position of the sample sequence in the registers, the time interval, and the enable signal to update the samples in FIFO and registers.

According to section 3.2., the output sample $y[j]$ is determined by its corresponding input sample sequence and time interval $u[j]$, the input sample sequence is denoted as $x'[J-n], \ldots, x'[J], \ldots, x'[J+n]$. Assuming that the number of parallel paths is $k$, the FIFO outputs up to $k$ samples per clock. However, $k$ FIR filter banks need at most $2n+k$ input samples in a unit time. Therefore, we use registers to provide input samples for FIR filter banks. As shown in Fig. 5, Reg(0) ~Reg(l-1) are $l$ registers with a storage capacity of $k$ samples.

$$l = \left\lfloor \frac{2n+k}{k} \right\rfloor .$$

(5)
The phase controller judges the current input sample sequence and the time interval through the phase accumulator. The control word of the phase accumulator is determined by the ratio of the variable sampling frequency and the fixed sampling frequency of the DAC:

$$\omega = \frac{f_s}{f_{DAC}} = \frac{T_{DAC}}{T_s} < 1$$

(6)

The phase of the output samples can be expressed as:

$$\eta(km) = \eta(km-1) + \omega$$
$$\eta(km+1) = \eta(km-1) + 2\omega$$
$$\cdots$$
$$\eta(km+k-1) = \eta(km-1) + k\omega$$

(7)

where $\eta(0) = 0$.

In FPGA, we need to quantize and normalize $\omega$ and $\eta(m)$, and add a sign bit to the highest bit, we denote this process as \([\cdot]\). The change of the sign bit can determine whether the input sample sequence needs to be updated. As shown in Fig. 6, the $y[0]$ is the initial samples and its phase is 0, and the phases of $y[1]$, $y[2]$, $y[3]$ are $\frac{T_{DAC}}{T_s}$, $\frac{2T_{DAC}}{T_s}$, $\frac{3T_{DAC}}{T_s}$ respectively. Because $2T_{DAC} < T_s < 3T_{DAC}$ in Fig. 6, the phase of $y[0]$,$ y[1]$,$ y[2]$ does not exceed 1, and the sign bit
has not changed, so their corresponding input sample is \( x[0] \); the phase of \( y[3] \) exceeds 1, the sign bit changes, and the corresponding input sample is updated to \( x[1] \).

![Fig. 6. The relationship between phase and input samples.](image)

The enable of input sample update can be expressed as:

\[
\begin{align*}
\text{en}[km] &= \text{sgn}[\eta(km-1)] \oplus \text{sgn}[\eta(km)] \\
\text{en}[km+1] &= \text{sgn}[\eta(km)] \oplus \text{sgn}[\eta(km+1)] \\
&\quad \vdots \\
\text{en}[km+k-1] &= \text{sgn}[\eta(km+k-2)] \oplus \text{sgn}[\eta(km+k-1)]
\end{align*}
\]

(8)

where \( \text{sgn} \) denotes the sign bit and \( \oplus \) represents exclusive OR (XOR).

Regarding \( l \) registers as waveform memories with a length of \( lk \), the position of input sample \( x[m] \) in the register group is recorded as \( \text{base}[m] \), then:

\[
\begin{align*}
\text{base}[km] &= \text{en}[km]?\text{base}[km-1]+1:0 \\
\text{base}[km+1] &= \text{en}[km+1]?\text{base}[km]+1:0 \\
&\quad \vdots \\
\text{base}[km+k-1] &= \text{en}[km+k-1]?\text{base}[km+k-2]+1:0
\end{align*}
\]

(9)

where "?" represents the logical condition judgement. In the next unit time, if the position of the first sample of the first path exceeds the position of the first register:

\[
\text{if base}[km+1]-n > k-1.
\]

(10)

It proves that the samples in the first register \( \text{Reg}(l-1) \) have been useless, so the samples in this register need to be updated. The shift registers implement shifting in steps of \( k \) samples, and the samples in the first register are replaced by \( k \) new samples output by FIFO. At the end, update the base address of the sample sequence.

\[
\begin{align*}
\text{Reg}(0) &= \text{Reg}(1) \\
\text{Reg}(1) &= \text{Reg}(2) \\
&\quad \vdots \\
\text{Reg}(l-1) &= \text{FIFO}_\text{out} \\
\text{base}[k(m+1)] &= \text{base}[k(m+1)]-k
\end{align*}
\]

(11)

The above is the process that the phase controller realizes the control of the input sample sequence through the sign bit of the phase. And the time interval is determined by the numerical
bits of the phase as below:

\[
\mu[km] = \text{num}\left[\eta(km)\right]
\]

\[
\mu[km + 1] = \text{num}\left[\eta(km + 1)\right]
\]

\[
\ldots
\]

\[
\mu[km + k - 1] = \text{num}\left[\eta(km + k - 1)\right]
\]

(12)

4. Experiment

In order to verify the performance of the designed arbitrary waveform synthesis module based on digital resampling, we have designed an experimental test bench to measure its spurs as shown in Fig. 7. An RTO 1024 Rohde&Schwarz oscilloscope, with a 2 GHz bandwidth and a sampling rate up to 10 GSa/s was used to observe the time-domain diagram of the output waveform. A 53230A Keysight Cymometer, with a bandwidth of 6 GHz and 11-bit measurement resolution, was used to measure the frequency of the output waveform. An N9010A Agilent EXA Signal Analyzer, with a frequency measurement range from 10 Hz to 26.5 GHz was used to measure the spurious performance. The designed waveform synthesis circuit board is shown in Fig. 8 is connect to the industrial computer through the PCI-e golden finger interface and integrated in the chassis called the designed circuit module. An external screen is used to display the user interface of the waveform editing software, which is used to edit the waveform samples and generate control commands. The designed circuit board includes six parts: Power module, DDR3 SDRAM, FPGA, DAC, clock module and analog module. The power module provides energy for the entire circuit board. DDR3 SDRAM is used to store data of waveform samples. FPGA is the core of the entire circuit board. Its functions include obtaining waveform samples and control instructions from an industrial computer, completing digital waveform synthesis, digital resampling, and finally transmitting the waveform samples to the DAC (AD9136) through a high-speed interface. The FPGA we chose is xcku060-ffva1156-2e of Xilinx company, and the resource utilization are as the below table:

<table>
<thead>
<tr>
<th>Resource</th>
<th>BRAM Number</th>
<th>LUT 12120</th>
<th>Slice flip flops 24240</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage</td>
<td>7%</td>
<td>5%</td>
<td>5%</td>
</tr>
</tbody>
</table>

Table 1. Output frequency after resampling test results.

Fig. 7. Experimental test bench.
Firstly, we verify the quality of the single tone waveform generated by the proposed waveform synthesis system. In order to facilitate the comparison of experimental results, we have set up two modes, with digital resampling (proposed by this paper in Fig. 3) and without digital resampling (DDWS). In the digital resampling mode, the fixed sampling rate of DAC is set to 2 GSPS and the resampling module is a FIR filter bank with 11 filters with 5-order according to [23]. In the mode without digital resampling, the sampling rate of DAC is set to the sampling rate in variable clock domain. We acquire a sine waveform samples with a period length of 64 samples through the calculation formula equation (4) and set the variable sampling rate to 983 MSPS, 1.228 GSPS, and 1.474 GSPS respectively. Therefore, the frequency of the generated waveforms should be 15.359375 MHz, 19.1875 MHz, and 23.03125 MHz. The accuracy of the crystal clock source we chose is 1ppm, so the output frequency should be within the range of 15.359359640625 MHz~15.359390359375 MHz, 19.1874808125 MHz~19.1875191875 MHz, and 23.03122696875 MHz~23.03127303125 MHz. The frequencies of the output waveforms measured by the cymometer are shown in Table 2, all within the effective range. Therefore, digital resampling does not change the frequency characteristics of the generated waveform.

Then, we verify the improvement of digital resampling on spurious. We compare the spectrums of the waveforms generated by the two modes. Figure 9 is a spectrum test diagram of sine waveforms of 15.35938 MHz generated by the two modes. If we set the low-pass filter with a cut-off frequency of 1 GHz (half of the DAC’s maximum sampling rate). In the without digital resampling mode (DDWS), the image component is in the passband of the low-pass filter and cannot be filtered out, the measurement of SFDR is 22 dBc in Fig. 9(a). In the digital resampling mode, the sampling rate is 2 GSPS, so the image component is at 1986.4062 MHz (\(f_s-f_c\)) that is in the cut-off band of the low-pass filter, the measurement of SFDR is 65 dBc in Fig. 9(b). Additionally, in Fig. 9(c, e), it can be found that the position of the image component will change greatly due to the different sampling rate. In Fig. 9(d, f), the image component of the output waveform after digital resampling appear at 1.980 GHz and 1.9756 GHz, respectively. Because the frequency measurement of the signal analyzer is not very accurate,
the frequency point of the image component basically satisfies the equation \(2\text{GHz} \times f_s\). Therefore, the arbitrary waveform synthesis structure based on digital resampling can make the sampling rate of the DAC be a fixed value. According to the Nyquist's theorem, the effective component of output is at range of \(0 \sim 0.5 f_s\), which is about \(0 \sim 0.4 f_s\) in practical applications. And the frequency of the image component is at \(f_s - f_s\). Therefore, if the cut-off frequency of the filter is set to half of the sampling frequency, the effective component can be effectively retained, and the image component can be filtered out. The SFDR of the output signal will not be affected by the image component, and is only related to the performance of the DAC.

![Figure 9](image_url)  
Fig. 9. Spectrums of the output waveforms generated by two modes.

The above experiment proves that the proposed waveform synthesize method based on digital resampling can realize the generation of single tone waveform effectively. And next we will verify the method’s ability to synthesize multi-tone waveform. First under a 2 GSPS...
sampling rate, get sweep waveform samples with a variation range of 10 MHz~114.4 MHz, the test results under time and frequency domain are shown in Fig. 10.

![Sweep waveform under time domain](image1)
![Sweep waveform under frequency domain](image2)

(a) sweep waveform under time domain (fs=2 GSPS) (b) sweep waveform under frequency domain (fs=2 GSPS)

Fig. 10. Test results of sweep waveform.

Then, under the mode without digital resampling, set the sampling rate to 983 MSPS, 1.228 GSPS and 1.474 GSPS, the spectrums are shown in Fig. 11.

![Spectrums without digital resampling](image3)
![Spectrums without digital resampling](image4)
![Spectrums without digital resampling](image5)

(a) fs=983MSPS (b) fs=1.228GSPS (c) fs=1.474GSPS

Fig. 11. Spectrums without digital resampling.

Enable the digital resampling, the spectrums are shown in Fig. 12.

![Spectrums with digital resampling](image6)
![Spectrums with digital resampling](image7)
![Spectrums with digital resampling](image8)

(a) fs1=983MSPS (b) fs1=1.228GSPS (c) fs1=1.474GSPS

Fig 12. Spectrums with digital resampling.

We can see from Fig. 11 that the image components are at 1.0428 GHz, 1.298 GHz, 1.5576 GHz before resampling, and are at 2.0584 GHz, 2.0724 GHz, 2.0856 GHz, 2.1164 GHz after, which fit the theoretical values. We can conclude from the results that the digital resampling module can alter waveform samples under variable sampling rate to waveform samples under fixed sampling rate of 2 GSPS without changing the output frequency. The image components
can be moved from a relatively low frequency to the vicinity of 2 GHz for subsequent low-pass filter to filter it out and improve the spectrum purity of the output waveforms.

5. Conclusions

We realized an arbitrary waveform synthesize method based on digital resampling method, avoiding the limitations exist in DDFS and DDWS. We separate the waveform synthesis process to two parts, a variable delay filter based on the farrow structure is used in the digital waveform synthesis part to realize the conversion of sampling rate so to avoid the phase truncation error that will appear in DDFS. The digital-to-analog conversion part is realized at a fixed sampling rate, compared to DDWS, we greatly restrain the image components. At the same time, in order to improve the sampling rate of the system, the waveform sample shift register and phase controller are used in FPGA to realize the parallel output of non-uniform resampling samples. As a result, we improved SFDR by 43 dBc and the output waveform’s spectrum purity is improved, compared to DDWS. The effectiveness of the proposed arbitrary waveform synthesis structure was verified, and it is clear that this new method can be applied to generated waveforms with much lower spurious and high sampling rate.

References


Wen Hao Zhao received the bachelor’s degree in automation engineering from the University of Electronic Science and Technology of China, Chengdu, China in 2015 and master’s degree in automation engineering from the University of Electronic Science and Technology of China, Chengdu, China in 2018. Currently, he is studying at the School of Automation Engineering, the University of Electronic Science and Technology of China. His current research interests include digital waveform synthesis and digital signal processing.

Shulin Tian was born in Sichuan province, China, in 1968. He received M.S. and Ph.D. degrees both from University of Electronic Science and Technology of China (UESTC), China, in 1991 and 2009 respectively. He is currently a Professor of the School of Automation Engineering, UESTC. He has led a number of projects including high-speed, high-precision data acquisition, network and communication testing, testing bus technology, and testing system integration. He has published more than 50 research papers, and owns ten patents in China. His research interests include testability analysis and fault diagnosis of electronic device systems, and broadband time-domain testing technology and instruments and Electrochemical Sensor.
Guangkun Guo was born in Shandong, China, in 1982. He received his B.S. degree from the Chengdu University of Technology (CDUT) in 2006, and the M.S. degree in measurement technology and instruments from the University of Electronic Science and Technology of China (UESTC) in 2010. His Ph.D. degree was received in instrument science and technology from UESTC in 2019. He is now a Research Assistant in UESTC. His research interests include atomic clock, highly stable frequency synthesis, and highly stable frequency transfer over free-space or underwater link.

Jiajing You was born in Chongqing, China, in 1995. She received the bachelor’s degree from China University of Petroleum (Beijing), Beijing, China in 2018 and the master’s degree from the University of Electronic Science and Technology of China, Chengdu, China in 2021. Her research interests are digital waveform synthesis and sampling rate transformation.

Qiong Wu was born in Sichuan province, China, 1997. She received the bachelor’s degree in automation engineering from Nanjing University of Aeronautics and Astronautics in 2019. She’s currently studying for a master degree at the School of Automation on Engineering, the University of Electronic Science and Technology of China.

Ke Liu was born in Sichuan, China, in 1978. He received the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China, China, in 2003 and 2010, respectively, where he is currently a Professor with the School of Automation Engineering. His research interests include frequency synthesis, measurements and instruments, and signal processing.