http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 9 (4): 686-691, 2010
ISSN 1812-5638

Bit Synchronization of PCM/FM Signals at Low SNR

Zhilu Wu, Qiyang Wang, Nan Zhao and Guanghui Ren
School of Electronics and Information Technology, Harbin Institute of Technology,
Harbin, Heilongjiang 150001, China

Abstract: In this study, a novel bit synchronization method for the telemetry digital Intermediate Frequency (IF) receiver of PCM/FM signals at low Signal-Noise Ratios (SNR) is proposed. Bit synchronization is also known as symbol synchronization, which is an important technique in the communication systems nowadays and it is essential in both baseband transmission and radio-frequency transmission systems. An imperfect bit synchronization algorithm will affect the performance of the communication system and even cause the communication system work abnormally. As SNR in PCM/FM telemetry system is much lower than other conventional communication systems, traditional bit synchronization algorithms can't meet the requirements. So by means of the mean filtering technique in image processing, a novel bit synchronization algorithm at low SNR for PCM/FM based on FPGA is proposed and has already been implemented in our telemetry digital IF receiver. Via simulations, it is shown that this algorithm can achieve excellent performance at low SNR.

Key words: PCM/FM, bit synchronization, low SNR, FPGA, mean filtering

INTRODUCTION

Synchronization is one of the most important techniques in the digital communication systems nowadays (Li *et al*., 2009), which will determine the performance of the Intermediate Frequency (IF) receivers. Efficient data detection requires the receiver know when one symbol (bit) begins and when it ends and then the receiver can make a decision in the middle of the symbol to detect it correctly. An efficient detection of the symbols requires a local synchronized clock, with its frequency accurately equal to the frequency of the transmitted baseband symbols and its phase according with the best decision time. This process of generating the local synchronized clock is known as the bit synchronization (Ossieur *et al*., 2009). Generally, bit synchronization can be realized by the pilot frequency method and data-derived method. The first method requires extra signal power and frequency spectrum, therefore it is not often used. In the data-derived method (Zicari *et al*., 2006), the receiver includes extracts bit-synchronized clock directly from the transmitted signals and the main advantage of this method is that no extra power expenditure and frequency spectrum are required. However, in the case of low SNR, none of them are available.

In this study, by means of the mean filtering technique in image processing (Zhang *et al*., 2008), a bit synchronization algorithm at low SNR for PCM/FM is proposed and applied to the telemetry digital IF receiver (Jordi *et al*., 2008) due to the disadvantages of the above two methods. It is implemented based on FPGA using VHDL language. By simulating in software of Quartus II, it is shown that the performance of this algorithm is excellent when the SNR (E_b/N_0) above 5 dB with its maximum synchronized error less than 1/14 of the baseband-symbol period. It also has been used in our telemetry digital IF receiver of PCM/FM signals successfully.

PRINCIPLES OF TELEMETRY DIGITAL IF RECEIVER

Synchronization is a fundamental problem in the digital communications. Synchronization algorithms depend on the initial error very much. When the initial error propagates, the convergence time of synchronization becomes unpredictable and unstable, so the performance of the communication systems deteriorates. In the digital communication systems, bit synchronization is a crucial technique for the best symbol detection. It analyzes the noisy and distorted demodulated signal waveforms to detect the corresponding logic value and extract the bit synchronized clock. As SNR in the PCM/FM telemetry system is lower than other conventional communication systems, the traditional bit synchronization algorithms can't meet the requirements.

Corresponding Author: Zhilu Wu, School of Electronics and Information Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China

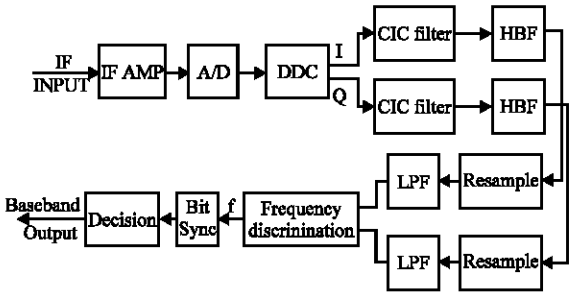

Fig. 1: Digital IF receiver diagram

The bit synchronization algorithm researched in this study is one part important of our PCM/FM (Li *et al*., 2006; Lee and Ra, 2008) telemetry digital IF receiver. The structure of the receiver is shown in Fig. 1.

In Fig. 1, the input of the receiver is the IF PCM/FM signal with its carrier at 70 MHz. The power of input signal is adjusted by the IF Amplification (IF AMP). Then the signal is under sampled by A/D at 56 MHz. The received under sampled signal can be written as:

$$S(n) = \sum_{m=-\infty}^{+\infty} A_0 g(n-m)\cos\left[(\omega_c + a_m \Delta\omega)n\right] \quad (1)$$

where, $\Delta\omega$ is the frequency deviation to the carrier and a_m is the value of the mth transmitted symbol, which can be +1 or -1.

The signal in Eq. 1 can be carried to zero IF is by the Digital down Conversion (DDC). The obtained baseband quadrature signals are as follows:

$$X_I(n) = \sum_{m=-\infty}^{+\infty} A_0 g(n-m)\cos(a_m \cdot \Delta\omega \cdot n) \quad (2)$$

$$X_Q(n) = \sum_{m=-\infty}^{+\infty} A_0 g(n-m)\sin(a_m \cdot \Delta\omega \cdot n) \quad (3)$$

In the receiver, the sampling frequency should be decimation, so the frequency discrimination can be implemented easily. Therefore, cascaded integrator-comb (CIC) filter (Dolecek, 2009), half band filter (BHF) (Yekta, 2008) and resample (Sheikh and Masud, 2010) blocks is used in the receiver to convert the sampling frequency to $14f_b$, where f_b is the rate of the baseband symbols. The signal from DDC is filtered and decimated by CIC filter and BHF and then the sampling frequency is converted to $14f_b$ to $28f_b$. Then the signal is resampled in the resample block at the frequency $14f_b$.

After resampled and filtered by the Low-Pass Filter (LPF), the signal is demodulated by frequency discriminator, which can be expressed as:

$$f(n) = X_I(n-1)\times X_Q(n) - X_I(n)\times X_Q(n-1) \quad (4)$$

Implementation of Eq. 4 requires only multiplication and subtraction and FPGA is suitable for it.

Finally, the signal f output from frequency discriminator is the input signal of the bit synchronization block which is researched in this study.

BIT SYNCHRONIZATION ALGORITHM AT LOW SNR

Conventional bit synchronization methods can work normally at high SNR; however when the SNR becomes lower its performance degrades rapidly. Therefore, in this study a novel bit synchronization algorithm which can adapt to low SNR is proposed. This algorithm takes inspiration from the mean filtering techniques in field of image processing.

The mean filtering technique has excellent performance in noise suppression. So, we can use the average value of former n-bit synchronized sequence a_1, a_2,..., a_n to adjust the position of the (n+1)th point a_{n+1}. Then the position of the (n+1)th bit synchronized point can be obtained as follows:

$$a_{n+1} = (a_1 + a_2 + ... + a_n)/n = a_{n+1} - \frac{a_{n+1}-a_1}{n} - \frac{a_{n+1}-a_2}{n} - ... - \frac{a_{n+1}-a_n}{n} \quad (5)$$

It is shown in Eq. 5 that the value of the (n+1)th point can be calculated using a_{n+1} subtract the mean difference between a_{n+1} and a_i, i=1,..., n. In our telemetry system, each baseband symbol is sampled for 14 points, so we calculate a_{n+1} modulo14, which will adjust the position of a_{n+1} using a_i, i = 1,..., n.

In this study, a novel bit synchronization algorithm is proposed based on the principle of the mean filtering technique described in Eq. 5. The algorithm can be achieved by four modules, including zero-crossing detection, coarse adjustment, fine adjustment and synchronized points completion modules. Its flowchart is shown in Fig. 2.

Zero-crossing detection: When SNR is lower, the signal f generated from the frequency discriminator becomes worse, which will cause zero points of the signal f offset or even pseudo-zero points appear. Therefore, the zero-crossing detection made directly through the zero points is not accurate enough. In this paper, a novel zero-crossing detection method with threshold is proposed, in which 20 continuous samples are detected each time (14 points are sampled in one symbol). If 11111111110000000000 or 00000000011111111 are detected, a bit synchronized pulse (syn) is generated, which lasts one period of the sample clock; otherwise, no

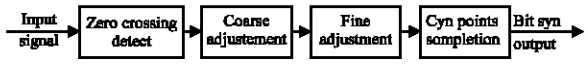

Fig. 2: Bit synchronization diagram

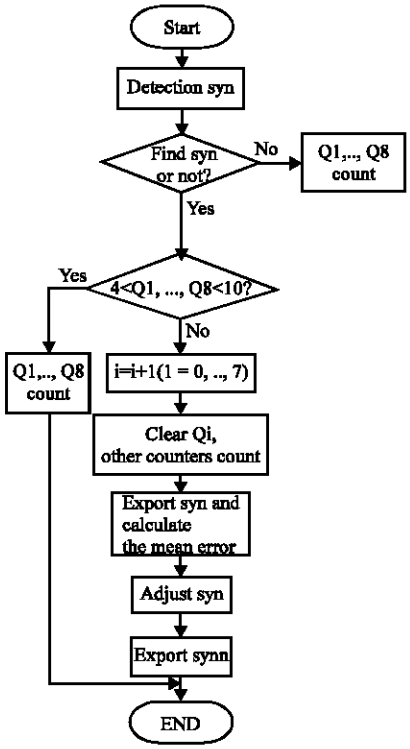

Fig. 3: Flowchart of the coarse adjustment algorithm

pulses are generated. Through this method, most of the pseudo-zero points can be removed.

Coarse adjustment: Through the zero-crossing detection, most of the pseudo-zero points are removed; however, some of the zero points are still pseudo and the offsets of the zero points should be reduced. Therefore coarse adjustment is applied to resolve the problems, which is based on the principle of the mean filtering technique. The flowchart of the coarse adjustment algorithm is shown in Fig. 3.

As in Fig. 3, the coarse adjustment algorithm can be described as the following steps:

- **Step 1:** From the 9th bit synchronized pulse syn out of the zero-crossing detection, calculate the number of the samples between it and the 8 former bit synchronized pulses respectively, then mod the results by 14. This can be done by using 8 modulo-14 counters to count the number of the samples between these pulses

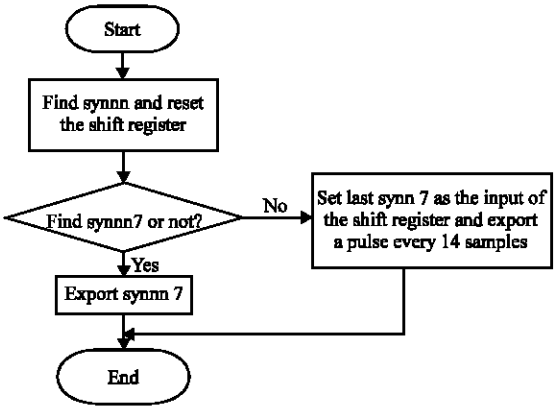

Fig. 4: Flowchart of the synchronized points completion

- **Step 2:** Analyze the results from Step1, if the number of the samples between the current bit synchronized pulse and any one of the former 8 pulses is larger than 4 or smaller than 10, delete this bit synchronized pulse because its offset is relatively big; otherwise, output this pulse and calculate the mean error (errorave) based on the former 8 pulses
- **Step 3:** Adjust the output from Step 2. If 0<errorave<5, bring forward the pulse by errorave samples; otherwise, delay it by (14-errorave) samples

Through coarse adjustment, the offset of the bit synchronized pulse is within 4 samples and it still can not meet the requirements at low SNR. Hence, further adjustment should be performed.

Fine adjustment: Similar to the coarse adjustment, the fine adjustment calculate the number of the samples between the current bit synchronized pulse synn and the former 32 bit synchronized pulses respectively, then mod the results by 14. As almost all the pseudo-zero points have been removed in the coarse synchronization, the bit synchronized pulse synn should only be adjusted by the mean error calculated from the former 32 pulsed in the fine adjustment and synnn can be obtained.

After the fine adjustment, the positions of bit synchronized pulses are excellent with its offset no more than 1 sample and the systems requirements are meet.

Synchronized points completion: The bit synchronized pulses obtained through the above modules is accurate; however some of the bit synchronized pulses are missed during the processing or don't exist because the transmitted symbols are continuous 1 or 0. Therefore, the missing bit synchronized pulses should be filled in synchronized points completion. The flowchart of the synchronized points completion is described in Fig. 4.

In Fig. 4, synnn7 is the delayed signal by synnn by 7 samples.

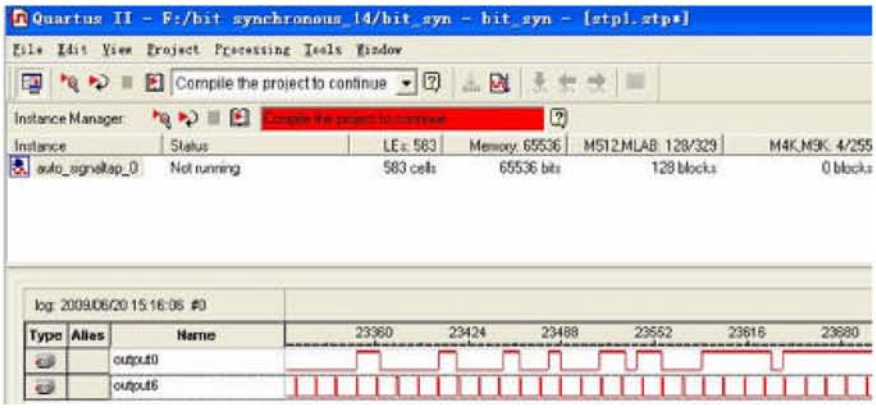

Fig. 5: SignalTap II waveform

SIMULATION RESULTS

Before giving the simulation results, we define the communication system as follows:

- PCM-FM modulation mode, with the carrier frequency of input IF signal 70 MHz
- Baseband symbol rate is variable and the range of it is from 100 bps to 2 Mbps
- The frequency of the system clock is 56 MHz and the minimum value of SNR which is expressed by E_b/N_0 is 5 dB

The proposed bit synchronization algorithm is implemented based on FPGA using VHDL language. Compile the project using the software of Quartus II and 815 ALUs are occupied with the maximal working frequency 91.95 MHz.

SignalTap II, which is an embedded logic analyzer in Quartus II, is used in the hardware debugging. The waveform from SignalTap II is shown in Fig. 5.

In Fig. 5, the signal output0 is the input frequency signal f which is the output from frequency discriminator and the signal output6 is the finally bit synchronized pulsesS. The data in the SignalTap II waveform can be exported as a tbl file and the format of the tbl file is depicted in Fig. 6.

So, we can analyze the tbl file by means of MATLAB and the sequence value of bit synchronized pulses can be obtained to analyze the performance of bit synchronization.

The error of the bit synchronization and the number of the effective bit synchronized pulses obtained by the fine adjustment are analyzed in Fig. 7 to 9 with the value of E_b/N_0 5 dB, 10 dB and 15 dB, respectively. The number of input baseband symbols is 20000 and the number of symbol-jumping edges is 10087.

```
Pattern
   0.0> 0 1 0 = 000 0 0 0 0 0 0
  20.0> 1 1 1 = 000 0 0 0 0 0 0
  40.0> 0 1 0 = 000 0 0 0 0 0 0
  60.0> 1 1 0 = 000 0 0 0 0 0 0
  80.0> 0 1 0 = 000 0 0 0 0 0 0
```

Fig. 6: The exported tbl file

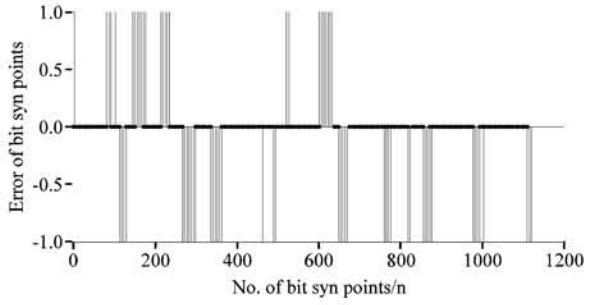

Fig. 7: Error and number of the bit synchronized pulses with E_b/N_0 5 dB

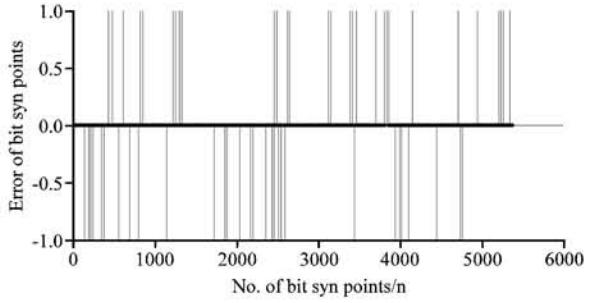

Fig. 8: Error and number of the bit synchronized pulses with E_b/N_0 10 dB

In Fig. 7-9, the number of the bit synchronized pulses and the error of every pulse are given. The simulation results show that the performance of the bit synchronization is excellent when E_b/N_0 is above 5 dB

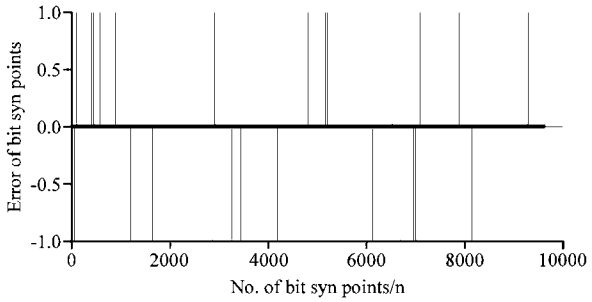

Fig. 9: Error and number of the bit synchronized pulses with E_b/N_0 15 dB

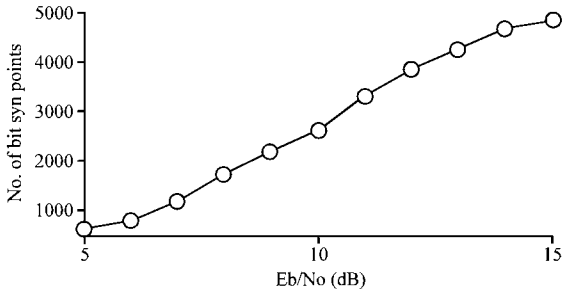

Fig. 10: Number of the bit synchronized pulses vs. E_b/N_0

with the maximal error less than 1 sample. It is also shown that the number of the bit synchronized pulses is increased as SNR becomes larger and the number of the bit synchronized pulses versus E_b/N_0 curve is explicitly depicted in Fig. 10 when 10000 baseband symbols are transmitted and 5024 symbol-jumping edges take place.

To further analyze the relationship of the performance of the bit synchronization and SNR, the relative error L is defined in however, the proportion relationship of bit synchronization and SNR can not proof the good performance of the bit synchronization enough. The effect of relative error caused by SNR should be analyzed. The relative error L is defined:

$$L = \frac{\sum |error|}{n} \quad (6)$$

In Eq. 6, error is the error of each bit synchronized pulse and n is the number of effective bit synchronized pulses which are extracted. So L reflects the average performance of the bit synchronization.

The simulation results in Fig. 11 show that the relative error is inversely proportional to SNR. Furthermore, the relative error is only 0.22 when SNR is 5 dB and that is to say, the average error is only 0.22 samples. Hence, the performance of proposed bit synchronization algorithm is pretty good at low SNR.

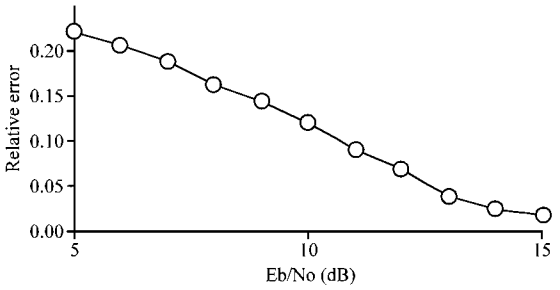

Fig. 11: SNR and relative error

CONCLUSIONS

Traditional bit synchronization algorithms have good performance in case of high SNR, but when the SNR becomes lower (5 dB), the performance gets worse. In the study, a novel bit synchronization algorithm at low SNR algorithm is proposed inspired from the mean filtering technique in image processing. The proposed bit synchronization algorithm consists of zero-crossing detection, coarse adjustment, fine adjustment and synchronized points completion, which enhances the performance obviously. The algorithm is implemented on FPGA and through software simulation and hardware debugging, it is shown that its performance is excellent when SNR is above 5 dB, which is much better than the traditional bit synchronization algorithm. The algorithm has already been implemented in our telemetry digital IF receiver of PCM/FM signals successfully.

REFERENCES

Dolecek, G.J., 2009. Simple wideband CIC compensator. Electr. Lett., 45: 1270-1271.

Jordi, S.R., S. Fredy and T.M. Oses, 2008. Simple and efficient inductive telemetry system with data and power transmission. Microelectron. J., 39: 103-111.

Lee, S.R. and S.W. Ra, 2008. Variable cuteoff frequency pre-modulation filter for PCM/FM transmission system. IEICE Trans. Commun. Ser., 91: 1387-1396.

Li, Q., D. Liu and S. Yuan, 2006. Research on the method of demodulation for PCM/FM signal based on instantaneous frequency measurement. Proceedings of IMACS Multiconference on Computational Engineering in Systems Applications, Oct. 4-6, Beijing, China, pp: 93-96.

Li, X., Y.C. Wu and E. Serpedin, 2009. Timing synchronization in decode-and-forward cooperative communication systems. IEEE Trans. Signal Process., 57: 1444-1455.

Ossieur, P., J. Bauwelinck and X. Yin, 2009. A dual-rate burst-mode bit synchronization and data recovery circuit with fast optimum decision phase calculation. AEU Int. J. Electr. Commun., 63: 931-938.

Sheikh, F. and S. Masud, 2010. Sample rate conversion filter design for multi-standard software radios. Digital Signal Process., 20: 3-12.

Yekta, M.M.J., 2008. Half-band FIR fractional delay filters with closed-form coefficient formulas and modular implementation based on Lagrange interpolators. Signal Process., 88: 2913-2916.

Zhang, X.M., Z.P. Yin and Y. Xiong, 2008. Adaptive switching mean filter using conditional morphological noise detector. Electr. Lett., 44: 406-407.

Zicari, P., P. Corsonello and S. Perri, 2006. An efficient bit-detection and timing recovery circuit for FPGAs. Proceedings of 13th IEEE International Conference on Electronics, Circuits and Systems, Dec. 10-13, Nice, France, pp: 168-171.