

CLOCK EXTRACTION

David Spreadbury, Plextek Ltd.
djs@plextek.co.uk

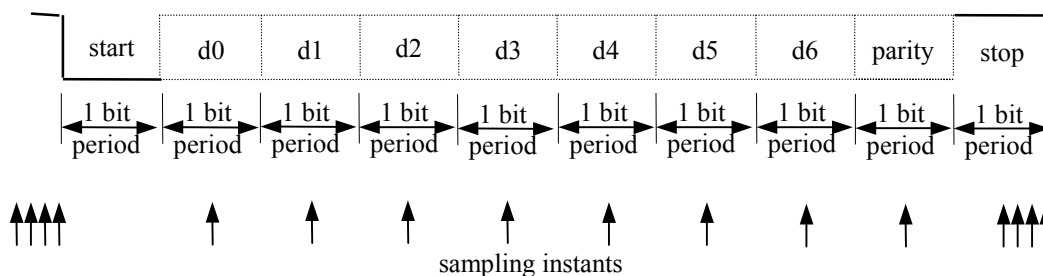
Instruct a wise man and he will be wiser still Proverbs 9:9

Introduction

DSP has exploded onto the radio scene, especially in the areas of modulation and demodulation, where increasingly complex or precise techniques are being employed to improve performance. Many of the standard techniques have been characterised in text books by a BER against E_b/N_0 plot. In order to approach these curves, not only must the symbols be optimally extracted from interferers and noise, but the instant at which the recovered signal is sampled must be correct. This is a less publicised but non-trivial issue which warrants careful design. Rather than reproduce theory from books, the following is a qualitative introduction to the techniques which the author has found useful in 6 years at Plextek Ltd.

Synchronising Events

The simplest form of clock extraction is probably that used by the familiar serial communication port on every PC (although this interface predates PCs by many years). Each character of say 7 bits, with its (optional) appended parity bit, is preceded by a start bit (space) and followed by a stop bit (mark), thus:



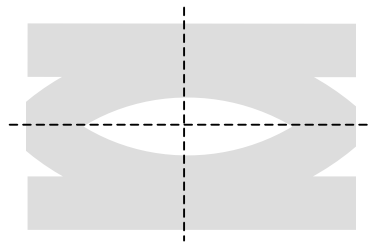
“Mark” was historically the condition under which current flowed in the line (this caused disconnected printers to chatter, which was a good fault indication).

This construction contains 2 features: first, a defined beginning (the first mark to space transition), and second, defined periods when each bit is valid. The receiver has only to detect the start transition (typically sampling the signal at 16 times the bit rate) and then time out initially 1.5 bit times and then exact bit times thereafter in order to be able to shift in each bit in turn and so reconstruct the character.

This system has worked well for many years and is still in common use today where there is little distortion or noise, and it is this requirement that limits its application to short distances in relatively quiet environments. Baud rates (the inverse of the bit time) have risen from 50 to 19200 and above. There is good tolerance to clock frequency error, accepting (say) a 3/16 bit slip over 9.5 bit periods or about 2%. The synchronisation bits occupy 20% of the raw bit rate.

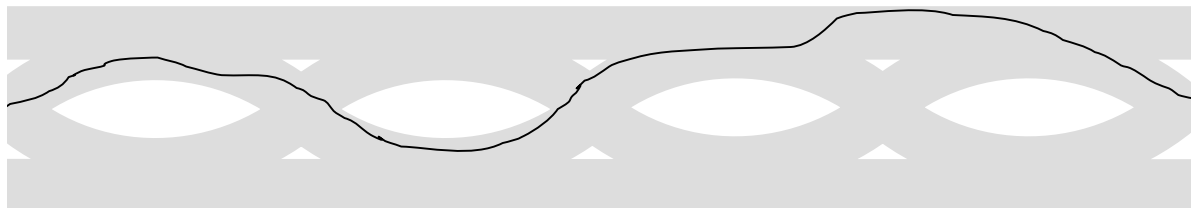
Data transmitted by radio is a different matter, as the need to conserve power and bandwidth becomes ever more important.

The eye diagram for the serial data above is rectangular, and the sampling instant requires only a cumulative accuracy of better than (approximately) half a bit time. The typical eye diagram that we deal with today looks more like this



which can be deceptive: it gives the impression that the threshold and timing instant are obvious. This is due to our tendency to visually integrate the traces and see the underlying mean, which of course the demodulator cannot see unless it has the luxury of being able to store the whole transmission before processing it. Furthermore in a real system there are other factors like carrier offset and component drift which make it unlikely that a great many symbols can actually be laid on each other exactly, i.e. the ability of a demodulator to find a mean is limited. The eye confirms that the timing requires far better accuracy than half a symbol time - the vertical eye opening starts to reduce as soon as the timing departs from the ideal.

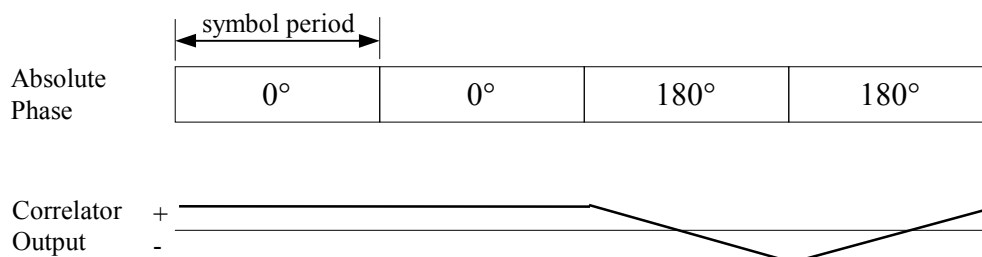
The signal thus comprises a sequence of symbols, each of which departs from the ideal in a different way:



It can be seen that any timing information derived from a single transition of this signal is likely to contain significant error (indeed, some transitions may not be seen at all). This error can be reduced by looking at more of the signal, so that the random effects of noise may be reduced.

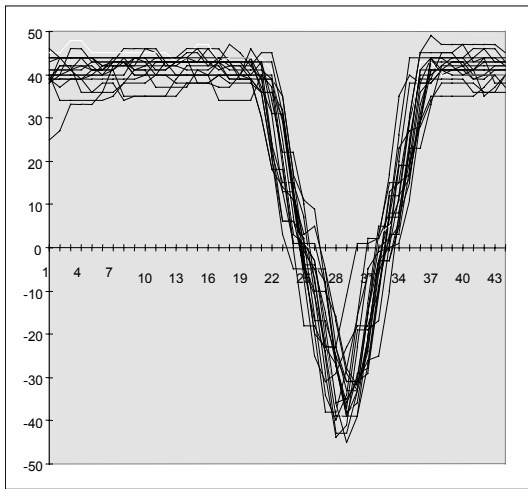
One simple way which has been employed in practice is to average the position of more than one transition.

Example: A BPSK signal from a limiting IF amplifier is sampled at 8 times the symbol rate. The last 8 samples are correlated with the previous 8 (each sample is multiplied by the sample 16 clocks before, and the result of the most recent 8 multiplies summed) to give a positive output on continuous carrier and a negative peak when a phase transition occurs, thus:

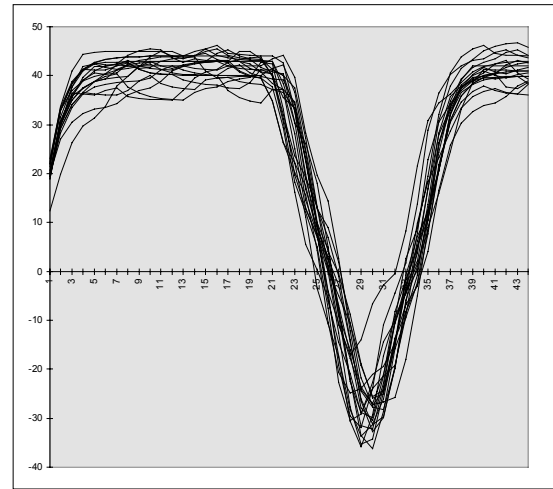


On signals with 12dB SNR the correlator output from a transmission's first synchronising phase reversal was observed to exhibit significant variation at 12dB NR, so it was passed through a simple digital low pass filter:

(i) Output from practical correlator

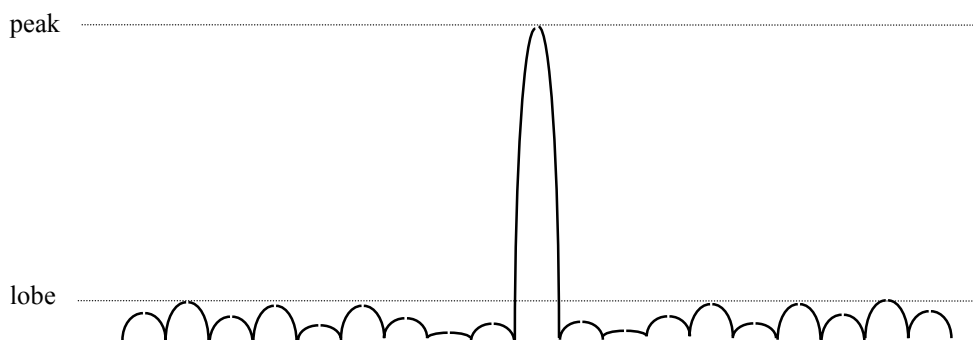


(ii) Lowpass filtered output



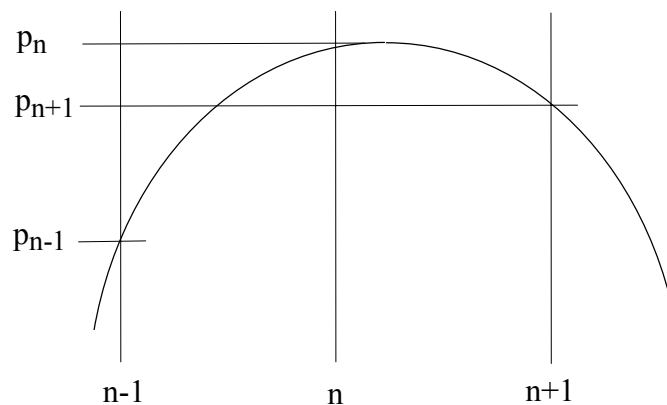
In the final FPGA design a counter was initially zeroed and allowed to count up only when the signal above fell below a threshold (about 10) for the first time. When the signal next exceeded the threshold the count was halved and subsequently allowed to count up modulo 8. This positive threshold reduces the effect of the unstable negative peaks (phase reversals being less precise than CW) and the halving effectively averages the positions of the two threshold crossings. Signal symbols were sampled each time the count passed through zero.

A better way of obtaining timing form an extended section of signal is to correlate against a known pattern. Suitable patterns are often employed in burst (TDMA) transmissions to allow training (e.g. in the centre of a GSM burst), and comprise symbol sequences whose complex modulation trajectories have very narrow autocorrelation peaks. These sequences can be precalculated and the conjugate of each complex sample stored in memory as a reference. The reference sample rate must be the same as the signal sample rate, which must be high enough to represent the complete waveform (i.e. more than one, typically 2 samples per symbol). The correlation processes can be thought of as a matched filter and should be performed each time a new signal sample is received (although not necessarily in real time). The magnitude of the complex correlation result will exhibit a sharp peak when the input signal and the reference pattern match:



The peak-to-lobe ratio is nominally $\sqrt{(\text{correlation length in samples})}$, and although the SNR is increased by the same ratio, noise will degrade the correlation, as will any phase rotation in the signal along the length of the correlation incurred by frequency offsets. If the total carrier phase rotation over the length of the correlation is θ radians then the peak will be scaled by the factor $\sqrt{(2-2\cos\theta)}/\theta$. Thus a longer correlation offers potentially more gain but is more sensitive to frequency offsets; in practice, lengths of 8 or 16 symbols (16 or 32 samples) used.

Note that the response curve above is continuous, the peak occurring when the input and reference modulations align *exactly*. Since both are sampled, but the signal sampling is uncontrolled, it is more likely that the peak will appear asymmetrical, with the maximum value bracketed by unequal neighbours:



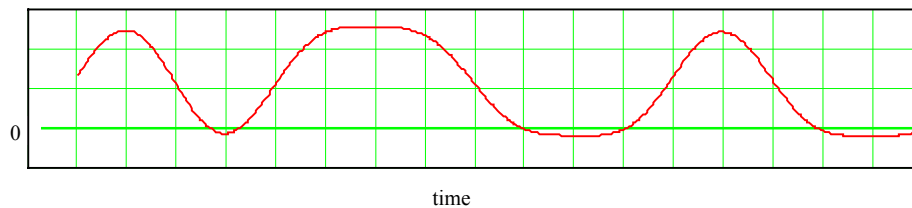
It can be seen that the inequality between P_{n-1} and P_{n+1} is due to the fact that the sampling is not symbol-aligned. It has been found that the expressions of the form

$$(P_{n+1} - P_{n-1})/P_n \quad \text{or} \quad (P_{n+1} - P_{n-1})/(P_{n+1} + P_{n-1})$$

give an signed estimator which is usefully proportional to the distance between the peak sample (n) and the ideal peak. This can be used either to adjust the sampling process or, if the existing samples have to be processed, to select an interpolation filter which will adequately realign the sampled data to the symbol centres (it is interesting to note that if the training mentioned earlier is actually performed, this has the result of time-aligning the signal's symbols to the sampling in a similar way). This technique has actually been used in a piece of equipment designed to test the accuracy of a GSM transmission. The received signal was mixed down to baseband I and Q and captured in hardware, the above correlation being performed in a PC. When a training pattern was detected in the central part of the burst memory the above process was used to realign the waveform and make detailed phase measurements.

The correlation described above is optimum in that it is a completely coherent process, that is all the phase information is retained and the gain in signal to noise ratio is maximised. It is possible, however, to perform similar albeit less efficient pattern matching techniques on signal envelopes.

Consider a low speed FSK signal being observed through a very narrow band filter centred at one of the frequencies. The result will appear as an am modulated carrier, where the carrier is either on or off. Taking the magnitude and low pass filtering appropriate to the modulation rate will generate an NRZ signal of the form



Here, the signal was sampled 64 times per symbol (the trace above shows a half- symbol horizontal grid). To detect the synchronisation pattern 10110010, which was appended to the start of each transmission, a history of 8 symbols (512 samples) was kept, and the following processing was performed as each new sample n arrived:

- take the greatest of samples $n-64$, $n-256$, $n-288$, $n-320$ and $n-448$ (these should all be 0) and save as *maxlo*
- take the least of samples n , $n-128$, $n-160$, $n-192$, $n-384$ (these should all be equal to the signal amplitude) and save as *minhi*
- compute $eye = minhi - maxlo$

If *eye* is positive, then it is possible to select a threshold such that the data sequence 10110010 can be extracted from the waveform, i.e. the pattern is present. As this process continues for successive samples, the value of *eye* will be seen to rise to a peak and then fall back to zero. It is at this peak that the pattern is most well matched i.e. aligned. In the somewhat more complex application worked on by the author, a centre of gravity algorithm was used on this set of positive eye openings to establish the best sampling position (with no interpolation being required because of the 64 times oversampling already in use).

The signal-to-noise improvement in this method is obtained entirely by the narrow-band and low-pass filtering. Coherent processing was not an option because the FSK transmitter used a VCO so that no periods of carrier separated by a frequency shift could be assumed to have any phase relationship. This full application runs in a SHARC DSP and is in everyday public use.

Extractable Characteristics

Some modulation schemes generate signals which possess a measurable characteristic which has a known relationship to symbol timing. Examples of this are amplitude modulation, which can arise intentionally in PAM or as a product of root-raised-cosine filtering in QPSK or $\pi/4$ -QPSK, and spectral components at the symbol rate or a harmonic, as in the symbol rate component present in a squared MSK signal (used by the author in a proprietary DSP modem). All of these are data dependent, and are usually emphasised at the beginning of a transmission by the use of a preamble with an appropriate symbol sequence. The extracted characteristic can be filtered and amplified to initialise a timing circuit, or perhaps set off a high Q resonator which will ring adequately for the duration of a burst.

Phase-locked loops

Any of the above can be successfully used to initialise or direct a digital phase-locked loop, which, as in analogue systems, can be used to track a frequency in noise by giving it a low loop bandwidth.

In essence the operation is similar to the analogue equivalent, but there are differences in detail.

The VCO is replaced by a phase accumulator, which in its simplest form is a counter which wraps round at the end of the count. A 4 bit counter, for example, counts through the sequence 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 0, 1, 2, etc. and has effective phase steps of $\pi/8$ radians. Its (“local oscillator”) frequency is given by the clock frequency divided by the count length. Usually the clock frequency should be an integer multiple of the wanted frequency. The phase resolution (the precision with which symbol sampling can be achieved) is limited to the phase step size. This can be thought of not so much as a variable frequency oscillator but more as a fixed frequency oscillator which can be nudged in phase, which is appropriate for the known symbol-rate applications considered here.

The phase adjustments are effected by momentarily modifying the usual phase step. In the 4 bit example above the count increments by 1. If this were to be replaced by 0 or 2 then the phase would be adjusted by $-\pi/8$ or $\pi/8$ radians respectively. Note that this adjustment will cause count values to be momentarily duplicated or missed out, and any circuitry which operates on the counter output must be able to cope with this.

The phase detector can be constructed in many ways, but the simplest is to sample the count at a given point in the signal to be tracked (a zero crossing, for example). The difference between the sampled count value and a design value (derived from the wanted lock condition) is used as an error signal. In this case, a positive error should result in a -ve phase adjustment and vice versa. Note that this method only responds to zero crossings when they occur - otherwise the counter continues normally and the system free-runs. Thus a digital PLL can lock to waveforms other than tone, e.g. data. Locking to one or both edges is possible.

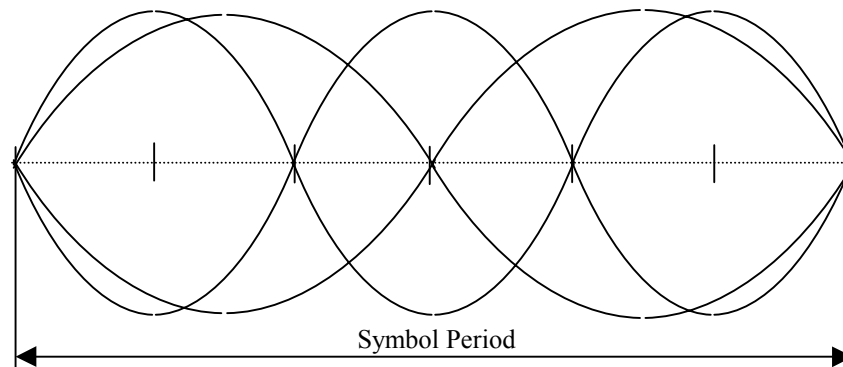
Rather than filter the error signal, it is simpler to accumulate it until it reaches some threshold which then triggers a phase adjustment. The threshold and the size of the adjustment affect the agility and stability of the loop, and tend to be optimised for a given application by simulation (particularly for high speed integer hardware implementations). These characteristics can be changed on the fly so that the agility of a digital PLL can be wide to assist acquisition and narrowed as lock is achieved.

Example 1:

The BPSK example above derived an initial estimate of symbol timing by averaging the timing of two edges. The counter used for this purpose was in fact part of a digital PLL, and each subsequent crossing of the threshold by the filtered correlator output was used to sample the count. The error was defined as the difference between this count and 4.5, these errors being summed until a total of ± 8 was reached, at which point a phase adjustment of ± 1 count was made and the summation reduced by ± 8 . This approach balanced the ability to recover from a poor initial timing estimate with susceptibility to noise. The mechanisms and their parameters were set by repeated simulation runs, on realistic transmissions with interferers and noise, over the whole operating envelope, observing both the internal circuit nodes and the overall bit error rate.

Example 2:

A quite different approach was taken with a 1200 bits per second MSK signal, where the symbols were accurately synthesised as 1 cycle at 1200 Hz or 1.5 cycles at 1800 Hz, with the symbol boundaries located on zero crossing points.



The signal was sampled at 96 times the symbol rate, and a digital PLL with an LO of 7200 Hz (clock rate ÷ 16, symbol rate × 6) was locked to the zero crossings (with this form of modulation, the zero crossings can only occur at 1/6th symbol intervals).

This leaves an ambiguity as to which of six possible LO cycles occurs at the beginning of a symbol. This was resolved by building a histogram of zero crossings in each of the six positions. Each message was preceded by dotting (alternate 0s and 1s) which should give the following distribution of zero crossings:

Position through symbol	Start (a)	1/6 (b)	2/6	3/6	4/6	5/6 (c)
% zero crossings	40	0	20	20	20	0

From this not only was the symbol position established, but the heuristic metric

$$q = (2a - b - c - 2) / (2a)$$

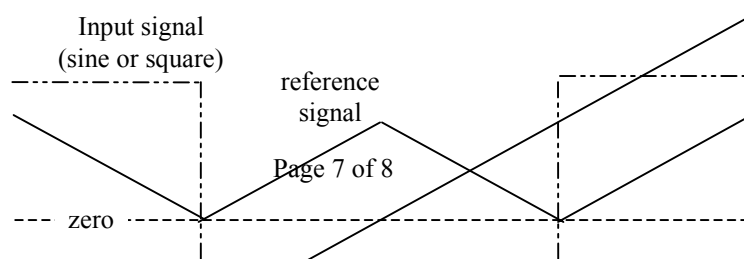
was established to estimate signal quality, where if $q \geq 0.5$ it was worth proceeding with the demodulation process.

The histogram process was adjusted through the message to make it less agile so that it could resolve the symbol ambiguity quickly but then not be disturbed by random noise.

Example 3:

Yet another phase estimation approach was used in a DSP MSK modem, where the symbol frequency was extracted by narrow-band filtering the squared signal.

The PLL counter had its sign discarded, half the peak amplitude subtracted, and the sign discarded again, to give the following reference signal:



The slope of the reference signal is measured on each input signal zero crossing, and this is used to select the sign of a $\pi/32$ radian correction to the ramp. The ramp polarity is also tested, and if a false lock is detected a $\pi/2$ radian correction is applied instead.

Summary

Methods of generating an accurate clock for the extraction of data from a digital radio signal have been presented qualitatively and by example. It should be apparent that the application of DSP has allowed this task to be performed in both traditional and novel ways, bring the twin benefits of precision and versatility to meet increasing performance demands.

Bibliography

Ziemer, R.E. and Peterson, R.L.; "Introduction to Digital Communication", Maxwell 1992, ISBN 0-02-946431-5.

Proakis, G.P.; "Digital Communications, Third Edition", McGraw-Hill 1995, ISBN 0-07-113814-5.

David Spreadbury graduated from The City University in 1971 with a first class honours degree in Electrical and Electronic Engineering. An experienced hardware designer, low level programmer, and systems engineer, he has spent twenty five years "doing DSP", in both military and commercial environments. Most of his projects employ novel algorithms or architectures, some in ASIC, to achieve speed, size, performance or cost targets. David is married, with two teenage children. He is actively involved in his local church and enjoys fixing things, reading, maths, origami, and a strictly limited amount of sport.

Plextek Ltd. is an independent communications technology consultancy, located in the village of Great Chesterford near Cambridge, which specialises in digital telecommunications and mobile radio communications. The principal services offered are product and system development, with hardware, software and ASIC design capabilities, and radio system planning and survey. Plextek also offer industry, product and technology studies and undertake standards and regulatory assignments.