AN ALGORITHM FOR MINIMUM BANDPASS SAMPLING FREQUENCY FOR MULTIPLE RF SIGNALS IN SDR SYSTEM

Satyabrata Sen and Vikram M. Gadre

Department of Electrical Engineering
Indian Institute of Technology, Bombay
Powai, Mumbai, India 400076
e-mail: satyabrata@iitb.ac.in, vmgadre@ee.iitb.ac.in

ABSTRACT

The principal idea behind the design of a software radio is to place the analog-to-digital and digital-to-analog converters as near the antenna as possible, such that most of the radio functionalities can be implemented on a programmable digital signal processor. One way to achieve this is by direct bandpass sampling of the desired RF signal band to baseband frequency. However, the design of a software radio receiver becomes more complicated when two or more distinct RF signals are to be received. The traditional approach for this case has been to bandpass sample a continuous span of spectrum containing all the desired RF signals. The disadvantage with this approach is that the required sampling rate depends upon the span of spectrum, instead of the information bandwidths of the signals. In this paper, we present an efficient algorithm to compute the minimum bandpass sampling frequency for direct downconversion of multiple distinct RF signal bands simultaneously. The novelty of this algorithm is that, at each iteration, it not only checks for a valid sampling frequency, but also determines the next possible minimum value of sampling frequency.

1. INTRODUCTION

The design of a software radio is based on two simple design goals [1]. First, the analog-to-digital converter (ADC) should be placed as near the antenna as possible in the chain of RF front-end components. Second, the resulting samples should be processed *softly* on a reconfigurable digital domain via digital signal processors (DSPs) or field programmable gate arrays (FPGAs) [2]. Ideally, in a software radio different air interfaces can co-exist on a common hardware platform. The desired channel and its corresponding signal processing operations can then be selected and carried out through software programming [3].

Generally, a RF front-end consists of multiple stages of amplification, filtering, and downconversion to process a single RF transmission, whereas in the direct digitization configuration, the RF signal is sampled directly without any downconversion. Hence, the challenge lies in designing these front-end components in such a way that these can operate across a spread of frequencies containing multiple signal bands, as would be the case in an *ideal* software radio.

However, for most radio applications, the required sampling rate for direct downconversion of a RF signal would be impractically high if Nyquist sampling is employed. Detailed specifications of some of the available ADCs can be found in [4], [5]. One alternative could be to include multiple frequency translation stages, however, that would add additional hardware between the antenna and ADC, contrary to the software radio design philosophy. The other alternative is the utilization of bandpass sampling.

Bandpass sampling is a special form of undersampling that translates (or deliberately aliases) a high frequency bandpass signal to baseband frequency [6]. The required sampling frequency depends on the signal bandwidth, rather than on its highest frequency component. The main advantage of this is, therefore, the reduced requirement of the sampling frequency and of the associated signal processing capability.

In this regard, D. M. Akos et al. [2] proposed a method to compute the bandpass sampling frequency for direct downconversion of multiple RF signals. This method is, however, computationally intensive as it requires an exhaustive test of all frequencies up to the Nyquist rate, and at each such frequency we need to check $2N + \binom{N}{2}$ constraint equations. This computational complexity can be alleviated to a certain extent using the graphical approach of N. Wong et al. [7], where the search is restricted within the intersections of the valid ranges of sampling frequencies of individual signal band. Apparently, this modified span of search will reduce with the number of bandpass signals, as the intersection will be lesser. But because of inherent discreteness of any graphical approach, and lack of any analytical formulation, this algorithm may not produce the exact values of the valid frequency ranges. C. H. Tseng et al. [8] proposed another method based on all the possible orders of spectral replicas of the sampled signal. However, the successful implementation of this algorithm lies in determining all the possible orders of spectral replicas, which varies as $N! \times 2^N$ with the number of RF signals. Hence, for large number of RF signals the possible spectral ordering may turn out to be very cumbersome and that could lead to failure of this algorithm.

In this paper, an efficient algorithm is proposed to compute the *minimum* sampling frequency for direct downconversion of multiple distinct RF signal bands. In this method, we start with the theoretical minimum sampling frequency, which is twice the summation of all information bandwidths, and check for its validity. If it is not found to be a valid sampling frequency, then we determine the next minimum value of sampling frequency, based on the given band specifications, and repeat the check for validity. Compared to conventional approaches, this method is superior in the sense that, instead of checking some arbitrary frequencies, it computes the next possible minimum value of sampling frequency at each iteration.

2. PROPOSED APPROACH

In this section, we present an algorithm for determining the *minimum* bandpass sampling frequency for direct downconversion of multiple distinct RF signal bands. For that we consider the bandpass signals $f_i(t)$ $(i = 1, 2, \ldots, N)$, with f_{li}, f_{ui} and B_i representing lower bound, upper bound and bandwidth of signal $f_i(t)$, respectively. We present this algorithm in two different subsections: first we present it for two signal bands, explaining elaborately all the necessary computations, and then extend that for multiple signal bands in the next subsection.

2.1. Algorithm of minimum f_s for two signal bands

The algorithm for two signal bands is as follows

1. Select initial sampling frequency as $f_s = 2(B_1+B_2)$, which is the minimum possible sampling frequency for two bands.

2. Check out whether any integer multiple of the chosen $f_s/2$ falls within any of these two bands. If *yes*, then increase the sampling frequency by Δf_s and repeat this step, *otherwise* move on to next step.

3. Perform bandpass sampling operation of both the bands, with the chosen sampling frequency of step 2.

4. Check out whether the bands overlap over each other in the sampled bandwidth $(0 - f_s/2)$. If *yes*, then again increase the sampling frequency by Δf_s and go back to step 2, *otherwise* the chosen sampling frequency represents one of the valid sampling frequency for direct downconversion of two RF signals.

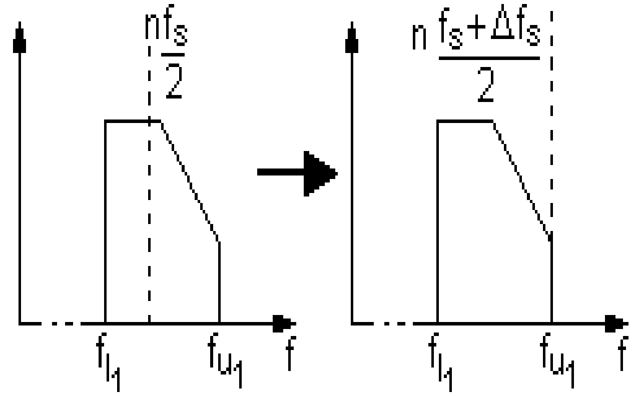

Fig. 1. Choosing Δf_s in step 2 for single band .

The efficiency of this algorithm is solely dependent on the value of Δf_s that needs to be chosen in step 2 and 4. If we choose Δf_s to be a fixed value, i.e., if we uniformly increase the sampling frequency from $2(B_1 + B_2)$, then we may miss the minimum sampling frequency. Hence we need to develop some analytical formulations to specify the values of Δf_s in step 2 and 4.

2.1.1. Determining value of Δf_s for step 2

In step 2, we actually try to find a sampling frequency such that none of the bands alias with itself. To avoid aliasing with itself each individual band needs to satisfy two constraint equations [2, eqn. (2), (3)]. We can combine these two constraints into a simple constraint as

$$rem(f_u, f_s/2) > B \tag{1}$$

If this constraint is not satisfied then one of the integer multiple of the chosen $f_s/2$ will lie within the signal band, i.e., $f_l < nf_s/2 < f_u$, where n is an integer. In that case, we need to increase the sampling frequency to f_s' such that the nth multiple of this new $f_s'/2$ would lie beyond f_u, i.e., $nf_s'/2 \geq f_u$. This logic is schematically depicted in Fig. 1. Hence, the modified sampling frequency $(f_s + \Delta f_s)$ is given as

$$f_s + \Delta f_s \geq \left(\frac{2f_u}{n}\right) \tag{2}$$

For two band case this can be written as

$$f_s + \Delta f_s \geq max\left[\left(\frac{2f_{u1}}{m}\right), \left(\frac{2f_{u2}}{n}\right)\right] \tag{3}$$

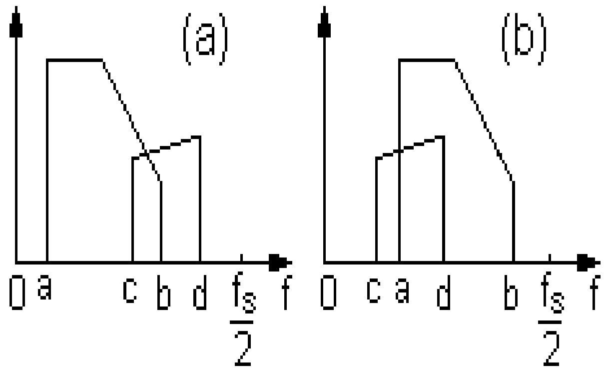

Fig. 2. Band overlap when both m and n are even.

where

$$m = floor\left[f_{u1}/(f_s/2)\right], \ n = floor\left[f_{u2}/(f_s/2)\right] \quad (4)$$

are two integer values. While computing the *minimum* sampling frequency we must use (3) with the equality condition.

2.1.2. Determining value of Δf_s for step 4

In step 4, we try to compute a sampling frequency such that the two bands do not overlap over each other within the sampled bandwidth ($0 - f_s/2$). For that, we need to consider eight different spectral orientations of these two bands, as presented in [8]. However, we will group these orientations in four different subgroups depending on whether m and n are even or odd, where

$$m = floor\left[f_{l1}/(f_s/2)\right], \ n = floor\left[f_{l2}/(f_s/2)\right] \quad (5)$$

When both m and n are even, there are two possible spectral orientations, as shown in Fig. 2. The alias version of signal $f_1(t)$ resides within (a, b) and that of signal $f_2(t)$ within (c, d) in the sampled bandwidth.

For the orientation of Fig. 2(a), let us compute the band overlap Δ, at the chosen sampling frequency f_s, as

$$\begin{aligned} b &= f_{u1} - m\left(\frac{f_s}{2}\right) \\ c &= f_{l2} - n\left(\frac{f_s}{2}\right) \\ \Delta &= b - c \end{aligned}$$

Now let us increase the sampling frequency to $(f_s + \Delta f_s)$, and compute the band overlap Δ', at this new sampling frequency $(f_s + \Delta f_s)$, as

$$\begin{aligned} \Delta' &= b' - c' \\ &= \Delta + (n - m)\left(\frac{\Delta f_s}{2}\right) \end{aligned}$$

Table 1. Expression for $(f_s + \Delta f_s)$ at step 4.

m and n	**Band Overlap** $a < c < b < d$	**Band Overlap** $c < a < d < b$
m even n even	no further improvement	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$
m even n odd	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$	no further improvement
m odd n even	no further improvement	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$
m odd n odd	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$	no further improvement

As from (5) we have $n > m$, and the chosen value Δf_s must always be a positive quantity, we get

$$\Delta' > \Delta \quad (6)$$

Hence band overlap increases for any increase of f_s for this spectrum orientation. Let us denote this condition as 'no further improvement' case.

Similarly, for the band orientation of Fig. 2(b), we compute the corresponding band overlap Δ at the chosen f_s as

$$\begin{aligned} a &= f_{l1} - m\left(\frac{f_s}{2}\right) \\ d &= f_{u2} - n\left(\frac{f_s}{2}\right) \\ \Delta &= d - a \end{aligned}$$

Then we increase the sampling frequency by Δf_s, and compute new band overlap Δ' as

$$\begin{aligned} \Delta' &= d' - a' \\ &= \Delta - (n - m)\left(\frac{\Delta f_s}{2}\right) \end{aligned}$$

To make the band overlap (Δ') zero we need to choose Δf_s as

$$\Delta f_s = \left(\frac{2}{n - m}\right) \Delta \quad (7)$$

Putting the value of Δ, we can write the expression for modified sampling frequency as

$$f_s + \Delta f_s = \left(\frac{2}{n - m}\right) (f_{u2} - f_{l1}) \quad (8)$$

Performing similar analysis for other cases, we get eight different expressions for $(f_s + \Delta f_s)$, all of which are tabulated in Table 1. As mentioned before, the signal $f_1(t)$ resides within (a, b) and the signal $f_2(t)$ within (c, d) in the sampled bandwidth, and the overlap conditions are expressed in terms of (a, b) and (c, d).

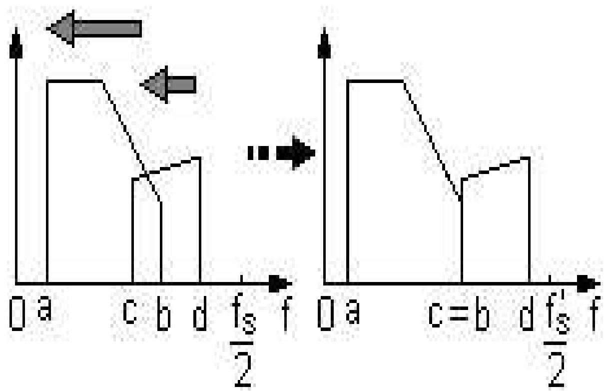

Fig. 3. Changes in band orientations (m even, n even) [Configuration I].

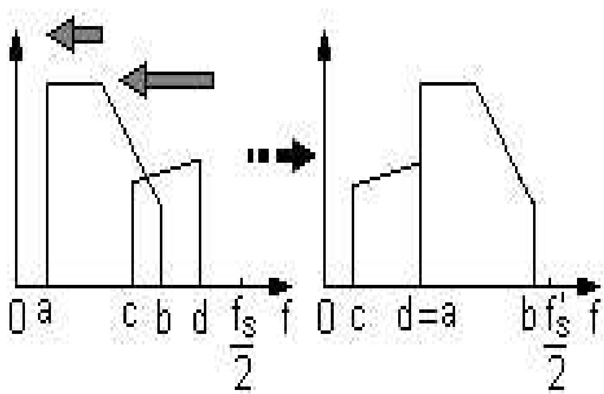

Fig. 4. Changes in band orientations (m even, n even) [Configuration II].

2.1.3. Decision making at 'no further improvement' cases

In this subsection, let us address the issue that when one of such 'no further improvement' situation arises what the algorithm should do. As the goal of this algorithm is to find the *minimum* sampling frequency, it can not take any random decision about the value of Δf_s, even at the 'no further improvement' cases. We discuss these situations, as before, depending on whether m and n, defined by (5), are even or odd.

When both m and n are even, let us re-consider the expressions for the band positions (a, b) and (c, d) within the sampled bandwidth. For any increase in the value of f_s, all the values (a, b) and (c, d) decrease, which is depicted in the Fig. 3, 4. Hence, ultimately we can have two different band orientations having zero band overlap.

However, if we consider the incremental changes in the values of (a, b) and (c, d), due to an incremental change in

Table 2. Modified expression for $(f_s + \Delta f_s)$ at step 4.

m and n	**Band Overlap** $a < c < b < d$	**Band Overlap** $c < a < d < b$
m even n even	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$
m even n odd	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$
m odd n even	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$
m odd n odd	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$

Table 3. Final expression for $(f_s + \Delta f_s)$ at step 4.

m and n	**(n - m)**	**Expression for $(f_s + \Delta f_s)$**
both even or both odd	even	$2\left(\frac{f_{u2}-f_{l1}}{n-m}\right)$
one is even and other is odd	odd	$2\left(\frac{f_{u1}+f_{u2}}{n+m+1}\right)$

sampling frequency (Δf_s), we get

$$\Delta a = \Delta b = -m\left(\frac{\Delta f_s}{2}\right)$$

$$\Delta c = \Delta d = -n\left(\frac{\Delta f_s}{2}\right)$$

Since $n > m$, from (5), therefore $\Delta c,\ \Delta d > \Delta a,\ \Delta b$. This suggests that the rate of decrease of the second band $f_2(t)$ is more than that of the first band $f_1(t)$, and hence the orientation of Fig. 3 is infeasible.

Then considering the band orientation of Fig. 4, we see that we need to choose the next sampling frequency $(f_s + \Delta f_s)$ in such a way that the band positions 'a' and 'd' become the same. Equating the expressions of 'a' and 'd' we get

$$f_{l1} - m\left(\frac{f_s + \Delta f_s}{2}\right) = f_{u2} - n\left(\frac{f_s + \Delta f_s}{2}\right)$$

$$\Rightarrow f_s + \Delta f_s = \left(\frac{2}{n-m}\right)(f_{u2} - f_{l1}) \qquad (9)$$

Performing similar analysis for the other 'no further improvement' cases, as mentioned in Table 1, we get Table 2 with modified (or improved) expressions of $(f_s + \Delta f_s)$. But because of the similarities between 1st and 4th row and between 2nd and 3rd row, we can represent these eight conditions by two simple cases, as presented in Table 3, depending on $(n - m)$.

At each iteration, using these values of $(f_s + \Delta f_s)$ from Table 3, we can specify the next possible minimum sampling frequency. Therefore, at a particular iteration, if the chosen sampling frequency is found to be not valid, then this algorithm can specify the next frequency to be tested for validity. This is not the case with the existing approaches, where any chosen frequency is checked for validity, and if found to be not valid then again some arbitrary frequency is chosen for the next iteration. Therefore, to avoid this ad-hoc search it is better to use the result of Table 3, to know about the next possible minimum frequency.

2.2. Algorithm of minimum f_s for multiple signal bands

Extending the logic of the previous section, we can formulate the algorithm for multiple signal bands as follows

1. Select initial sampling frequency as $f_s = 2\sum_{i=1}^{N} B_i$

2. Check out whether any integer multiple of the chosen $f_s/2$ falls within any of these bands. To do that, check out

$$rem(f_{ui}, f_s/2) \; > \; B_i \tag{10}$$

   for $i = 1, \ldots, N$. If this constraint is satisfied for all i, then move on to next step. Otherwise, compute the next modified sampling frequency as

$$f_s + \Delta f_s \; \geq \; max\left(\frac{2f_{ui}}{m}\right) \tag{11}$$

   where $m = floor\left[f_{ui}/(f_s/2)\right]$, and repeat this step.

3. Perform bandpass sampling operation of all the bands, with the chosen sampling frequency of step 2.

4. Check out whether the bands overlap over each other in the sampled bandwidth $(0 - f_s/2)$. If any two signal bands 'i' and 'j' ($i, j = 1, \ldots, N$ and $i \neq j$) alias within the sampled bandwidth then, using Table 3, compute the next modified sampling frequency for that pair of signal bands (i, j). Let us denote that as $(f_s + \Delta f_s)_{i,j}$. Then overall the next modified sampling frequency can be considered as

$$f_s + \Delta f_s = max\,(f_s + \Delta f_s)_{i,j} \tag{12}$$

   and with this modified frequency go back to step 2. On the other hand, if none of the bands overlap over each other, then the chosen sampling frequency represents the minimum sampling frequency for direct downconversion of these N signal bands.

2.2.1. *Computational complexity of the algorithm*

We can analyze the computational complexity of this algorithm from two aspects: one is the number of constraint equations to be checked out, and the other one is the computation involved in determining the next modified sampling frequency $(f_s + \Delta f_s)$. In step 2, we need to check out (10) for each band separately, i.e., for N signal bands we need to check out N such constraint equations. Further in step 4, to check whether any two bands overlap or not within the sampled bandwidth, we need to check out $\binom{N}{2}$ constraint equations. Hence, at a particular iteration, we need to do $N + \binom{N}{2}$ constraint equations check, instead of $2N + \binom{N}{2}$ checking of [2]. Moreover, if a particular frequency does not satisfy step 2, then the number of constraint equations only restrict to N.

In the calculation of the next modified sampling frequency, both in step 2 and 4, we need to find the maximum out of a set of values. In step 2, we need to find the maximum out of N values, on the assumption that every signal band includes one of the integer multiple of the chosen $f_s/2$. Similarly, in step 4, at worst case, if all bands overlap over each other, then we need to find the maximum out of $\binom{N}{2}$ values. However, practically it is very unlikely that every signal band would include one of the integer multiple of $f_s/2$ and/or all the signal bands would alias with each other within $(0 - f_s/2)$. Hence, actually we have to find the maximum out of a set of values which is much lesser than N and $\binom{N}{2}$, in step 2 and 4, respectively.

3. SIMULATION RESULTS

To get a better feeling of the effectiveness of this algorithm, let us consider a hypothetical situation where we try to incorporate the GSM and IS-95 CDMA standards over a single system. The GSM standard operates over (890 - 915) / (935 - 960) MHz [9], whereas the IS-95 CDMA operates over (824 - 849) / (869 - 894) MHz [10]. We applied this algorithm for both uplink and downlink bands separately, and the results are summarized in Table 4. We extended this analysis to three and four bands cases, with different combinations of GSM and CDMA up- and downlink bands. These results are also included in Table 4.

4. CONCLUSIONS

While expanding the digital signal processing boundary toward the antenna, in a software radio implementation, the application of bandpass sampling technique can be very helpful to achieve the design goal. In this paper, we have presented an algorithm to determine the minimum bandpass sampling frequency for direct downconversion of multiple distinct RF signals. The essence lies in the determination of

Table 4. Minimum sampling frequency for multiple signal bands.

No. of bands	Frequency ranges (in MHz)	Minimum sampling frequency (in MHz)
2	824 - 849 890 - 915	117.6
	869 - 894 935 - 960	120.0
3	824 - 849 869 - 894 935 - 960	206.0
	824 - 849 890 - 915 935 - 960	213.3
4	824 - 849 869 - 890 890 - 915 935 - 960	274.3

a sampling frequency closest to the theoretical lower limit $[2\sum_{i=1}^{N} B_i]$. This minimization, indirectly, is also critical in estimating the computational requirements, one of the primary bottlenecks in software radio design.

Apart from this bandpass sampling process, the initial stages of digital signal processing, namely the digital downconversion, the sample rate adaptation, the dispreading and channelization, are very crucial functionalities in a software radio design, since these have to be performed at relatively high data rate. Moreover, as we try to get closer to the ideal SDR, many advanced but complex signal processing techniques, such as multipath minimization, digital beamsteering, space-time adaptive processing etc., also become critical. So a careful consideration over these factors, along with the issues like power consumption, software complexity, cost of initial deployment etc., are required for the promising development of SDR technology.

5. REFERENCES

[1] J. Mitola, "The Software Radio Architecture," *IEEE Communications Magazine*, pp. 26–38, May 1995.

[2] D. M. Akos and J. B. Y. Tsui, "Direct Bandpass Sampling of Multiple Distinct RF Signals," *IEEE Transactions on Communications*, vol. 47, no. 7, pp. 983–988, July 1999.

[3] T. Hentschel, M. Hanker, and G. Fettweis, "The Digital Front-End of Software Radio Terminals," *IEEE Personal Communications*, pp. 40–46, Aug. 1999.

[4] R. H. Walden, "Performance Trends for Analog-to-Digital Converters," *IEEE Communications Magazine*, pp. 96–101, Feb. 1999.

[5] ——, "Analog-to-Digital Converter Survey and Analysis," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 539–550, Apr. 1999.

[6] R. G. Vaughan, N. L. Scott, and D. R. White, "The Theory of Bandpass Sampling," *IEEE Transactions on Signal Processing*, vol. 39, no. 9, pp. 1973–1984, Sept. 1991.

[7] N. Wong and T. S. Ng, "An Efficient Algorithm for Downconverting Multiple Bandpass Signals Using Bandpass Sampling," in *IEEE International Symposium on Circuits and Systems*, vol. 3, June 2001, pp. 910–914.

[8] C. H. Tseng and S. C. Chou, "Direct Downconversion of Multiple RF Signals Using Bandpass Sampling," in *IEEE International Conference on Communications, 2003*, vol. 3, May 2003, pp. 2003–2007.

[9] A. Mehrotra, *GSM System Engineering*. Boston and London: Artech House Inc., 1997.

[10] V. K. Garg, *IS-95 CDMA and cdma2000 Cellular/PCS Systems Implementation*. Asia: Pearson Education, 2002.