where $[x]$ represents the maximum integer number no greater than x.

If $M > N$, from theorem 1, $s_{M,j} = s_{N,j}$ for $j = 1, 2, \ldots, N-1$, and from eqn. 2, $x_M^T x_M = x_N^T x_N = 1$, we have

$$\sum_{j=1}^{N} s_{M,j}^2 = \sum_{j=1}^{N} s_{N,j}^2$$

Thus $s_{M,j}^2 = s_{N,j}^2$. Obviously, only when $s_{M,N} = -s_{N,N}$, $x_M \neq x_N$. This means that $M \leq N + 1$. If $M = N + 1$, it must satisfy eqn. 2, $x_{N+1}^T x_N = p$, or,

$$\sum_{j=1}^{N} s_{N,j} s_{N+1,j} = p$$

Using theorem 1 and eqns. 10 and 11 we have

$$\sum_{j=1}^{N-1} \left(\frac{p\sqrt{(1-p)}}{\sqrt{\{[1+(j-1)p] \cdot [1+(j-2)p]\}}} \right)^2 - \left[\sqrt{(1-p)} \cdot \sqrt{\left(\frac{1+(N-1)p}{1+(N-2)p} \right)} \right]^2 = p$$

Simplifying the above equation we obtain

$$1 + N \cdot p = 0 \quad \text{or} \quad p = -\frac{1}{N}$$

This means that only when $p = -(1/N)$ is the maximum number of vectors in the generated vector set greater than N, i.e. equal to $N + 1$. From eqn. 3 it is easy to prove that the first N vectors are linearly independent. Summarising the above results we obtain the following theorem:

Theorem 3: Given an N-dimensional orthonormal vector set $\{c_1, c_2, \ldots, c_N\}$, a normalised complete vector set $\{x_1, x_2, \ldots, x_M\}$ with the given percentage of correlation p, can be generated according to eqn. 3 and the maximum number of vectors M satisfies

$$M = \begin{cases} 1 & p = 1 \\ N & -\frac{1}{N} < p < 1 \\ \left[1 - \frac{1}{p}\right] & -1 \leq p \leq -\frac{1}{N} \end{cases}$$

From theorem 3 it is known that only when $-(1/N) \leq p < 1$ does the constructed vector set have N linearly independent vectors, and it can be used as a base to represent any N dimensional signals.

Conclusions: We have proposed a new algorithm which can generate a complete vector set in which any two vectors have a given percentage of correlation. Under the special condition the constructed vector set can be used as a nonorthogonal base which has many applications in signal representation. If the constructed vectors are used as row vectors of an $N \times N$ matrix, this matrix can also be used as a nonorthogonal transform which has many applications in signal processing, image compression, and other areas. Applications will be discussed in detail in other papers.

14th September 1992

X. Yu, J. Wang, N. K. Loh, G. A. Jullien and W. C. Miller (*Department of Electrical Engineering, University of Windsor, Windsor, Ontario, Canada N9B 3P4*)

References

1 GENOSSAR, T., and PORAT, M.: 'Optimal bi-orthonormal approximation of signals', *IEEE Trans.*, 1992, **SMC-22,** (3), pp. 449–460

2 GABOR, D.: 'Theory of communications', *J. Inst. Elect. Engr.*, 1946, pp. 429–459

3 EINZIGER, P. D., RAZ, S., and SHAPIRA, M.: 'The Gabor representation and aperture theory', *J. Opt. Soc. Am. A.*, 1986, **3,** (4), pp. 508–522

4 FRIEDLANDER, B., and PORAT, B.: 'Detection of transient signals by the Gabor representation', *IEEE Trans.*, 1989, **ASSP-37**

5 BEHAR, J., PORAT, B., and ZEEVI, Y. Y.: 'Image reconstruction from localized phase', *IEEE Trans.*, 1992, **SP-46,** (4), pp. 736–743

6 PORAT, M., and ZEEVI, Y. Y.: 'The generalized Gabor scheme in biological and machine vision', *IEEE Trans.*, 1988, **PAMI-10,** (4), pp. 452–468

EFFICIENT SPREADING CODE ASSIGNMENT ALGORITHM FOR PACKET RADIO NETWORKS

K.-W. Hung and T.-S. Yum

Indexing terms: Radio communication, Packet radio networks

Multihop spread spectrum packet radio networks with a large number of stations would require a large number of spreading codes, even with code reuse beyond the interference range. In the Letter, the code assignment problem is related to the graph colouring problem and a very efficient algorithm is proposed for assigning codes to the stations. A very tight lower bound on the number of codes needed is also derived.

Introduction: Spread spectrum signalling opens up a new dimension for protocol design and performance tradeoff in packet radio networks (PRNs). When receiver-based or transmitter-based spreading code protocols are used, a multihop PRN with a large number of stations requires a large number of codes and hence a large channel bandwidth. Because a spreading code assigned to a station needs to be unique only to its neighbours, the codes could be reused by the stations which are farther apart.

It is important to find an efficient algorithm for assigning as few codes to the PRN stations as possible because the smaller the number of codes used the smaller the bandwidth needed. In this Letter we first transform the code assignment problem to the familiar graph colouring problem. This allows us to consider the possible use of the graph colouring algorithms for assigning codes in PRNs. We then design a heuristic code assignment algorithm making use of some special properties of PRNs. We also obtain a lower bound on the chromatic number, which in our case is the minimum number of codes required. Finally the performance of this new algorithm is assessed by making comparisons to the bound as well as to one of the best heuristics for graph colouring.

Code assignment and graph colouring: The colouring of a graph G means assigning colours to the vertices of G so that adjacent vertices have different colours. The chromatic number of a graph is defined as the miniumum number of colours needed to colour the graph. There are numerous papers on graph colouring. It was shown that this problem is NP-complete in the sense of Karp [1]. Therefore to solve large size graph colouring problems, many heuristics are proposed. One particularly good heuristic is the degree saturation or Dsatur algorithm [2]. To measure the performance of the heuristics, various methods have been proposed to estimate the chromatic number of graphs [3, 4].

The code assignment problem in PRNs is as follows. Let there be N fixed stations in a packet radio network. Each station is assigned a code which is unique only in its neighbourhood covering all one-hop and two-hop neighbours. Beyond this neighbouring area, which we call the local range, they can be reused.

To relate the code assignment problem to the graph colouring problem, the network structure must first be represented by a graph. The stations in a multihop PRN are treated as the vertices and an edge is formed between two vertices when the two stations are neighbours. Because stations separated by two hops also cannot be assigned to the same code, edges

between all vertices which are two hops apart are added to reflect this requirement. The code assignment problem then becomes the colouring problem of the graph formed.

In graph theory the degree of a vertex is defined as the number of edges incident on that vertex. In PRNs we refer to the degree of a station as the number of stations within that station's local range. Let C denote the chromatic number of this network (i.e. the minimum number of codes needed). Let $k_1, k_2, \ldots, k_N$ be the number of neighbours of station 1, 2, ..., N; then $k = \max(k_1, k_2, \ldots, k_N)$ is the size of the largest neighbouring group.

All neighbours of a particular station, say A, must be within the local range of each other. Hence they cannot use the same code. Therefore the number of codes needed must be no smaller than the size of the largest neighbouring group plus one (i.e. station A itself). Hence a lower bound on the chromatic number C, denoted as C_L, is $k + 1$. This bound is very useful in solving the assignment problem.

Algorithm description: In designing the code assignment algorithm, the following criteria were followed:

(1) When building a K colouring of a graph, we can ignore all vertices of degree less than K, because once the other vertices are coloured, there will always be at least one colour available for each of these vertices.

(2) The lower bound of the chromatic number obtained in the preceding Section is a good starting point.

(3) When there is more than one available code, we choose the code which gives the minimum binding in assigning codes to other stations.

Let c be the total number of codes currently used. When a station, say station A, is assigned to code x, this assignment will affect the assignment freedom of the set of stations in station A's local range and might increase the number of codes needed. For the set of stations within station A's local-range let n_i, $i = 0, 1, \ldots, c-1$, be the number of stations with i codes to choose from after A chooses code x and let $D(A)$ be the degree of station A. The degree of freedom left for the assignment of the remaining codes can be measured by the binding function F defined as follows:

$$F(A, x) = \sum_{i=0}^{c-1} n_i D(A)^{-i}$$

It is easy to see that the smaller the $F(A, x)$, the larger the freedom of assigning codes to the stations within A's local range and hence the smaller the number of additional codes required.

The code assignment algorithm is as follows:

(1) Find the station with maximum number of neighbours and denote it as station S^*. Let the number of neighbours of station S^* be k. Assign code 1 to station S^*. Assign code 2 to code $k + 1$ to the neighbours of station S^*. Let $c = k + 1$ where c is the total number of codes currently used.

(2) Rank all the stations in order of decreasing degrees and denote them as $S_1, S_3, \ldots, S_N$ where N is the number of stations in the network.

(3) $j := 1$.

(4) If station S_j is assigned, go to (7).

(5) If $D(S_j) < c$, go to (7).

(6) Find the codes available to S_j. (Find the codes which are not used by the stations in the local range of S_j).

(*a*) If no code is available then $c := c + 1$ and assign code c to S_j.

(*b*) If only code is available, assign that code to S_j.

(*c*) If more than one code is available, choose the code with the minimum F value and assign it to S_j.

(7) If $j < N$ then $j := j + 1$ and go to (4).

(8) To assign codes to the remaining stations, repeat steps (3) to (7) but skip step (5). (This second loop is for assigning codes to stations with degree less than c.)

Results and discussion: Many random networks are generated to compare our code assignment algorithm with the Dsatur algorithm [2] (which is similar to our algorithm but without steps (1) and (6c)) and the lower bound on chromatic number. The stations in the networks are randomly located within a $20 \times 20\,\text{km}^2$ region. The transmission range is 4 km. The number of stations in the network ranges from 40 to 160. For each case 50 random distributions are generated. Table 1 summarises the assignment results.

Our code assignment algorithm runs very quickly. For a 160 station network the assignment is completed within a few seconds using a PC/AT. Out of a total of 200 cases, there are 90 cases where our code assignment algorithm requires fewer codes than the Dsatur algorithm but only 6 cases where our code assignment algorithm requires more. When using our code assignment algorithm the average numbers of codes needed are 9·32, 16·32, 23·48 and 29·60 for the 40, 80, 120 and 160 station network the assignment is completed within a few seconds using a PC/AT. Out of a total of 200 cases, there are 90 cases where our code assignment algorithm requires fewer codes than the Dsatur algorithm but only 6 cases where our the time. This means that our code assignment algorithm gives the optimal result at least 88% of the time. When the number of stations is increased to 160, there are still 30% of cases reaching the lower bound and the average additional codes required is only 1·22. This therefore shows that both the code assignment algorithm and the bound on chromatic number are very good indeed.

1st October 1992

K.-W. Hung (*Department of Computing, Hong Kong Polytechnic, Hung Hom, Kowloon, Hong Kong*)

T.-S. Yum (*Department of Information Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong*)

Table 1 CODE ASSIGNMENT RESULTS

Network type[1]				
Network parameters	1	2	3	4
Number of stations N	40	80	120	160
Maximum size of neighbouring group k[2]	8·20	14·90	21·90	27·38
Maximum degree D_{max}[2]	15·02	33·28	53·26	71·04
Average code size[2]	9·32	16·32	23·48	29·60
Number of cases reaching lower bound	44	32	26	15
Maximum additional codes above lower bound	1	2	2	4
Number of cases with smaller code size than Dsatur algorithm	2	21	30	37
Number of cases with larger code size than Dsatur algorithm	0	1	1	4
Maximum number of codes saved compared to Dsatur algorithm	1	2	3	4

[1] 50 sample networks were generated for each network type

[2] averaged over the 50 samples

References

1 KARP, R. M.: 'Reducibility among combinatorial problems', *in* '*Complexity of computer computations*' Plenum Press, 1972, pp. 85–103

2 BRELAZ, D.: 'New methods to color the vertices of a graph'. *Comm. ACM*, 1979, **22**, (4), pp. 251–256

3 MITCHEM, J.: 'On various algorithms for estimating the chromatic number of a graph', *The Computer Journal*, 1976, **19**, (2), pp. 182–183

4 MCDIARMID, C.: 'Determining the chromatic number of a graph', *SIAM J. Comput.*, 1979, **8**, (1), pp. 1–14

PERFORMANCE EVALUATION OF CMOS COMPATIBLE BIPOLAR TRANSISTORS AND VERTICAL JUNCTION FETs FOR ADVANCED VLSI TECHNOLOGY

A. Nouailhat, M. Mouis, A. Marty, A. Granier, N. Degors, J. Kirtsch and A. Chantre

Indexing terms: Bipolar transistors, Field-effect transistors, Large-scale integration

A submicrometre CMOS technology, with selfaligned silicide (salicide) and a high energy implanted retrograde well, has been used to develop bipolar transistors and vertical junction FET devices based on the same *p*MOS selfaligned structure. Good performance devices have been obtained, with no need for buried layers and epitaxy, allowing low-cost multidevice integration.

Introduction: The trend in VLSI is for multidevice integration on-chip. For instance, BiCMOS technologies incorporate bipolar transistors where high speed and large currents are required. The vertical JFET is another very interesting device which could be usefully integrated for its very low noise level and high power driving capability [1]. In contrast with most CMOS technologies, bipolar technologies generally need epitaxy on an n^+ buried layer in order to obtain low collector resistance and capacitance. The cost of mixing these two technologies is therefore high. However, owing to device shrinkage and the emerging use of retrograde wells [2], both technologies can be made fully compatible. The same arguments are valid for the JFET. In this Letter we describe a very attractive family of devices based on the same *p*MOS selfaligned structure, and which can therefore be easily integrated. Our devices are processed in a submicrometre CMOS technology, with high energy implantation for retrograde well formation and titanium salicidation. We show that almost optimal performance can be obtained without changing the basic CMOS process.

Process integration: As shown in Fig. 1, the bipolar transistors are classically derived from the *p*MOS structure by suppression of the gate oxide and addition of a base implantation [3]. However, the usual collector buried layer is replaced here by a retrograde well similar to the *p*MOS well. Vertical junction FETs (v-JFETs) are very simply obtained by skipping the additional base implant (Fig. 1). This new JFET structure takes advantage of the submicrometre lithography (highly doped narrow channel, large transconductance) and of the salicidation of the gate and source regions (reduced sheet resistance). For the two types of device, we compared a standard *p*MOS well (implant dose: 7×10^{12} P/cm^2, 900 keV) and a less resistive well obtained with a higher implant dose (7×10^{13} P/cm^2) and a rapid thermal anneal which gives almost the same doping concentration near the silicon surface (1130°C for 20 s instead of 1050°C for 150 min for the standard well).

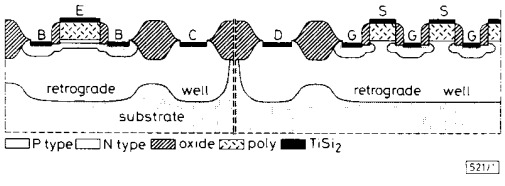

Fig. 1 *Schematic view of bipolar and JFET structures derived from pMOS structure*

Bipolar transistor characteristics: The first noticeable result was that the electrical characteristics were unaffected by the high-energy implantation of the retrograde well. Devices showed near-ideal Gummel plots and normal behaviour when stressed under reverse E/B voltage. The aging of the transistors with a *p*MOS well or higher implant dose was similar. The device geometry was designed so as to minimise the collector resistance R_C and C/B capacitance C_{jC}. Table 1 summarises the results of DC and high frequency *s*-parameter measurements for the two collector wells. The improvement in the cutoff frequency f_T for the highly doped well was due to the reduction in the forward transit time (narrower base due to the highly doped collector) and in the $R_C C_{jC}$ time constant. However, devices using the standard *p*MOS well showed a similar f_{max} and ~60% shorter ECL gate delays (Fig. 2) due to a smaller C_{jC}. A gate delay of 100 ps can be obtained for a logic swign of 0.8 V and a CmA collector current (corresponding to the maximum in f_T). By comparison with the literature [4], we are able to estimate that the loss of performance due to the *p*MOS well in comparison with the n^+ buried layer and epitaxy is 30% [5].

Table 1 STATIC AND HIGH-FREQUENCY CHARACTERISTICS OF 1·1 × 11·3 (× 2) μm^2 EMITTER SIZE DEVICE FOR V_{ce} = 3 V

	pMOS	Highly doped well
Current gain	83	108
BV_{ceo} (V)	8	4·5
BV_{cbo} (V)	16	9
BV_{ebo} (V)	6·5	6·5
C_{je} (fF)	49	49
C_{jc} (F)	71	130
R_c (Ω)	110	23
max f_T (GHz)	6·6	10·3
f_{max} (GHz)	7·8	7·9

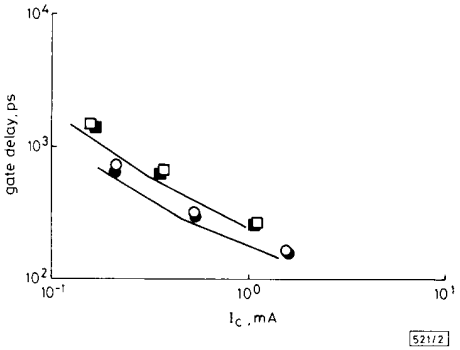

Fig. 2 *Gate delay of 7 stage ring oscillators (with 1·1 × 11·3 (× 2) μm^2 emitter size devices) against switching current, for logic swing of 0·8 V*

Highly doped collector
□ measurement
■ simulation
*p*MOS well
○ measurement
● simulation