ON THE POSSIBILITY OF DESIGN AND THE ALGORITHMICS FOR THE FAST VLSI-RANDOMNESS SOURCES

C. GLOWACZ
DEPT. OF COMPUTER SCIENCE
UNIVERSITY OF BONN

M. KARPINSKI*
DEPT. OF COMPUTER SCIENCE
UNIVERSITY OF BONN

H.T. VIERHAUS
GMD RESEARCH INSTITUTE
/E.I.S PROJECT
5205 ST. AUGUSTIN 1

SUMMARY

1. Design of Fast Randomized Algorithms.

The latest developments in the design of fast parallel algorithms lead to the fundamental new insight into the growing importance of randomness (cf. [F 79], [KUW 85], [MVV 87], [MS 82], [Fü 87]) as the computational resource. The randomization makes some algorithmic solutions possible for problems for which no deterministic solutions are known or possible. The randomization not only becomes widely applicable in combinatorial optimization algorithms but also in algorithmic coding theory, computer algebra, number theory ([AH 87]), encryption techniques, protocols, smart-card design, etc. All these applications assume the availability of a source of randomness in form of the bit sequence random generators working as rapidly as needed. These generators must not only be fast but also *random* (not pseudo-random, although sometimes pseudo-randomness is enough, cf. [B 87]), i.e. capable of producing unbiased and totally unpredictable bit sequences. The existence of such sources is an open question even if one accepts the modern Physics postulate on the existence of 'truly random' elementary processes. The model of imperfect but available physical sources of randomness, still good enough for all algorithmic applications, was recently developed in a series of papers by Vazirani and Santha [V 85], [SV 86], [V 87]. We base our development on models of slightly-random ([V 85]) and semi-random sources ([SV 86]).

*Supported in part by Leibniz Center for Research in Computer Science and the DFG Grant KA 673/2-1

We apply the phase-jitter of a free-running oscillator connected via an EXOR tree structure, and subsequently apply decoding techniques from the theory of BCH codes. The design is just being applied for the randomized combinatorial optimization chip (joint project of the University of Bonn and GMD-E.I.S.), which comprises the randomized MAXIMUM MATCHING subroutine. The details will be described in the full paper.

2. VLSI-Source of Randomness.

The conventional sources of pseudo-random numbers are not always suitable for algorithmic or cryptographic applications. In the number of applications the generation of "truly random" seeds of random numbers occurs to be necessary and the pseudo-random bit sequence generators ([BM 82]) are not a proper substitution for it (cf. [RS 85]). As proposed by [FMC 85], the phase-jitter of a free-running oscillator can be used for random number generation. This basic approach was considerably modified using a relaxation-type oscillator at 100 MHz (see Figure 1). It can be shown that the exploitable phase jitter from such an oscillator exceeds that from a feedback-gate oscillator by far at comparable demands in silicon area (see Figure 2). Analog sources were also considered and showed inferior performance due to limitations of available technology with respect to power dissipation and gain-bandwidth product.

As shifts of MOS threshold voltages will cause an inescapable bias in the oscillator output, an equal probability of 0 and 1 values over a long time requires further processing. This can be done by connecting several independent sources via an EXOR tree structure. We used a more area economical approach devised by Vazirani [V 87] instead and were able to prove its correctness for the first time. The main method we apply is on-chip implementation of BCH codes: Thus 127 independent oscillators were separately implemented on one chip using analog design techniques for optimum isolation and VDD/VSS decoupling.

The chip design (see Figure 3) finally yields 8 independent sources of truly random bits with negligible algorithmic bias. The pattern rate is controllable from the outside. True randomness is expected up to pattern rates of 100 KB per second, but the basic circuit design allows for an exploitation of rates up to 2 MB per second. By comparison, the AT&T LSI chip [FMC 85] works with 4 B per second rates, which gives our design speed-up of order 25 000. As an illustration in usual algorithmic applications, one needs a number of bits proportional to the square of the size of the input.

The chip was designed using 2.5 micron n-well CMOS technology (of Fraunhofer-Institut IMS, Duisburg, Germany) and contains about 9000 transistors using 18 smm of silicon area (see Figure 4). This includes an overhead for interfacing not necessary if the design's kernel is used in further applications where the random number generation and use of random bits take place on the same chip. Optimum testability for CMOS specific faults was implemented using novel scan-path elements with stuck-open testability (see Figure 5) (cf. also [Vi 87], [GMKMV 88]). The chip seems to be the first VLSI-design of this type for the fast, parallel, and 'provably good' on-chip random bit generation. It may find main applications in the design of fast random number generators for the randomized VLSI algorithms and potentially also in cryptography.

References

[AH 87] Adleman, L., and Huang, M., *Recognizing Primes in Random Polynomial Time*, Proc. 19 th ACM STOC (1987), pp. 462-469

[AM 83] Abidi, A.A., and Meyer, R.G., *Noise in Relaxation Oscillators*, IEEE J. of Solid-State Circuits, SC-18 (1983)

[B 87] Bach, E., *Realistic Analysis of Some Randomized Algorithms*, Proc. 19th ACM STOC (1987), pp. 453-461

[Bl 84] Blum, M., *Independent Unbiased Coin Flips from a Correlated Biased Source: a Finite State Mackoc Chain*, Proc. 25th IEEE FOCS (1984), pp. 425-433

[BM 82] Blum, M., and Micali, S., *How to Generate Cryptographically Sequences of Pseudo-Random Bits*, Proc. 23th IEEE FOCS (1982), pp. 112-117

[F 79] Freivalds, R., *Fast Probabilistic Algorithms*, Proc. MFCS '79, Springer LNCS 74 (1979), pp. 57-69

[Fü 87] Fürer, M., *The Power of Randomness for Communication Complexity*, Proc. 19th ACM STOC (1987), pp. 178-181

[GMKMV 88] Glovacz, C., Hübner, U., Krügel-Sprengel, B., Matthäus, C., and Vierhaus, H.T., *CMOS Fault Behaviour, Test Pattern Generation and Design for Testability*, submitted to EUROMICRO '88, Zürich

[K 60] Kleinrock, L., *A Program for Testing Sequences of Random Numbers*, MIT Lincoln Laboratory Report 51 G-0018 (1960), pp. 1-27

[KUW 85] Karp, R.M., Upfal, E., and Wigderson, A., *Constructing a Perfect Matching Is in Random NC*, Proc. 17th ACM STOC (1985), pp. 22-32

[M 72] Maddocks, R.S. et.al, *A Compact and Accurate Generator for Truly Random Binary Digits*, J. of Physics E: Scientific Instruments 5 (1972), pp 542-544

[MS 82] Mehlhorn, K., and Schmidt, E.M., *Las Vegas is Better than Determinism in VLSI and Distributed Computing*, Proc. 14 th ACM STOC (1982), pp 330-337

[MVV 87] Mulmuley, K., Vazirani, U.V., and Vazirani, V.V., *Matching Is as Easy as Matrix Inversion*, Proc. 19th ACM STOC (1987), pp. 345-354

[FMC 85] Fairfield, R.C., Martenson, R.L., and Coulthart, K.B., *An LSI Random Number Generator (RNG)*, in: *Advances in Cryptology - CRYPTO 84. Proceedings*, Springer LNCS, Vol. 196 (1985), pp. 203-230

[RS 83] Rivest, L.R., and Sherman, A.T., *Randomized Encrytion Techniques*, Research Report MIT/LCS/TM-234 (1983), pp. 1-20

[SV 86] Santha, M., and Vazirani, U.V., *Generating Quasi-Random Sequences from Semi-Random Sources*, J. Computer Systems and Sciences 33 (1986), pp. 75-87

[V 85] Vazirani, U.V., *Towards a Strong Communication Complexity Theory or Generating Quasi-Random Sequences from Two Communicating Slightly-Random Sources*, Proc. 17^{th} ACM STOC (1985), pp. 366-378

[V 87] Vazirani, U.V., *Efficiency Considerations in Using Semi-Random Sources*, Proc. 19^{th} ACM STOC (1987), pp. 160-168

[Vi 87] Vierhaus, H.T., *Rule-Based Design for Testability, the EXTEST Approach*. Proc. Compeuro, Hamburg 1987

[W 75] Wyner, A.D., *The Wire-Tap Channel*, Bell System Techn. Journal 54 (1975), pp. 1355-1387

Figure 1: Relaxation Oscillator: Circuit-Diagram

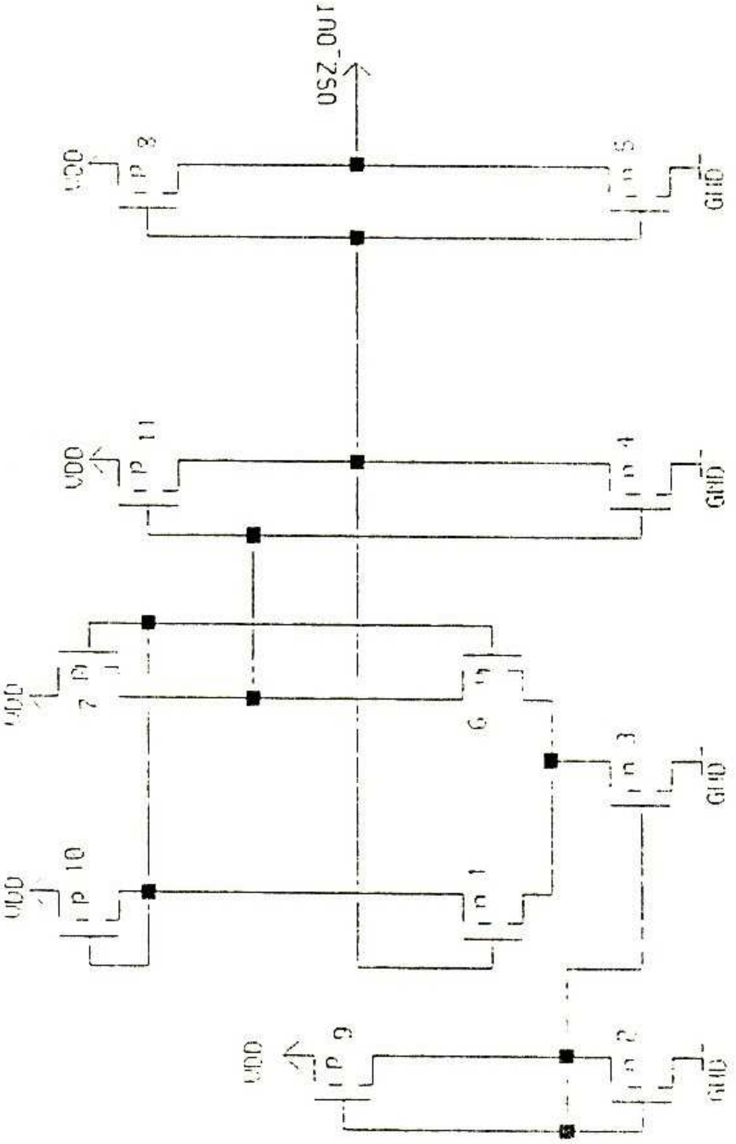


Figure 2: Relaxation Oscillator Layout

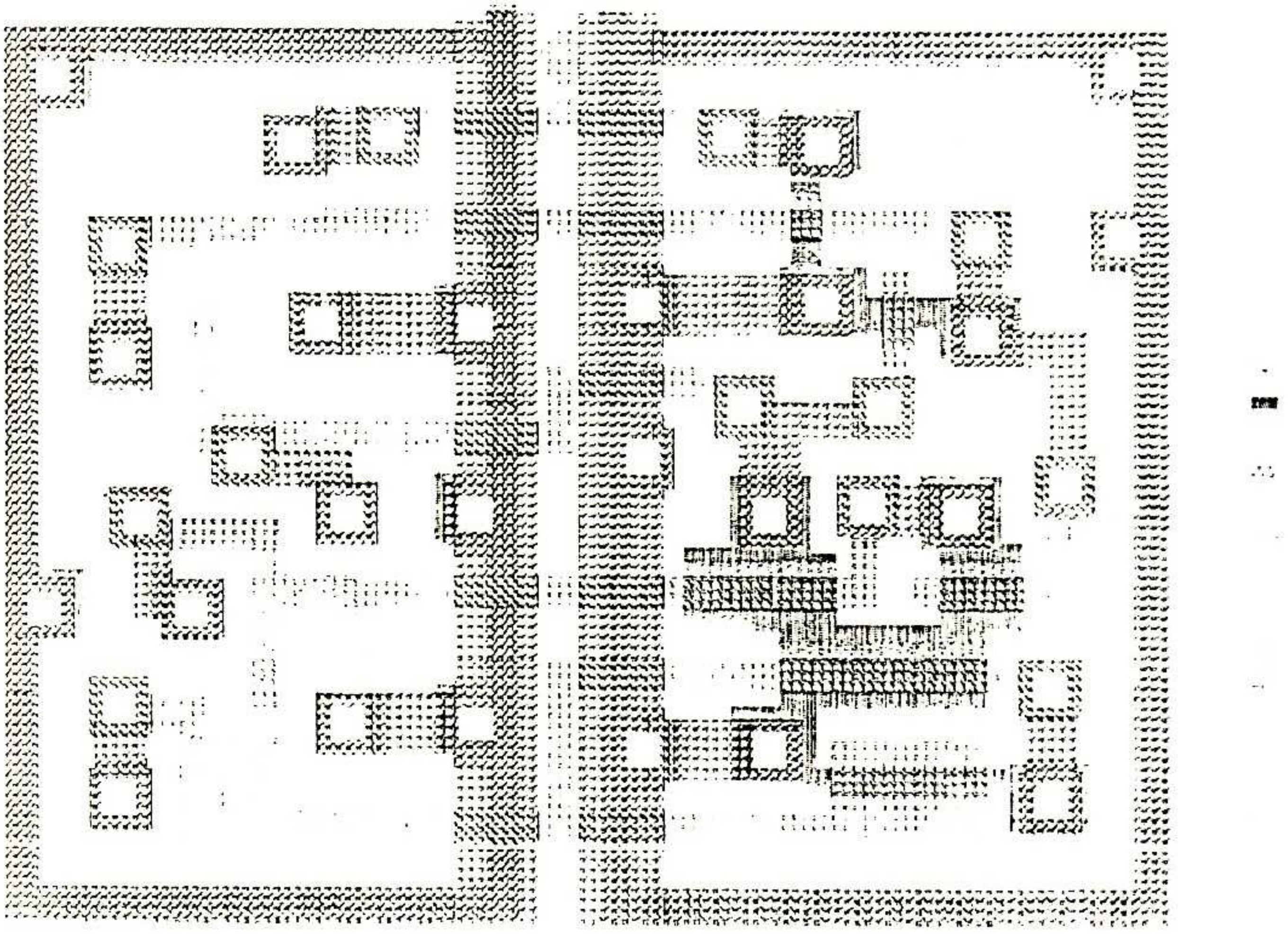


Figure 3: Structure of the Chip

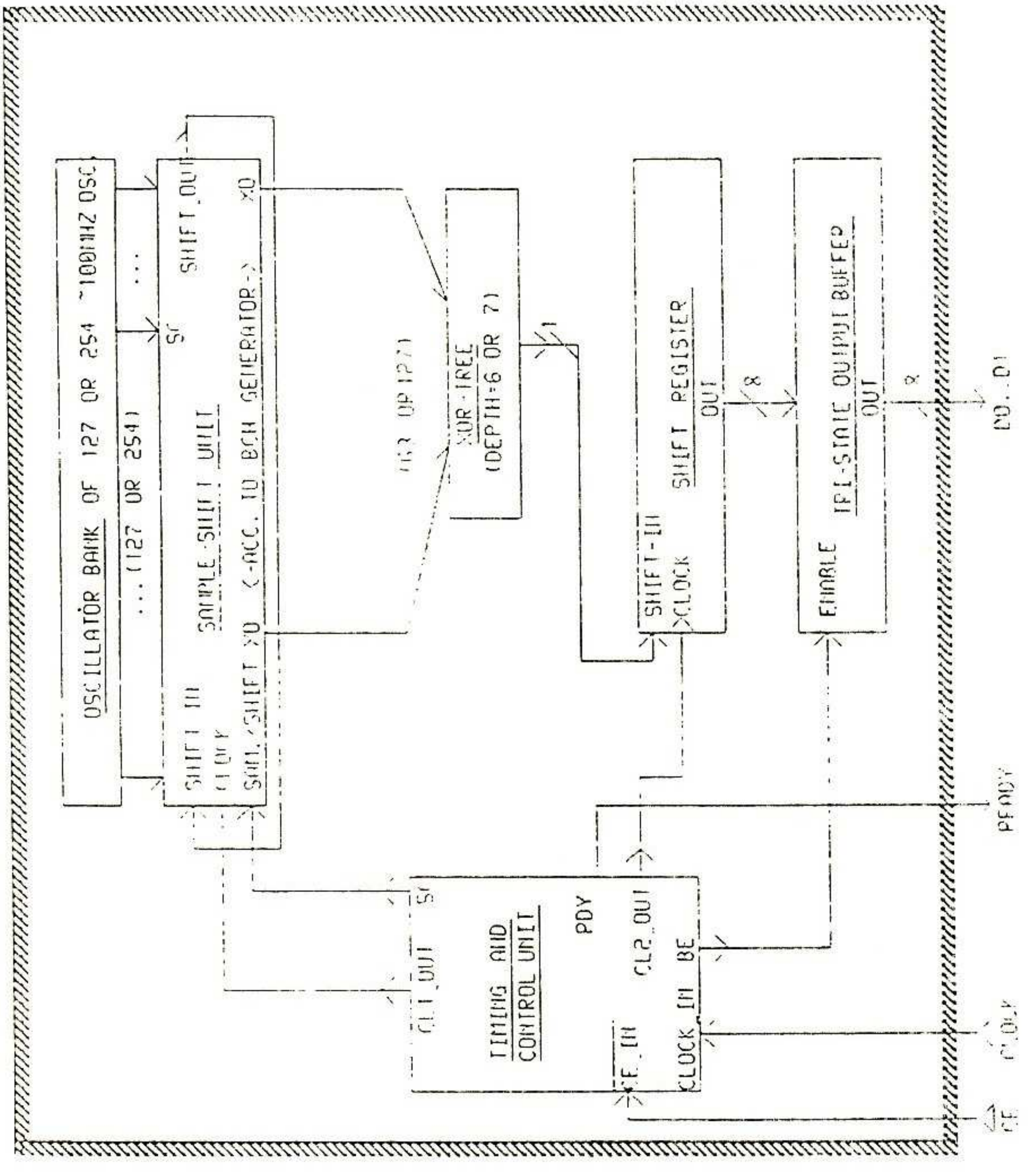


Figure 4: Layout of the Chip

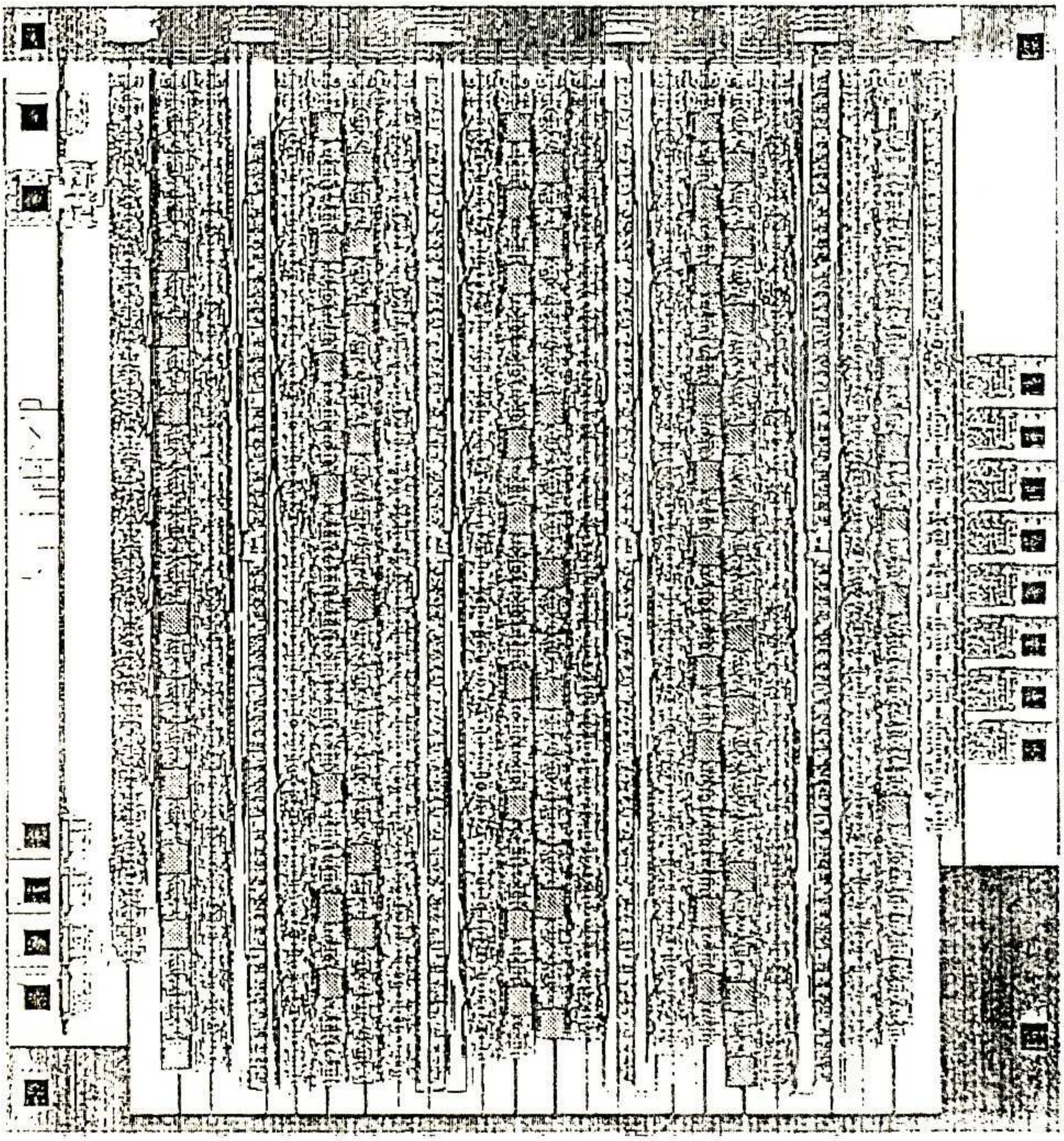
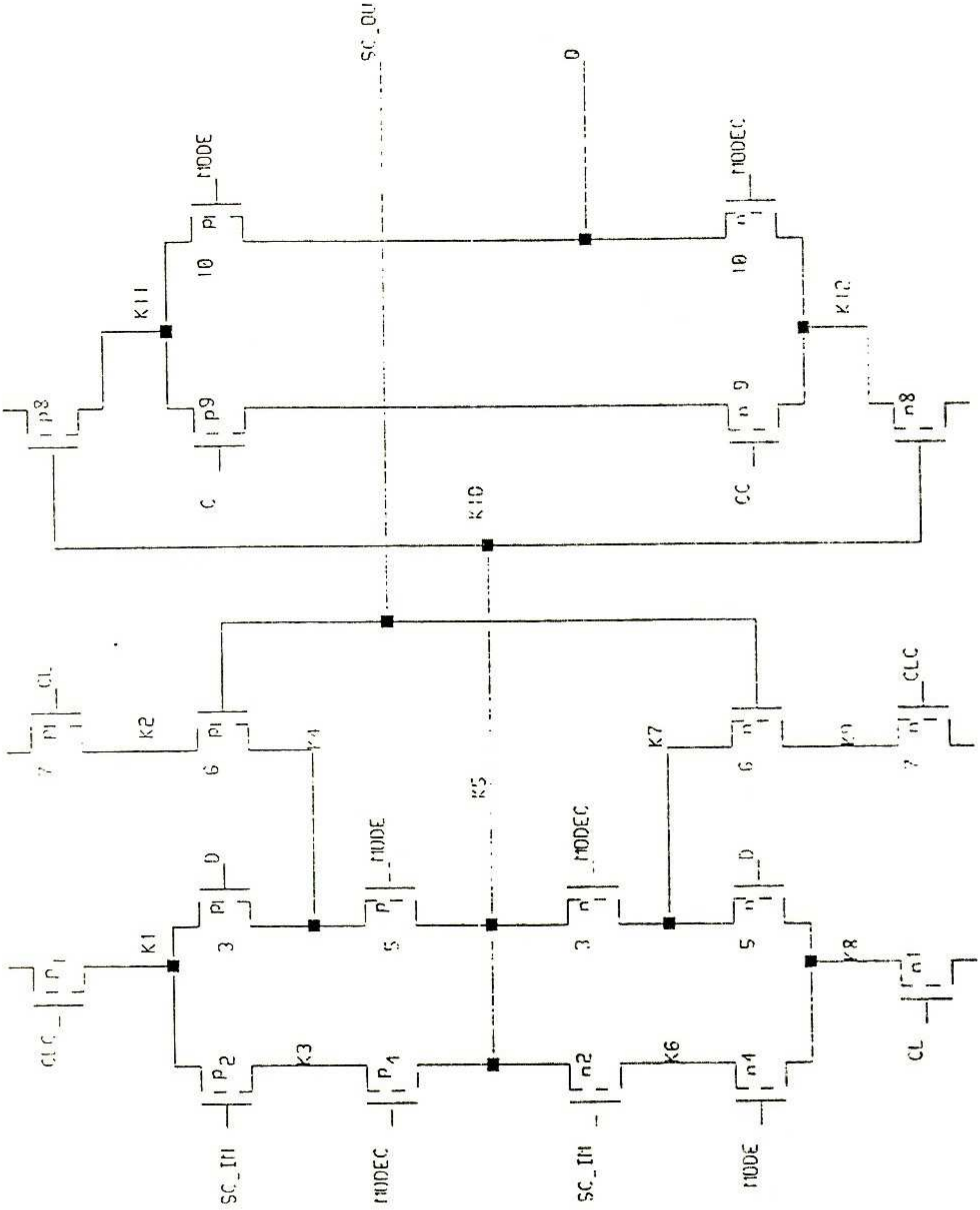


Figure 5: Testable Scan-Path D-Latch


SC_OU
Q
MODE
MODEC
K11
K12
p8
p9
n8
n9
C
CC
K10
CL
CLC
K2
K4
K5
K7
K9
K1
K3
K6
K8
D
p1
p2
p4
n1
n2
n4
SC_IN
MODEC
SC_IN
MODE