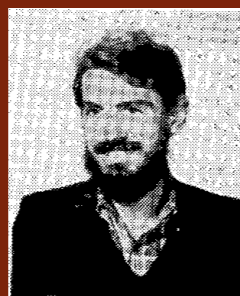


# A Brief History of Timing



Technical Content: *Too Many to name*

Presenter: J. White

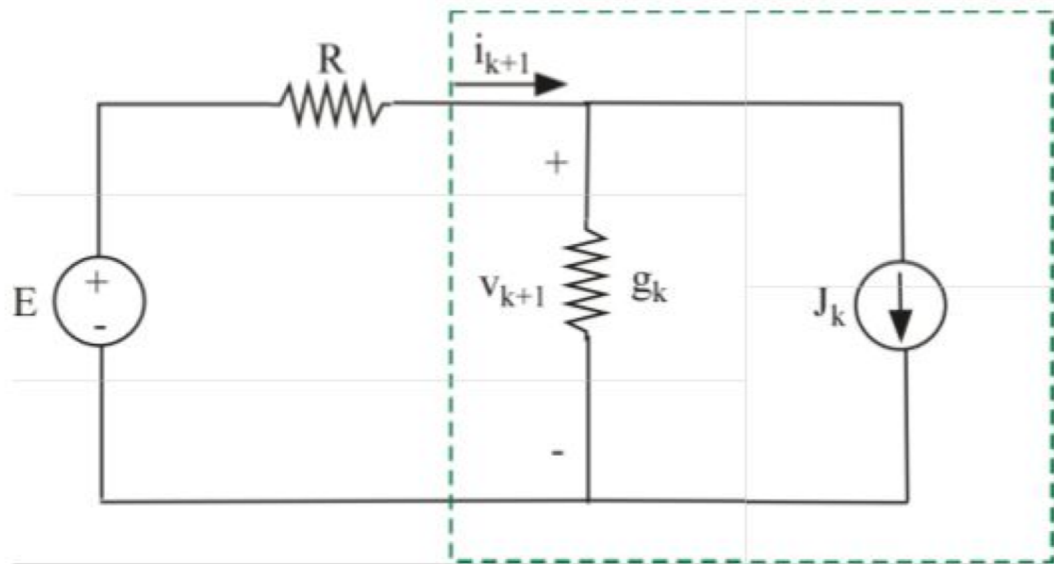
- In 1983:

- 0.1Mflop computers,  $\text{flop} > 10 * \text{MemoryRef}$
- BIG CIRCUIT: 1000 Transistors
- Exploitable Properties
  - Multi-rate, no bipolar, MOS unidirectional, hierarchy
- Key Techniques
  - Decomposition, multirate integration, simplified models

- In 2019

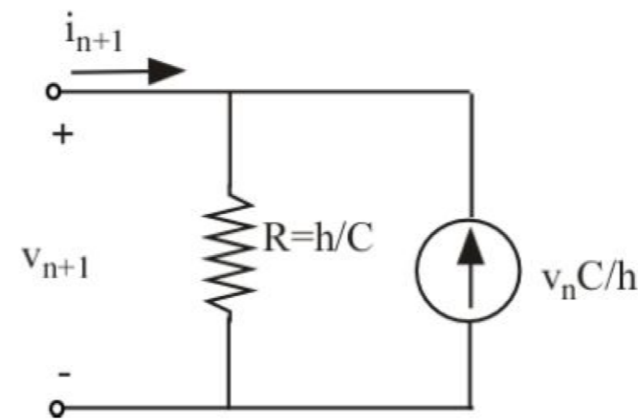
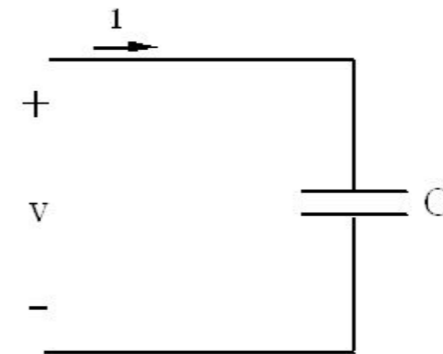
- 100 Gigaflops or more,  $\text{MemoryRef} > 10 \text{ Flop}$
- BIG CIRCUIT: 1,000,000,000 Transistors
- Exploitable Properties
  - Multi-rate, no bipolar, MOS unidirectional, HIERARCHY
- Key Techniques
  - Multi-rate, no bipolar, SIMD (GPU), REUSE

# Bad Companions

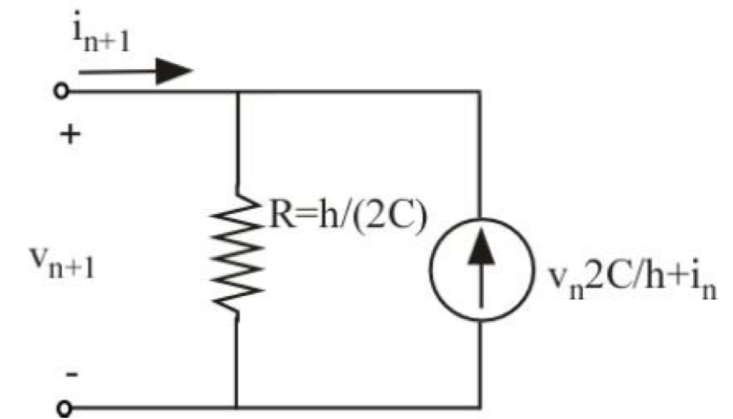


$$g_k = \frac{I_s}{V_t} e^{v_k/V_t}$$

$$J_k = I_s (e^{v_k/V_t} - 1) - V_k g_k$$



Backward Euler companion model at  $t=nh$



Trapezoidal companion model at  $t=nh$

**N models \* N methods = N<sup>2</sup> work**

$$\frac{dq(\mathbf{v}, t)}{dt} = i(\mathbf{v}, \mathbf{u}, t) ,$$

$$\mathbf{v}_{t+\Delta t} = \mathbf{v}_t + \Delta t f(\mathbf{v}_{t+\Delta t}, \mathbf{u}_{t+\Delta t}, t + \Delta t) \longrightarrow F(\mathbf{v}) = 0$$

$$\mathbf{v}^{i+1} = \mathbf{v}^i - J_F^{-1}(\mathbf{v}^i) F(\mathbf{v}^i) \longrightarrow J_F(\mathbf{v}_i) \mathbf{x} = F(\mathbf{v}_i).$$

# Circuit Simulators have become a Commodity



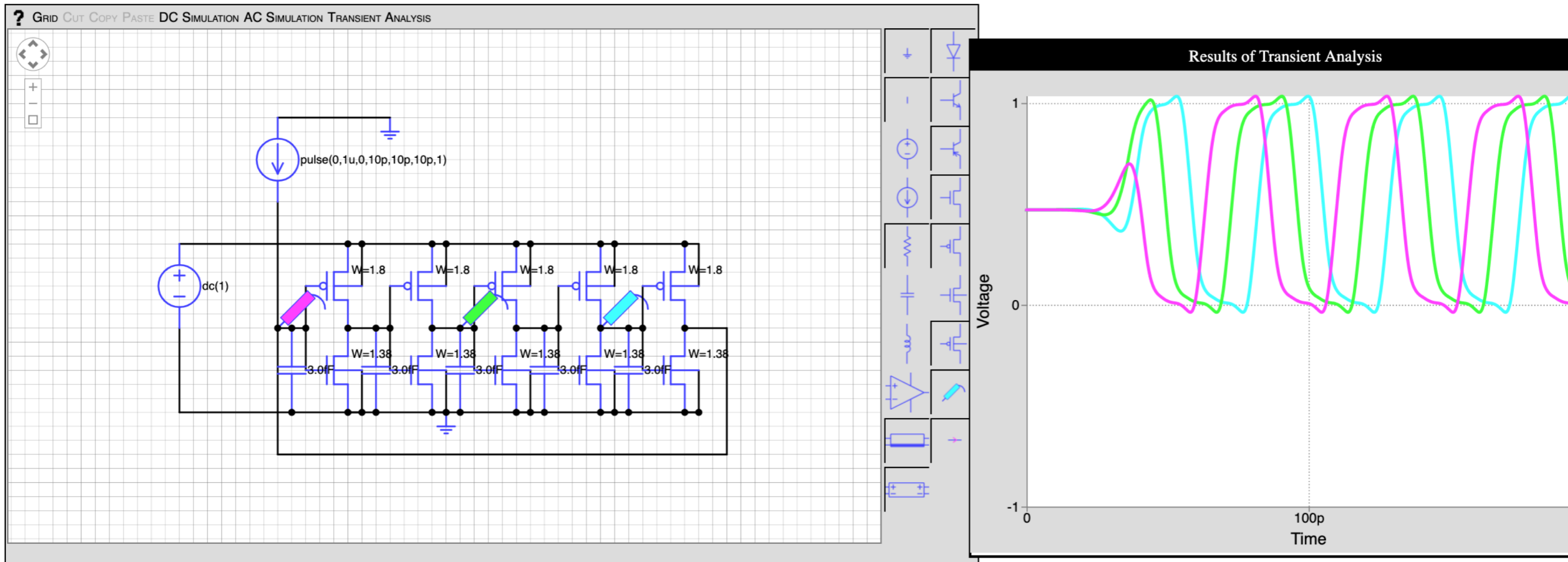
Chrome File Edit View History Bookmarks People Window Help 55% Tue Apr 16 1:24 PM white

International Symposium on Ph x Schematic Entry

Not Secure | scripts.mit.edu/~white/newtlines/schvsp.cgi?file=vsr5s

Apps https://www.ieee... (6/14) Queue Image result for a... Google 7 citi 27 filter http://libproxy.mit... Timing, Logic and...

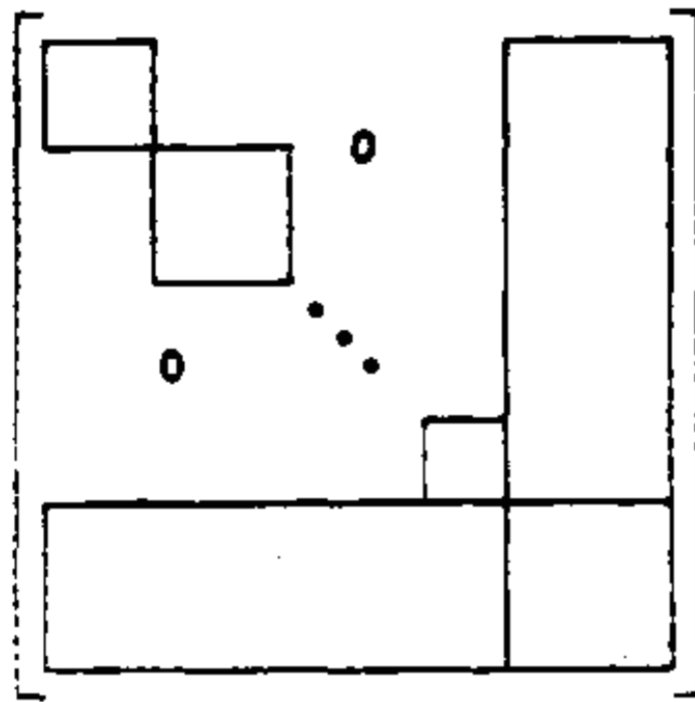
Schematic Name: vsr5s Load Save VS FET Parameter File: File Reload Choose File No file chosen



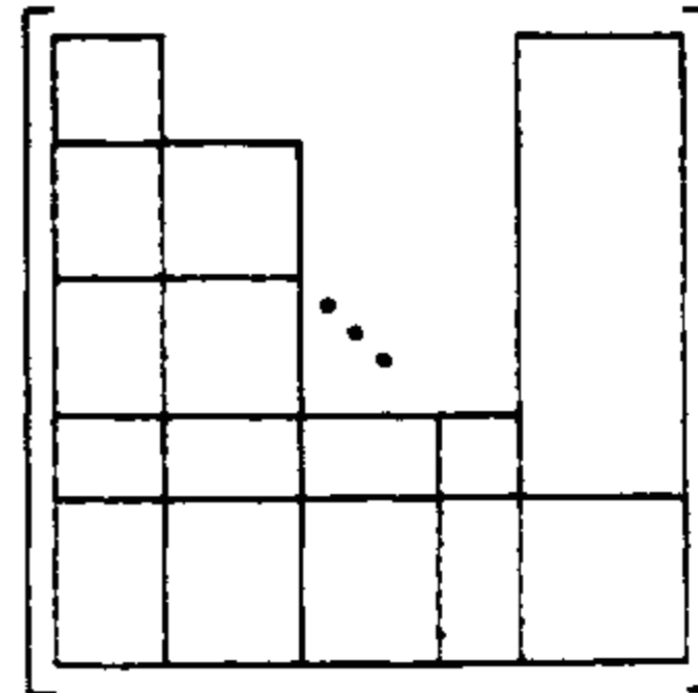
Javascript Simulator – 3 person weeks to write



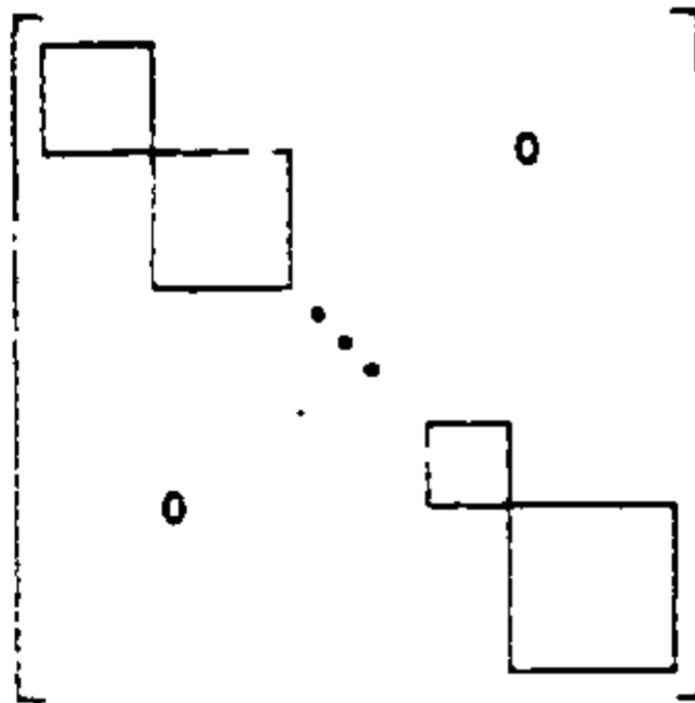
# BBD, BLBD, BD, LBD, .....



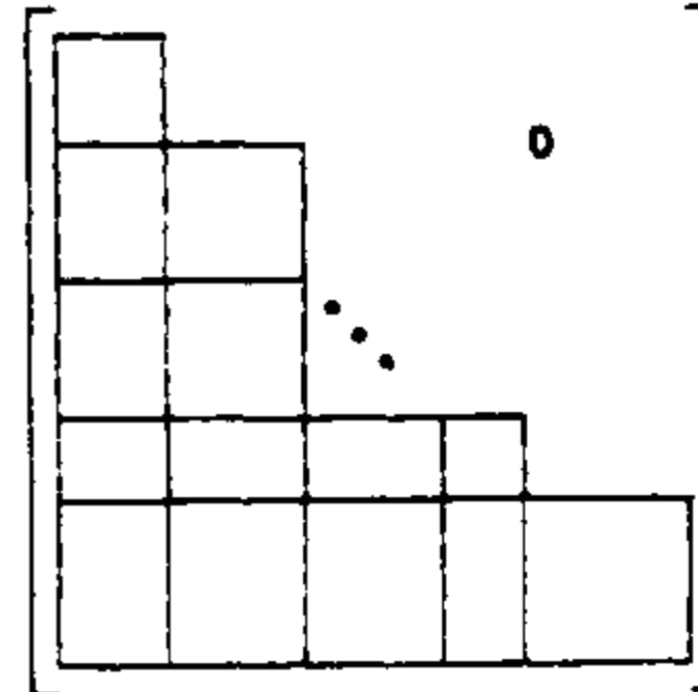
(a)



(b)



(c)



(d)



(a) Gauss–Jacobi integration algorithm:

$$[\mathbf{I} - h\mathbf{D}]\mathbf{x}_{k+1} = [\mathbf{I} + h(\mathbf{L} + \mathbf{U})]\mathbf{x}_k.$$

$$\mathbf{x}_{k+1} = \mathbf{M}_{\text{GJ}}(h)\mathbf{x}_k.$$

where  $\mathbf{I}$  is the identity matrix and

$$\mathbf{M}_{\text{GJ}}(h) = [\mathbf{I} - h\mathbf{D}]^{-1}[\mathbf{I} + h(\mathbf{L} + \mathbf{U})]$$

(b) Gauss-Seidel integration algorithm:

$$[\mathbf{I} - h(\mathbf{D} + \mathbf{L})]\mathbf{x}_{k+1} = [\mathbf{I} + h\mathbf{U}]\mathbf{x}_k$$

$$\mathbf{x}_{k+1} = \mathbf{M}_{\text{GS}}(h)\mathbf{x}_k$$

where

$$\mathbf{M}_{\text{GS}}(h) = [\mathbf{I} - h(\mathbf{D} + \mathbf{L})]^{-1}[\mathbf{I} + h\mathbf{U}]$$

# The Wrath of Kahan



(c) Modified symmetric Gauss–Seidel integration algorithm:

Let:

$$\mathbf{A}_L = \mathbf{L} + \frac{1}{2}\mathbf{D} \quad \mathbf{A}_U = \mathbf{U} + \frac{1}{2}\mathbf{D}$$

forward step:

$$\left[ \mathbf{I} - \frac{h}{4} (2\mathbf{L} + \mathbf{D}) \right] \mathbf{x}_{k+1/2} = \left[ \mathbf{I} + \frac{h}{4} (\mathbf{D} + 2\mathbf{U}) \right] \mathbf{x}_k$$

$$\left[ \mathbf{I} - \frac{h}{2} \mathbf{A}_L \right] \mathbf{x}_{k+1/2} = \left[ \mathbf{I} + \frac{h}{2} \mathbf{A}_U \right] \mathbf{x}_k$$

$$\mathbf{x}_{k+1/2} = \left[ \mathbf{I} - \frac{h}{2} \mathbf{A}_L \right]^{-1} \left[ \mathbf{I} + \frac{h}{2} \mathbf{A}_U \right] \mathbf{x}_k$$

backward step:

$$\left[ \mathbf{I} - \frac{h}{4} (\mathbf{D} + 2\mathbf{U}) \right] \mathbf{x}_{k+1} = \left[ \mathbf{I} + \frac{h}{2} (2\mathbf{L} + \mathbf{D}) \right] \mathbf{x}_{k+1/2}$$

$$\mathbf{x}_{k+1} = \left[ \mathbf{I} - \frac{h}{2} \mathbf{A}_U \right]^{-1} \left[ \mathbf{I} + \frac{h}{2} \mathbf{A}_L \right] \mathbf{x}_{k+1/2}$$

Combining (22) and (24) we obtain:

$$\mathbf{x}_{k+1} = \mathbf{M}_S(h) \mathbf{x}_k$$

# The Missing Iteration

## Relaxation-Based Circuit Simulation

## Standard Circuit Simulation

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2) \\ \dot{x}_2 &= f_2(x_1, x_2) \end{aligned}$$

Integration Formulae  
(e.g. Backward Euler)

$$\begin{aligned} g_1(x_1^k, x_2^{k-1}) &= 0 \\ g_2(x_1^k, x_2^k) &= 0 \end{aligned}$$

$$\begin{aligned} g_1(x_1, x_2) &= 0 \\ g_2(x_1, x_2) &= 0 \end{aligned}$$

Nonlinear Gauss-Seidel

Newton-Raphson

$$\begin{aligned} a_{11}x_1^k + a_{12}x_2^{k-1} &= b_1 \\ a_{21}x_1^k + a_{22}x_2^k &= b_2 \end{aligned}$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \end{aligned}$$

Linear Gauss-Seidel

Gaussian Elimination  
or LU Decomposition

**SPICE2 VS RELAX2.2 (DIRECT) VS RELAX2.2(WR) ON INDUSTRIAL CIRCUITS**

Circuit	Devices	SPICE2	RELAX2.2(DIRECT)	RELAX2.2(WR)
uP Control	232	1400s*	90s*	45s*
Cmos Memory	621	10400s*	995s*	308s*
4-bit Counter	259	4300s*	540s*	299s*
Inverter Chain	250	439s**	98s**	38s**
Digital Filter	1082	18000s*	1800s*	520s*
Encode-Decode	3295	115,000s*	5000s*	1500s*

**Theorem 1.5.1:** If, in addition to the assumptions of Theorem 1.2.1, the WR iteration equations are solved using a stable and consistent multistep method with a fixed timestep  $h$ , for a finite number of points, then there exists an  $h' > 0$  such that the sequences  $\{ x^k(n) \}$  generated by the Gauss-Seidel or Gauss-Jacobi discretized WR algorithm will converge for all  $0 < h < h'$ .