

Victory 및 SmartSpice SEE를 활용한 SRAM 셀의 물리적 3D Single Event Upset 시뮬레이션

소개

Victory 시뮬레이션 프레임워크는 현대 반도체 기술의 1D/2D/3D 시뮬레이션 툴을 포함합니다. Victory는 복잡한 3D 형상을 빠르고 정확하게 시뮬레이션하기 위해 완전한 사면체 메쉬 엔진을 구현합니다. 내장된 메쉬 세밀화 기준과 사용자가 정의하는 기준을 활용하여, 시뮬레이션 중에 메쉬를 적절하게 수정할 수 있습니다. 예를 들어, SEU (Single Event Upset) 시뮬레이션이 이에 해당됩니다. SEU 현상의 3D 시뮬레이션은 SEU 트랙 가까이에서 물리적 양이 점진적으로 크게 변하므로 매우 복잡합니다. 3D 구조에서 SEU의 영향을 정확하게 안정적으로 시뮬레이션하려면, 트랙에서 멀리 있는 곳의 메쉬는 엉성하게, SEU 트랙 중심 근처의 메쉬는 조밀하게 하는 것이 효과적입니다.

본 애플리케이션 노트에서 SEU의 영향을 받는 SRAM 셀을 정확하게 시뮬레이션하는 방법을 소개합니다.

구조 생성

2개의 NMOS와 2개의 PMOS로 구성된 3D 구조를 DevEdit 3D로 만듭니다. 소자 시뮬레이션을 위해 DevEdit 3D의 기능을 활용하여, 구조의 형상, 물질, 농도 표현 및 3D 사면체 메쉬를 구성합니다. 그림 1에서

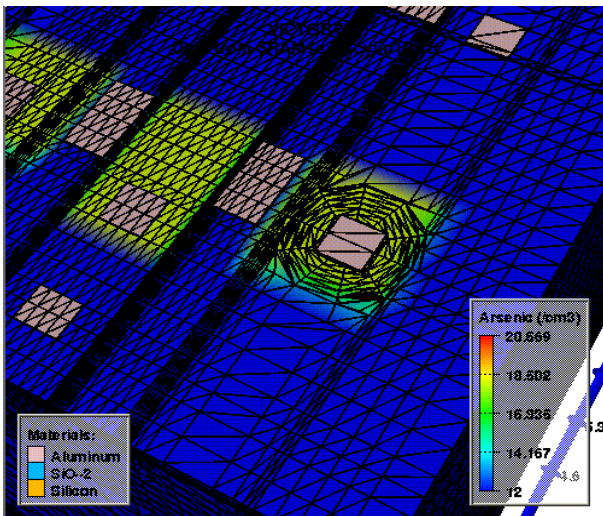


그림 1. SRAM 구조의 비소 농도 및 표면 메쉬

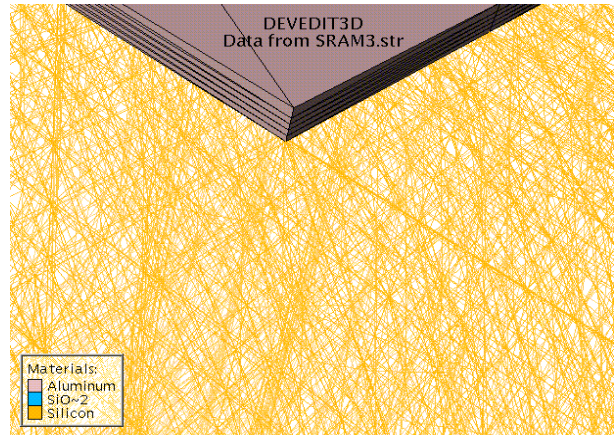


그림 2. SRAM 구조에서 컨택 주위의 사면체 메쉬

3D 구조의 도핑 및 표면의 메쉬를 나타냅니다. 영향을 받을 곳에서 메쉬를 원통형으로 재구성하는 것이 매우 중요합니다. 그림 2에서 컨택 주위를 확대하여 3D 구조의 사면체 요소를 나타냅니다.

그림 3에서 4개의 트랜지스터로 이루어진 SRAM 회로의 스키매틱을 나타냅니다. 수치 소자 시뮬레이션에 4개의 트랜지스터를 사용하여 트랜지스터 간의 결합 효과를 모두 고려합니다. SRAM 회로에서 어떤 트랜지스터도 SPICE 소자로서 시뮬레이션하지 않습니다.

시뮬레이션

Victory에 대한 시뮬레이션 입력을 다음과 같이 DeckBuild에서 정의합니다 (그림 4 및 5):

- Devedit 구조 로드
- 원통형 메쉬 재구성
- 모델 사양
- DC, 과도 분석 사양

SEU의 충격 방향은 입구와 출구 위치에 해당하는 (x,y,z) 좌표로 지정합니다 (그림 6).

SEU 트랙은 원통형으로 가정하며, 전하 생성 펄스의 폭 (시간)과 함께 시간 내 최대 전하 밀도의 위치를 지정할 수 있습니다. Victory에 기본 SEU 함수가 있지만, 사용자 정의 충격 함수를 만들 수도 있습니다 (그림 7).

SRAM 회로의 지정 DC 바이어스는 노드 3과 2를 각각 0.0 및 3.3V로 설정합니다. SRAM 회로의 DC 바이어스는 과도 분석의 초기 조건으로 사용합니다. 과도 분석은 10 펨토초의 초기 시간 간격으로 10마이크로초 동안 수행합니다. SEU 충격은 4피코초에서 2피코초 동안 최대 밀도가 됩니다.

결과

Victory 시뮬레이션 출력은 과도 시간의 함수로 노드 전압과 전류를 제공합니다. 또한 소자의 내부 동작 (예: 전위 및 전자 농도)을 과도 시간의 함수로서 분석할 수 있습니다.

노드 2와 3의 전압 대 과도 시간 플롯은 SEU 입자가 전압에 미치는 영향을 나타냅니다 (그림 8).

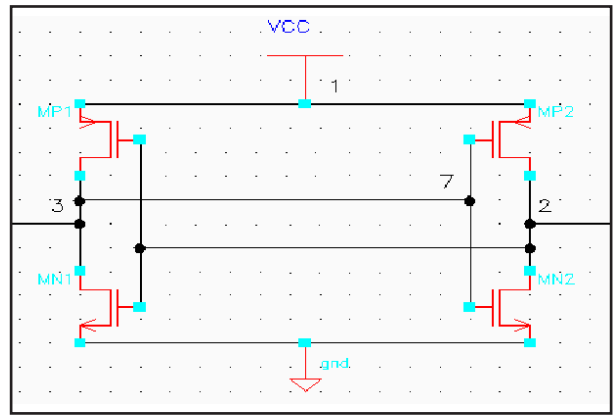


그림 3. SRAM 셀의 스키매틱

원통형 메쉬를 충격에 따라 세밀하게 해주지 않으면, 전압은 일정하게 유지되므로 시뮬레이션 결과가 정확하지 않습니다 (그림 9의 청색 곡선). 그림 9에서, 녹색 곡선과 적색 곡선은 각각 2개와 10개의 원통형 트랙에 따라 메쉬를 세밀하게 조정된 결과입니다.

그림 10은 2개 또는 10개의 원통형으로 메쉬를 재구성했을 때, 충격에 따른 메쉬 이산화 인자는 2D 플롯으로 표현합니다. 메쉬 이산화 인자는 메쉬를 세밀하게 해야 하는 영역을 찾는 데 도움이 됩니다. 적색은 상대적으로 선형인 척도에서 가장 비선형적인 요소를 나타내며, 이는 솔루션에서 미세 조정으로 가장 큰 차이를 내는 영역임을 의미합니다.

결론적으로, 2개의 원통형만으로 메쉬를 조정하면 충분히 정확한 결과를 얻을 수 있습니다.

```
Deckbuild V3.23.17.R - (NONE) (edited), dir: /home/eric/workki
File View Edit Find Main Control Commands Tools
solve vgate=0.1 vstep=0.1 vstop=3.3 name=gate
solve vsource=0.1 vstep=0.1 vstop=3.3 name=source
# contact definition
contact name=source current
contact name=bulk current COMMON=bulk WORK=4.35
contact name=drain current COMMON=bulk WORK=4.35
contact name=substrate current COMMON=source WORK=4.35
contact name=fgate current COMMON=source WORK=4.25
contact name=base current COMMON=bulk WORK=4.25
contact name=anode current COMMON=source
contact name=vss COMMON=gate
contact name=vgg COMMON=collector
contact name=cgate current COMMON=bulk
models consrh auger bgn cvt klassen
method bicgst.rcm
method linear.fill_level=2
method linear.scaling_parameter=2
solve isource=0
solve ibulk=0
beam f.radiate=ion_c_inter_v_LET_3.5.1lb
log outf=SRAM.log
solve tfinal=1e-5 tstep=1.e-14
next line stop cont run quit Line: 2
paste init pause clear restart kill Stop: None
Thu Feb 2 10:52:32 2006
Executing on host: trieves
Starting: DEVICE module.
VICTORY>
VICTORY started VICTORY
```

그림 5. 모델, DC, 과도 분석 사양

```
Deckbuild V3.23.17.R - (NONE) (edited), dir: /home/eric/workki
File View Edit Find Main Control Commands Tools
go victory
mesh inf=SRAM.str hybridx
contact NAME=gate
contact NAME=source
contact NAME=bulk
contact NAME=drain WORK=4.35 COMMON=bulk
contact NAME=substrate WORK=4.35 COMMON=source
contact NAME=fgate WORK=4.25 COMMON=source
contact NAME=collector WORK=4.25 COMMON=bulk
contact NAME=base WORK=4.25 COMMON=bulk
contact NAME=anode COMMON=source
contact NAME=vss COMMON=gate
contact NAME=vgg COMMON=collector
contact NAME=cgate COMMON=bulk
models consrh auger bgn cvt klassen
method bicgst.rcm
method linear.fill_level=2
method linear.scaling_parameter=2
method NONLINEAR.MAX_ITERATIONS=30
regrid x.min=0.775 y.min=0 z.min=3.22 \
x.max=0.775 y.max=5 z.max=3.22 \
radius=0.05 \
lmax=20 rmax=10 num.cylinders=10 constraint
next line stop cont run quit Line: 2
paste init pause clear restart kill Stop: None
Thu Feb 2 10:52:32 2006
Executing on host: trieves
Starting: DEVICE module.
VICTORY>
VICTORY started VICTORY
```

그림 4. DevEdit 구조 로드 및 원통형 메쉬 재구성 정의

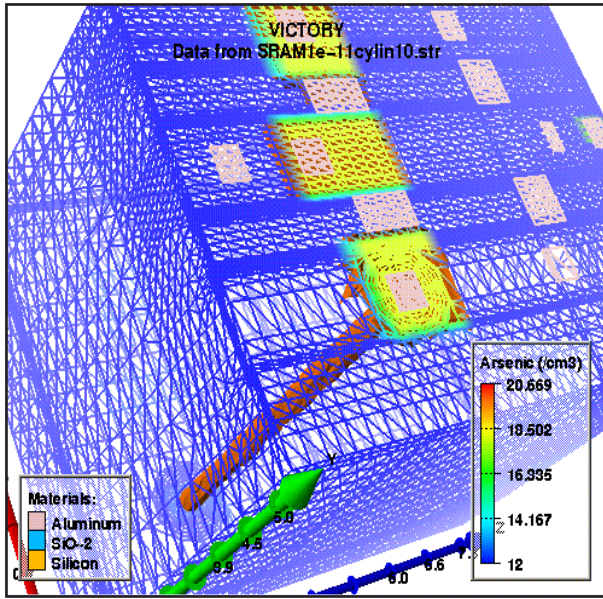


그림 6. 충격의 위치를 나타내는 3D SRAM 구조

```

Text Editor V3.6.2 FCS - NewDocument (edited), dir: /home/eri
File View Edit Find
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <ctype.h>
#include <malloc.h>
#include <string.h>

/*
 * Generation rate as a function of position (3D)
 * Statement: BEAM
 * Parameter: F.RADIATE
 * Arguments:
 * x location x (microns)
 * y location y (microns)
 * z location z (microns)
 * t time (seconds)
 * *rat generation rate per cc per sec.
 */
int radiate(double x, double y, double z, double t, double *rat)
{
    double LET; /* en pC/um² */
    double w0; /* en um² */
    double x0; /* en um² */
    double z0; /* en um² */
    double y0; /* en un faire attention au sens de propagation du faisceau z>0
    ou z<0 */
    double T0; /* en s² */
    double Tc; /* en s² */
    double Range; /* en um² */

    LET=0.035;
    w0=0.05;
    x0=0.775;
    z0=3.22;
    y0=0;
    T0=4e-12;
    Tc=2e-12;
    Range=40;

    *rat =
    ((LET*1e-12*1e4/1.6e-19)/
    (5.568327997*(w0*1e-4)*(w0*1e-4))*
    exp(-(
    (x-x0)*(x-x0)/
    
```

그림 7. C-인터프리터를 이용한 사용자 정의의 충격 함수

충격의 강도에 따른 노드 2와 3에서의 전압 대 과도 시간 플롯을 나타냅니다 (그림 11). 따라서 SRAM 셀의 동작은 충격 강도의 함수로 해석할 수 있습니다. 낮은 강도 (청색 곡선)에서 SRAM의 변화는 없습니다. 중간 강도 (녹색 곡선)에서 전압이 바뀝니다. 마지막으로 높은 강도 (적색 곡선)에서 SRAM은 불안정해집니다.

SRAM 셀에 문제가 발생했을 때, 충격을 받은 NMOSFET의 전자 전류 밀도의 변화를 표현하기 위해 전자 전류 밀도를 3D로 나타낼 수 있습니다. 처음에는 NMOSFET의 소스와 드레인 영역 사이에 전자 전류 밀도가 낮습니다 (트랜지스터 꺼짐).

SEU 충격이 NMOSFET에 들어가면, SEU 충격 경로를 따라 전자-홀 쌍이 생성되어 소자 전체의 전자 분포를 변화시켜, 결국 NMOSFET의 전자 전류 밀도가 증가합니다 (그림 12). SEU 충격이 구조에서 사라지면 외부 전하의 원천이 제거되고, 원래의 전자 전류 밀도가 다시 설정됩니다 (트랜지스터는 다시 꺼짐).

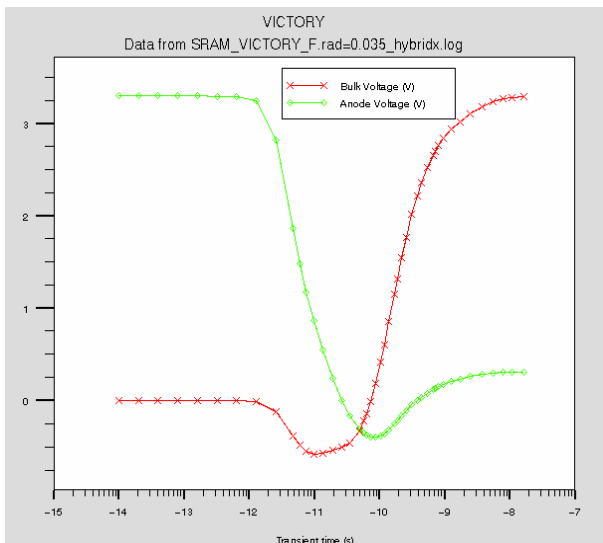


그림 8. 노드 2와 3의 전압 (그림 3 참조) 대 과도 시간. 양극 전압은 노드 2, 벌크 전압은 노드 3에 해당합니다.

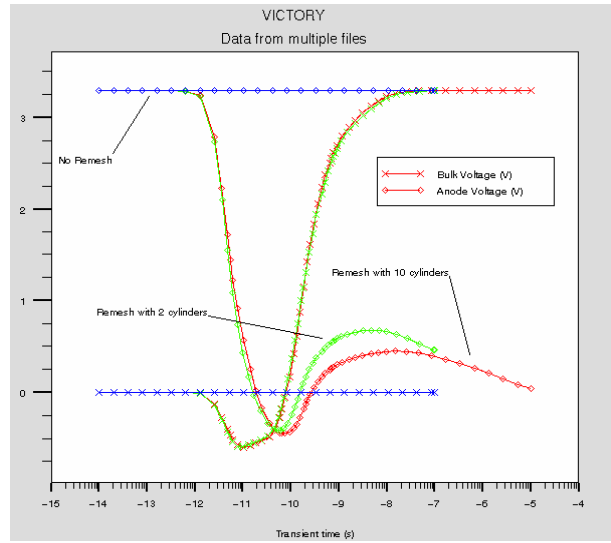


그림 9. 노드 2와 3의 전압 (그림 3 참조) 대 트랙에 따른 메시의 과도 시간 및 밀도. 양극 전압은 노드 2, 벌크 전압은 노드 3에 해당합니다.

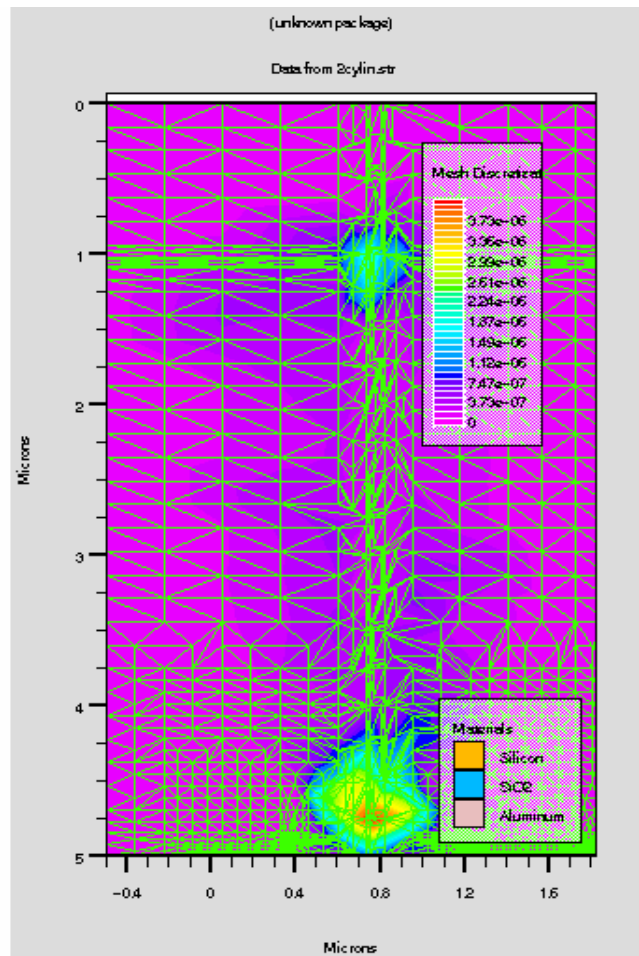
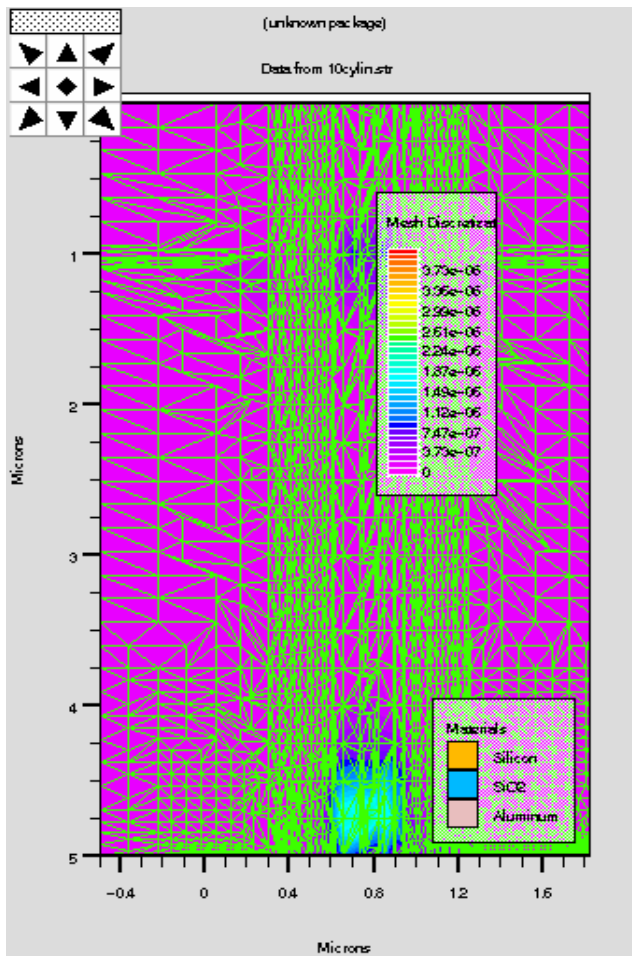


그림 10. 원통형 2개 (왼쪽), 원통형 10개 (오른쪽)를 사용한 노드 3에서의 메쉬 이산화 인자

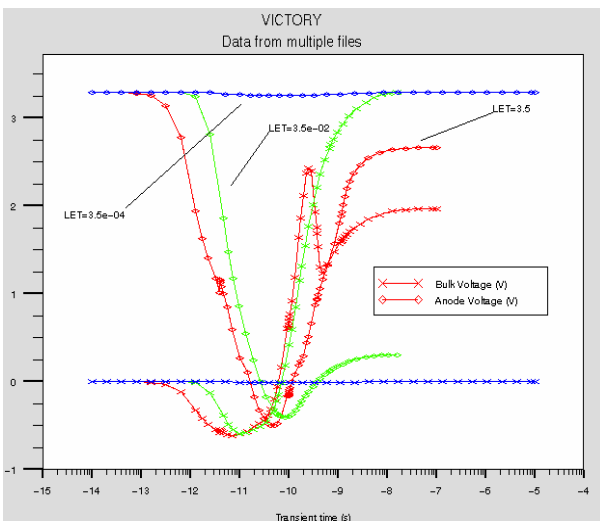


그림 11. 노드 2와 3의 전압 (그림 3 참조) 대 충격의 과도 시간 및 강도. 양극 전압은 노드 2, 벌크 전압은 노드 3에 해당합니다.

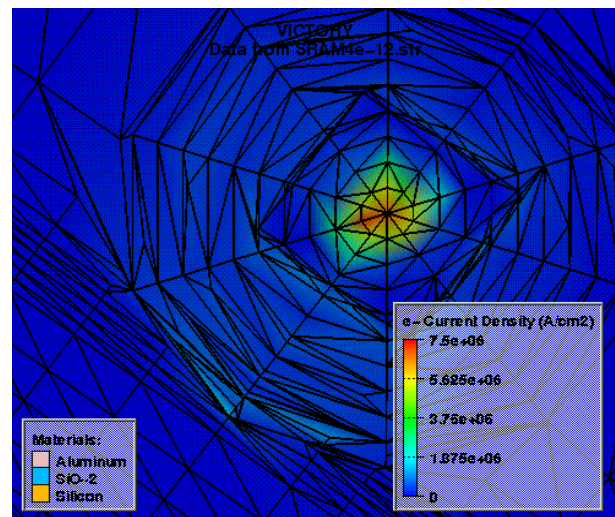


그림 12. $t=4e-12s$ 에서 NMOS 트랜지스터 (노드 3)의 드레인 전류 밀도

5. 회로 시뮬레이션

앞서 설명한 바와 같이, TCAD 시뮬레이션을 통해 트랜지스터 또는 몇 개의 트랜지스터로 구성된 회로의 동작을 자세히 분석하는 것은 매우 흥미로운 일입니다. 전위, 전자 농도, 전류 밀도 등의 양을 활용하여, 영향을 받는 소자의 동작을 자세히 연구할 수 있습니다. 그러나 더 큰 회로에서 SEU의 영향을 연구하고자 할 때, TCAD 시뮬레이션으로는 불가능합니다. 그래서 MOS (Bulk 및 SOI) 및 바이폴라 소자의 SEE (Single Event Effect)를 정확하게 시뮬레이션하기 위해, SmartSpice SEE 모듈을 개발하였습니다. 입사된 입자의 충격으로 전자-홀 쌍이 생성됩니다. 전류 발생기를 회로에 삽입하여, 입자 충돌의 결과로 가정한 노드에서 수집한 전하를 모델링합니다. ASIC의 경우, 민감한 노드의 위치를 알 수 있습니다. 생성된 전류의 형태는 SmartSpice에서 사용할 수 있는 이중 지수 소스에 근접하게 됩니다. 다른 파형도 SmartSpice에서 사용할 수 있습니다.

래치 회로의 전압 대 과도 시간 플롯은 SEU 입자에 의해 전압이 어떤 영향을 받는지 보여줍니다 (그림 13).

그림 13은 래치의 안정적인 상태 (상단 그림)와 $t=10\text{ns}$ 에서 충격의 영향 (중간 그림)을 나타냅니다. 이는 래치 상태에 영향을 미치지 않고 기생 효과만 생성합니다. 에너지(LEF)를 증가시키면 충격으로 인해 래치 상태가 변경되고, 회로에 오류가 발생합니다 (하단 그림).

SmartSpice SEE는 회로의 여러 노드에서 동시 또는 다른 시간에 여러 가지 영향을 시뮬레이션할 수 있습니다. 사용자는 모든 노드에서 동시에 절대 또는 상대 임계값을 사용하여 이상을 탐지할 수 있습니다. 상태 변화를 일으키는 임계 전하 (QCRIT)는 자동으로 계산할 수 있습니다. 자세한 내용은 SmartSpice 문서를 참조하십시오.

6. 결론

Victory의 IV 곡선으로 특정 메쉬 기능과 정확한 물리 모델을 활용하여 SEU 충격 전하 생성을 설명할 수 있습니다. Victory의 전류 특성을 SmartSpice 전류원에 맞추고, SmartSpice SEE 기능을 활용하여 모든 유형의 회로에서 SEU 이상을 예측할 수 있습니다.

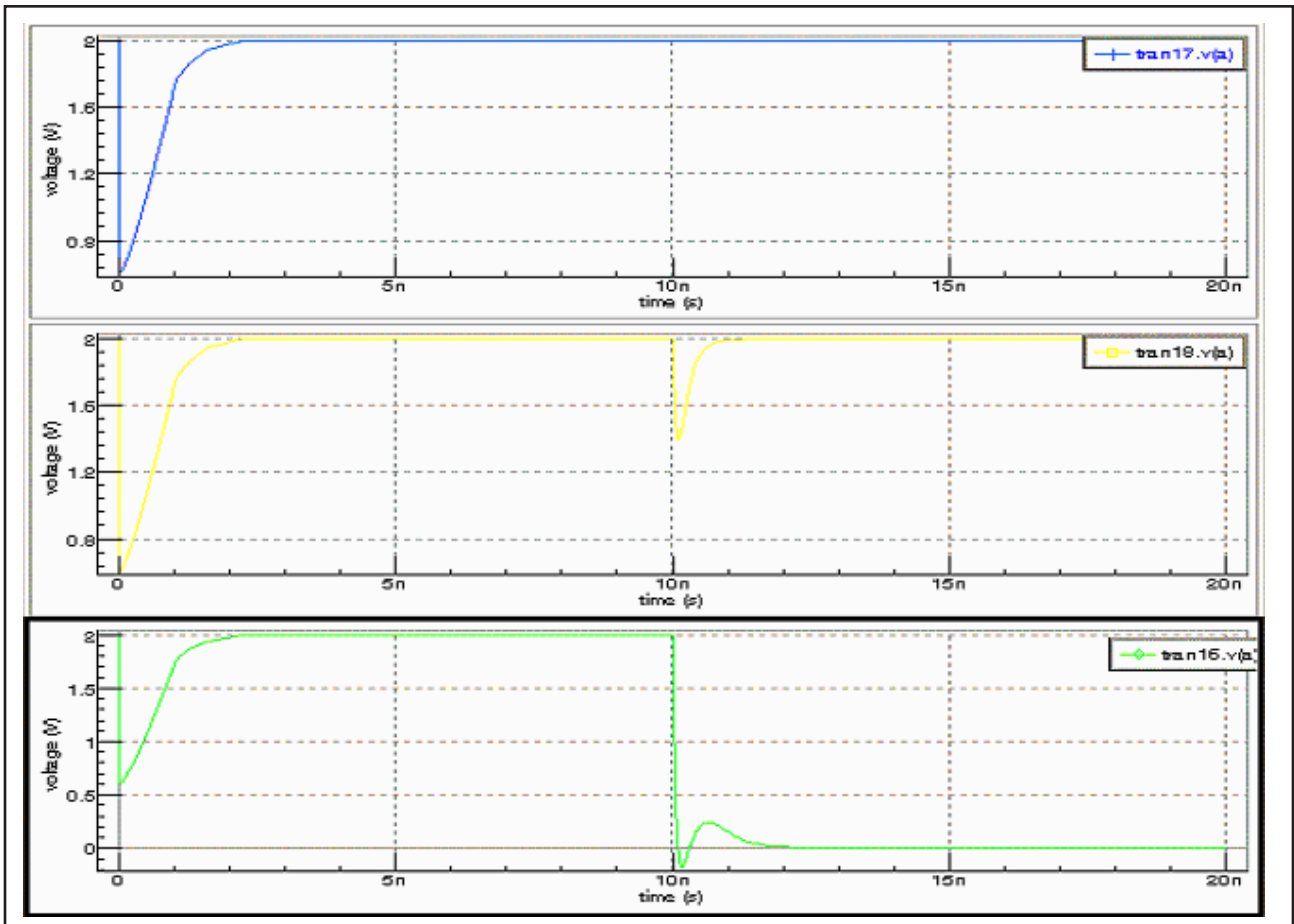


그림 13. SEU 입자의 영향을 받는 래치 회로의 동작