AIX-MARSEILLE CYBER SECURITY FORUM

Rétro-ingénierie exploitant des outils d'analyse de défaillance afin de faciliter les attaques matérielles

J.-M. Dutertre[1], R. Silva Lima[1, 2] , R. Viera[1] , M. Pommies[2], W. Magrini[2]

25 mai 2023

(1) Equipe Commune Systèmes et Architectures Sécurisées
Mines Saint-Etienne, Centre de Microélectronique de Provence
13541 Gardanne FRANCE

(2) Centre Technologique ALPhANOV – Systèmes Optiques

MINES Saint-Étienne
Une école de l'IMT

Institut Mines-Télécom

ALPhA NOV
Centre Technologique Optique et Lasers

This talk

Rétro-ingénierie exploitant des outils d'analyse de défaillance afin de faciliter les attaques matérielles

- Attaques matérielles ?

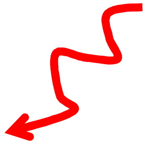
Vise un composant (circuit intégré, CI) sur lequel est exécuté une fonction de sécurité (algo. cryptographique, identification, etc.) à des fin d’attaque.

Cette présentation :

- Attaques par injection de fautes (1 famille att. HW) → i.e. attaque par perturbation à l’origine de l’apparition de fautes ou d’erreurs dans les opérations du circuit cible.
- Injection de fautes par illumination laser

- Attaques par injection de fautes - exemple

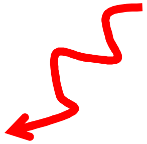
Routine de verification d’un mot de passe



```
Si passwd est égale à ref_passwd
alors
    accès = VRAI
Sinon
    accès = FAUX
Fin
```

▪ Attaques par injection de fautes - exemple

Routine de verification d'un mot de passe



```
Si passwd est different de ref_passwd alors
    accès = VRAI
Sinon
    accès = FAUX
Fin
```

▪ Attaques par injection de fautes - exemple

Routine de verification d’un mot de passe

```
Si passwd est égale à ref_passwd
alors
    accès = VRAI
Sinon
    accès = VRAI
Fin
```

Injection de fautes par laser

- Technique éprouvée (depuis 70s)
- Grande précision spatiale et temporelle
  - Perturbation locale
  - Capacité à fauter 1 bit/instruction unique
  - Durée : 1 période (ns) à la gamme des µs

Mais nombreux paramètres à ajuster (t, X, Y, puissance, durée, vulnérabilité, etc.)

→ Peut être très long, en particulier pour la recherche des points d'intérêts (zones sensibles permettant d'exploiter une vulnérabilité)

This talk

Rétro-ingénierie exploitant des outils d'analyse de défaillance afin de faciliter les attaques matérielles

- Analyse de défaillance ?

Outils permettant d'observer les propriétés physiques ou électriques des circuits jusqu'au niveau de leur transistors.

Nécessaires à la fiabilisation de la fabrication des CI (enjeu : atteindre les rendements de fabrication permettant la rentabilisation des fabs).

Cette présentation :

- Analyse de la photoémission → capture des photons émis lors des commutations des transistors
- Rétro-ingénierie spatiale et temporelle

This talk

❑ Failure analysis as hardware attack facilitation tools?

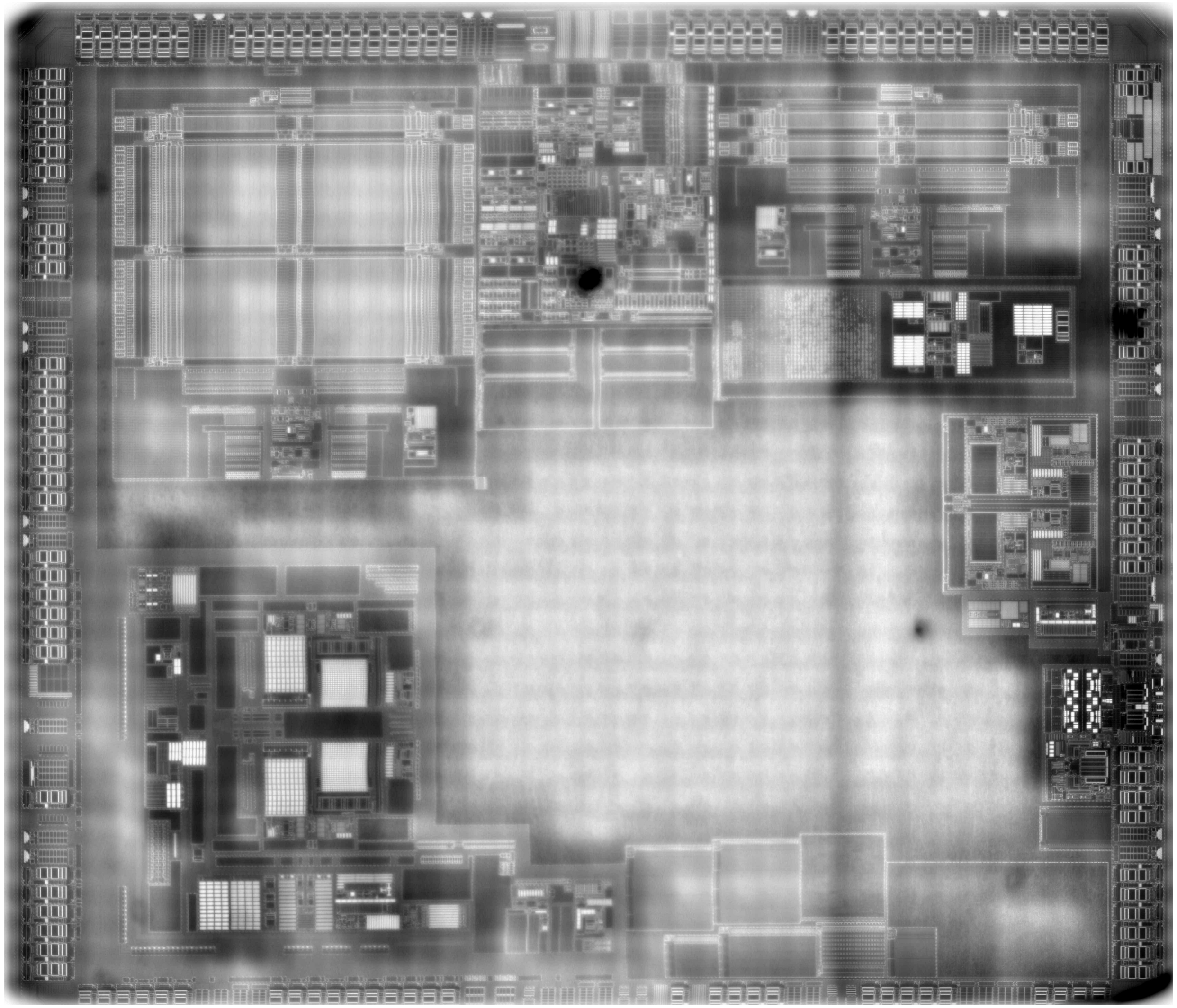
- Hardware attacks: laser fault injection
  - Accurate & local → POI identification = time consuming
- FA tool: photoemission analysis
  - Reverse engineering aims: where? and when?

This talk

- Considered POI:

Microcontroller target

- Flash memory (program) beq → bne
- RAM memory (data) FALSE → TRUE



Where? and When?

This talk

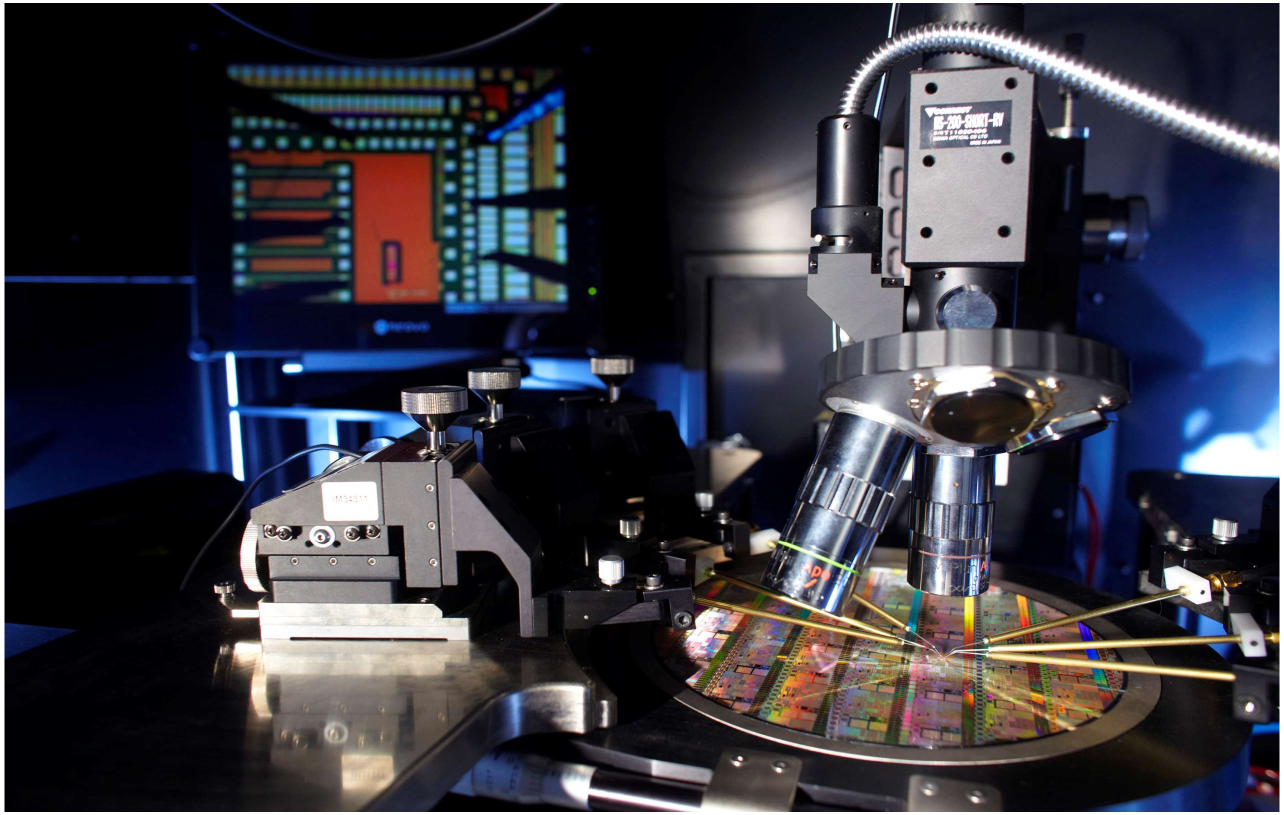
❑ Failure analysis as hardware attack facilitation tools?

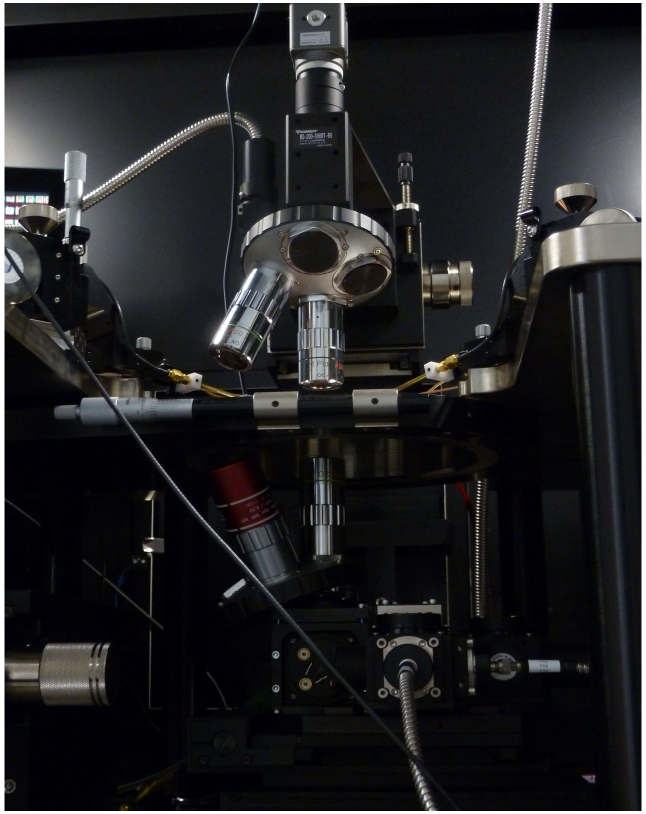
- Preliminary results
- PhD Hafsa El Alami, MSE – ST Microelectronics: secure microcontrollers provider
- PhD Rodrigo Silva Lima, MSE – Alphanov: laser benches provider

→ This talk: built from Rodrigo’s experiments (PE results)

❑ Laser fault injection?

- Pulsed lasers are used to inject faults into running secure devices for the purpose of retrieving secret information.

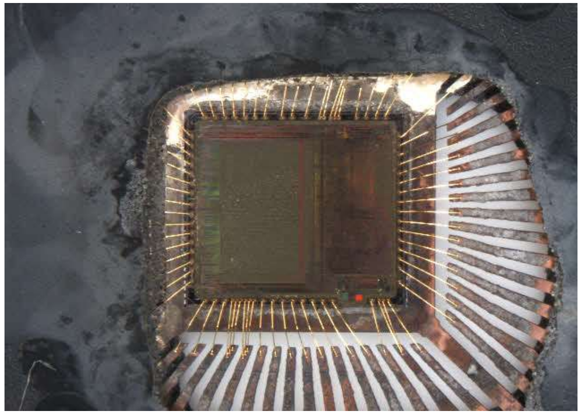




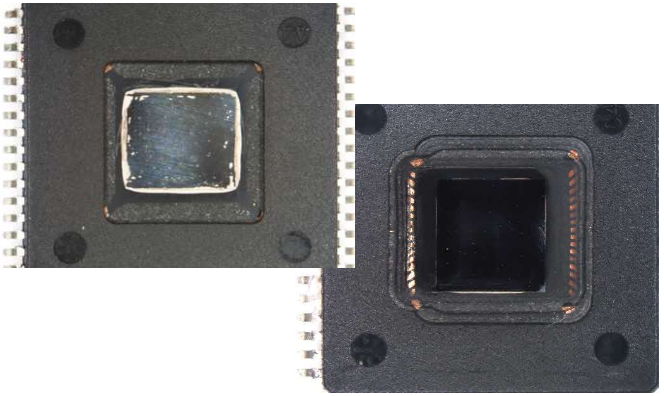
Laser fault injection attacks

❑ Physics of laser fault injection

▪ Laser beam: semi invasive (package mechanical/chemical opening)

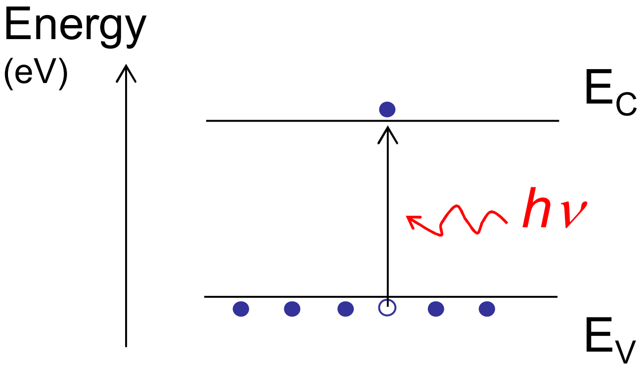


Front side



Backside

• laser – silicon interaction: the photoelectric effect



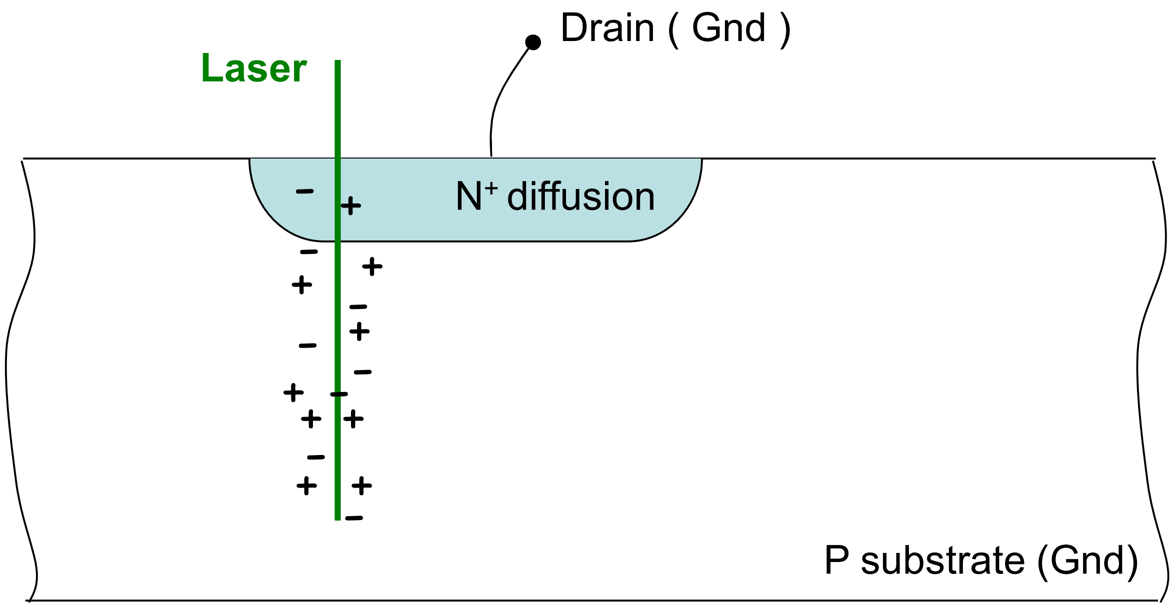
$$h\nu > E_g$$

$$\boxed{\lambda_{laser} < 1{,}1\ \mu m}$$

Laser fault injection attacks

- Photoelectric effect:

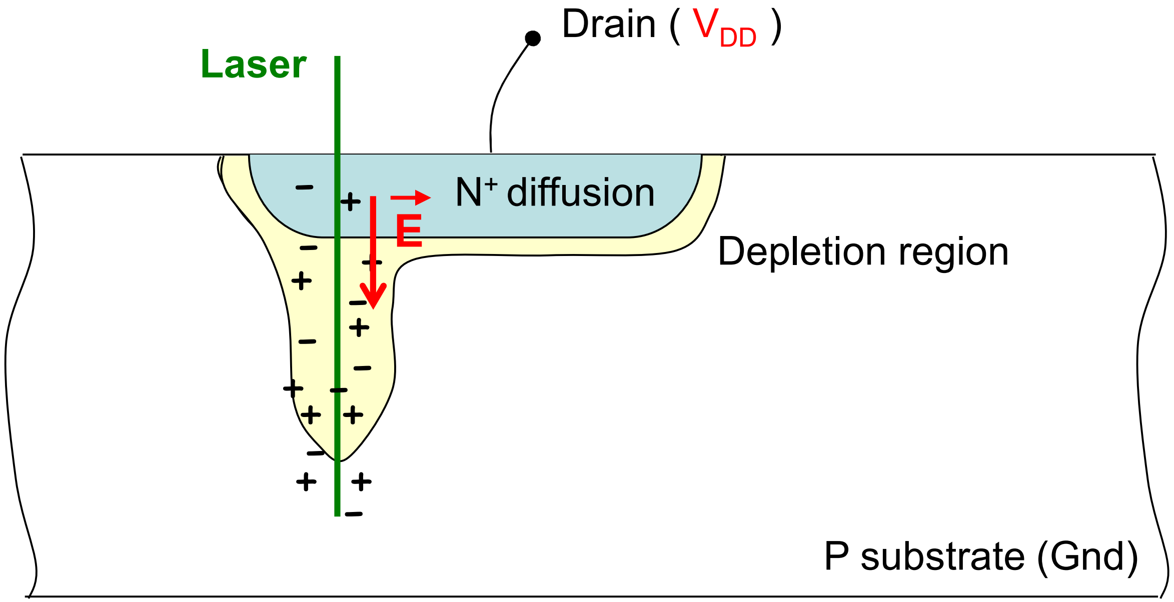
  from a laser pulse to transient current generation



Laser fault injection attacks

- Photoelectric effect:

  from a laser pulse to transient current generation



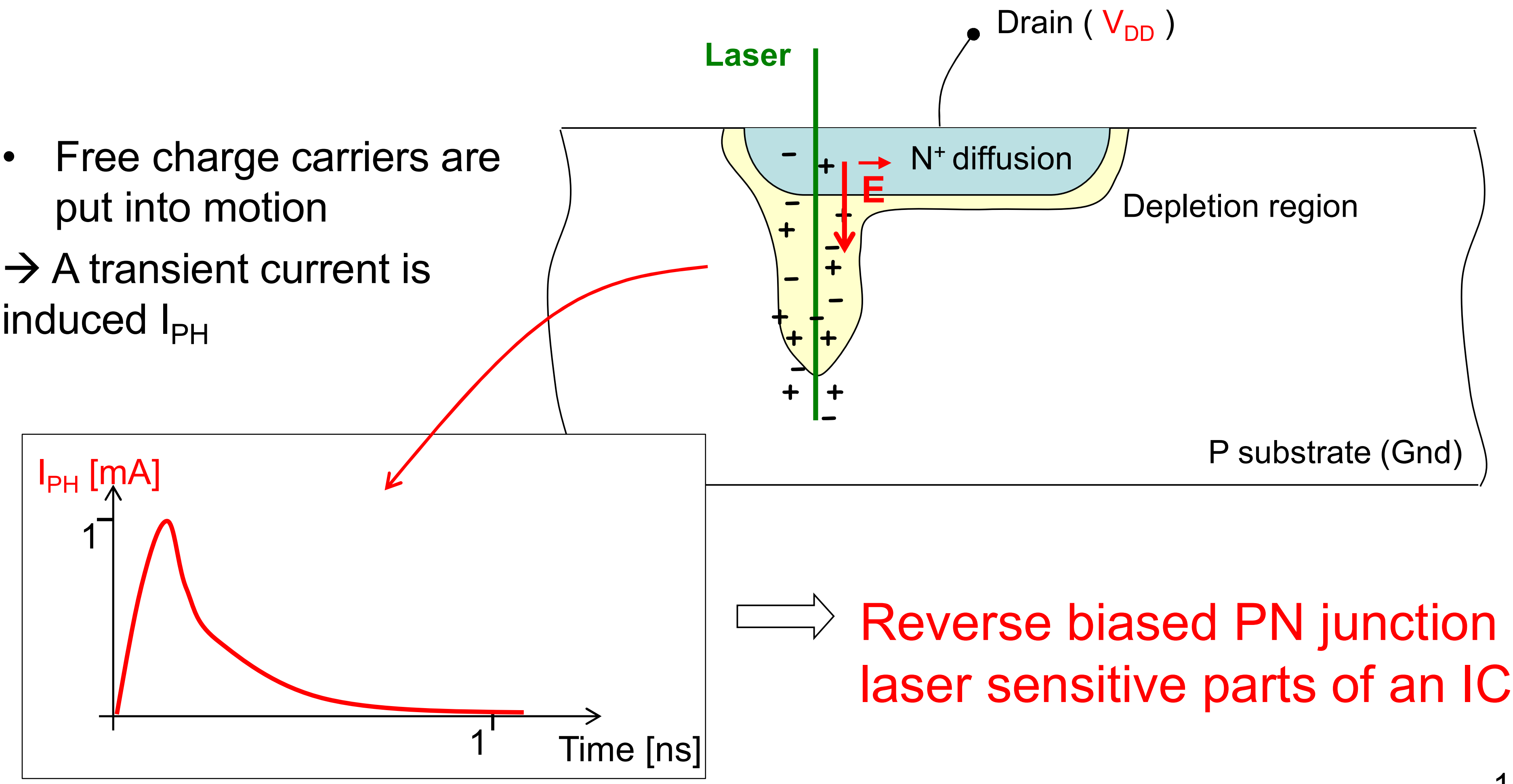
Laser fault injection attacks

- Photoelectric effect:

  from a laser pulse to transient current generation

- Free charge carriers are put into motion

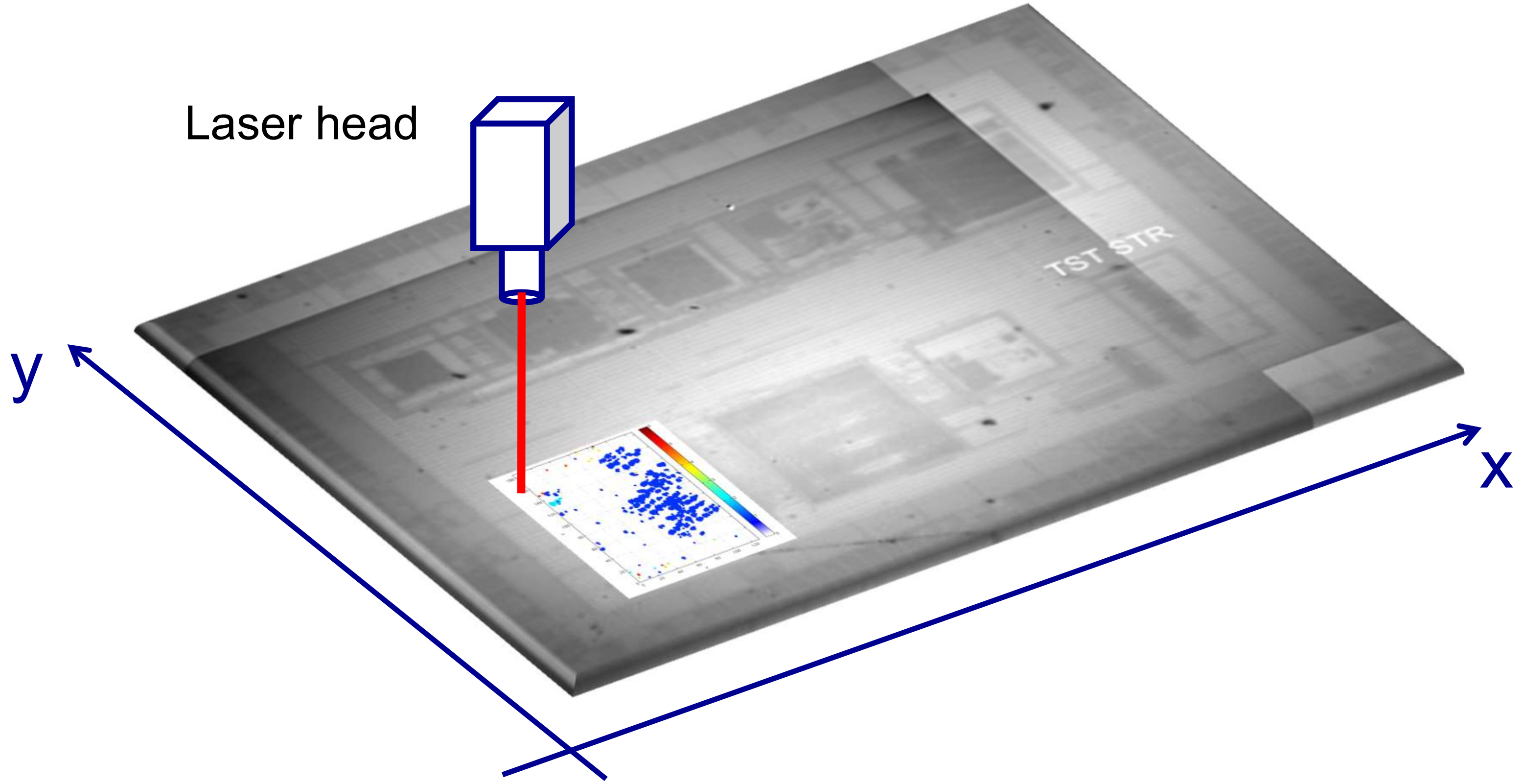
→ A transient current is induced I_{PH}



⇨ Reverse biased PN junction = laser sensitive parts of an IC

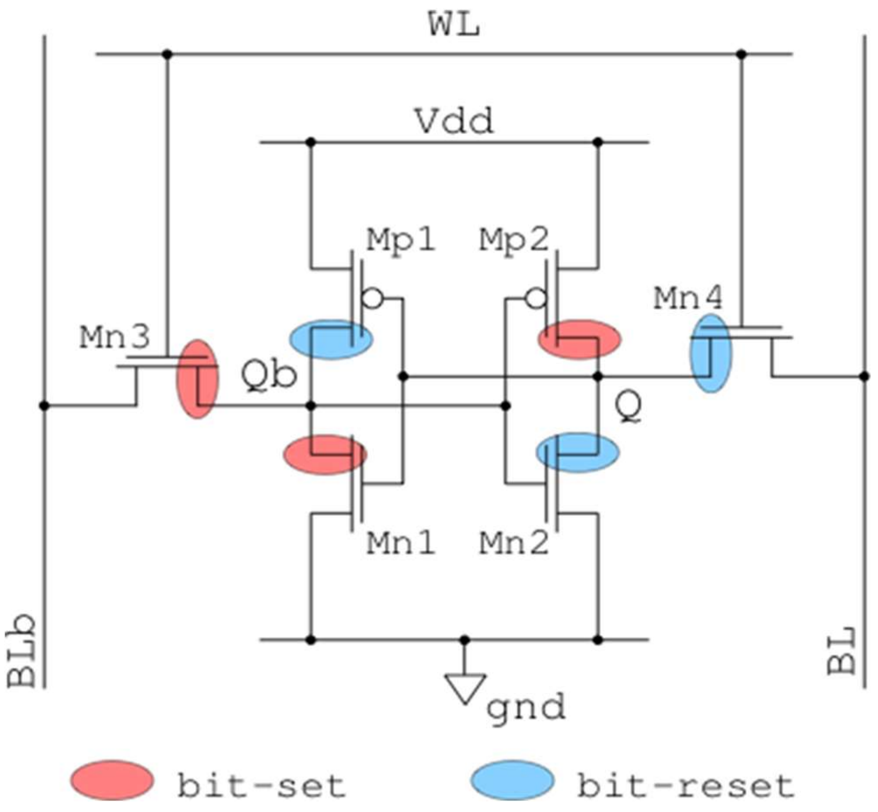
Laser fault injection attacks

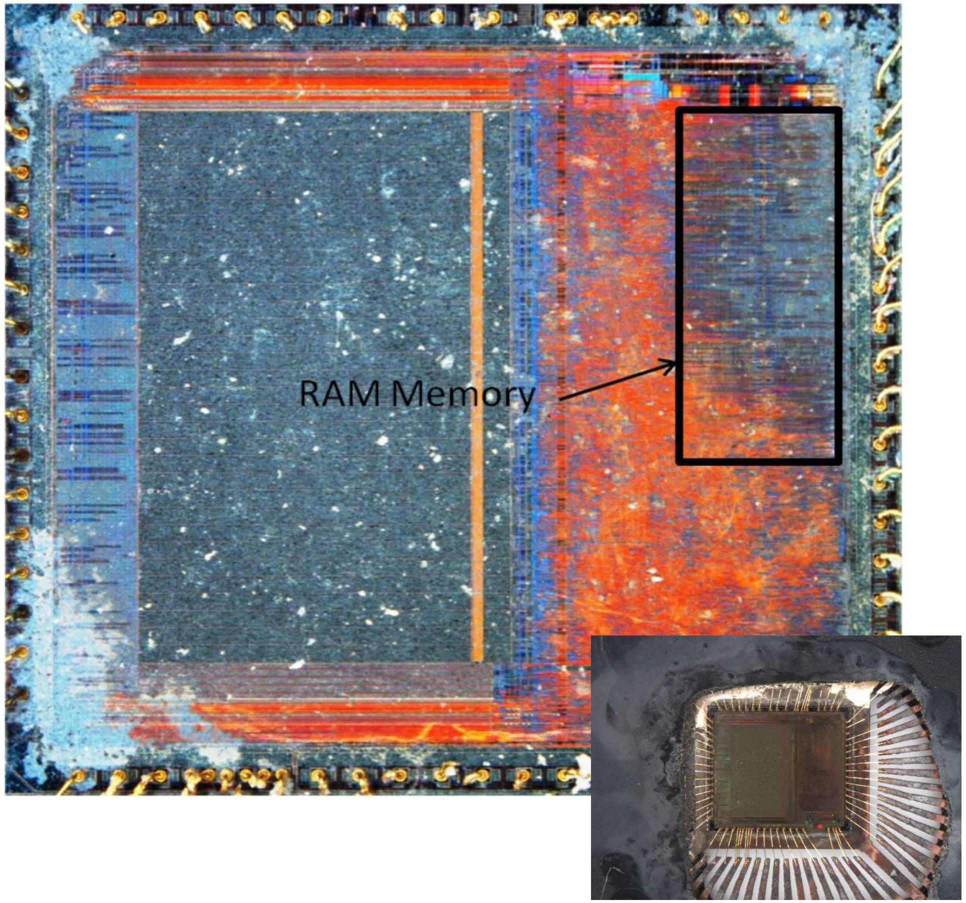
- Faulting data at rest
  - SRAM cells, registers, DFF



Laser fault injection attacks

- Faulting data at rest
  - RAM memory of an 8-bit µCTRL, CMOS 350 nm



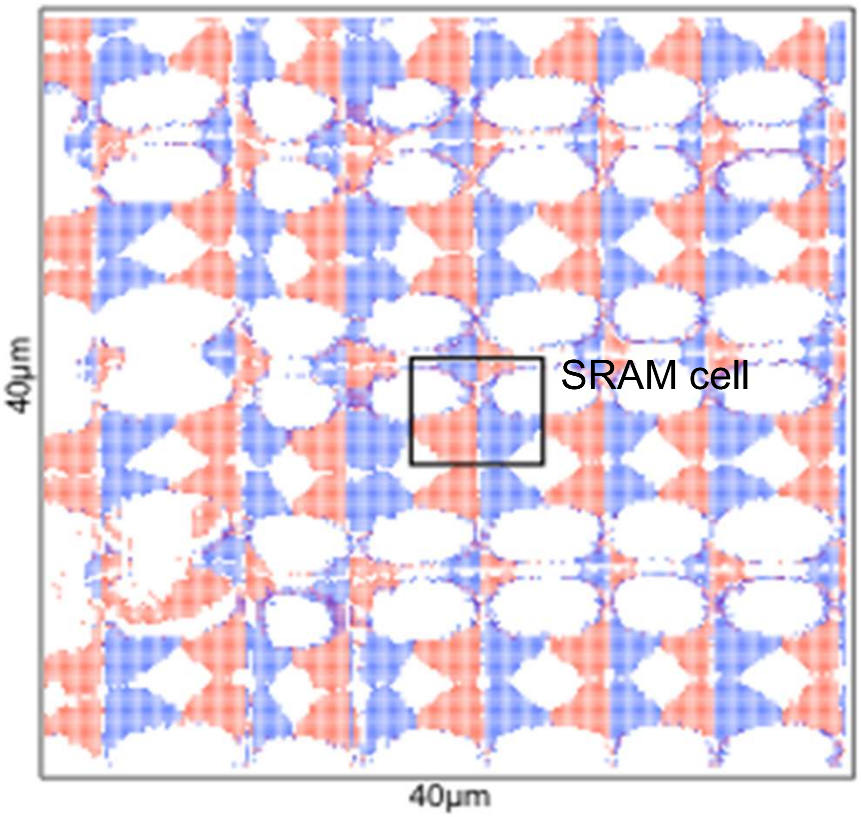


Laser fault injection attacks

- Faulting data at rest

- RAM memory of an 8-bit µCTRL, CMOS 350 nm

Static LFI – Parameters: 1 µm spot / 30 ps / 2.4 nJ / ∆xy = 0.2 µm / backside



● Bit-reset (1 → 0)

● Bit-set (0 → 1)

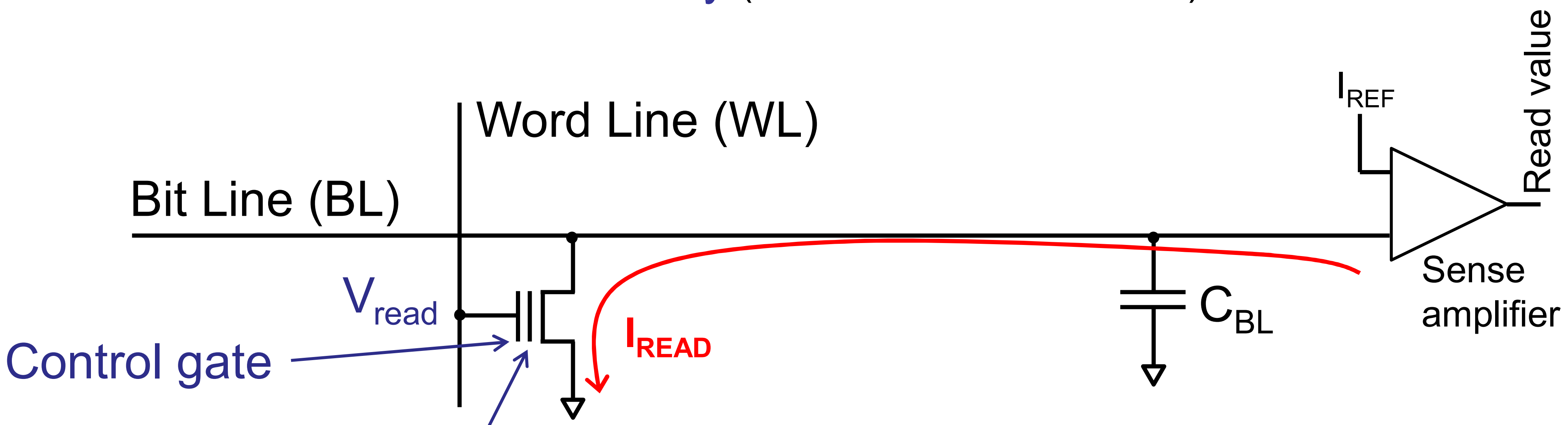
- LFI accurate and repeatable (100% success rate)
- In memory cells (SRAM, DFF)
  - Single-bit fault
  - Bit-set/reset FM (bit-flip also achievable)

Laser fault injection attacks

- Faulting data in motion
  - Combinatorial logic
  - Bit read from Flash memory (stored value unmodified)

Laser fault injection attacks

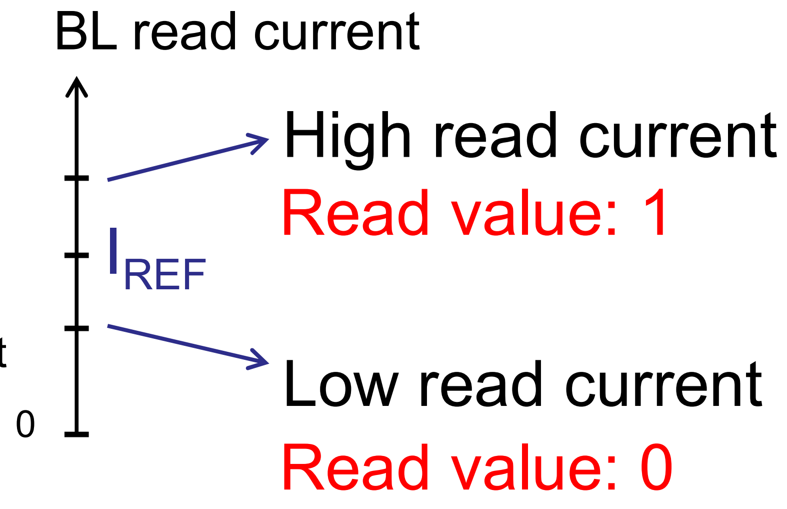
- Bit read from Flash memory (stored value unmodified)



Floating gate

Not charged (ie erased) → low V_t

Charged (ie programmed) → high V_t



High read current

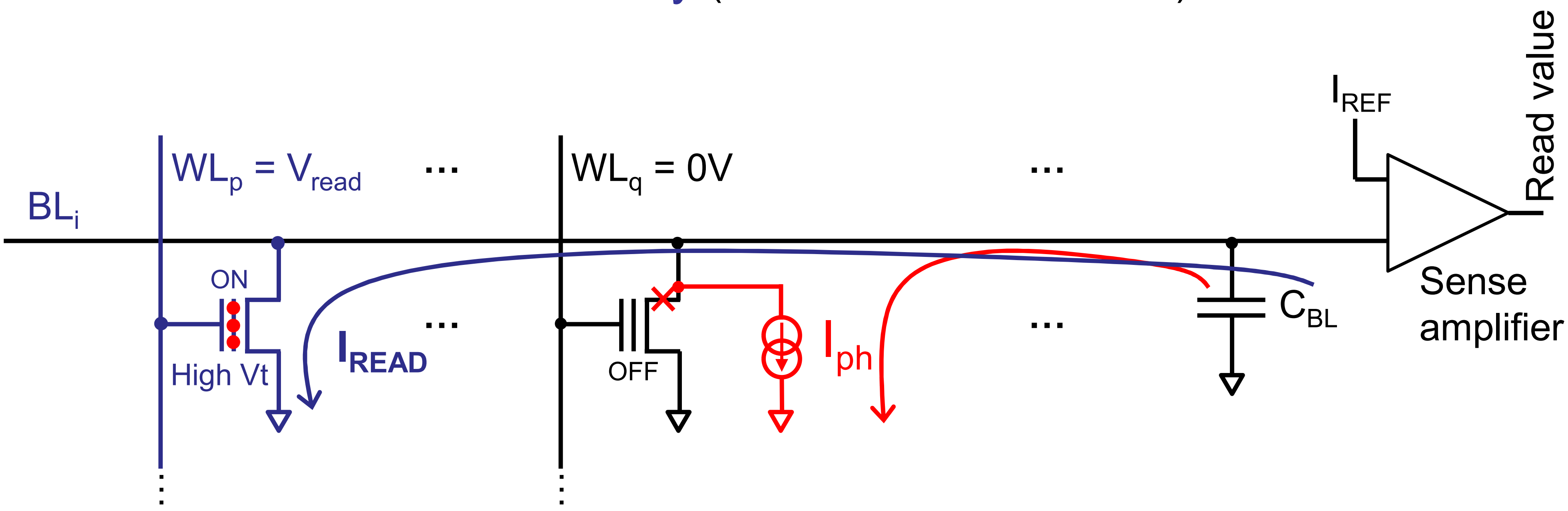
Read value: 1

Low read current

Read value: 0

Laser fault injection attacks

- Bit read from Flash memory (stored value unmodified)



- Floating gate T. prog. low read current → logic 0
- Additional I_{ph} current s.t. $I_{read} + I_{ph} > I_{REF}$ → logic 1

One-way (unidirectional) fault model

Bit-set fault model

× Laser hit

❑ Laser fault injection – wrap up

Accurate:

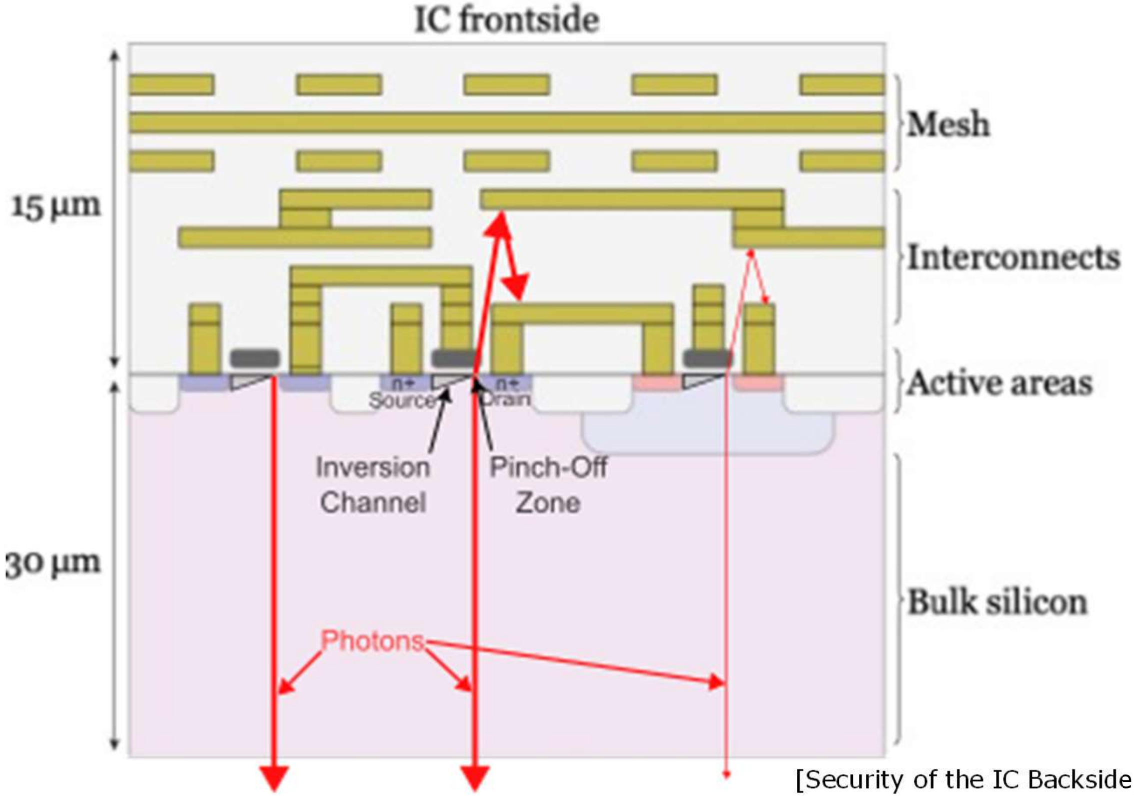
- from single-bit (local) to a wide area (spot size)
- with 100% repeatability
- in logic or memories
- in microcontrollers: instruction skip(s)

But:

- time consuming
- multi-space search: XYZ, timing, duration, power

❑ Photoemission analysis

- Source-drain electric field: charge carrier acceleration
- Kinetic energy released as photons
- NMOS$_{\text{emission}}$ > PMOS$_{\text{emission}}$
- Si substrate transparent to NIR
- Substrate thinning improves SNR

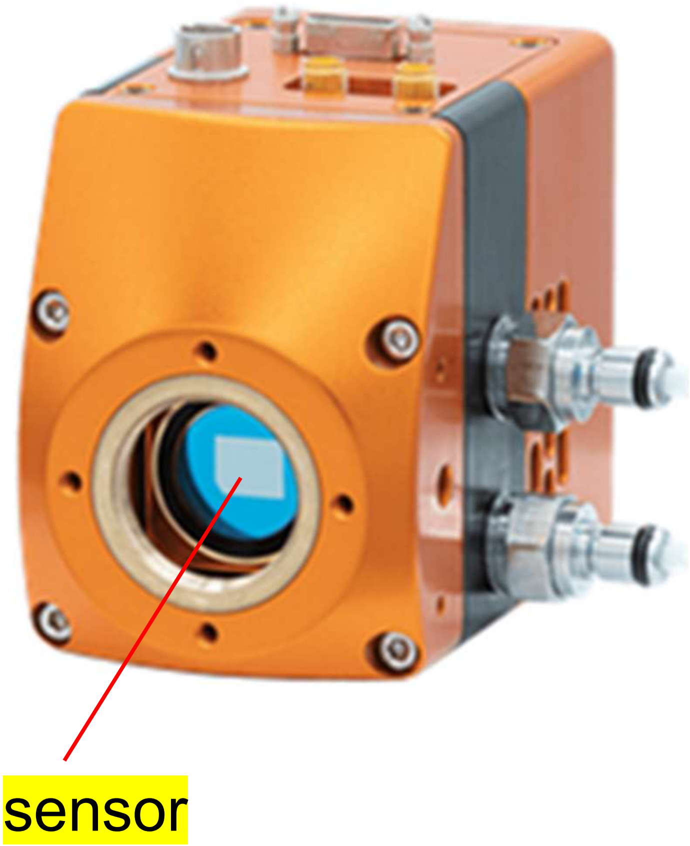


[Security of the IC Backside, D. Nedospasov, 2015]

500 to 1200 nm wavelength

Photoemission-based reverse engineering

❑ Camera: Ninox 640 II



- Typical readout noise (rms) : 18 e^-
- Typical dark current (@-15 °C) : < 750 e-
- 640x512 InGaAs sensor
- High sensitivity from 0.6 to 1.7 µm
- 15x15µm pixel pitch
- Peak Quantum Efficiency : >90% @ 1.3µm
- Air-cooled to -15 °C

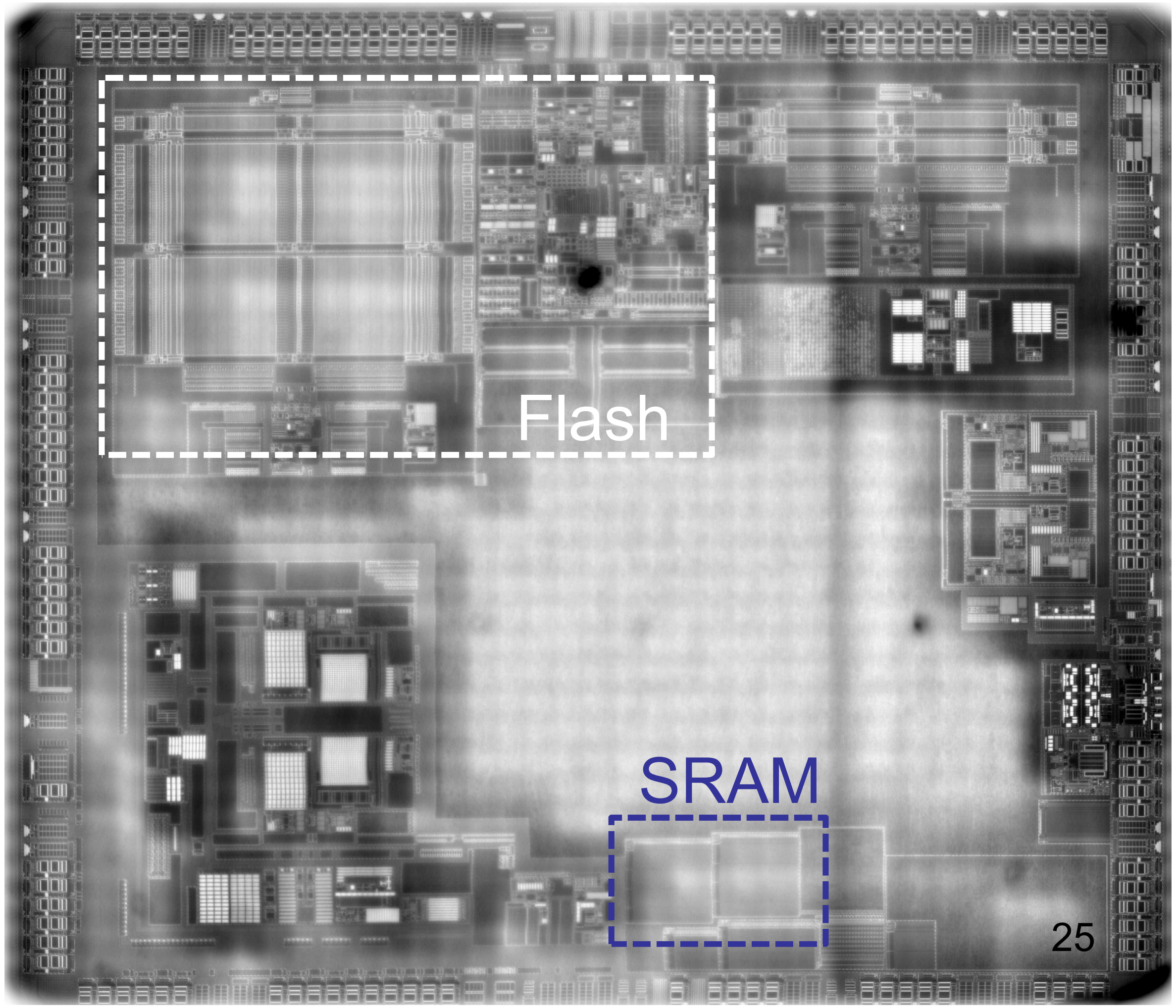
Photoemission-based reverse engineering

❑ Target

- Microcontroller: TM4C123GH6PM
  - ARM Cortex M4F
  - 32-bit CPU, 80 MHz
  - 256 kBytes Flash
    - page size = 1 kB
  - 32 kBytes SRAM
  - Si thickness: ~250 µm

(~ 50 µm when thinned down)

Backside IR view



Si die: 3,600 x 3,300 µm

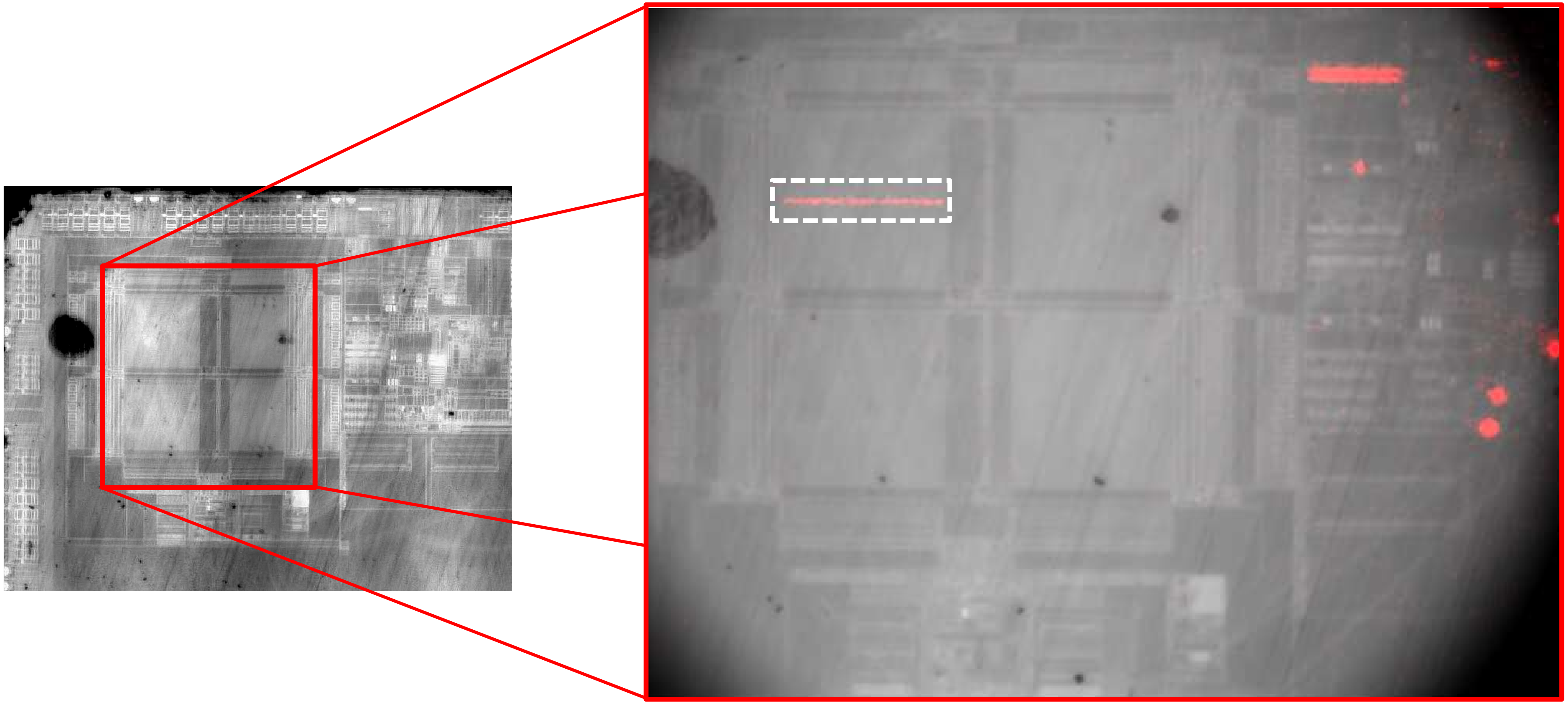
Flash: 330 x 310 µm (x4) – 5 bits/µm²

SRAM: 250 x 265 µm (x2) – 2 bits/µm²

Photoemission-based reverse engineering

❑ Flash memory

- Flash page location



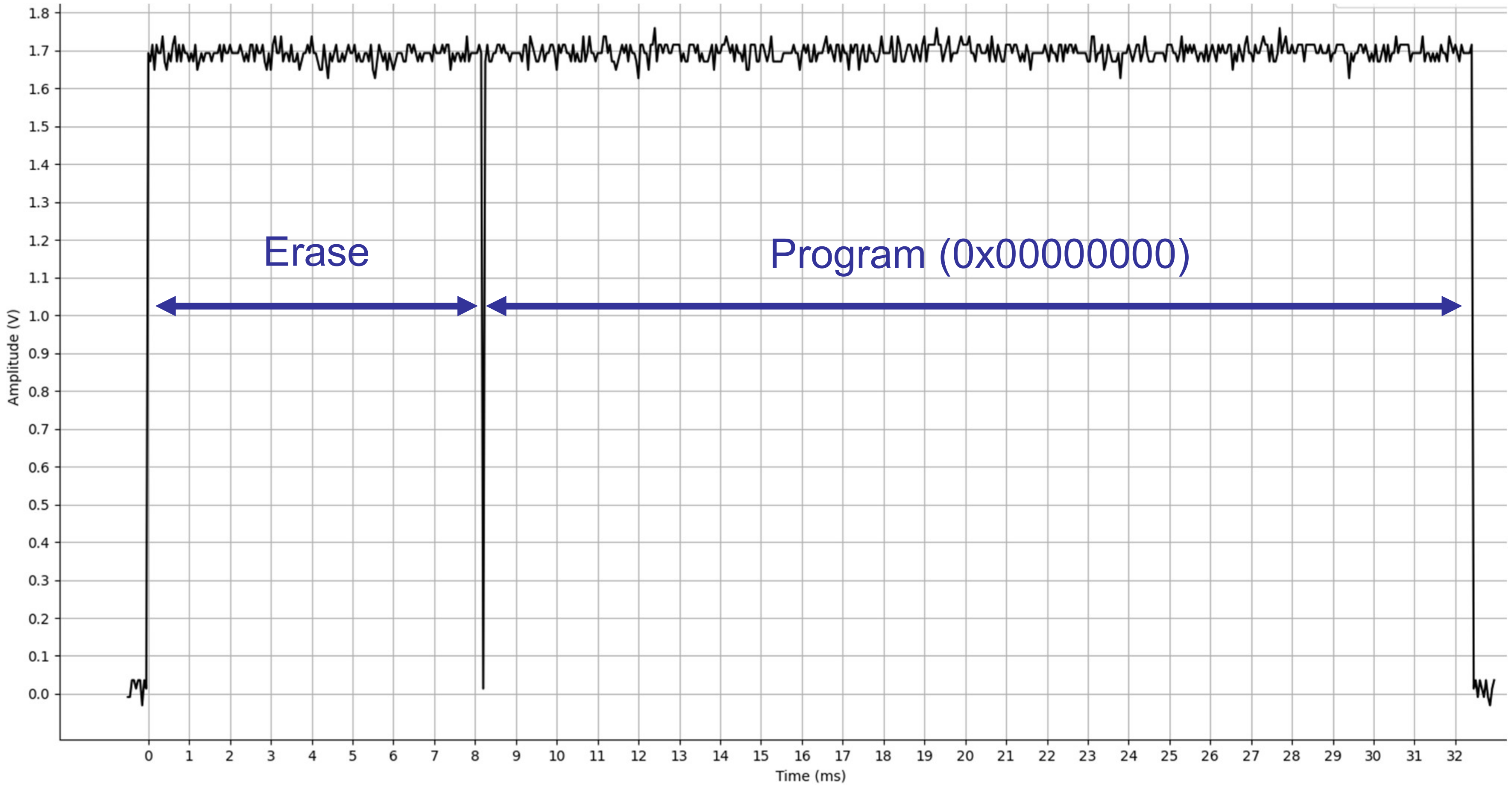
Photoemission map: erase + program cycles

Flash page #190, x5 lens, exposure 5 s

Photoemission-based reverse engineering

- Test code timing

- Erase + Program cycle time = 32 ms, i.e. ~150 cycles in 5 s



Photoemission-based reverse engineering

- Flash memory modes of operation: erase & program

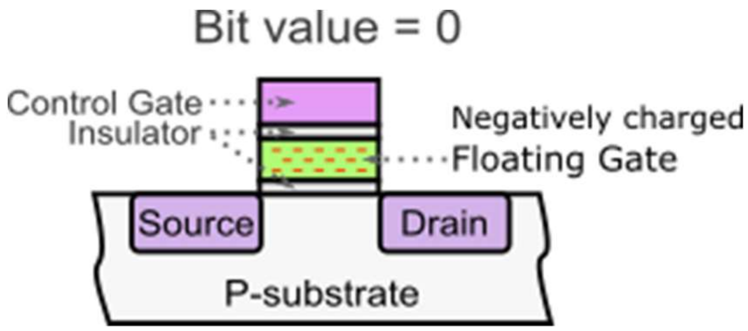
Writing in an embedded Flash is a complex 2 steps process

- Flash memories are …
- … erased at page level (e.g. 1 kB)
- Fowler-Nordheim tunneling effect

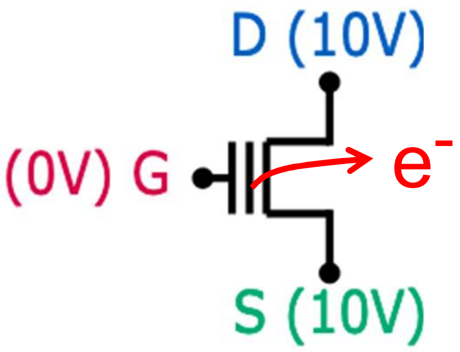
→ Set to 1 (or 0xFFFFFFFF at word level)

- … programmed (i.e. written) at word level
- Using channel-hot-electron injection

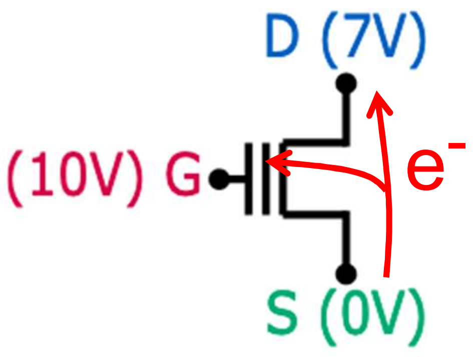
→ Set to 0 (or 0x00000000 at word level)



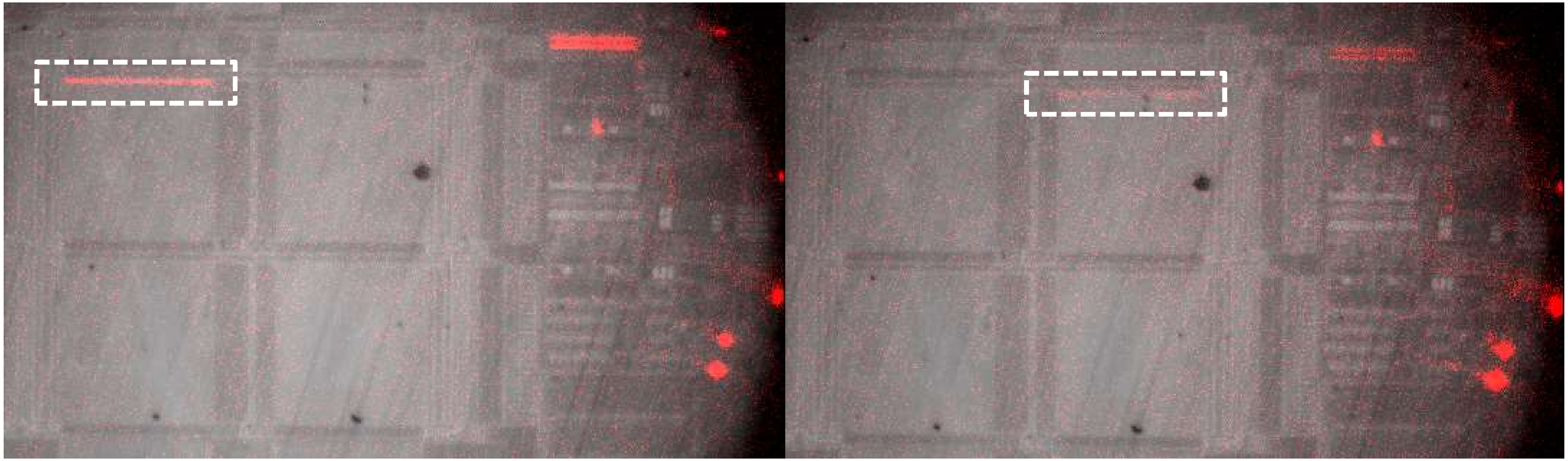
Erase



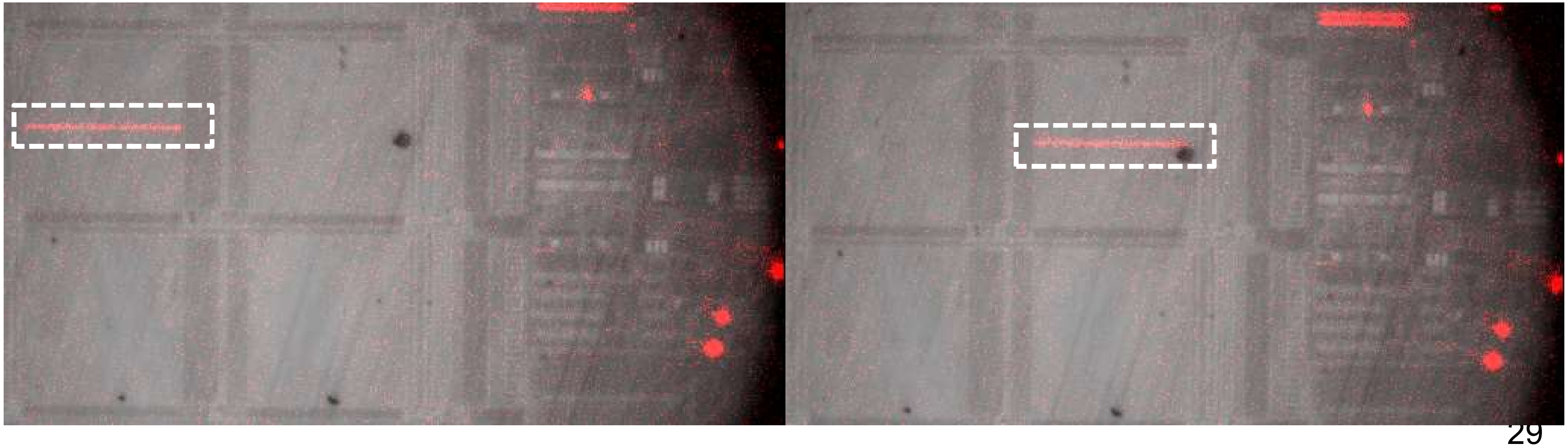
Program



Photoemission map: erase + program cycles



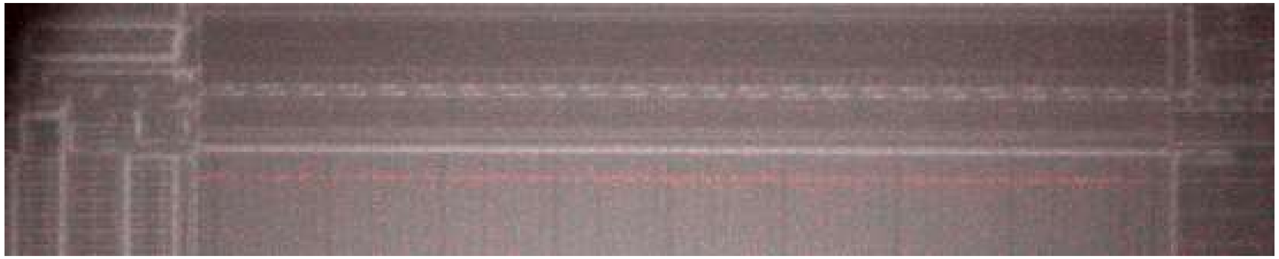
Flash page #254 (left) & #255 (right), x5 lens, exposure 5 s



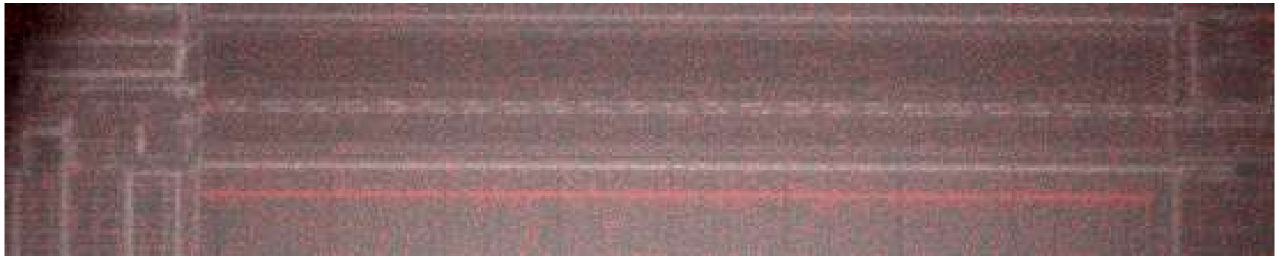
Flash page #190 (left) & #191 (right), x5 lens, exposure 5 s

Photoemission map: erase + program cycles

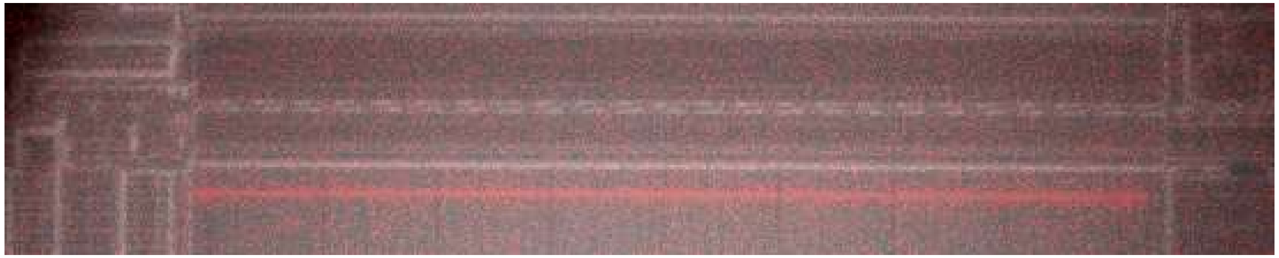
Intensity



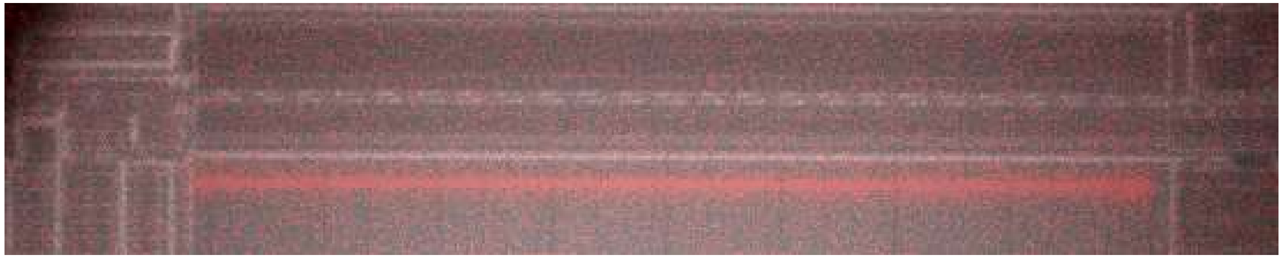
Page #254, 8 words



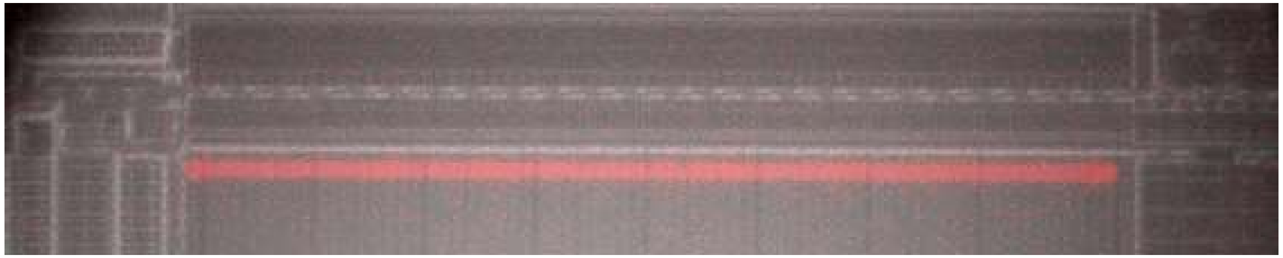
Page #254, 16 words



Page #254, 32 words



Page #254, 64 words

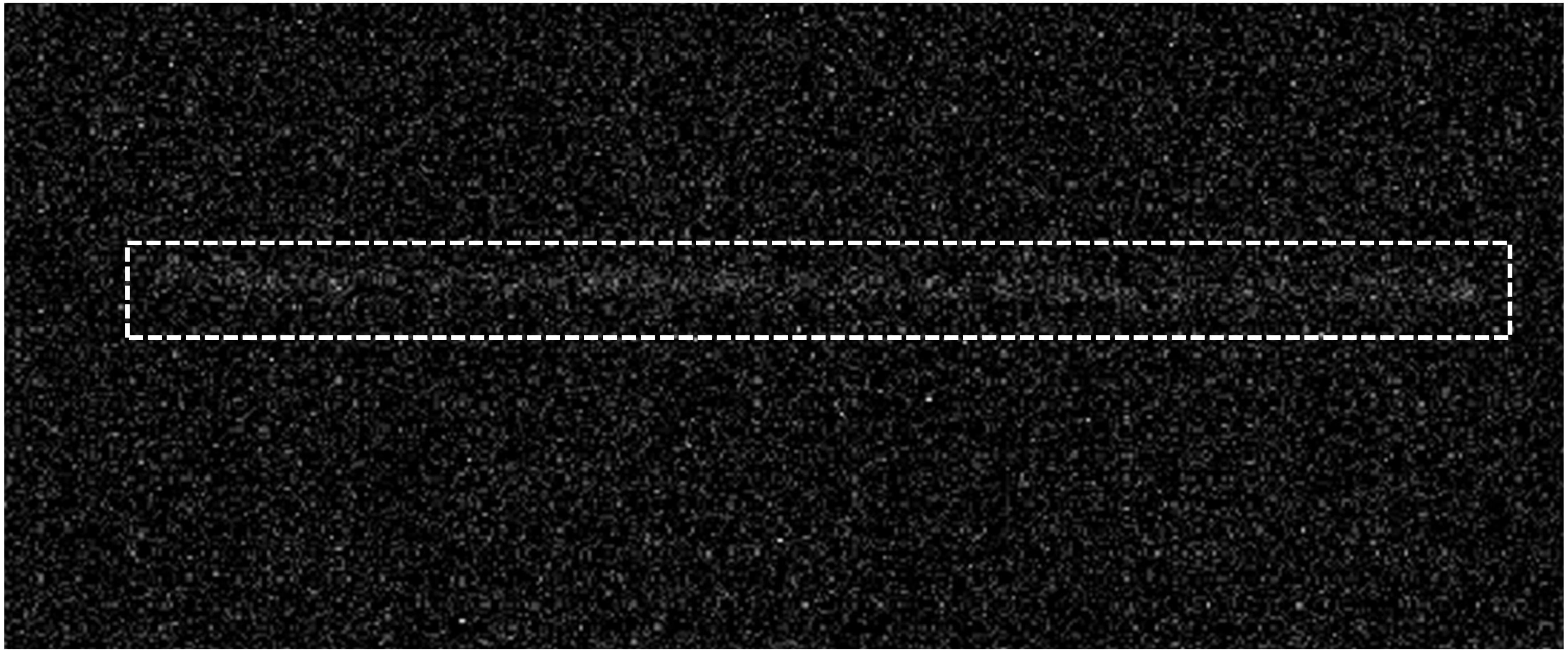


Page #254, 128 words

Photoemission map: erase + program cycles

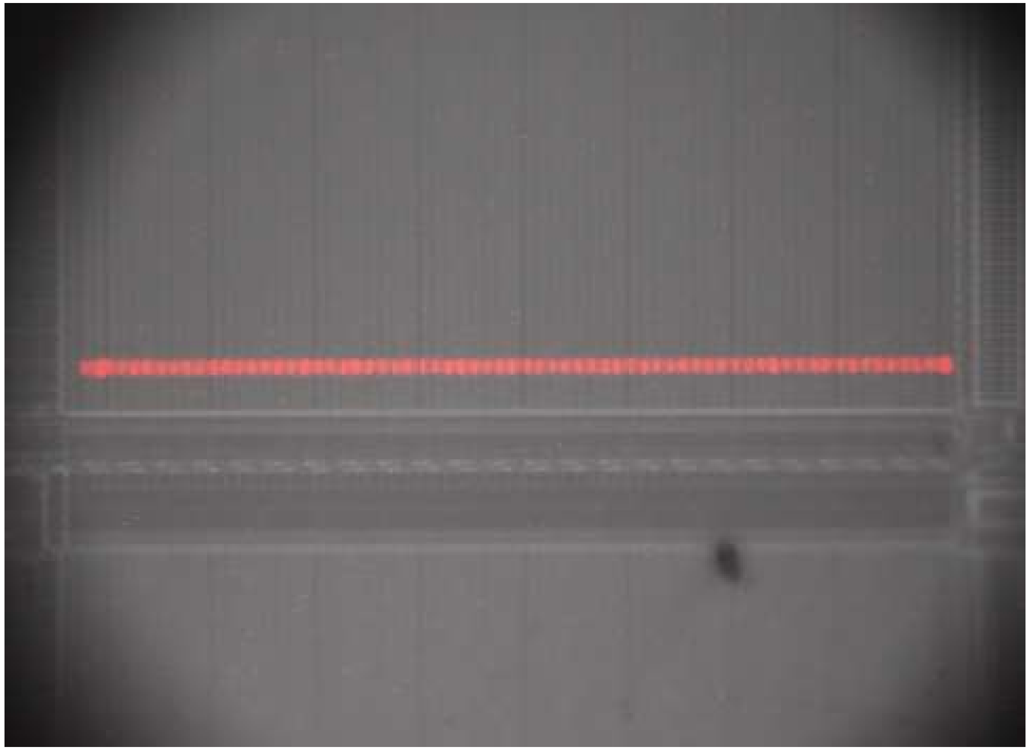
- Number of erase + program cycles needed for the information to emerge from noise?

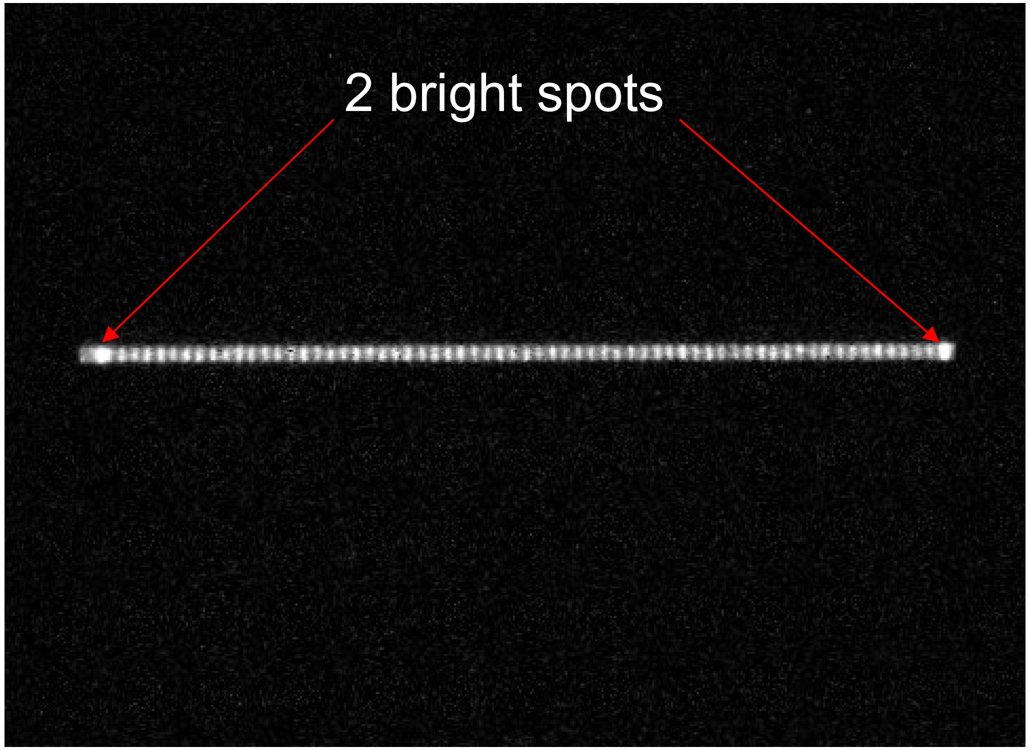
Photoemission map: 10 cycles, 500 ms, Page #255



Photoemission map: erase + program cycles

- Data dependency
  Target thinned down to 50 µm



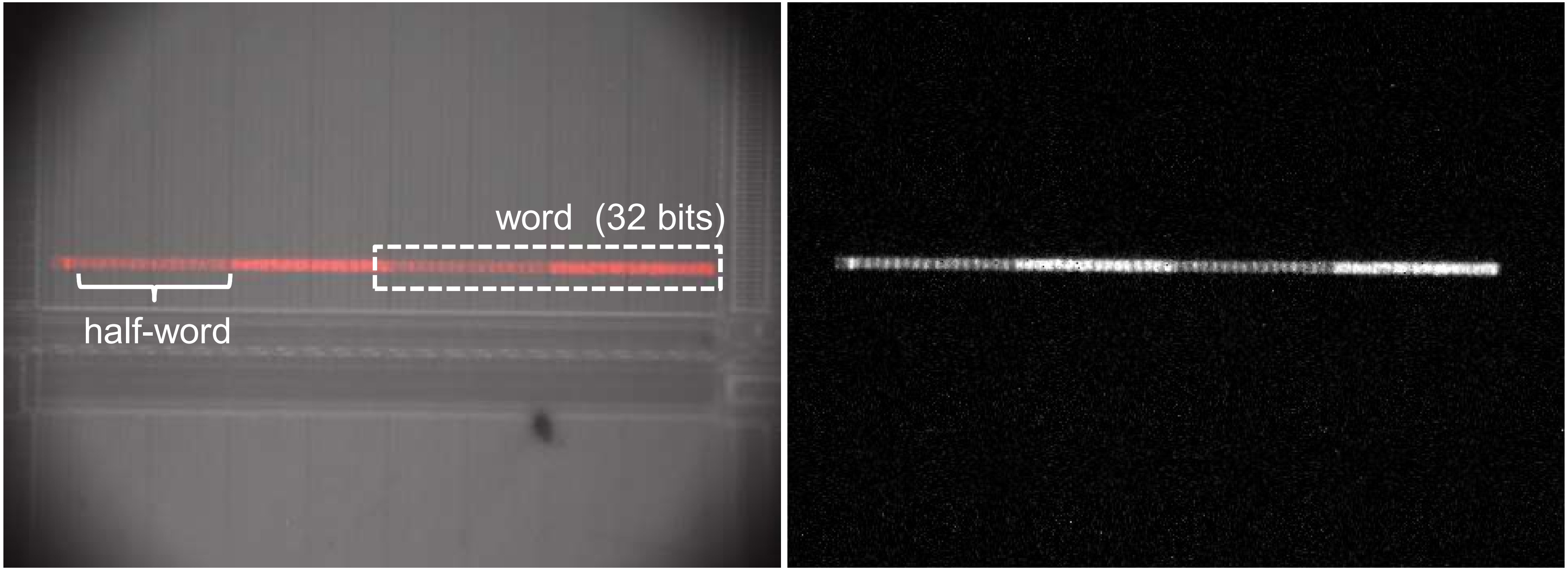


Page #120, 20x lens, exposure 2.5 s, program 0X00000000

Overlay (left) & camera output (right)

Photoemission map: erase + program cycles

- Data dependency

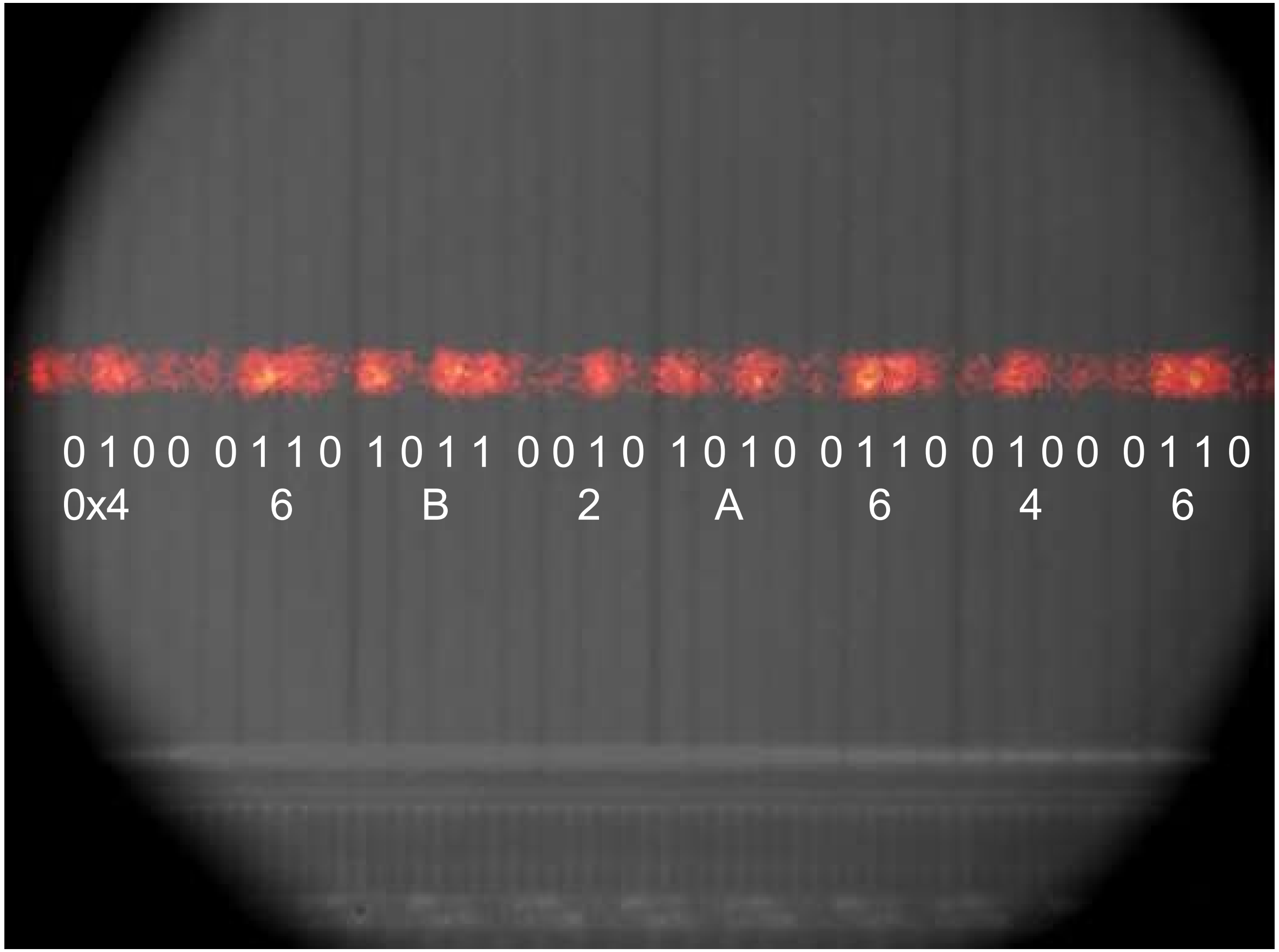


Page #120, 20x lens, exposure 2.5 s, program 0x0000FFFF

Overlay (left) & camera output (right)

Photoemission map: erase + program cycles

- Data dependency

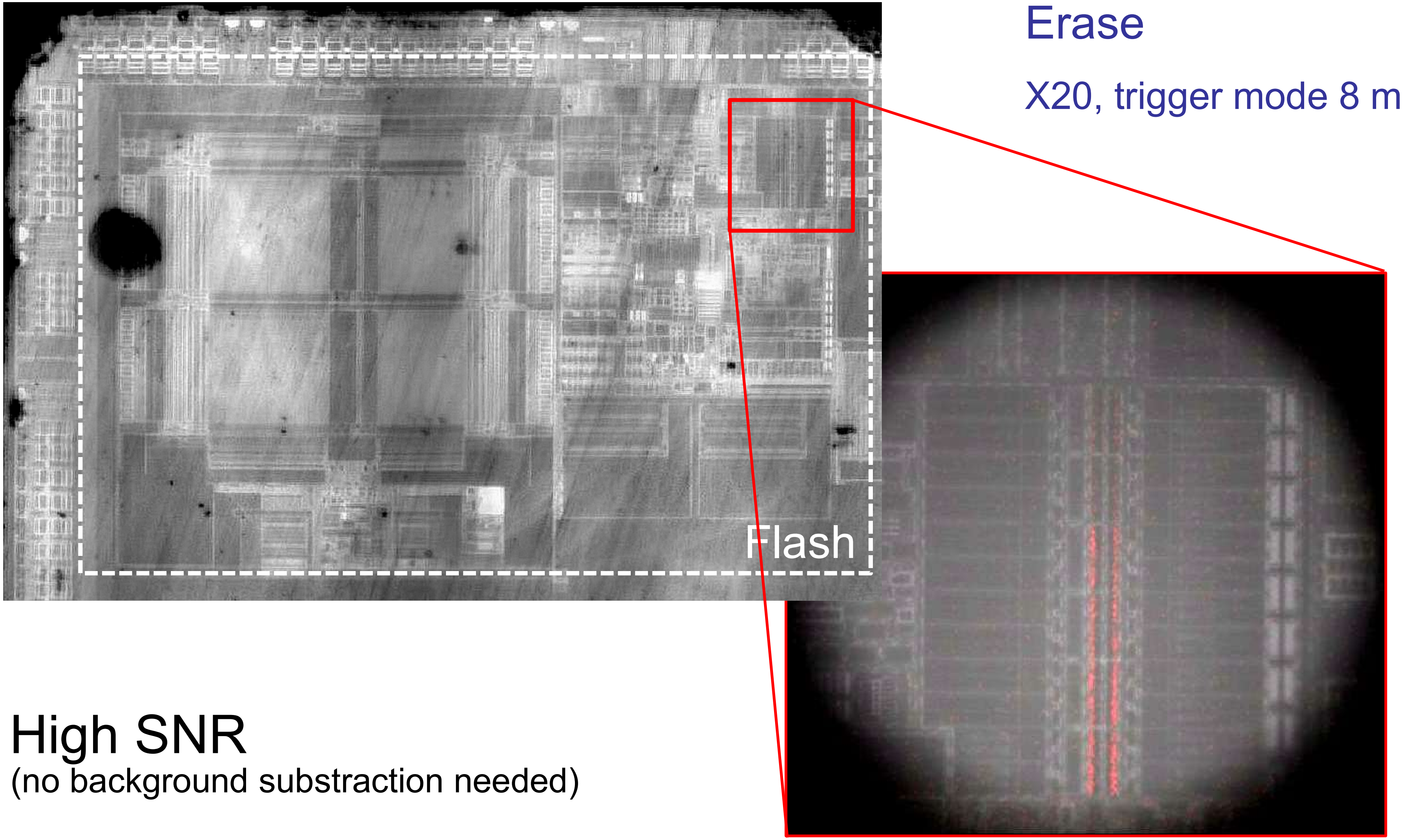


128x32-bit words

Page #120, 50x lens, exposure 2.5 s, program 0x46B2A646, Overlay

Trigger mode (ie one operation capture)

- Charge pump identification



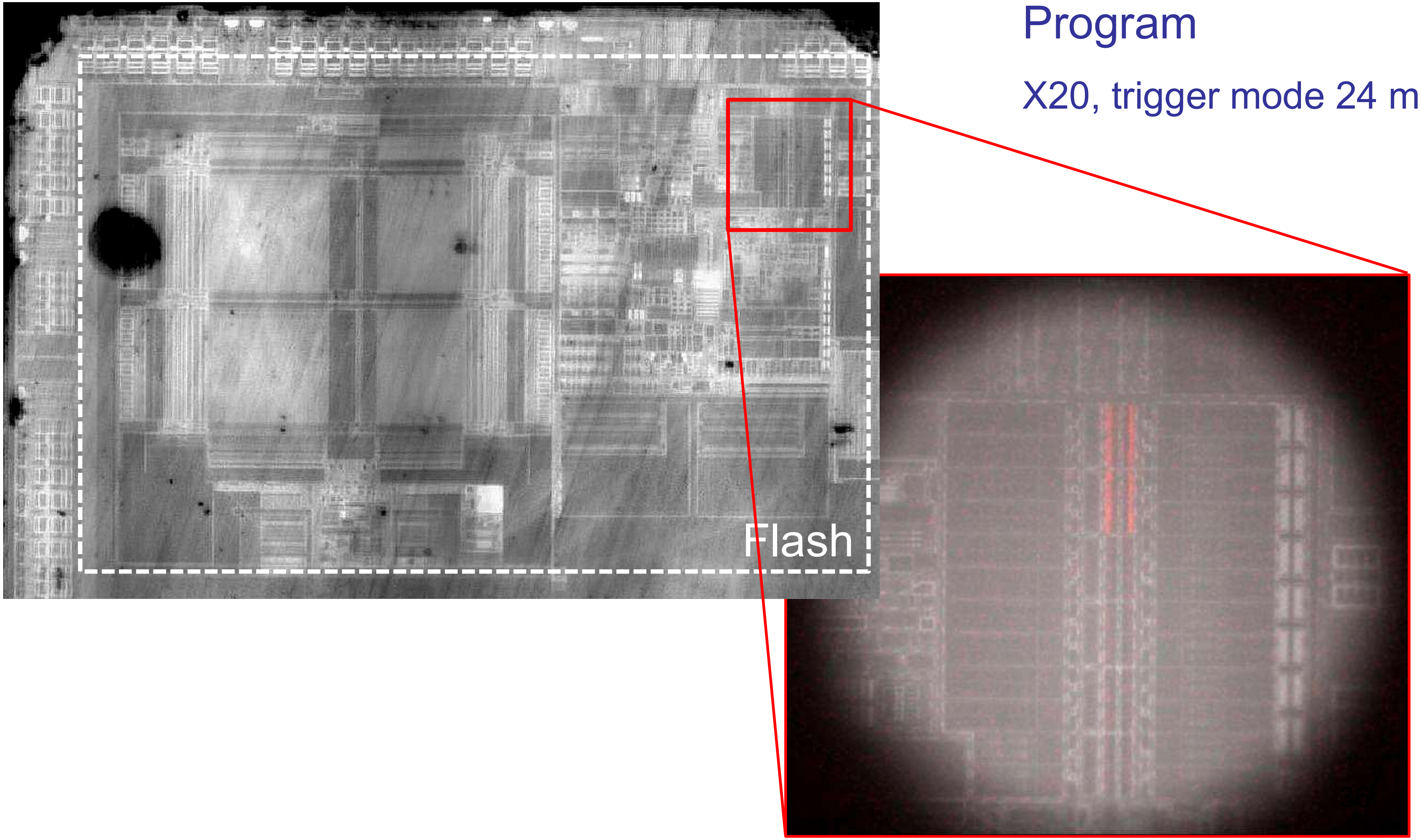
Erase

X20, trigger mode 8 ms

High SNR
(no background substraction needed)

Trigger mode (ie one operation capture)

- Charge pump identification

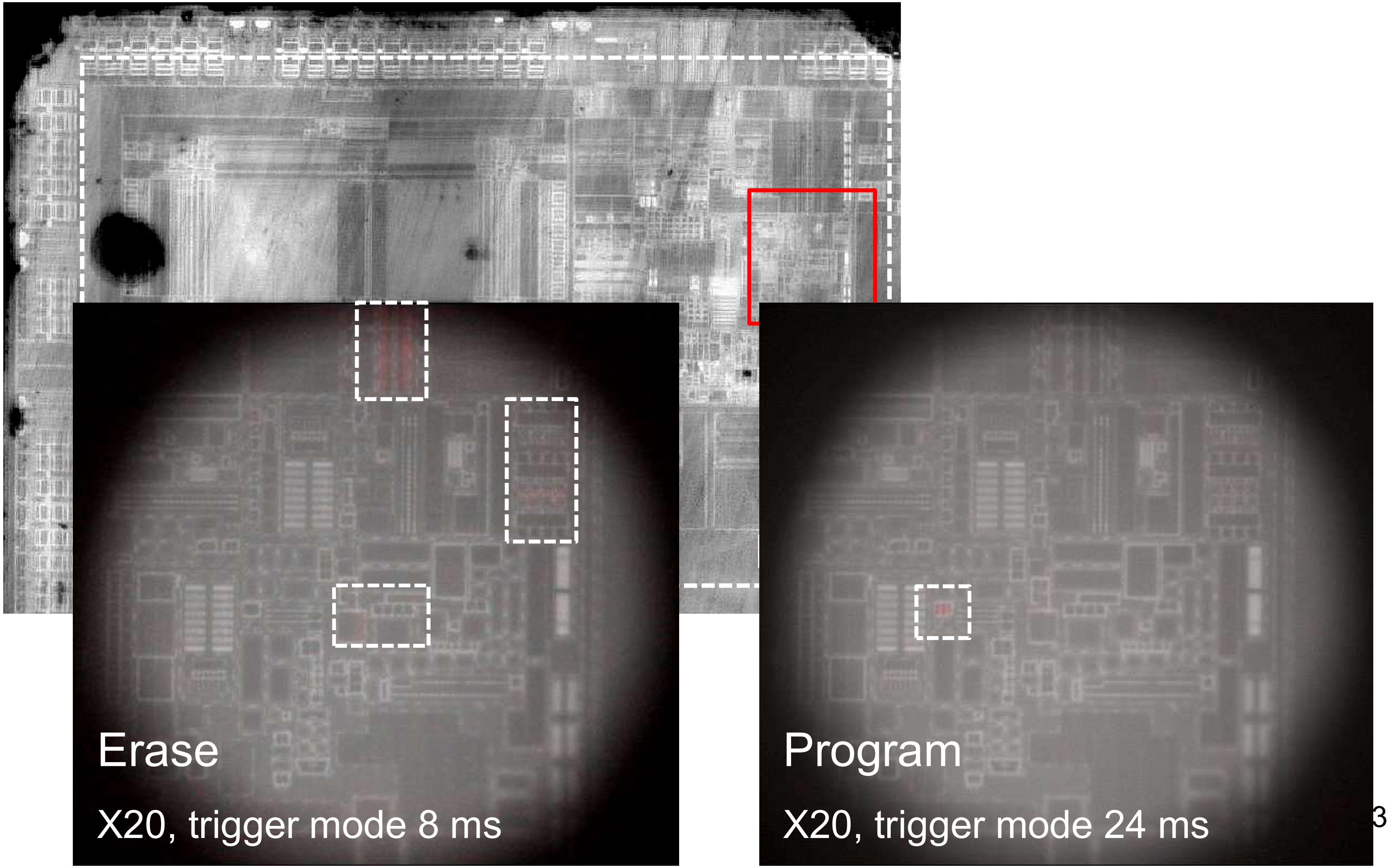


Program

X20, trigger mode 24 ms

Trigger mode (ie one operation capture)

- Charge pump logic identification



Erase

X20, trigger mode 8 ms

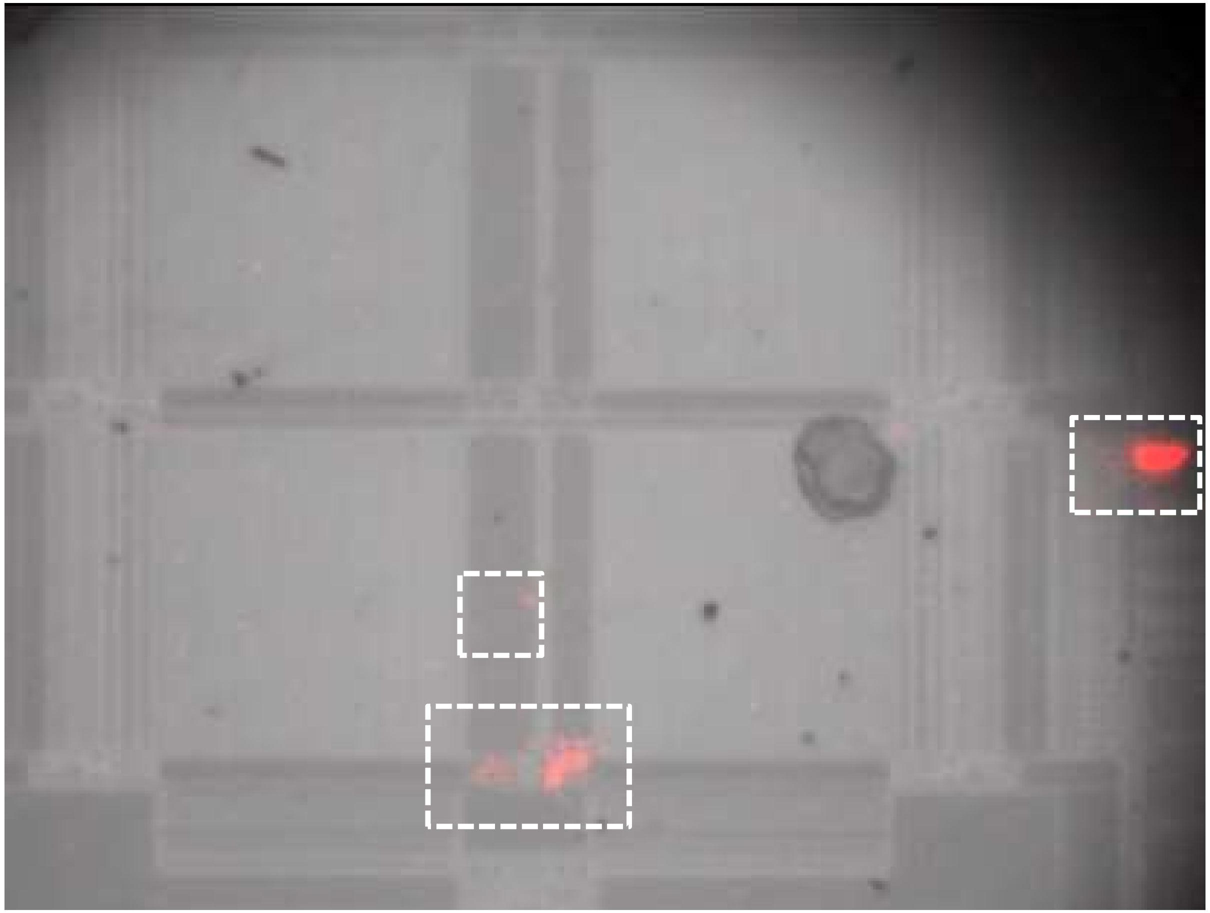
Program

X20, trigger mode 24 ms

Flash photoemission at read time

❑ Photoemission at read time

- Nothing to be seen in the floating gate transistors matrix
- Reading page #255



Flash page #255

x5 lens

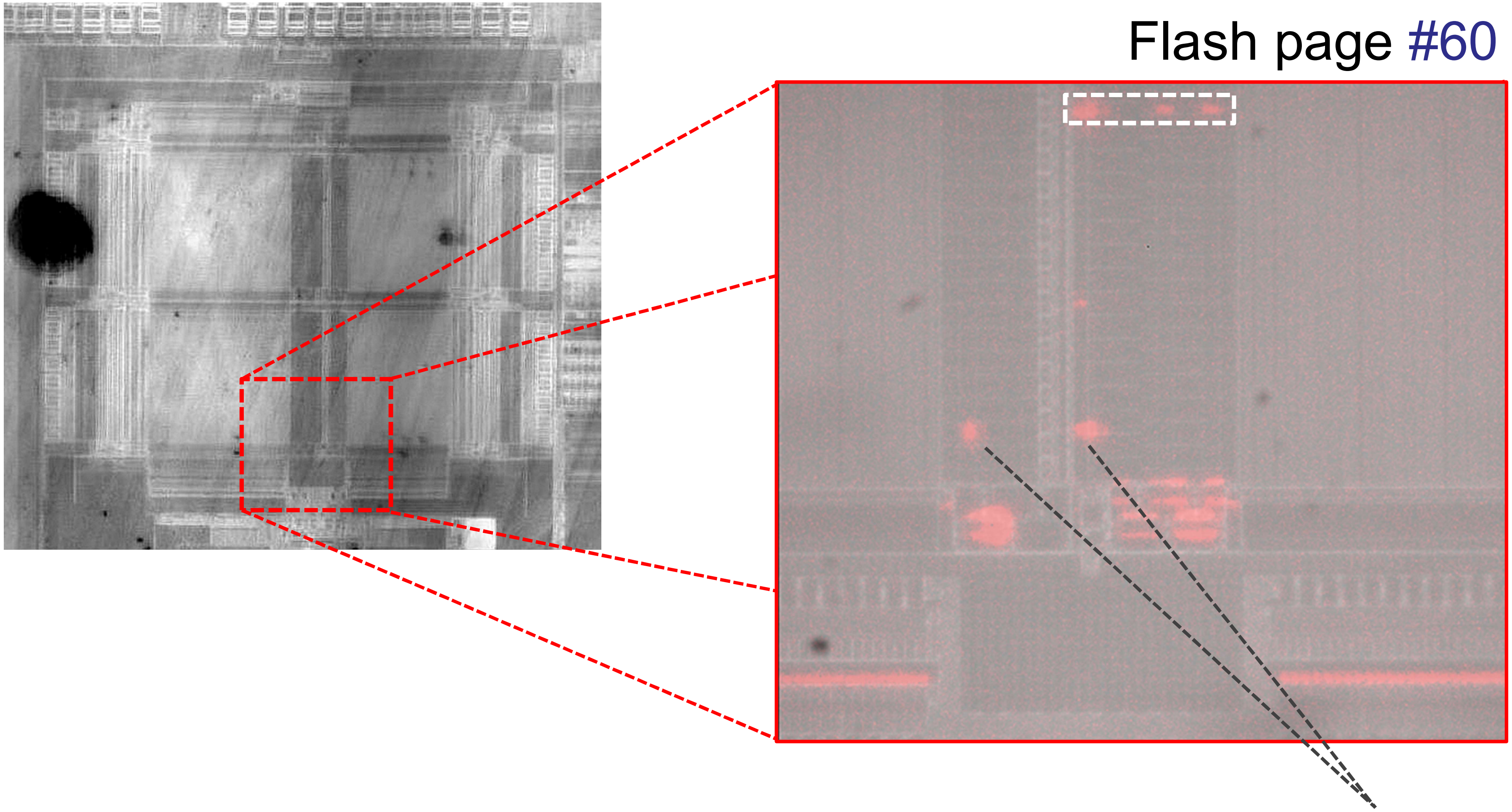
exposure 5 s

(whole page read time 57.8 µs)

Flash photoemission at read time

❑ Photoemission at read time

- Addressing logic (x20 lens, exposure 2.5 s)

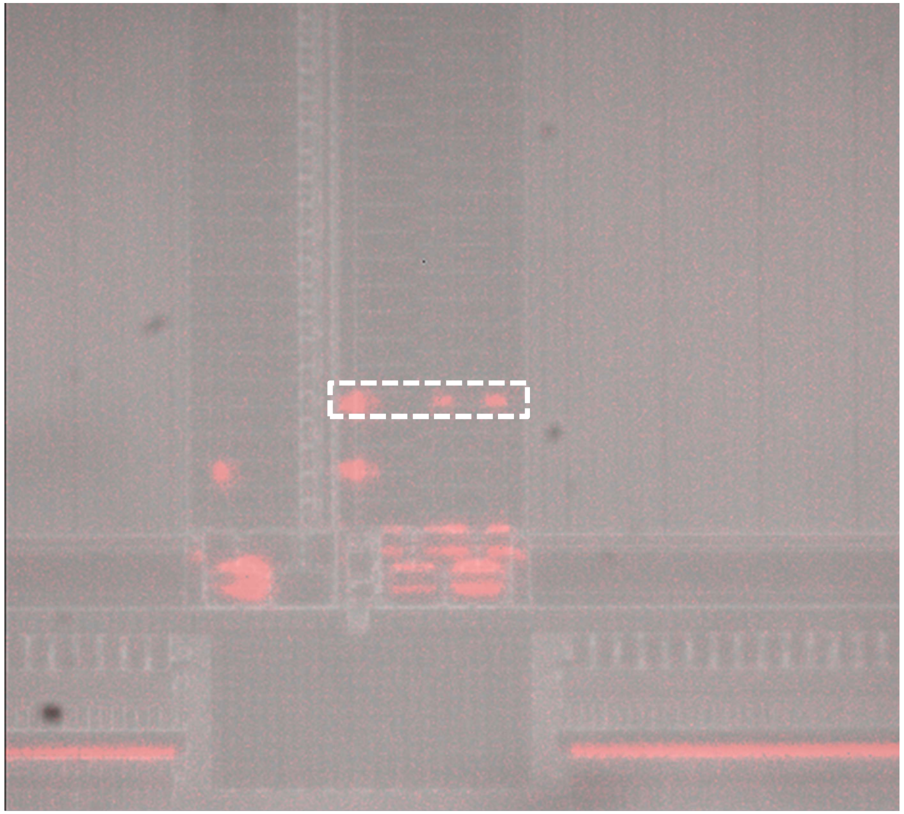


Flash page #60

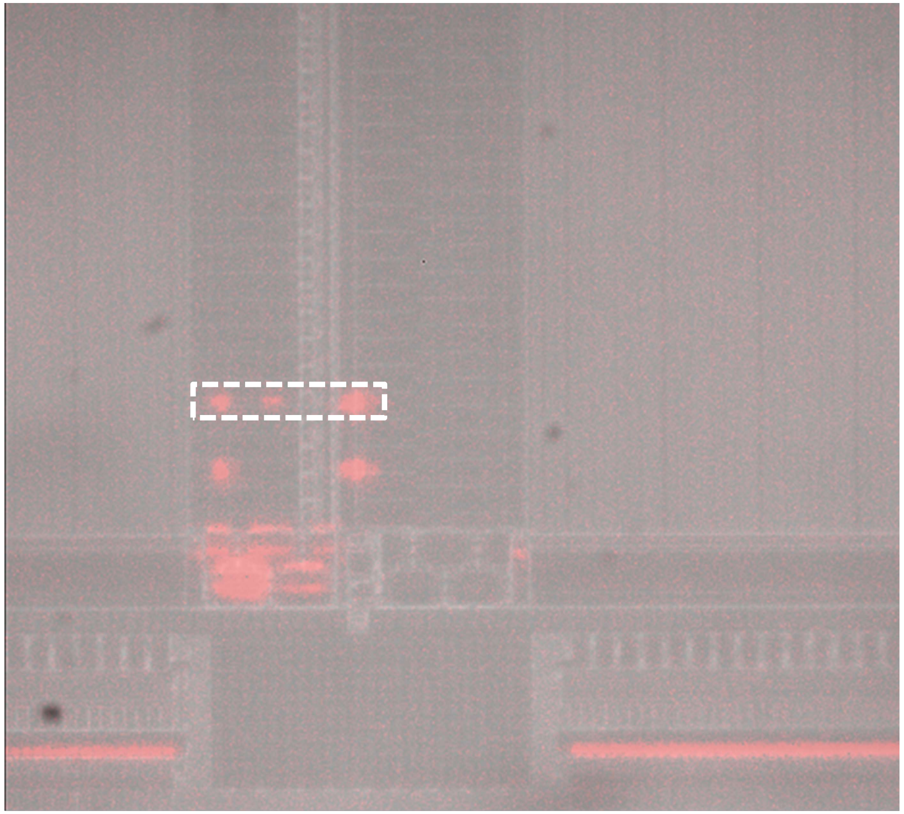
Program executed from page #5

- ❑ Photoemission at read time
  - ▪ Addressing logic (x20 lens, exposure 2.5 s)

Flash page #16



Flash page #17



SRAM memory photoemission

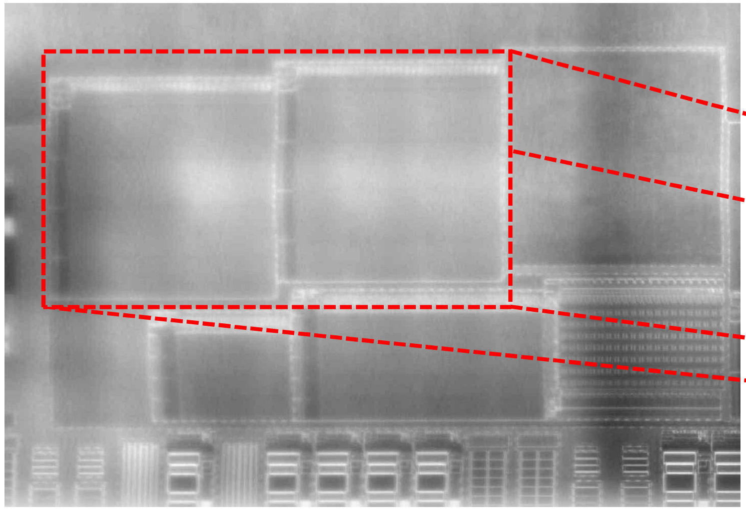
❑ SRAM memory

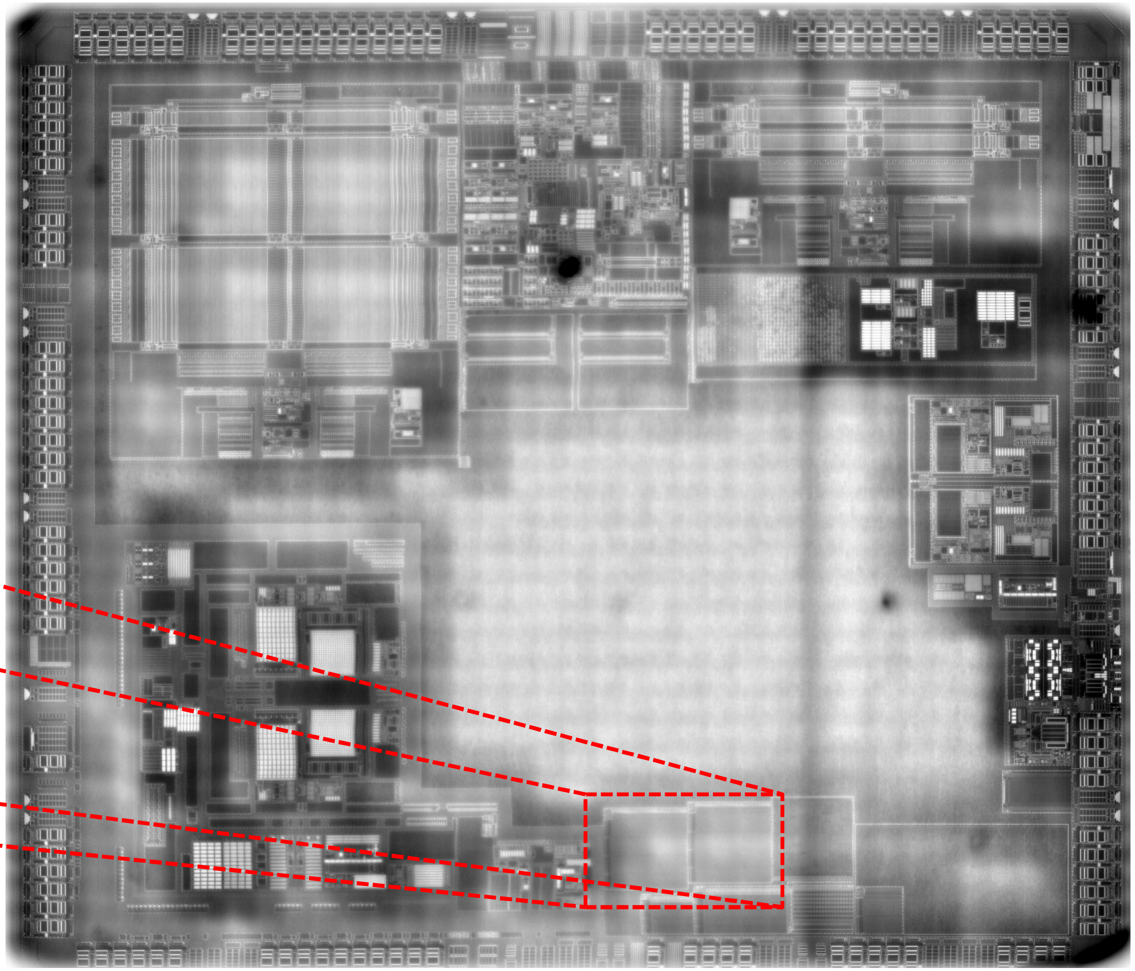
- 2x 16 kBytes SRAM

Left even @

Right odd @

0x20000000 – 0x20007FFF



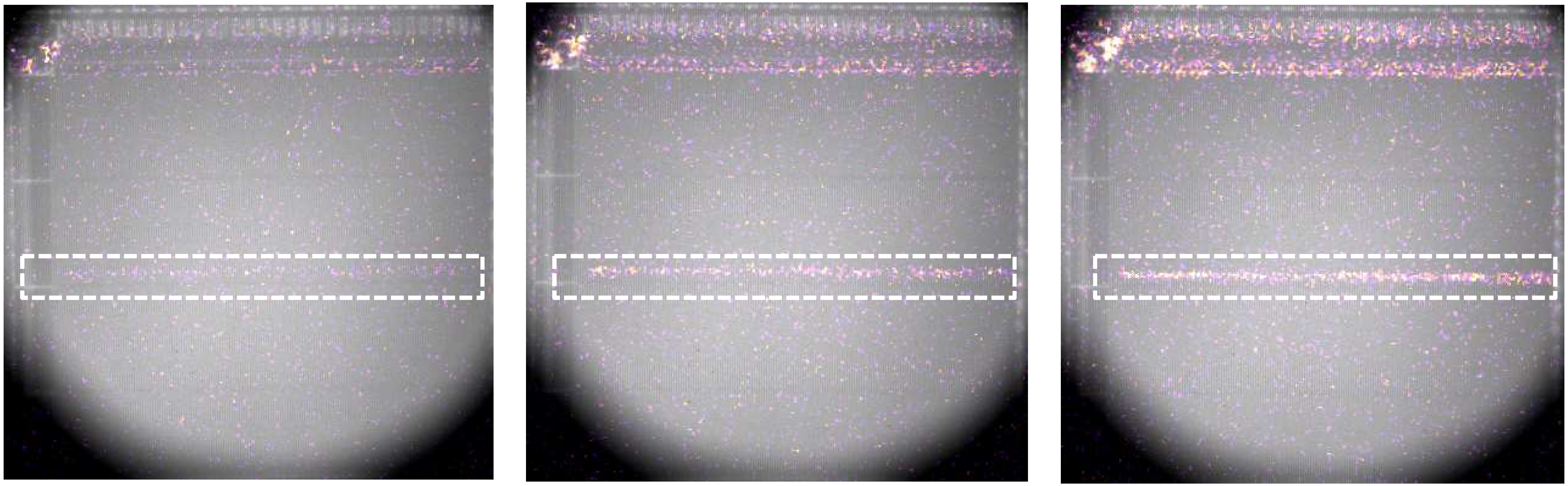


SRAM memory photoemission

❑ Photoemission at write time

Test code: write 0x00000000, then 0xFFFFFFFF

1 cycle ~550 ns

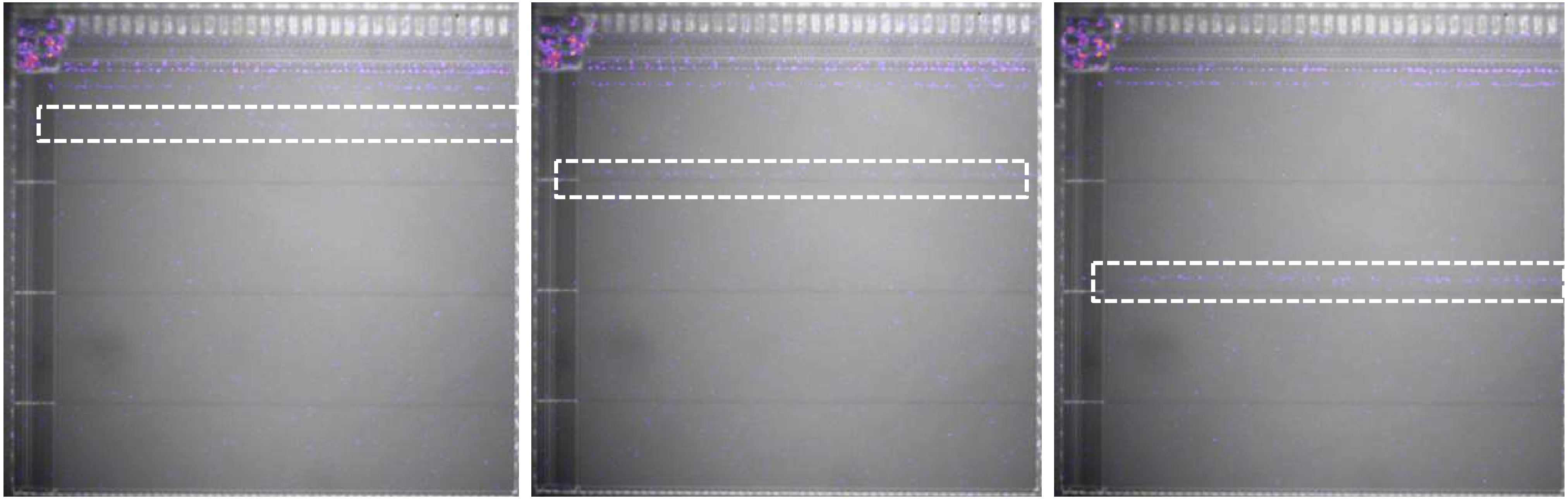


Photoemission map: 20x lens, exposure 5s, @: 0x20004000

1 word, 8 words, 64 words (left to right)

SRAM memory photoemission

- Photoemission at read time

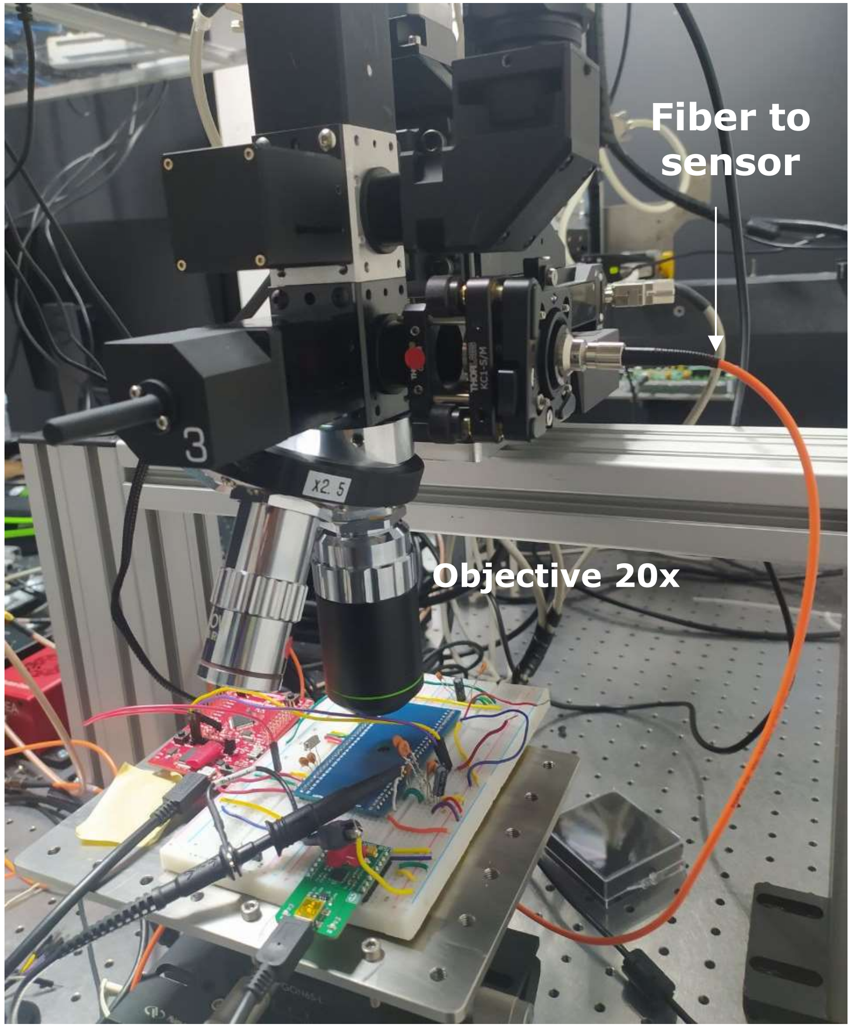


Photoemission map at read time: 20x lens, exposure 5s
@: 0x20001000 - 0x20003000 - 0x20004000 (left to right)

Weak emission at read time

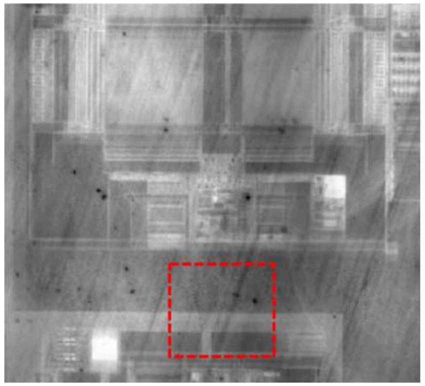
❑ Timing photoemission: observing one point

- Avalanche photodiode: InGaAs sensor
- Photons (0.9 to 1.1 µm) → e^- cascade in sensor
- Photoemission vs time
- Time: 32,768 bins (250 ps/bin min.)
- Measure on trigger

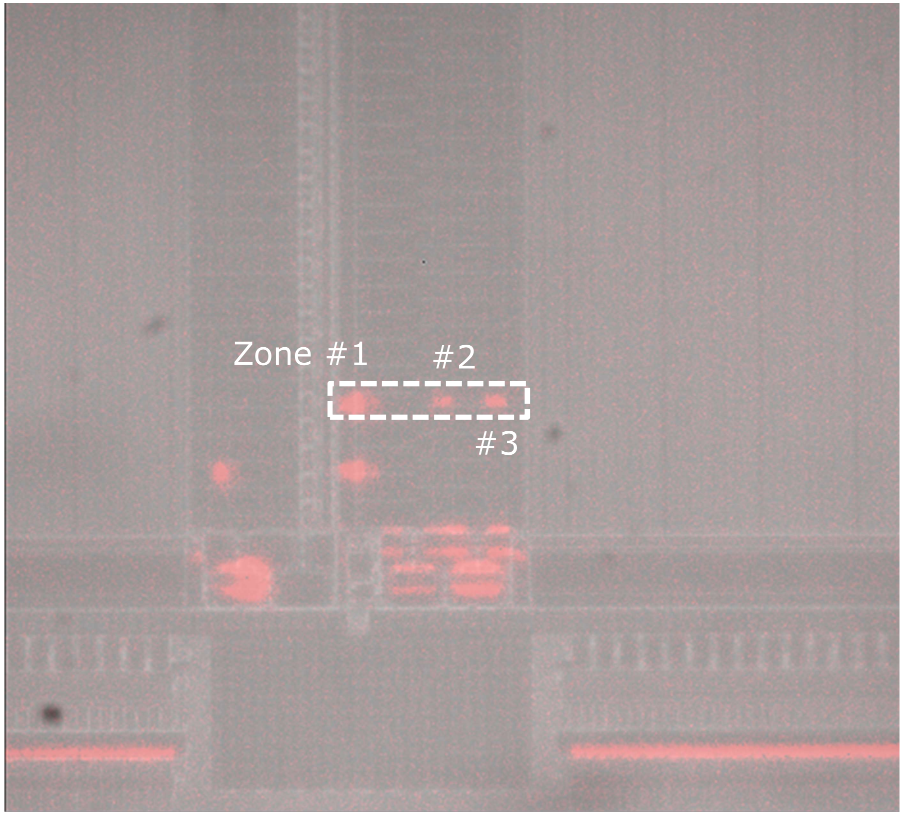


SPD OEM NIR sensor

- Flash memory at read time, addressing logic



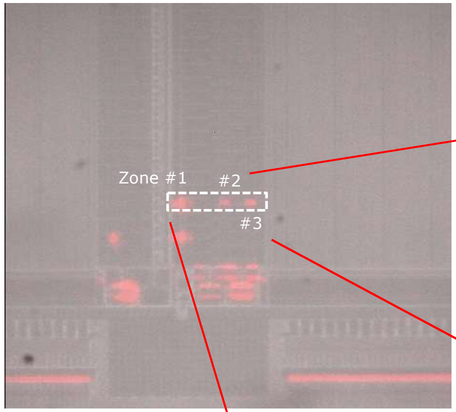
Flash page #16

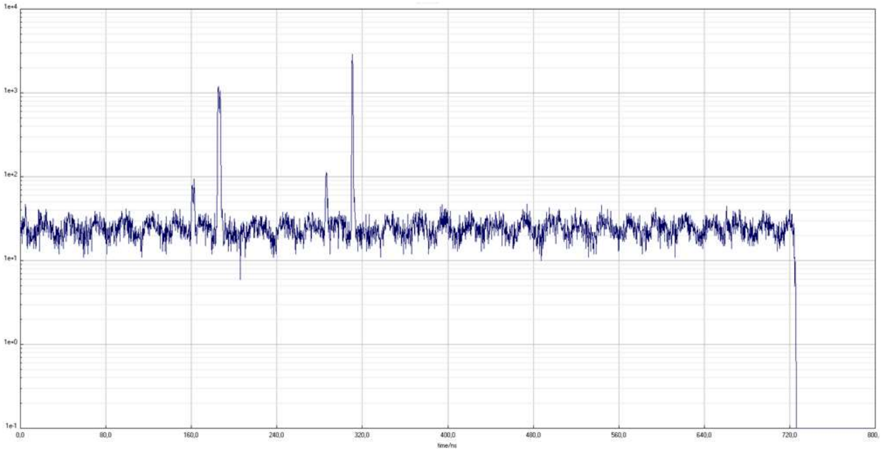


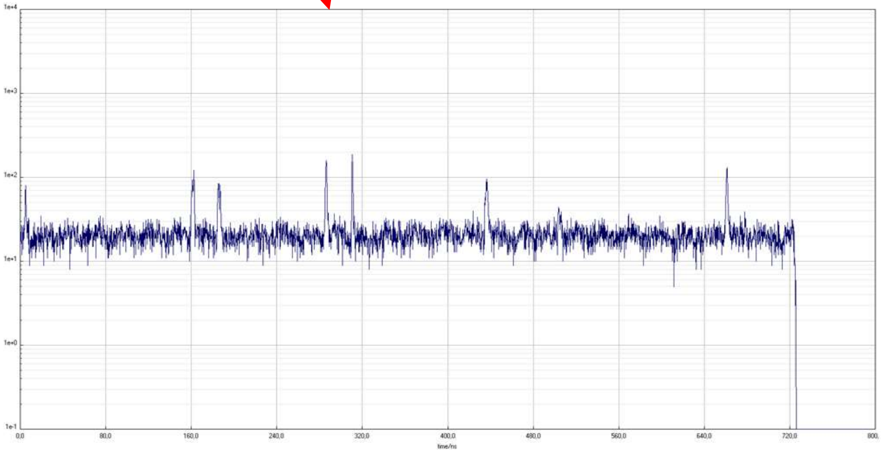
Timing photoemission

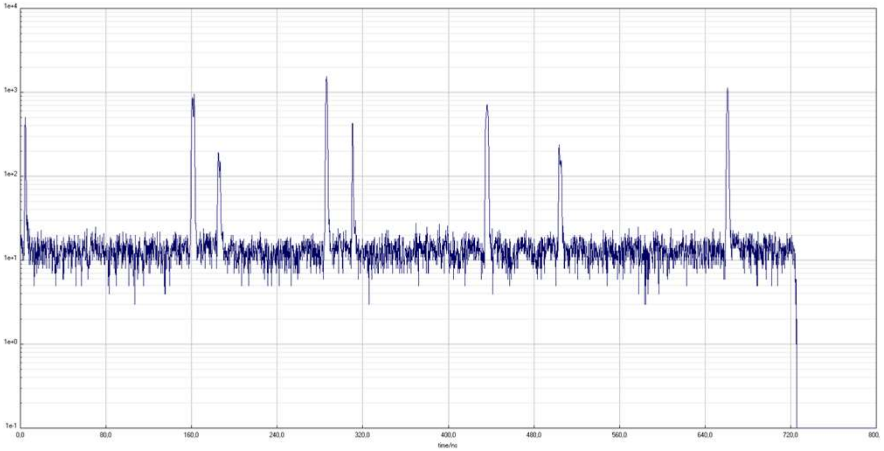
Read time: 2 words, Flash page #16, exposure 60 s, capture window 700 ns

Flash page #16



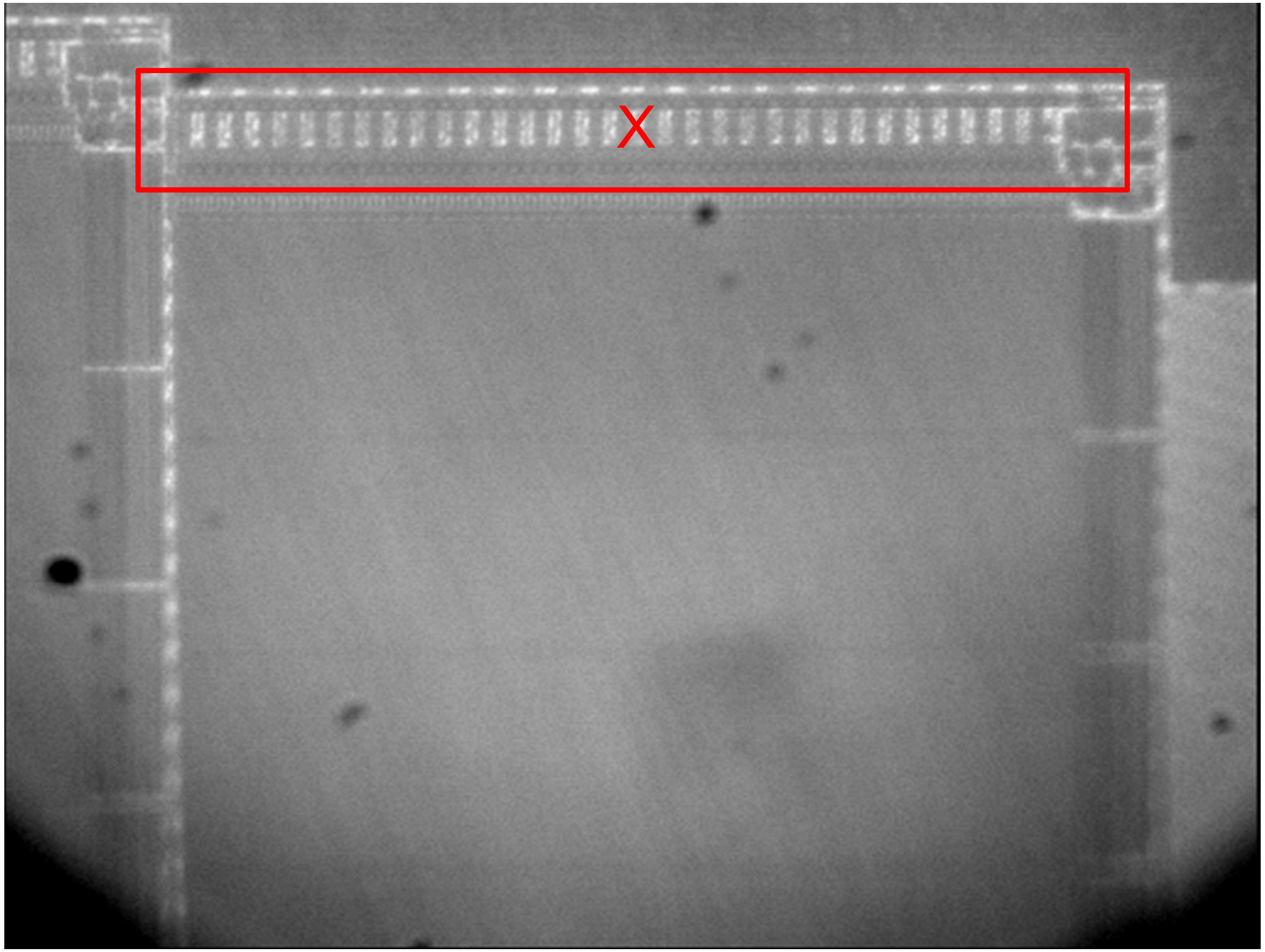






Timing photoemission

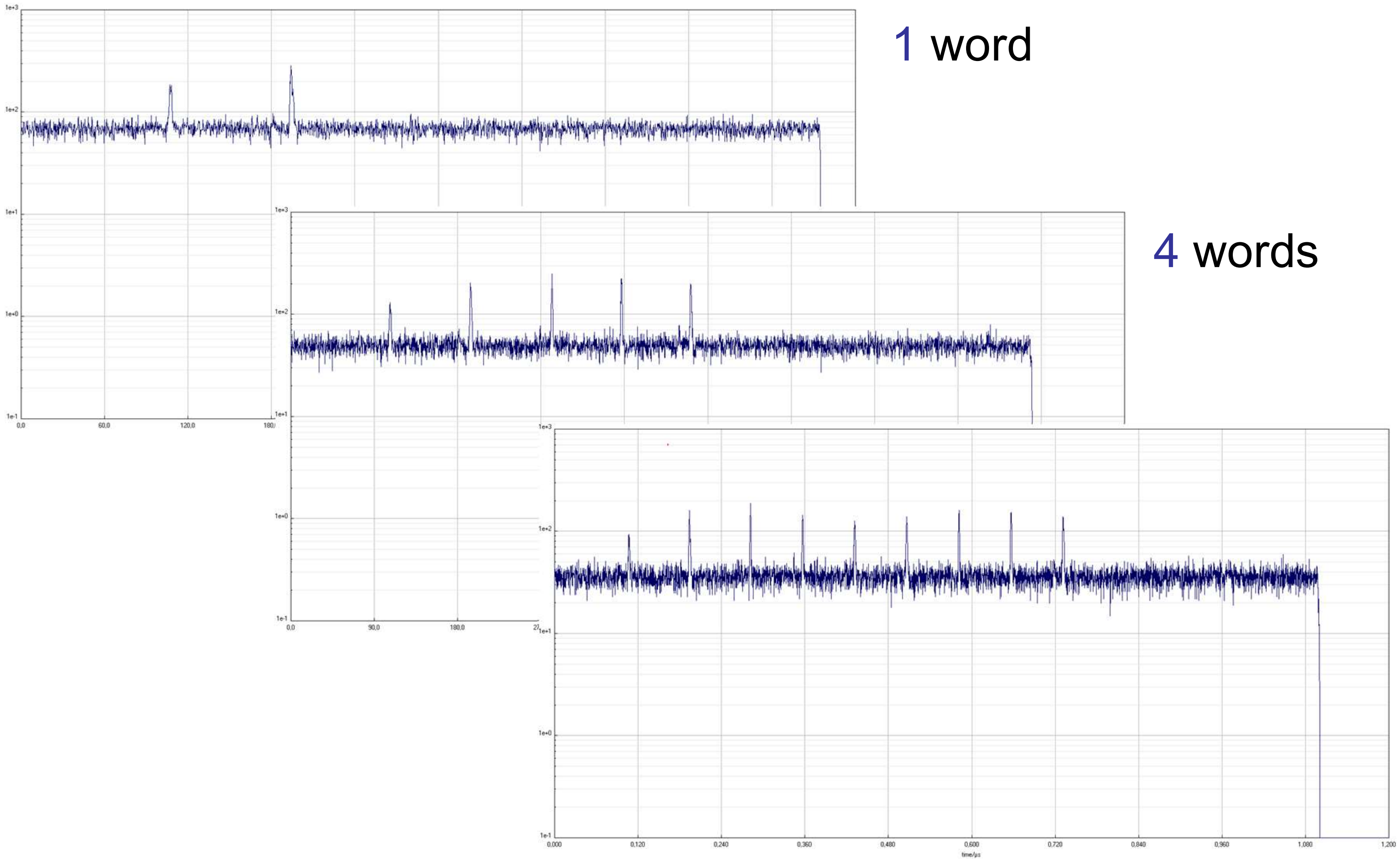
- SRAM memory
- Point of interest: read/write buffers (?)



Timing photoemission

- SRAM memory, write operation

Exposure 60 s, capture window 600 ns / 800 ns / 1,100 ns



1 word

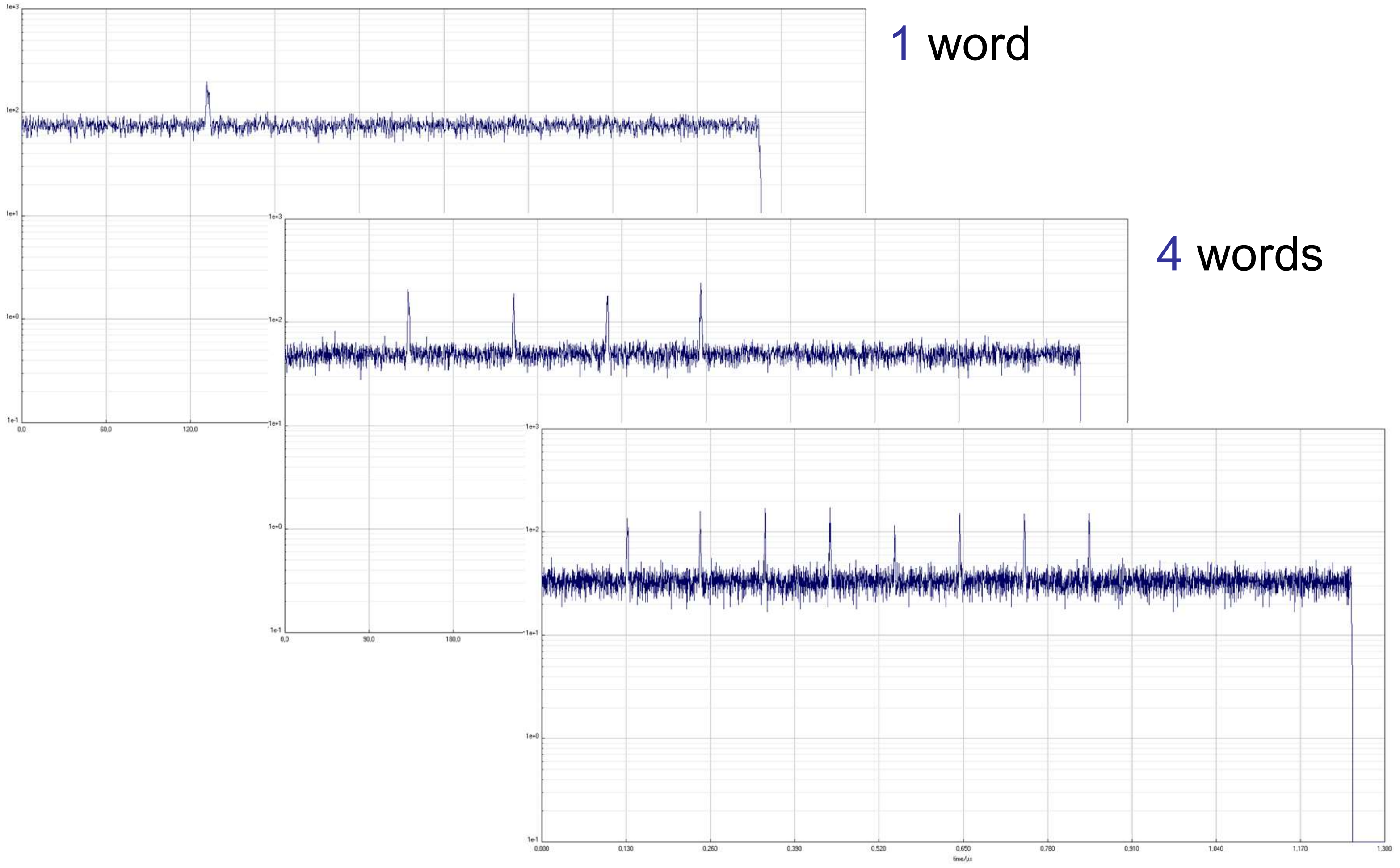
4 words

8 words

Timing photoemission

- SRAM memory, read operation

Exposure 60 s, capture window 500 ns / 850 ns / 1,200 ns



1 word

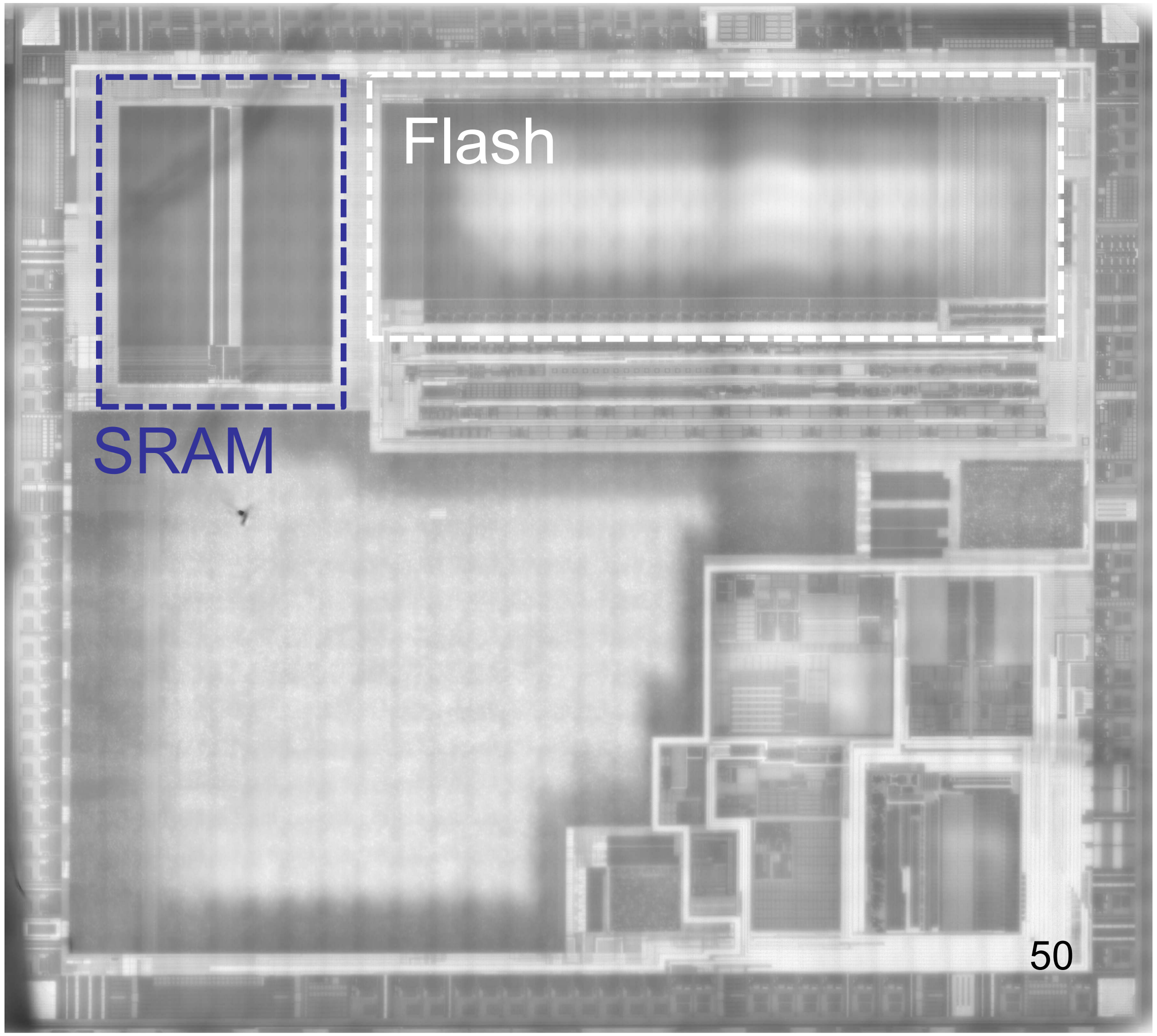
4 words

8 words

❑ Comparison with another target – Photoemission

- Target:
  - ARM Cortex M3
  - 32-bit CPU, 24 MHz
  - 128 kBytes Flash
    - page size = 1 kB
  - 4 kBytes SRAM
  - Si thickness: ~360 µm

Backside IR view



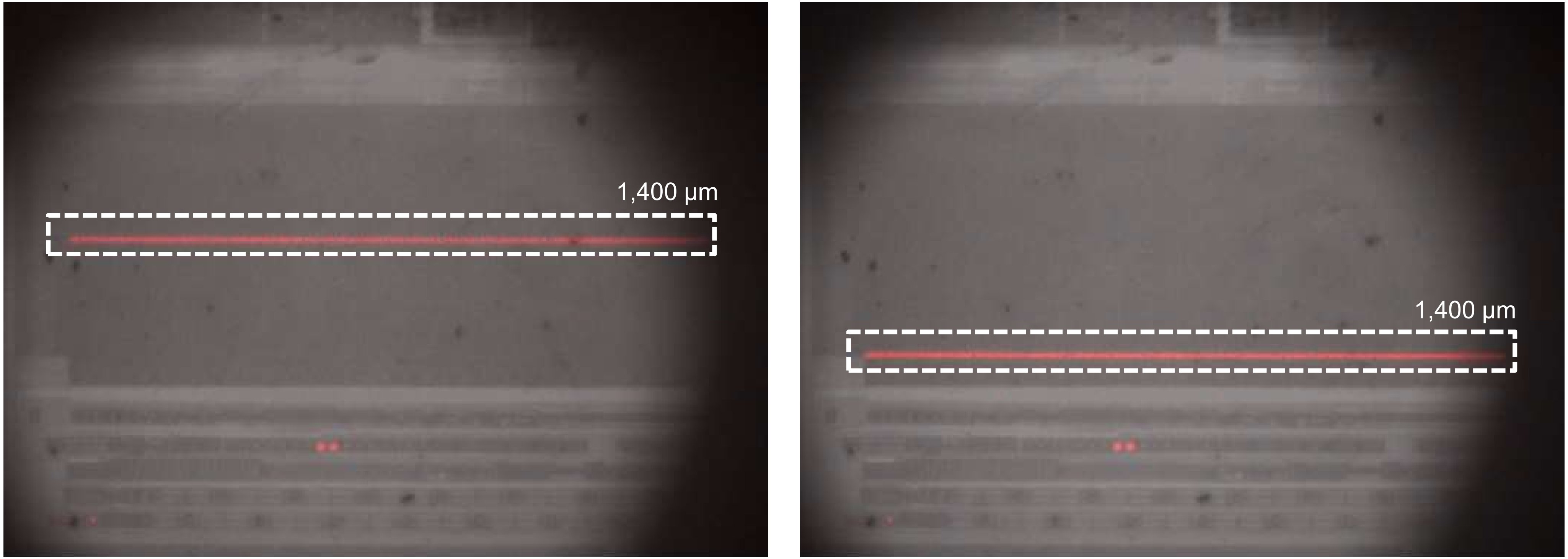
Si die: 3,000 x 2,500 µm

Flash: 1,400 x 550 µm – 1.3 bits/µm²

SRAM: 245 x 660 µm (x2) – 0,1 bits/µm²

❑ Flash memory – 2nd target

- Flash page location

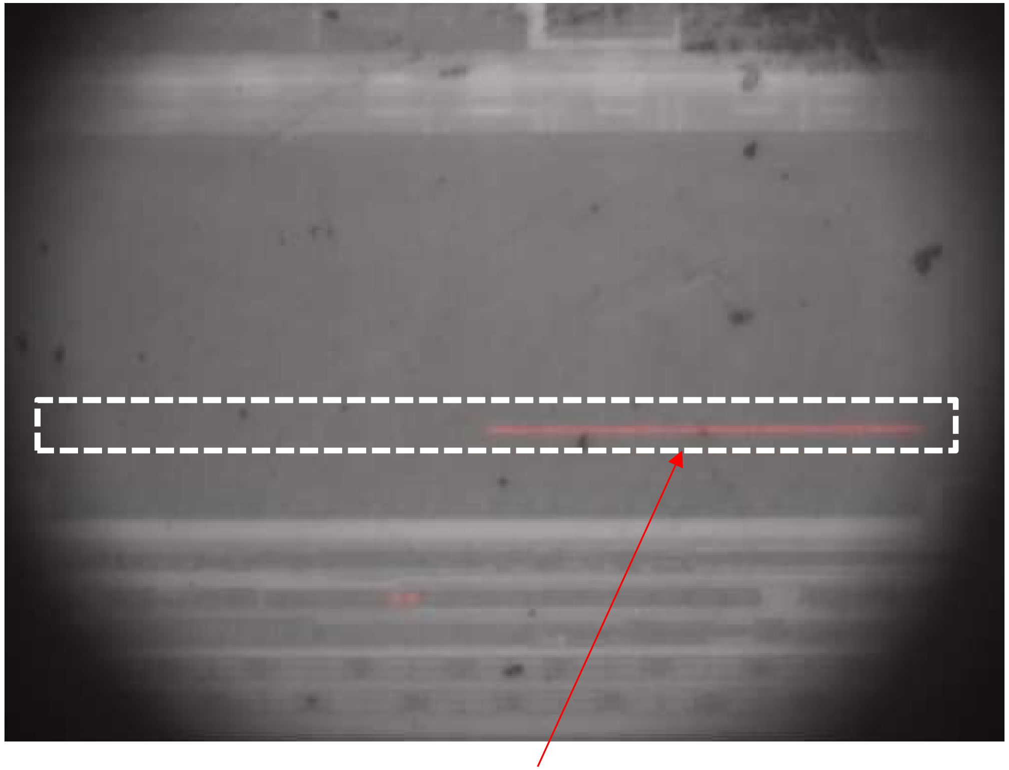


Photoemission map: erase + program (at 0x00000000) cycles
Flash page #64 (left) #127 (right), x5 lens, exposure 2.5 s, 50 cycles

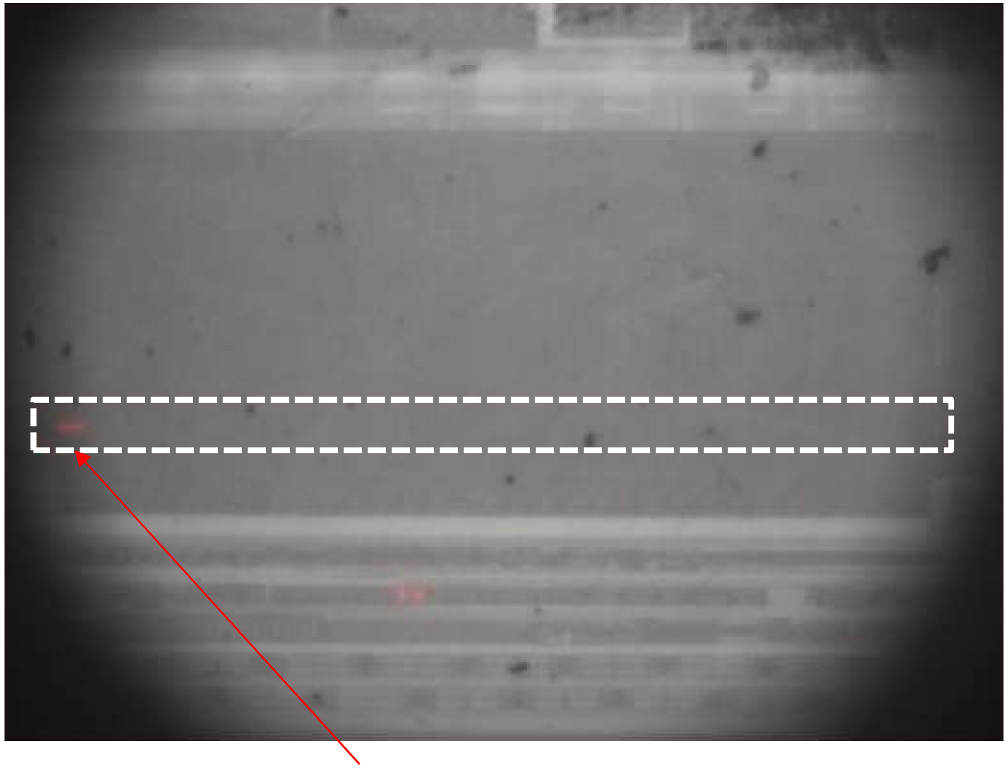
Photoemission map: erase + program cycles

- Data dependency

Flash page #120, x5 lens, exposure 2.5 s



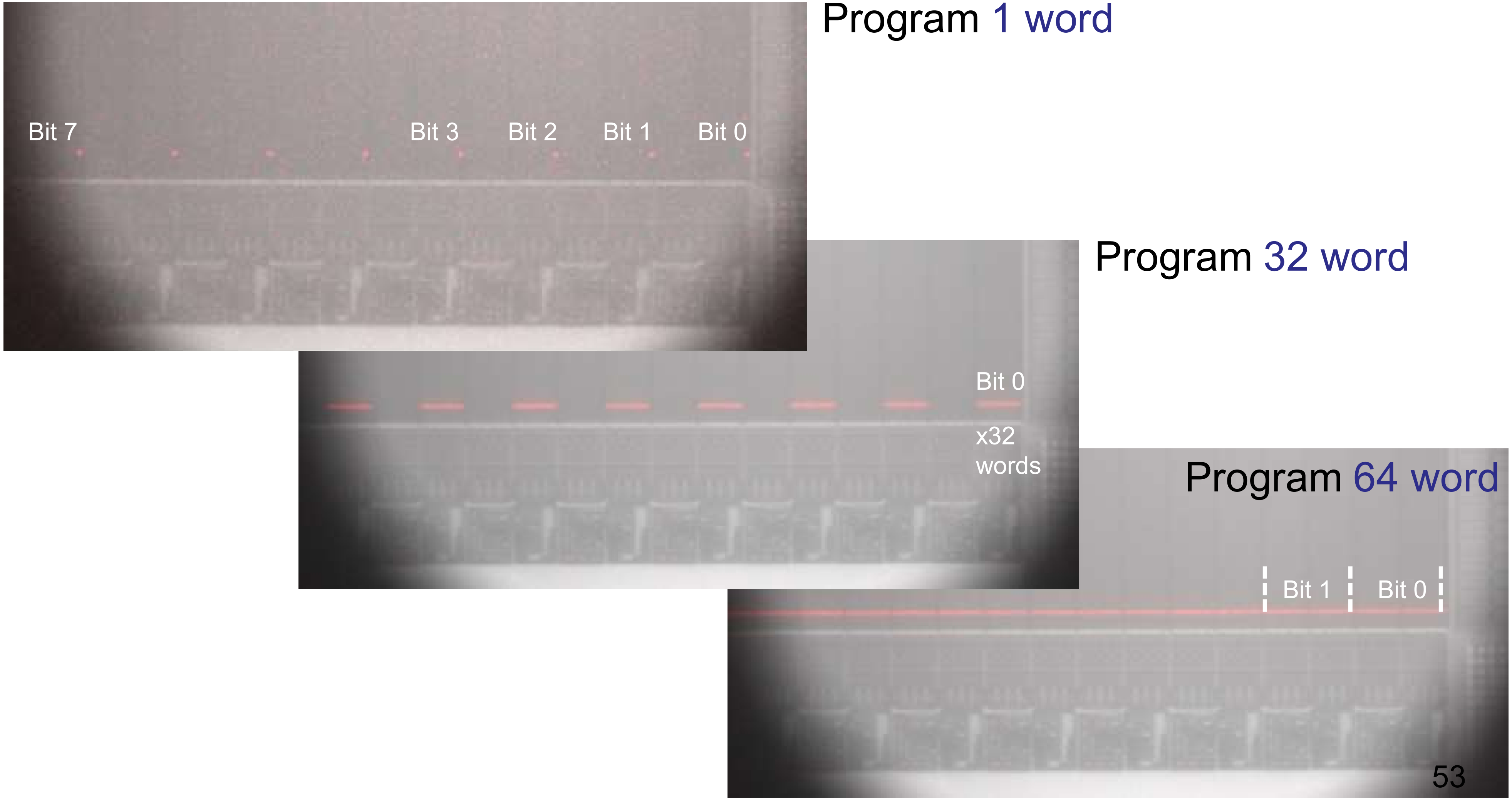
Program at 0xFFFF0000



Program at 0x7FFFFFFF

- Flash Bit-line architecture reverse

Flash page #120, x20 lens, exposure 2.5 s, program at 0x00000000 (all bits sign)



Program 1 word

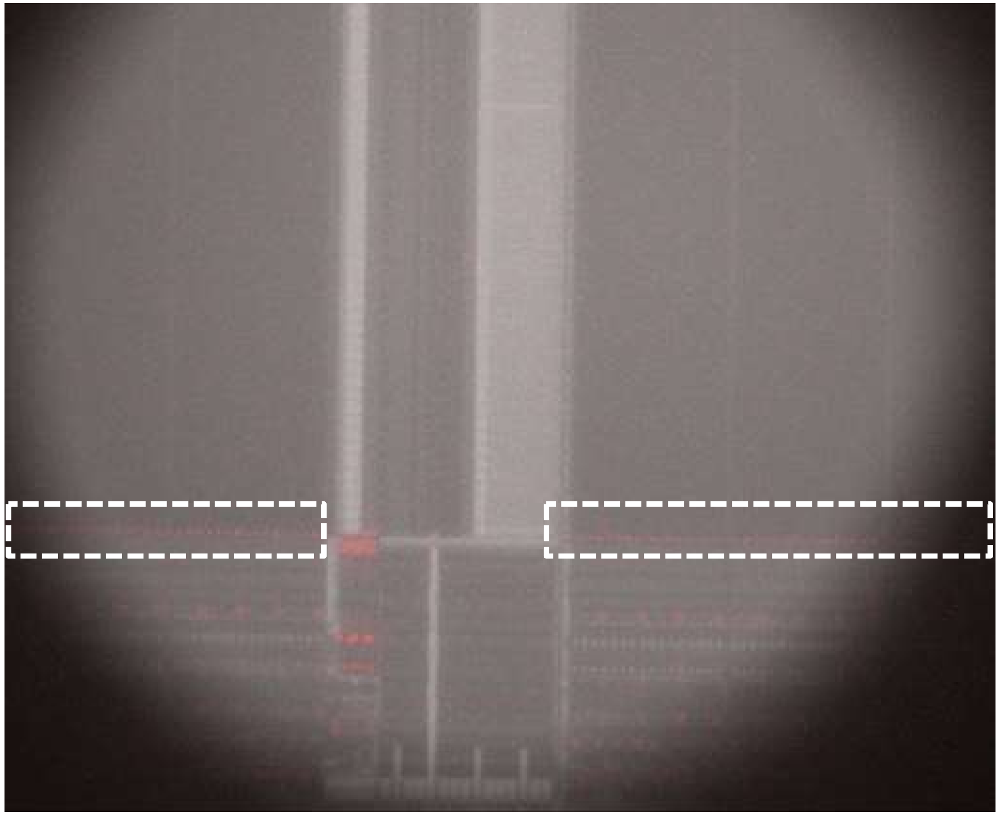
Program 32 word

Program 64 word

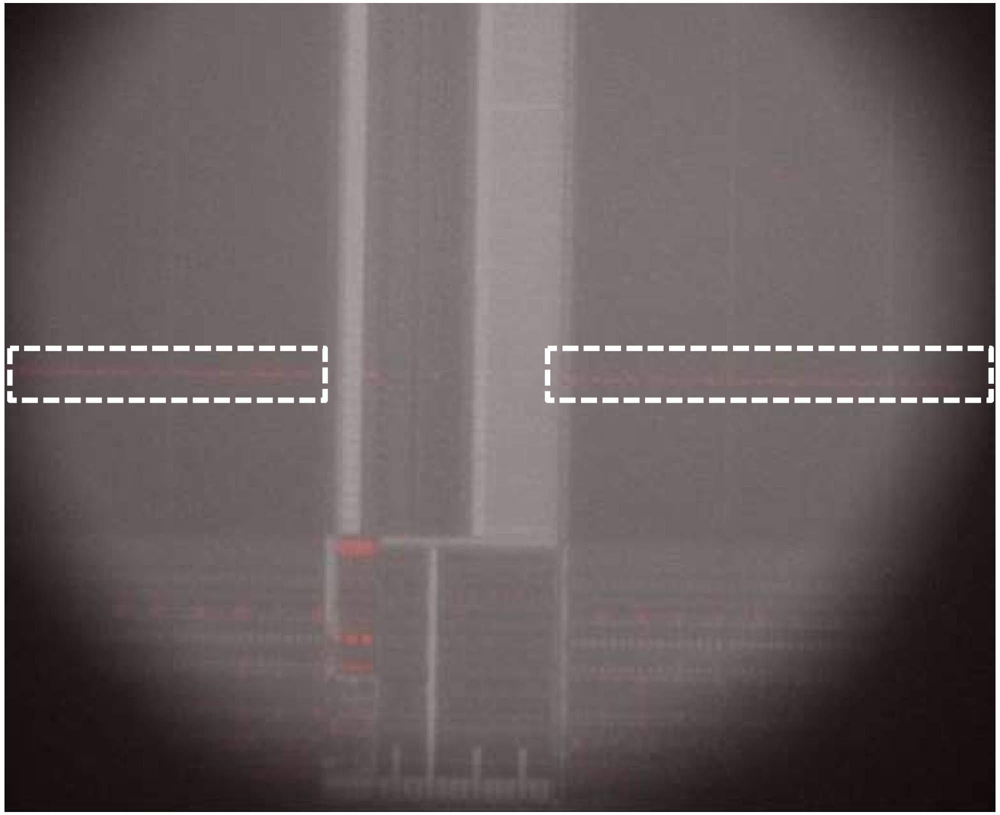
❑ Photoemission at write time – 2nd target

Test code: write 0x00000000, then 0xFFFFFFFF (loops)

Photoemission map: 20x lens, exposure 7.5s, 1word



Write: @ 0x20000000

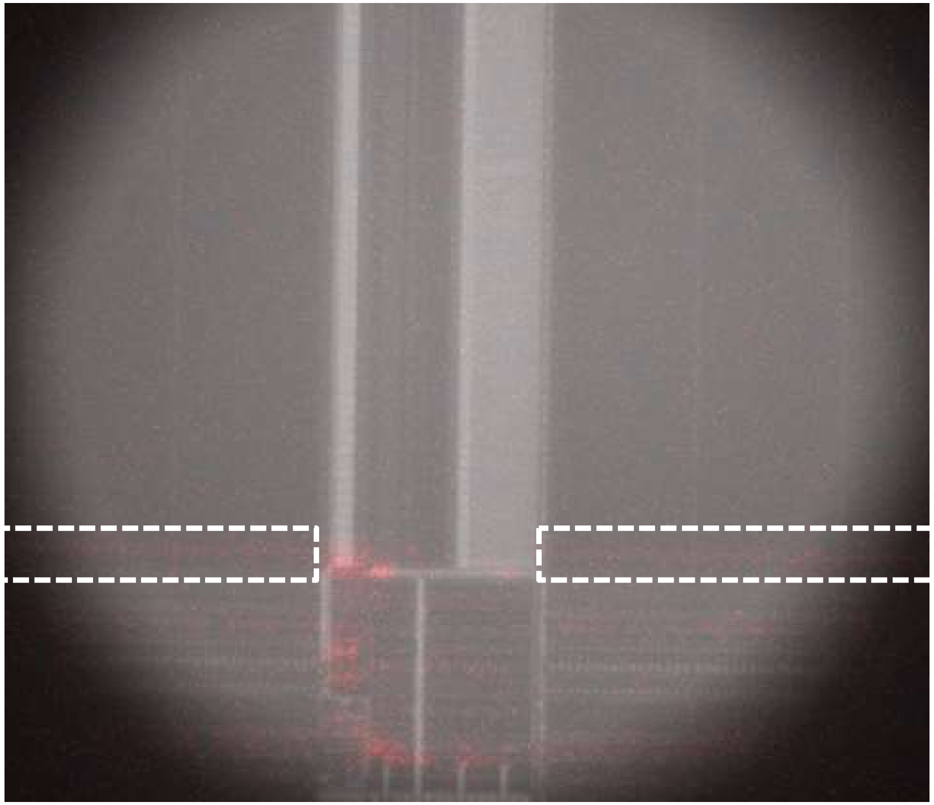


Write: @ 0x20000300

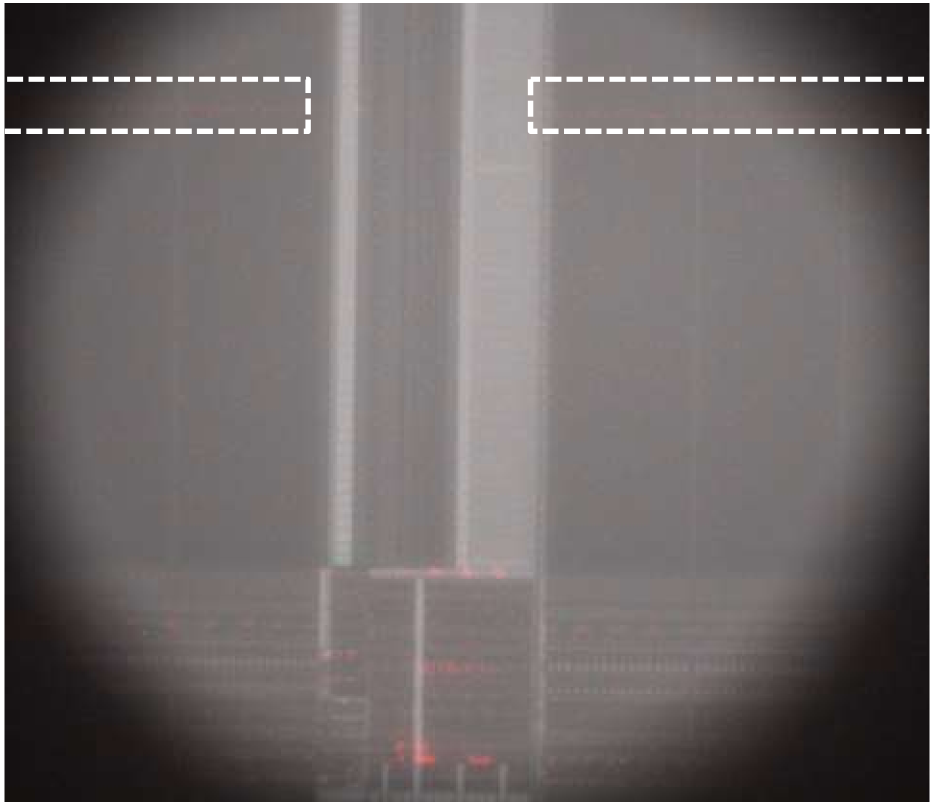
❑ Photoemission at read time – 2nd target

Test code: read 0x00000000 (loops)

Photoemission map: 20x lens, exposure 7.5s, 1word



Read: @ 0x20000000

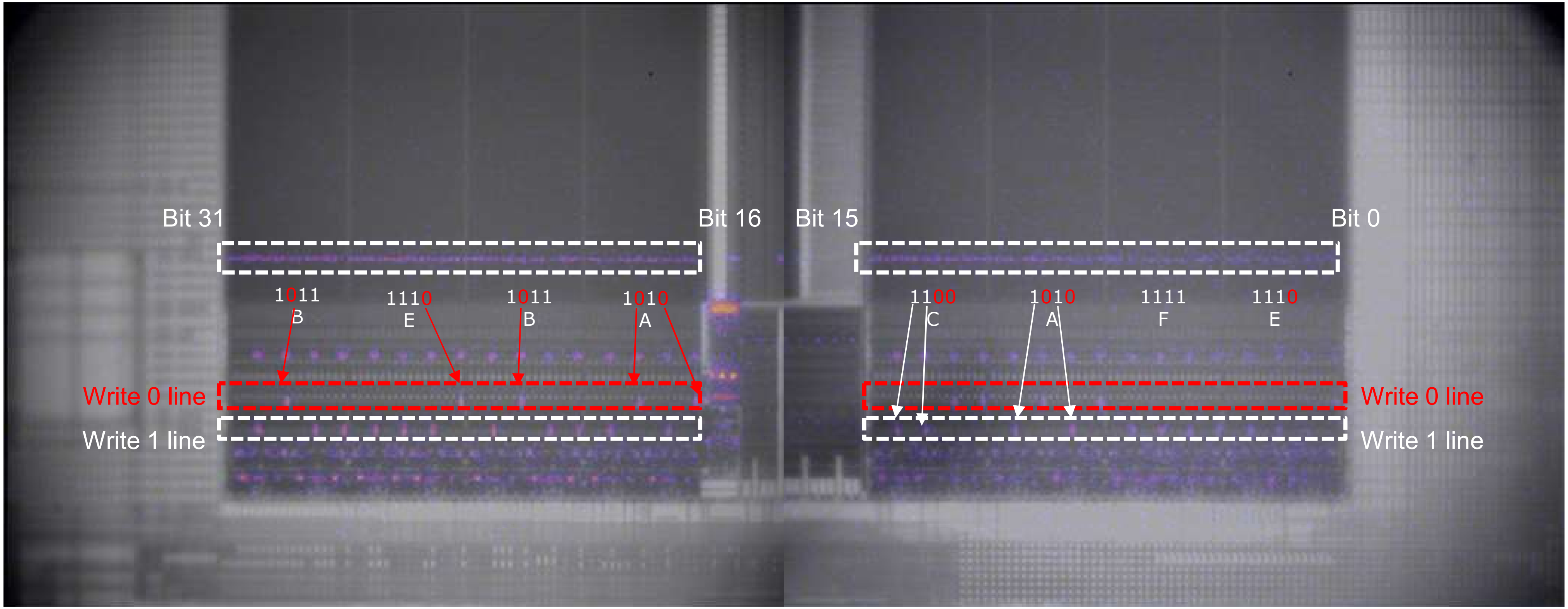


Read: @ 0x20000900

❑ Data leakage – 2nd target

Test code: write 0xBEBACAFE @0x20000100 (loops)

Photoemission map: 20x lens, exposure 2.5s, 1word

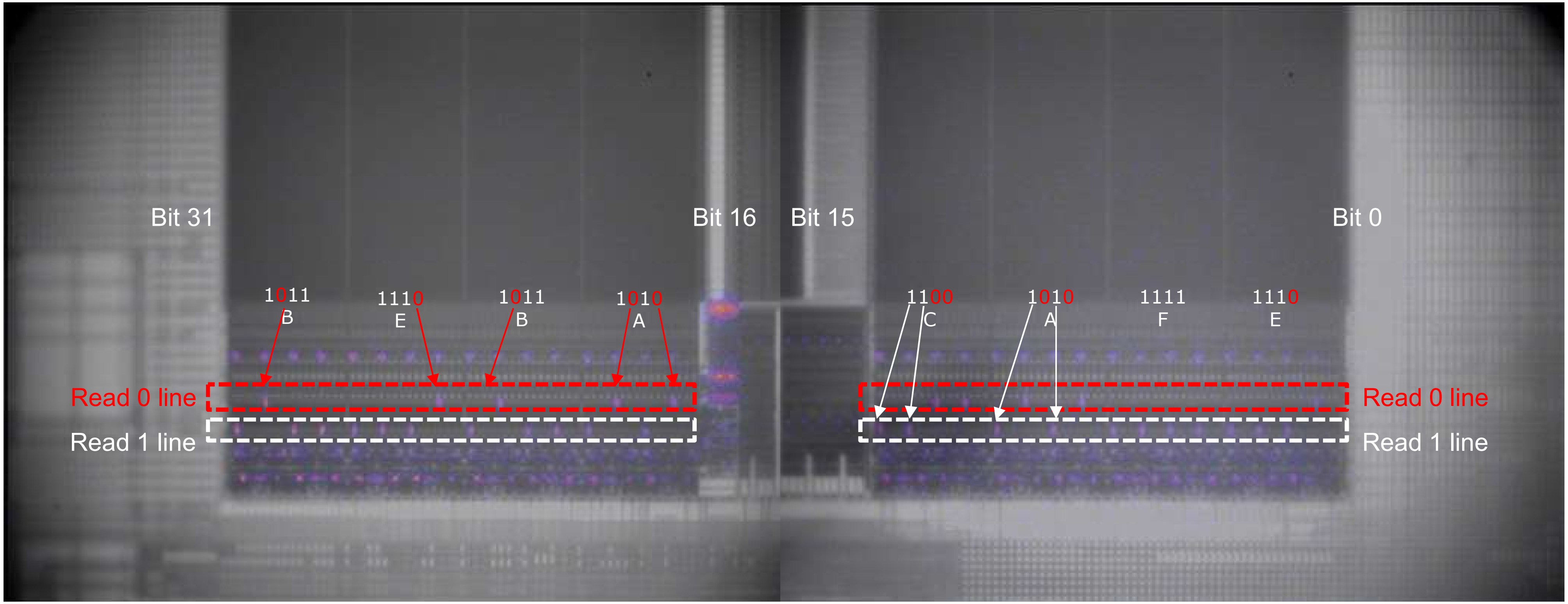


SRAM memory photoemission

❑ Data leakage – 2nd target

Test code: read 0xBEBACAFE @0x20000100 (loops)

Photoemission map: 20x lens, exposure 2.5s, 1word



❑ Photoemission reverse engineering capabilities

- POI identification (photoemission map)

! High level of variability from one target to the other

Flash memory:

Erase – program

- Page/word location
- A certain level of data dependency
- Bit-line architecture at transistor level
- Charge pump

Read operation

- Addressing logic
- Page/word location (target dependent)

Conclusion

❑ Photoemission reverse engineering capabilities

- POI identification (photoemission map)

! High level of variability from one target to the other

SRAM memory:

At write time

- Word location

At read time

- Word location (target dependent)

Data leakage at read/write time

- Read/Write logic can be leaky
- Target dependent

- ❑ Photoemission reverse engineering capabilities
  - ▪ Timing information (photoemission *waveforms*)

Flash:
- Addressing logic timing (read operation)

SRAM:
- Read/write timing

Not explored yet, perspective → direct data leakage

Contact:

dutertre@emse.fr

Département Systèmes et Architectures Sécurisées
Mines Saint-Etienne, Centre de Microélectronique de Provence
13541 Gardanne FRANCE