

# Reverse Engineering ICs

## ReCON 2012

**Dmitry Nedospasov**  
dmitry@nedos.net  
**@nedos**

# The Optics Expert

**Alexander Schlösser**

Optical Technologies

[schloesser@opttech.tu-berlin.de](mailto:schloesser@opttech.tu-berlin.de)

TU Berlin

# whoami

- Studied CE at TU Berlin
- PhD student - "Security in Telecommunications"
- IC security, crypto, low-cost attacks...
- Blog: <http://hwsec.net>

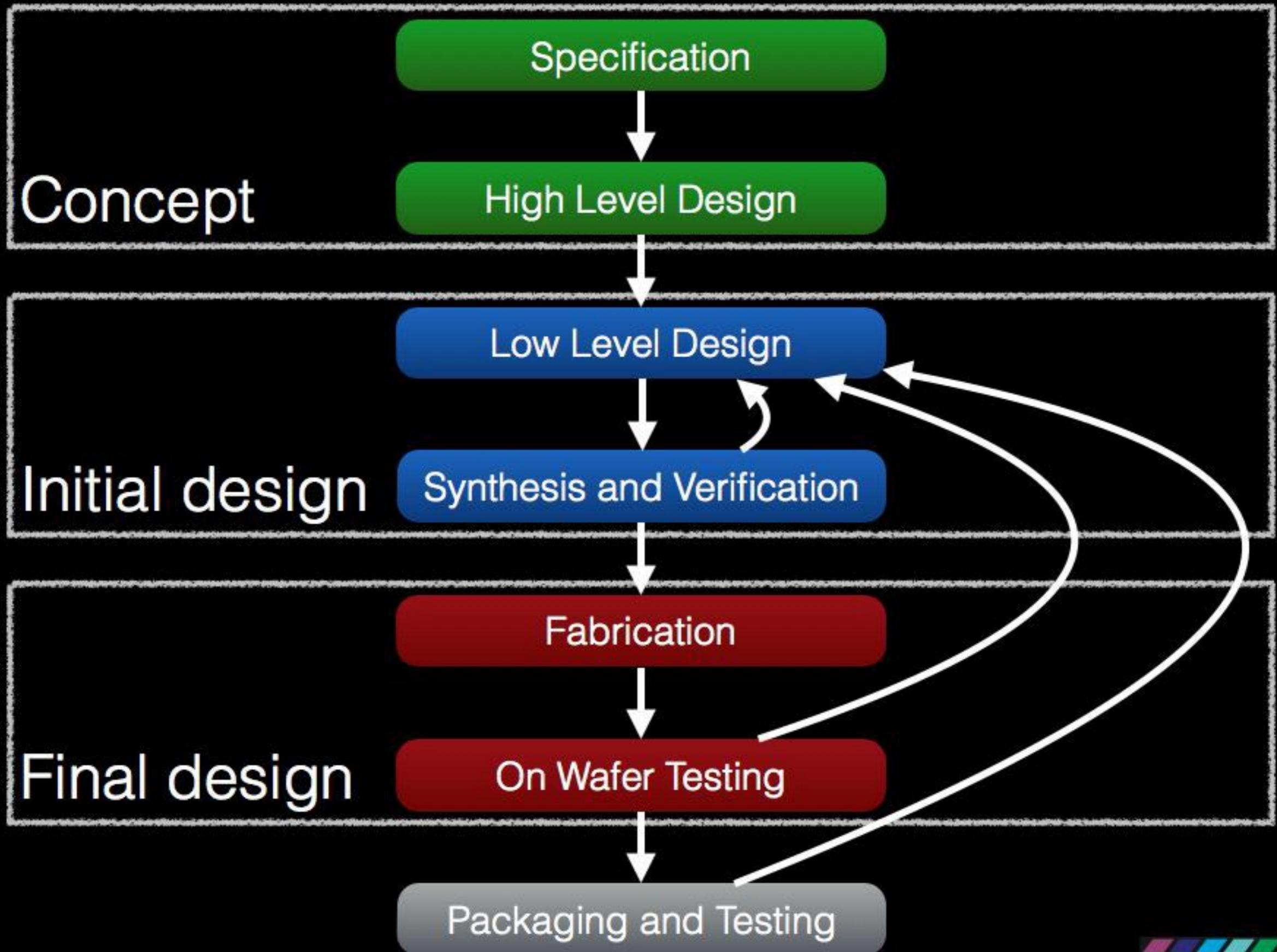
# RTFPapers

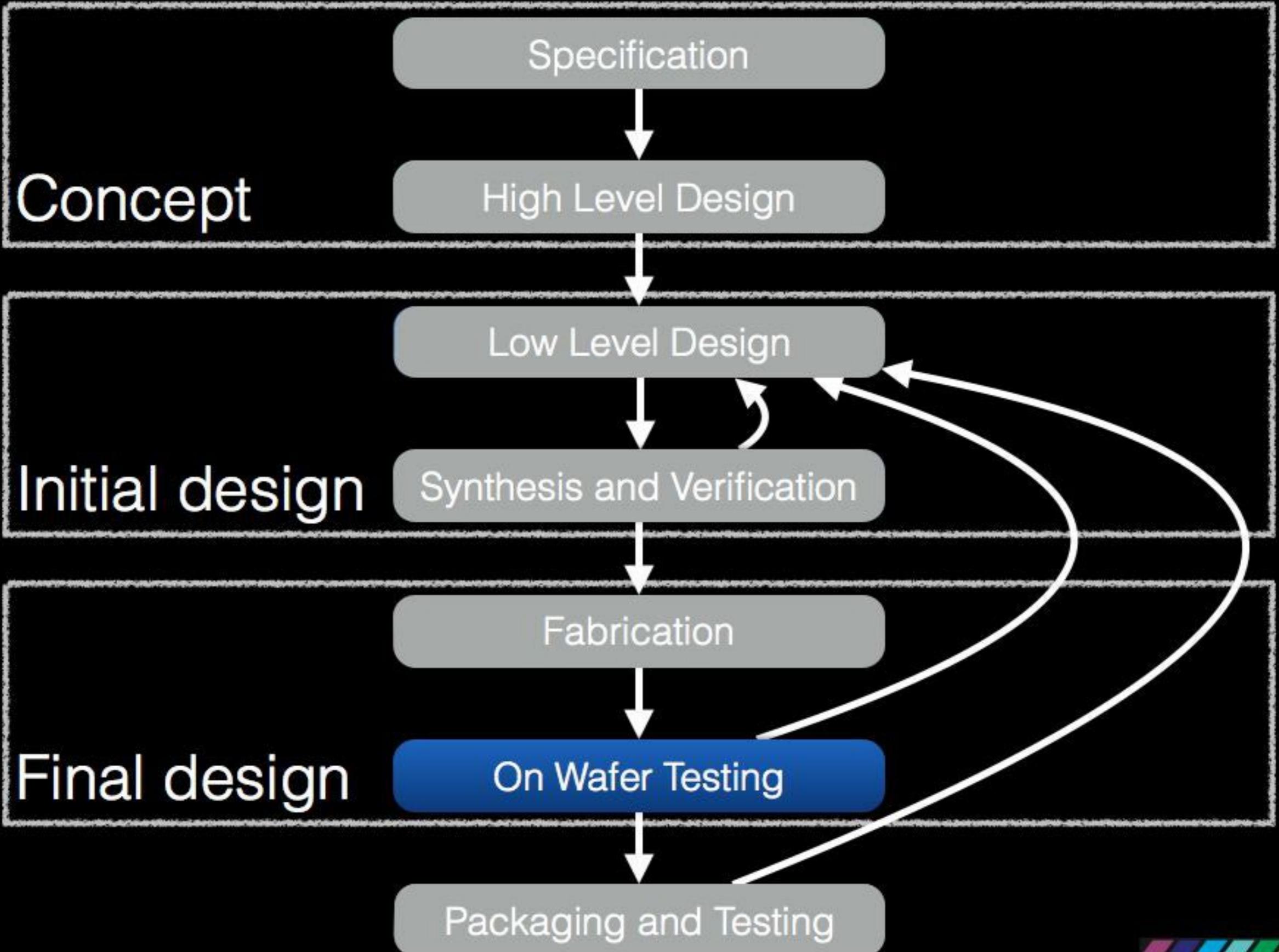
*Functional IC Analysis*  
IEEE HOST 2012

*Simple Photonic Emission Analysis of AES*  
CHES 2012

# Story time with Dmitry

A brief introduction to failure analysis





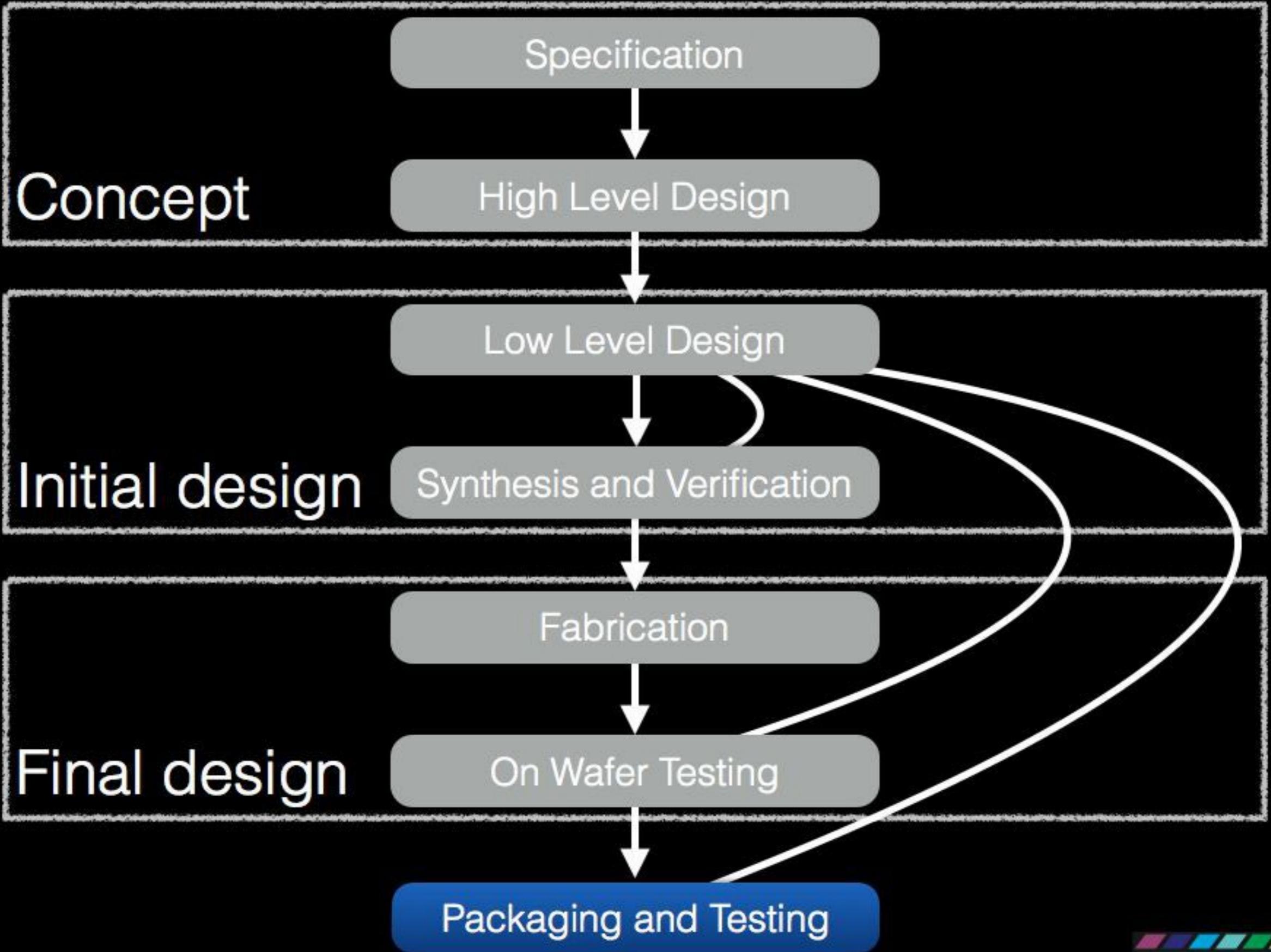
# On wafer testing

Copyright: Verigy

Source: [http://www.youtube.com/watch?v=de\\_LCVmJEE4](http://www.youtube.com/watch?v=de_LCVmJEE4)

# On-Wafer Testing

- Completely automated
- Pass/Fail Testing
- Test scan chains
- Can be performed during manufacturing



# FIB Circuit Edit

Copyright: FEI

Source: <http://www.youtube.com/watch?v=CF5vCsmuiAk>

# FIB

- Analyze quality of bonds
- Edit circuits
- Labor intensive, requires skilled operator
- > \$100,000

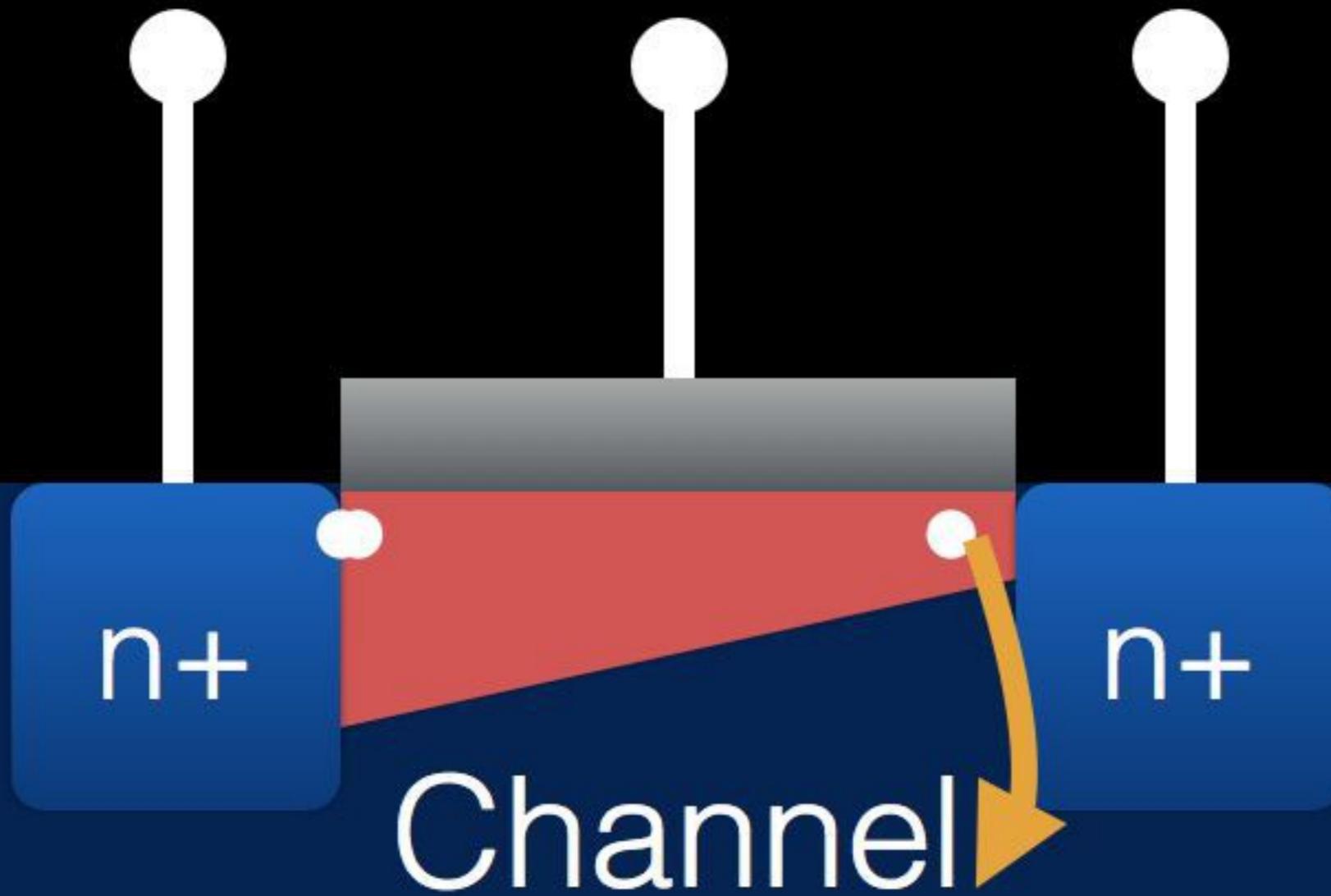
# More Exotic Techniques

- Laser stimulation
- Atomic force microscopy
- ★ Photonic emission analysis

Source

Gate

Drain



# Hamamatsu Phemos



- Can be used for optical emission analysis
- Backside is possible - no rebonding
- > \$1M

NIR CCD

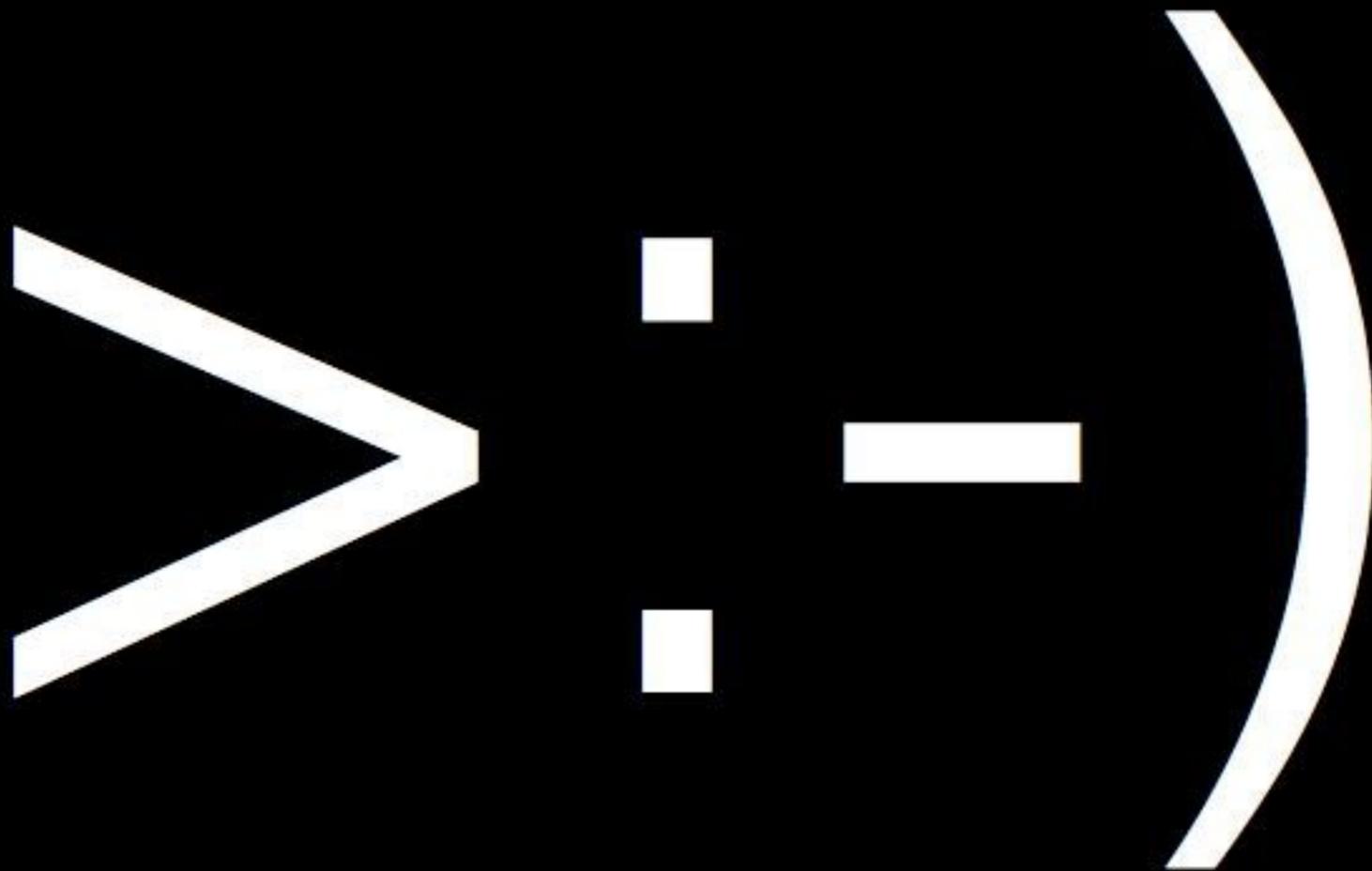


Microscope Optics



DUT

# Reverse Engineering



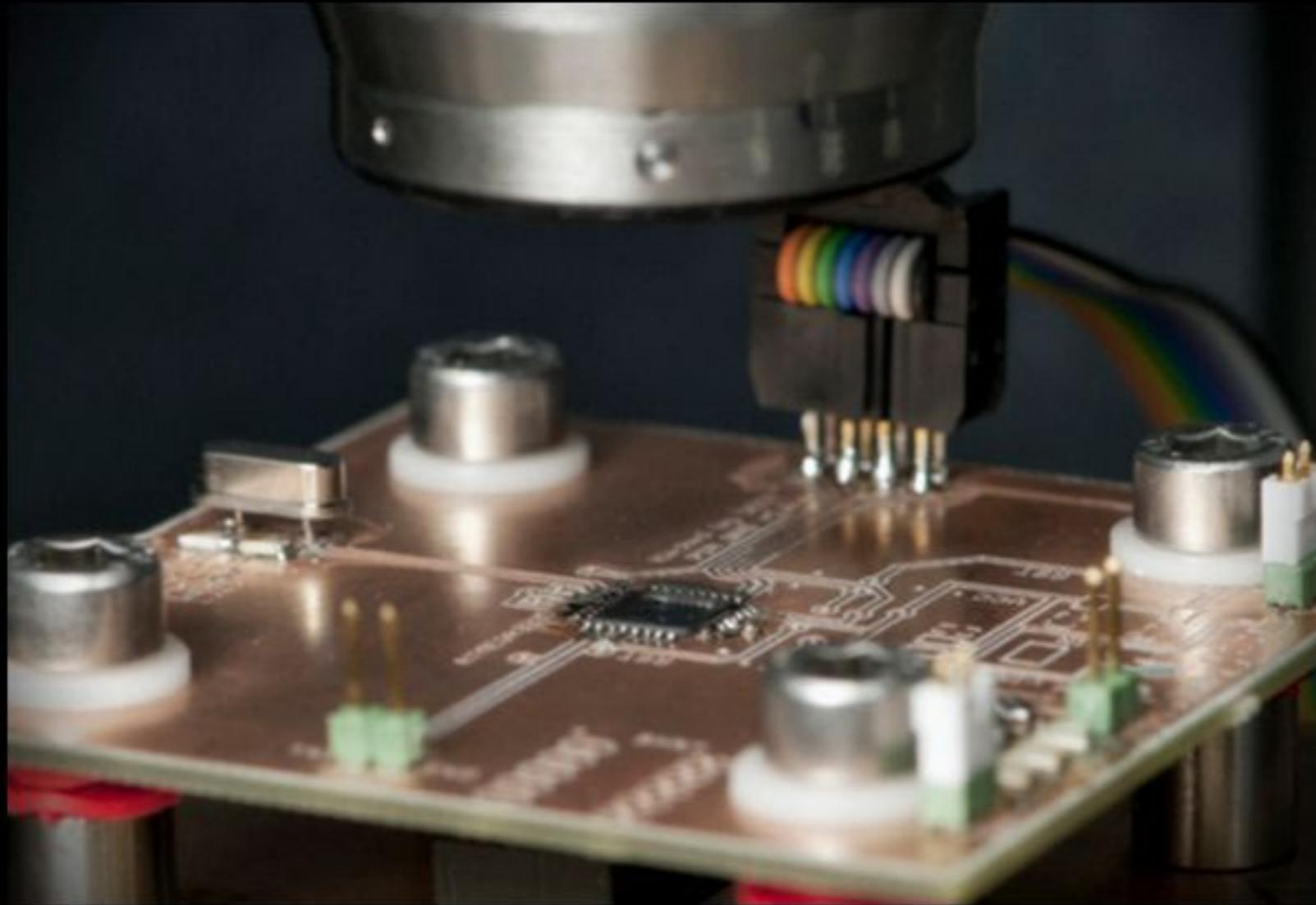
**Frontside**

$n+$

$n+$

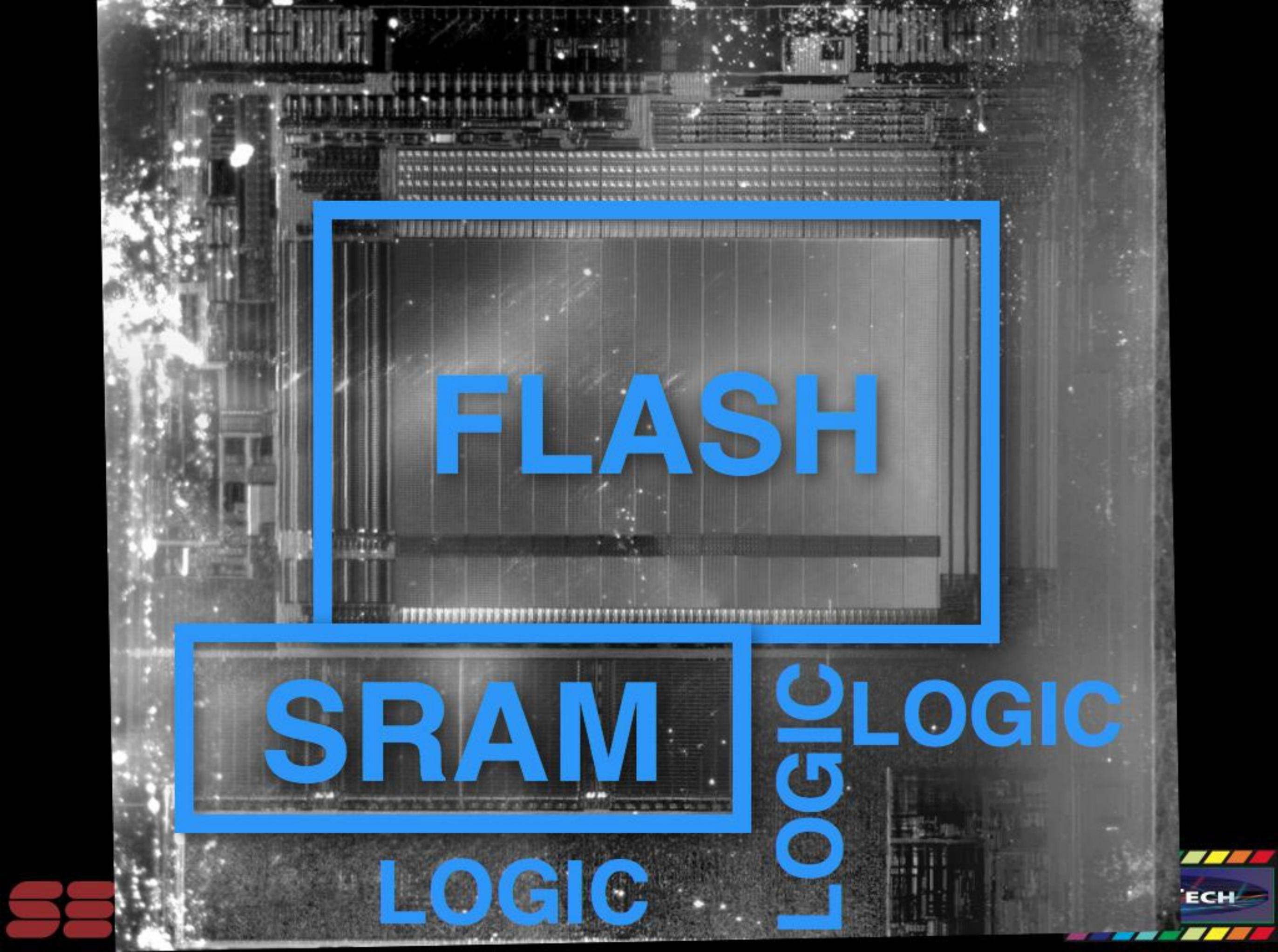
**Backside**

# Setup



A microscopic view of a circuit board, showing a grid of components and traces. A blue rectangular box highlights a central area containing the word "FLASH" in large, bold, blue capital letters. The background is a complex pattern of blue, green, and yellow, representing the underlying circuitry.

# FLASH

A grayscale micrograph of a microchip die with several regions highlighted by blue rectangular boxes. The largest box is centered in the upper half of the die. Below it, two smaller boxes are positioned horizontally, one on the left and one on the right. At the bottom, the word 'LOGIC' is written twice, once horizontally and once vertically.

**FLASH**

**SRAM**

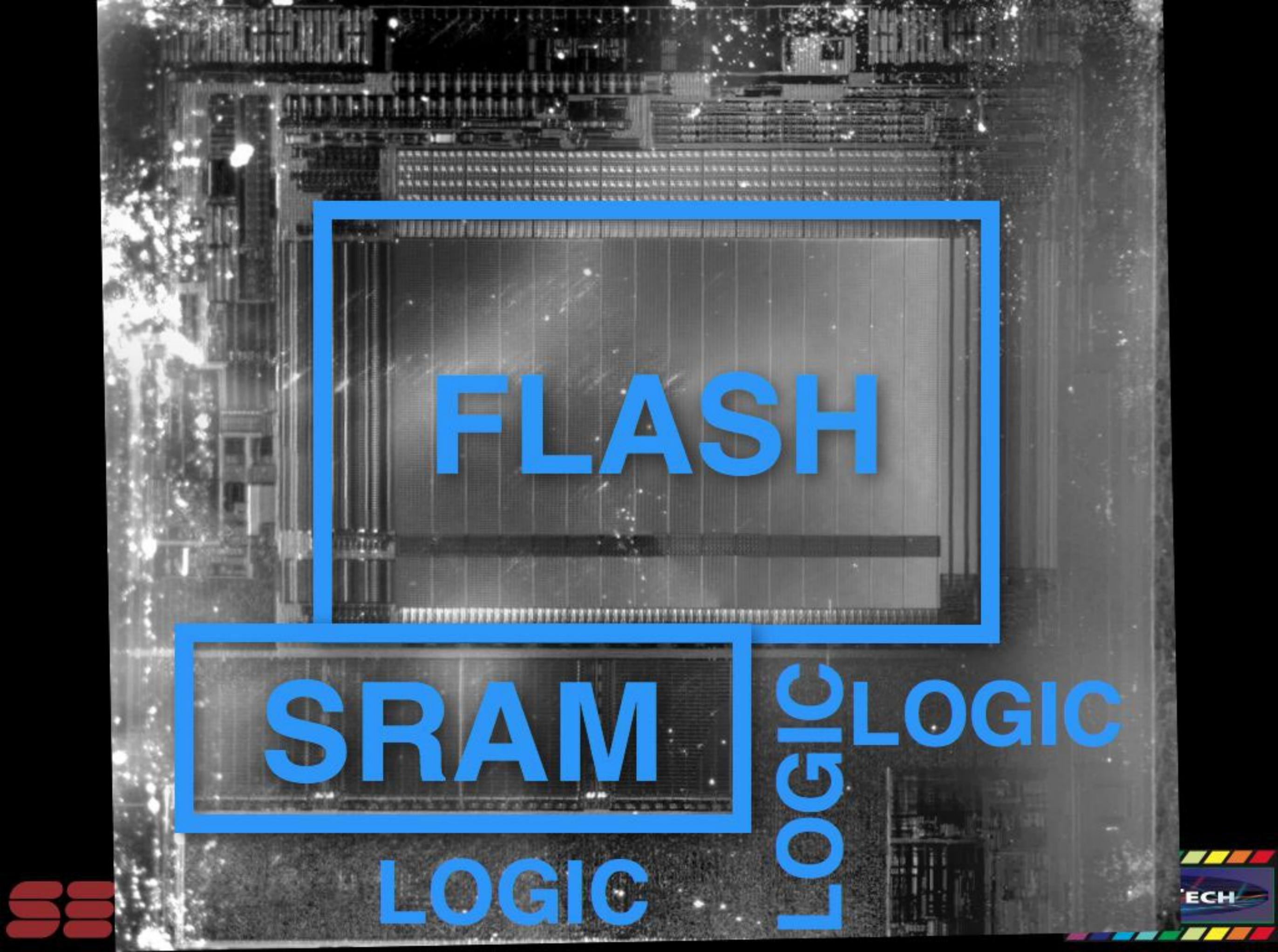
**LOGIC**

**LOGIC**

**LOGIC**



# Identifying logic

A grayscale micrograph of a microchip die with several regions highlighted by blue boxes. The largest box is centered in the upper half and contains the word 'FLASH'. Below it, a horizontal box contains 'SRAM'. To the right of the 'SRAM' box, the word 'LOGIC' is written vertically. Below the 'SRAM' box, the word 'LOGIC' is written horizontally. The background shows the intricate circuitry of the chip.

**FLASH**

**SRAM**

**LOGIC**

**LOGIC**



**FLASH**

**SRAM**

**LOGIC**

**LOGIC**

**LOGIC**

ldi r16,0x00

# What to look at



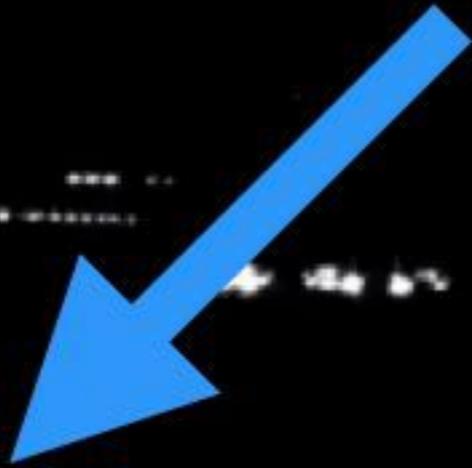
```
.global infloop  
infloop: rjmp infloop ; to self
```



# Memory map

```
; first parameter: r25:r24 - addr
; second parameter: r22 - value
.global memmap
memmap: movw r26,r24 ; addr to X
loop:   st   X,r22 ; write to [X]
        rjmp loop
```

0x100





0x180	0x380	0x580	0x780
0x100	0x300	0x500	0x700
0x880	0x280	0x480	0x680
0x800	0x200	0x400	0x600

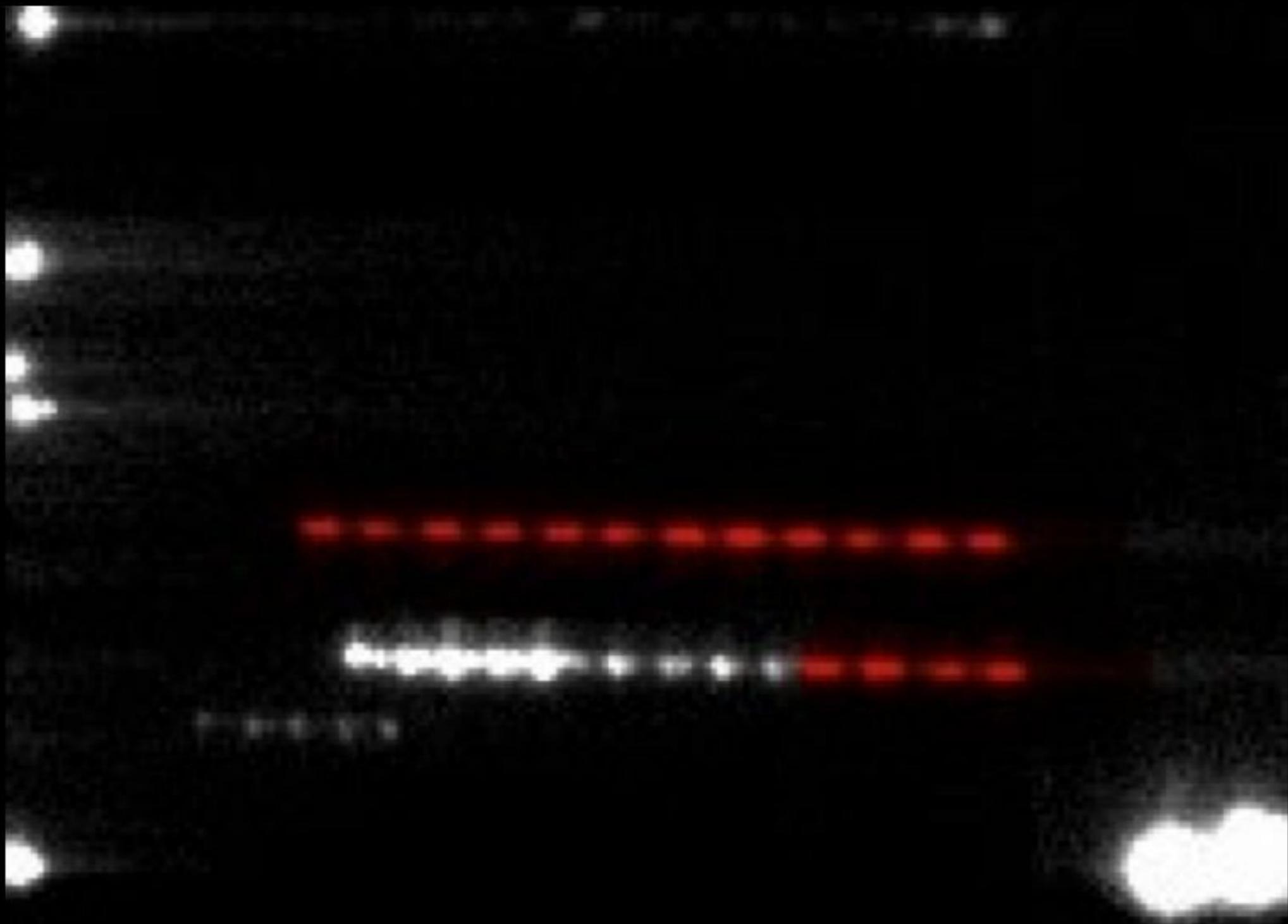
# Branching logic

```
.global setclrz  
setclrz: sez ; set Z-flag  
        clz ; clear Z-flag  
        rjmp setclrz
```





# Execution logic



0x0f5a: ldi r17,0x03



0x0f5a: ldi r17,0x03

0x0f5b: ldi r17,0x03

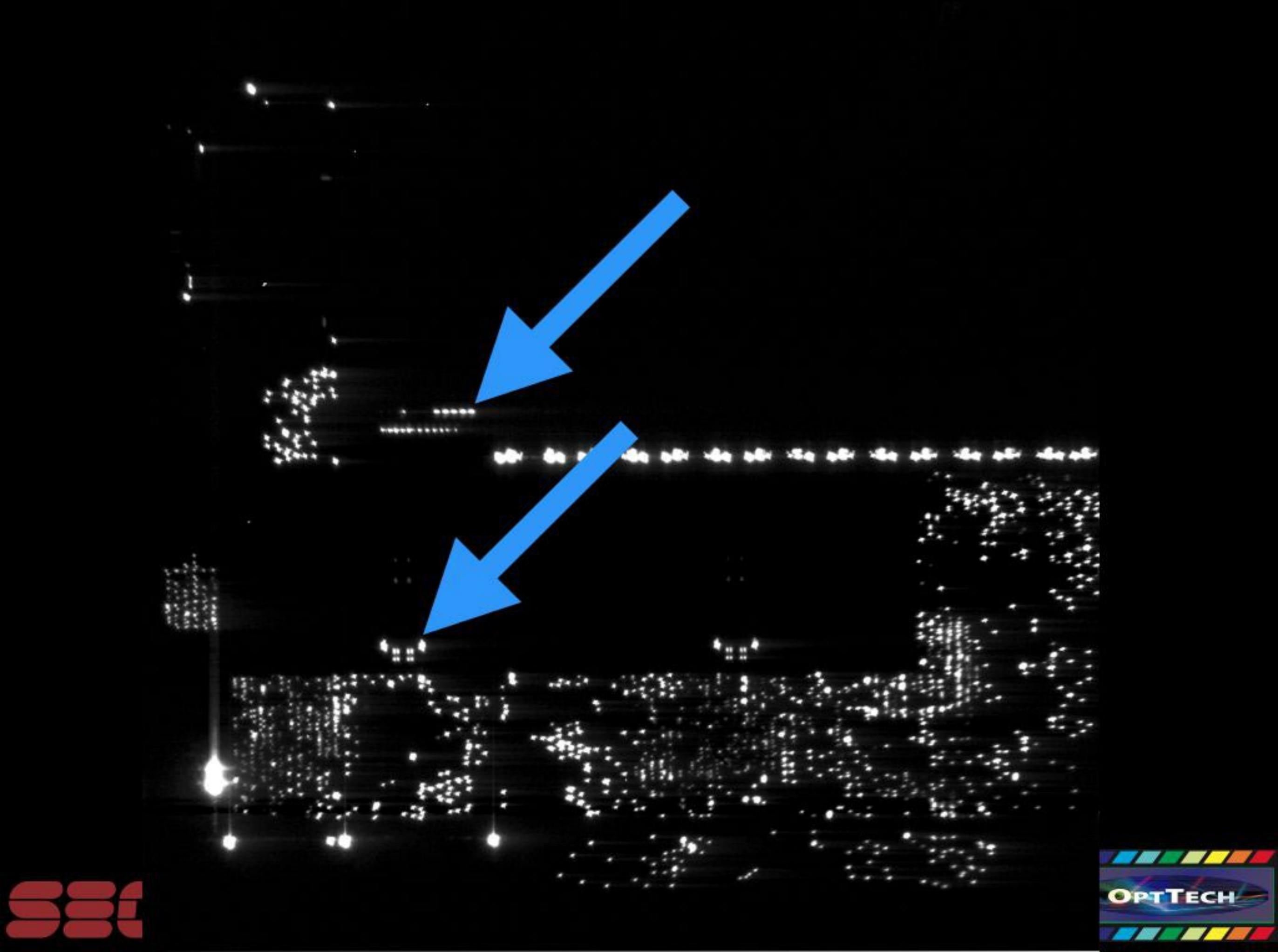
```
ldi r17, 0x01 ; 0xf6a - loop1
rjmp .-4 ; 0xf6c
ldi r17, 0x02 ; 0xf6e - loop2
rjmp .-4 ; 0xf70
ldi r17, 0x04 ; 0xf72 - loop3
rjmp .-4 ; 0xf74
```

Loop3 (0xf72)  
Loop2 (0xf6e)  
Loop1 (0xf6a)

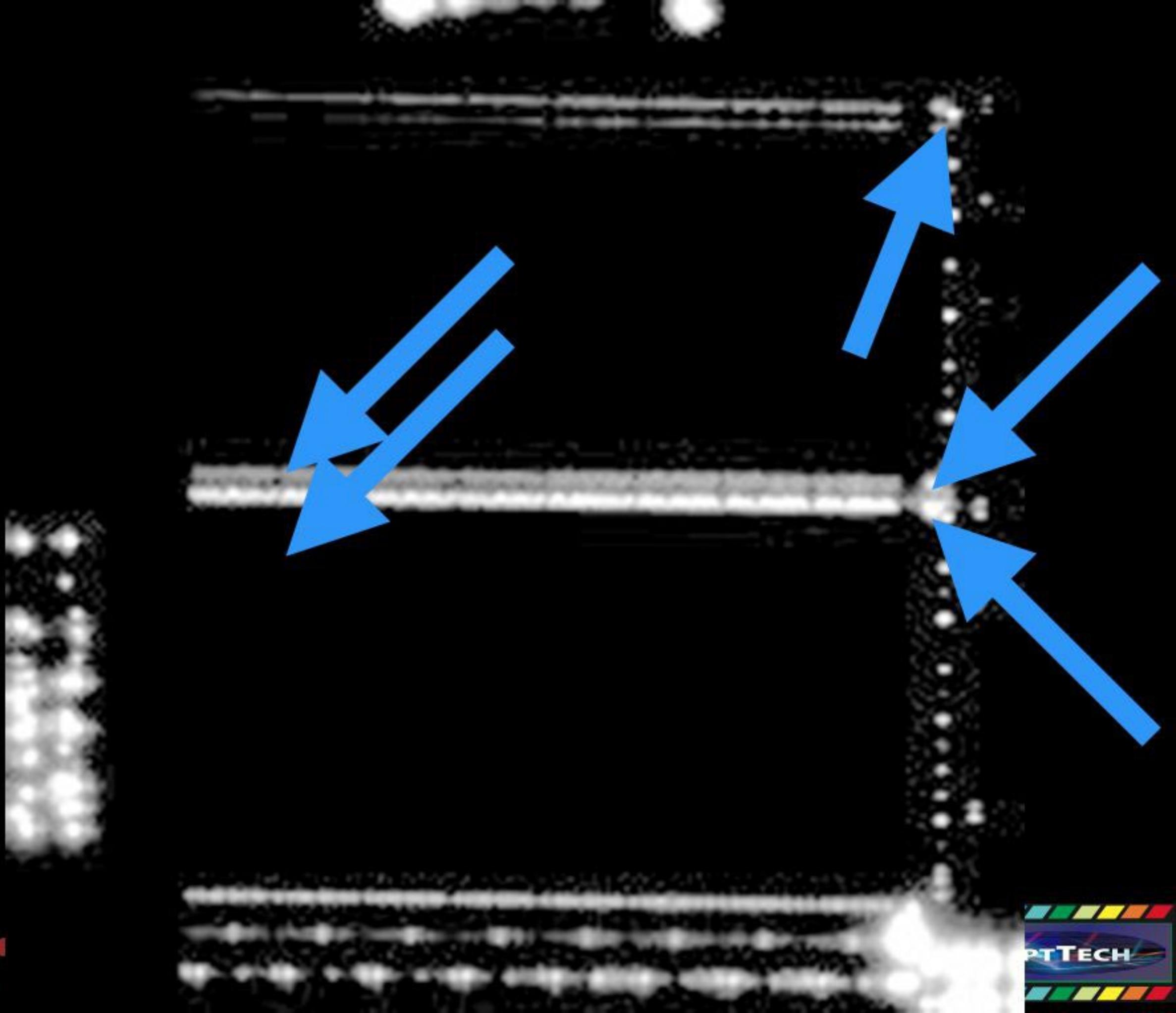
# Reverse engineering ICs?

- There's an industry for that!
- FA is similar to reversing
- Lots of literature
- Low cost is possible

# UART Hello World



# memcpy





0151

T-Mobile

54944671-9/185

n3

1. Enable AES interrupts (optional)
2. Select the AES direction, encryption or decryption.
3. Load the Key data block into the AES Key memory
4. Load the data block into the AES State memory
5. Start the encryption/decryption operation

progress.

Figure 23-2. The State memory with pointers and register



If more than one block is to be encrypted or decrypted **repeat** the procedure from step 3.

