ECE 459/559
Secure & Trustworthy Computer Hardware Design

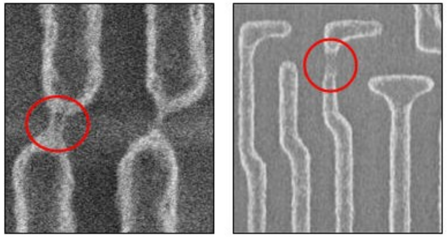
Hardware Metering

Garrett S. Rose
Spring 2017

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

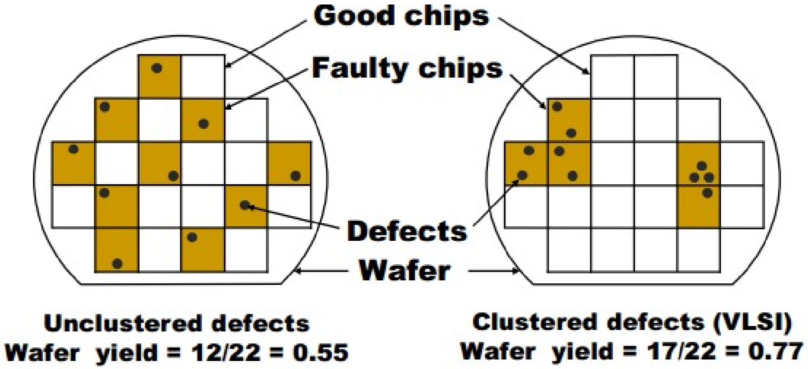
Background: Test and Yield

- Errors in fabrication process cause defects on chip which causes chip to malfunction
- Chips tested in order to detect defects
- Failing chips are discarded
- **Yield** – percentage of remaining good chips



$$Yield = \frac{total\,chips - discarded\,chips}{total\,chips}$$

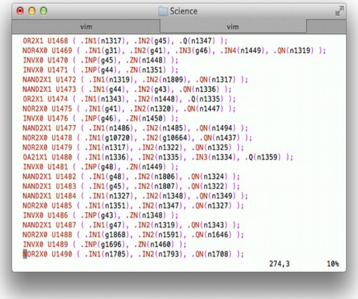
- Foundry decides/predicts yield



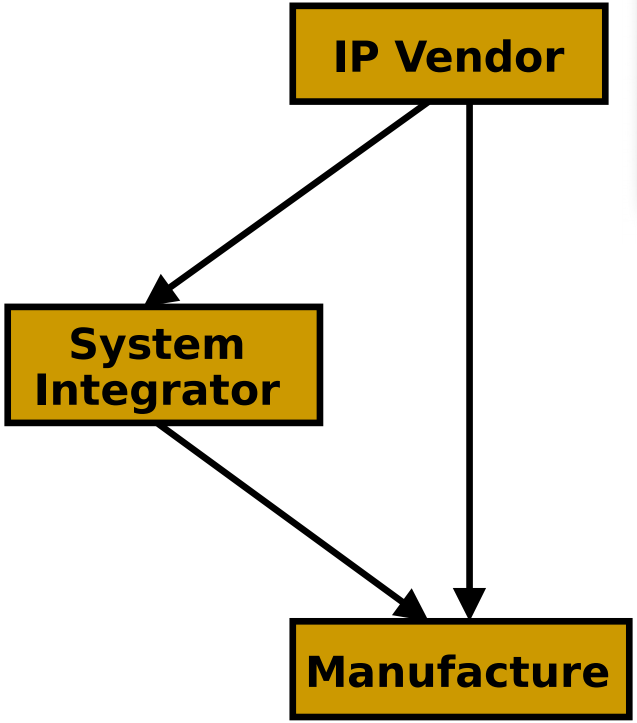
Unclustered defects
Wafer yield = 12/22 = 0.55

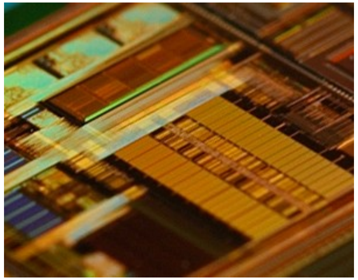
Clustered defects (VLSI)
Wafer yield = 17/22 = 0.77

Hardware Threats



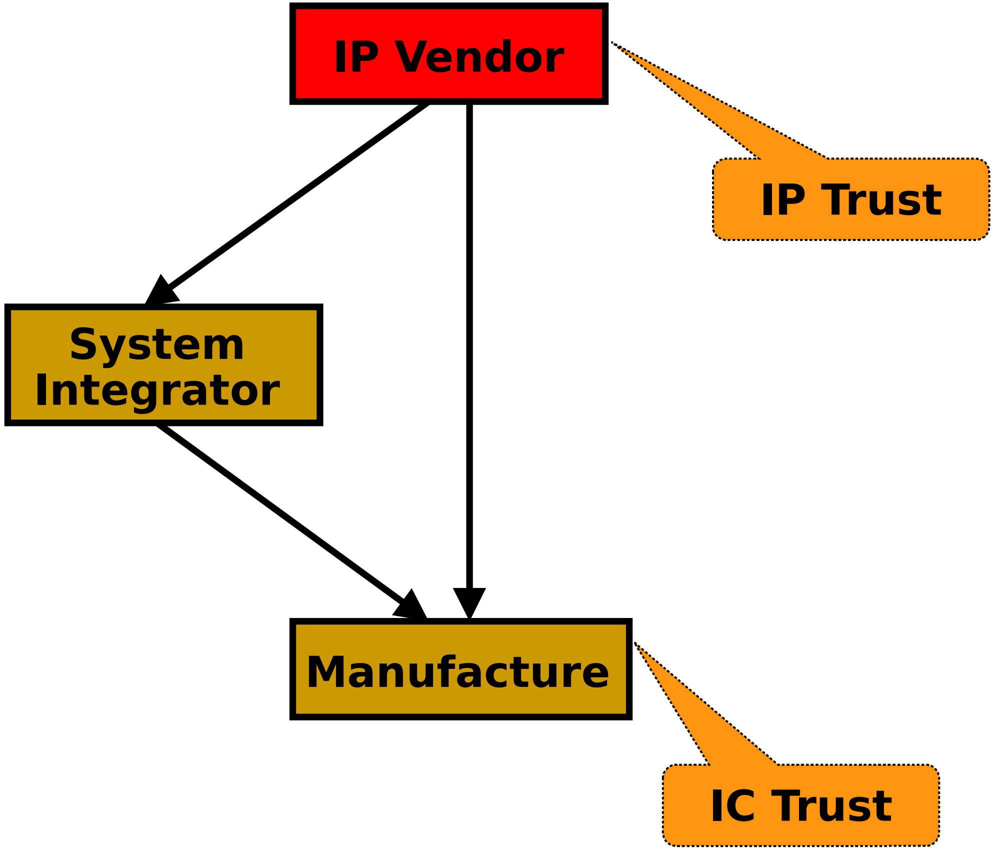
```
OR2X1 U1468 ( .IN1(n1317), .IN2(g45), .Q(n1347) );
NOR4X0 U1469 ( .IN1(g31), .IN2(g41), .IN3(g46), .IN4(n1449), .QN(n1319) );
INVX0 U1470 ( .INP(g45), .ZN(n1448) );
INVX0 U1471 ( .INP(g44), .ZN(n1351) );
NAND2X1 U1472 ( .IN1(n1319), .IN2(n1809), .QN(n1317) );
NAND2X1 U1473 ( .IN1(g44), .IN2(g43), .QN(n1336) );
OR2X1 U1474 ( .IN1(n1343), .IN2(n1448), .Q(n1335) );
NOR2X0 U1475 ( .IN1(g41), .IN2(n1320), .QN(n1447) );
INVX0 U1476 ( .INP(g46), .ZN(n1450) );
NAND2X1 U1477 ( .IN1(n1486), .IN2(n1485), .QN(n1494) );
NOR2X0 U1478 ( .IN1(g10720), .IN2(g10664), .QN(n1437) );
NOR2X0 U1479 ( .IN1(n1317), .IN2(n1322), .QN(n1325) );
OA21X1 U1480 ( .IN1(n1336), .IN2(n1335), .IN3(n1334), .Q(n1359) );
INVX0 U1481 ( .INP(g48), .ZN(n1449) );
NAND2X1 U1482 ( .IN1(g48), .IN2(n1806), .QN(n1324) );
NAND2X1 U1483 ( .IN1(g45), .IN2(n1807), .QN(n1322) );
NAND2X1 U1484 ( .IN1(n1327), .IN2(n1348), .QN(n1349) );
NOR2X0 U1485 ( .IN1(n1351), .IN2(n1347), .QN(n1327) );
INVX0 U1486 ( .INP(g43), .ZN(n1348) );
NAND2X1 U1487 ( .IN1(g47), .IN2(n1319), .QN(n1343) );
NOR2X0 U1488 ( .IN1(g1868), .IN2(n1591), .QN(n1646) );
INVX0 U1489 ( .INP(g1696), .ZN(n1460) );
NOR2X0 U1490 ( .IN1(n1705), .IN2(n1793), .QN(n1708) );
                                                   274,3          10%
```





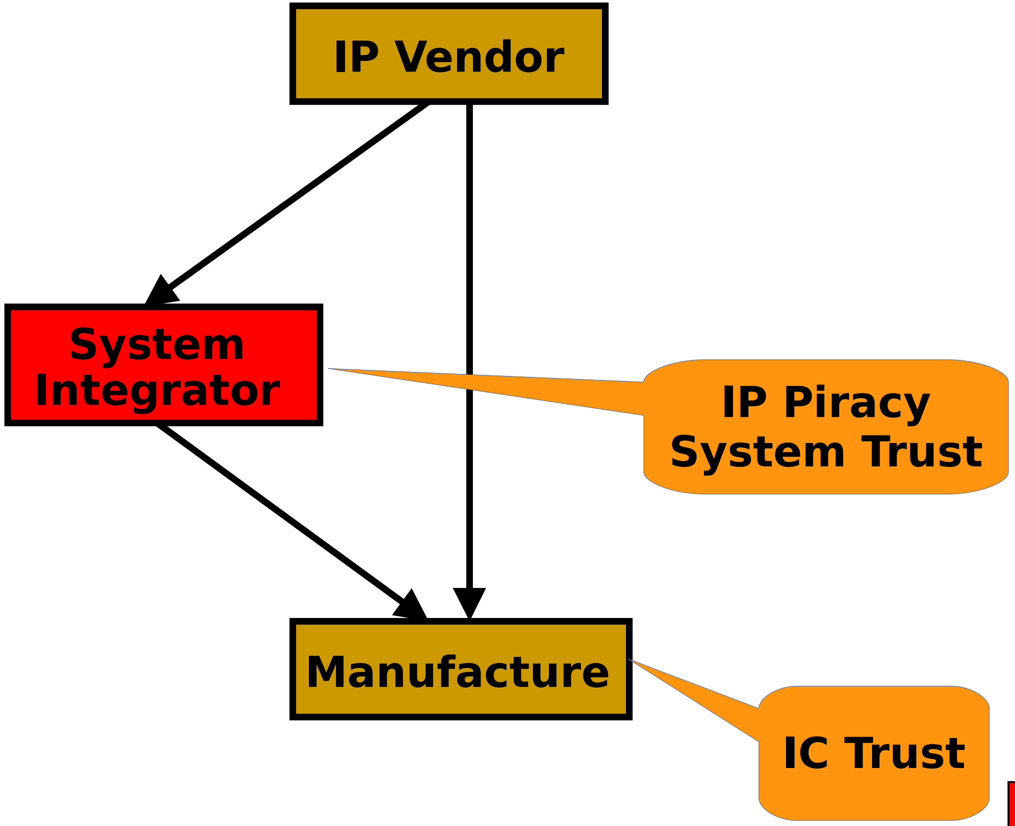
Any of these steps can be untrusted

Hardware Threats



Untrusted

Hardware Threats



IP Vendor

System Integrator

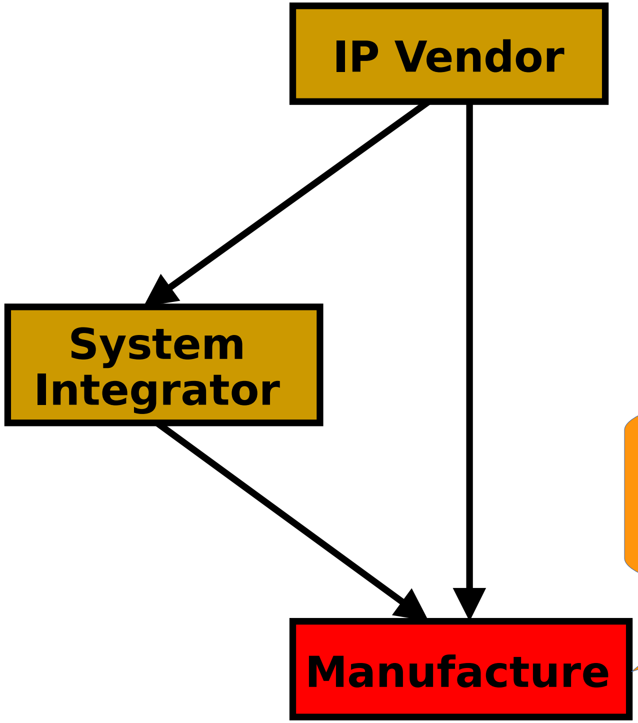
Manufacture

IP Piracy
System Trust

IC Trust

🟥 Untrusted

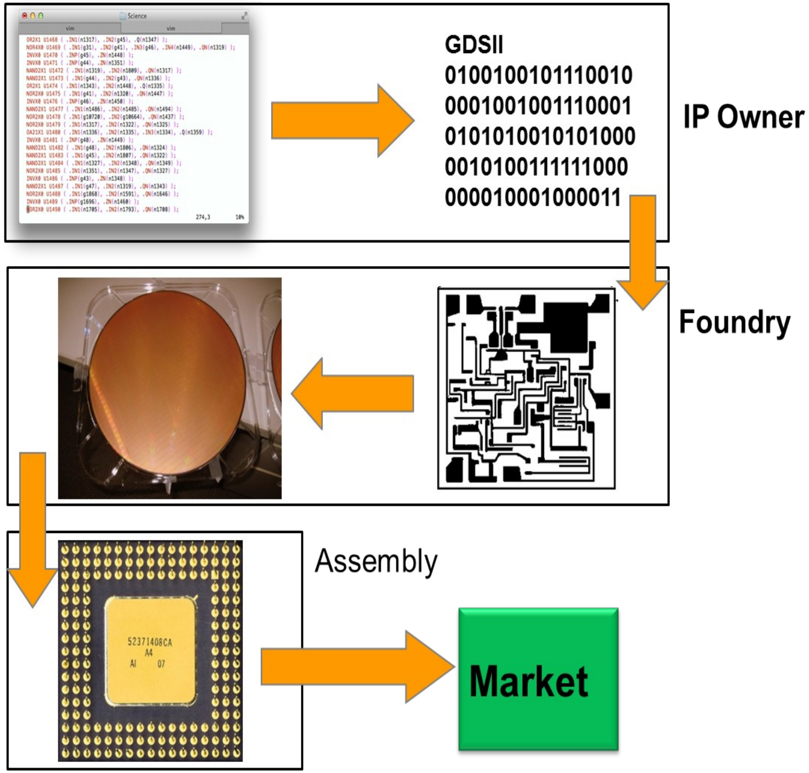
Hardware Threats



IC Trust
IC Piracy (Counterfeiting)
Secure Manufacture Test

🟥 Untrusted

Chip Production Flow

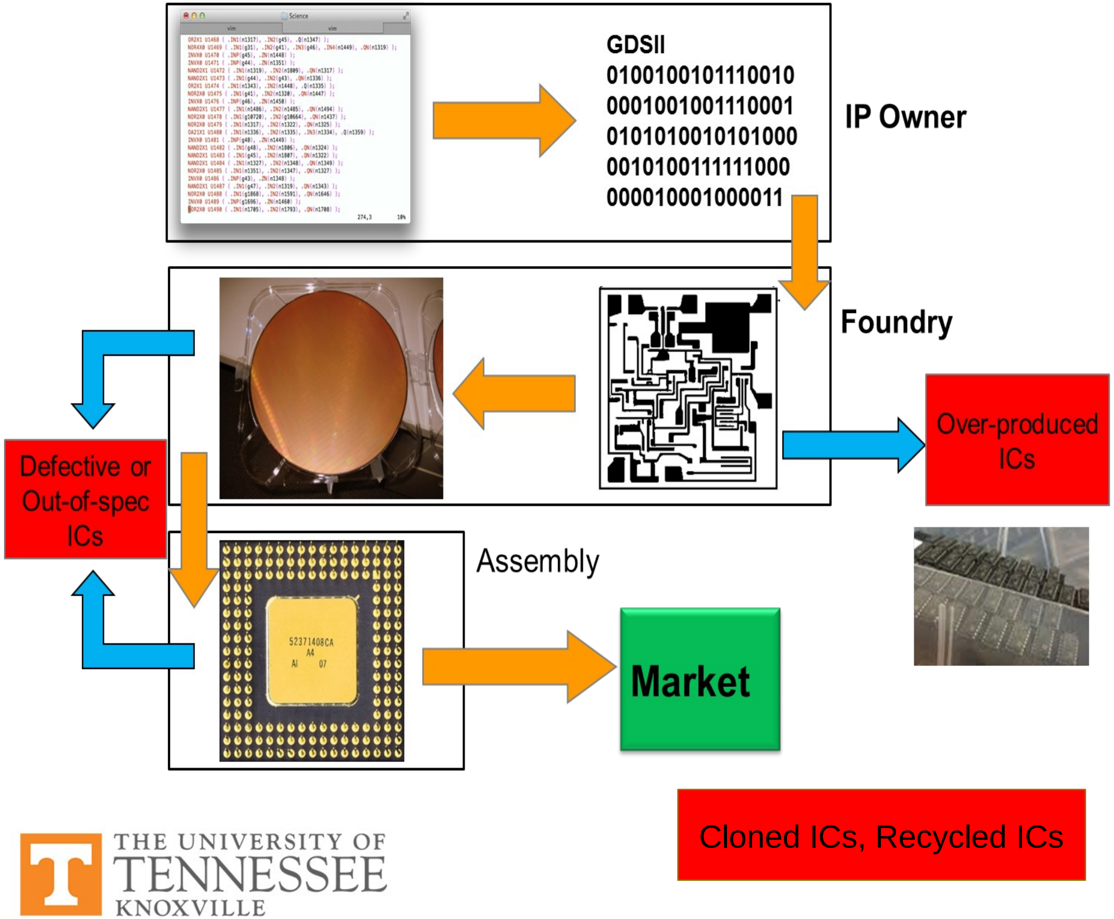


```
OR2X1 U1468 ( .IN1(n1317), .IN2(g45), .Q(n1347) );
NOR4X0 U1469 ( .IN1(g31), .IN2(g41), .IN3(g46), .IN4(n1449), .QN(n1319) );
INVX0 U1470 ( .INP(g45), .ZN(n1448) );
INVX0 U1471 ( .INP(g44), .ZN(n1351) );
NAND2X1 U1472 ( .IN1(n1319), .IN2(n1809), .QN(n1317) );
NAND2X1 U1473 ( .IN1(g44), .IN2(g43), .QN(n1336) );
OR2X1 U1474 ( .IN1(n1343), .IN2(n1448), .Q(n1335) );
NOR2X0 U1475 ( .IN1(g41), .IN2(n1320), .QN(n1447) );
INVX0 U1476 ( .INP(g46), .ZN(n1450) );
NAND2X1 U1477 ( .IN1(n1486), .IN2(n1485), .QN(n1494) );
NOR2X0 U1478 ( .IN1(g10720), .IN2(g10664), .QN(n1437) );
NOR2X0 U1479 ( .IN1(n1317), .IN2(n1322), .QN(n1325) );
OA21X1 U1480 ( .IN1(n1336), .IN2(n1335), .IN3(n1334), .Q(n1359) );
INVX0 U1481 ( .INP(g48), .ZN(n1449) );
NAND2X1 U1482 ( .IN1(g48), .IN2(n1806), .QN(n1324) );
NAND2X1 U1483 ( .IN1(g45), .IN2(n1807), .QN(n1322) );
NAND2X1 U1484 ( .IN1(n1327), .IN2(n1348), .QN(n1349) );
NOR2X0 U1485 ( .IN1(n1351), .IN2(n1347), .QN(n1327) );
INVX0 U1486 ( .INP(g43), .ZN(n1348) );
NAND2X1 U1487 ( .IN1(g47), .IN2(n1319), .QN(n1343) );
NOR2X0 U1488 ( .IN1(g1868), .IN2(n1591), .QN(n1646) );
INVX0 U1489 ( .INP(g1696), .ZN(n1460) );
NOR2X0 U1490 ( .IN1(n1705), .IN2(n1793), .QN(n1708) );
                                                        274,3          10%
```

GDSII
0100100101110010
0001001001110001
0101010010101000
0010100111111000
0000100010000011

- Little communication between IP owner and foundry
- Foundry *usually* trusted with full design
- Responsible for production of requested amount of chips
- IP holder provides foundry with all test patterns and responses

Chip Production Flow



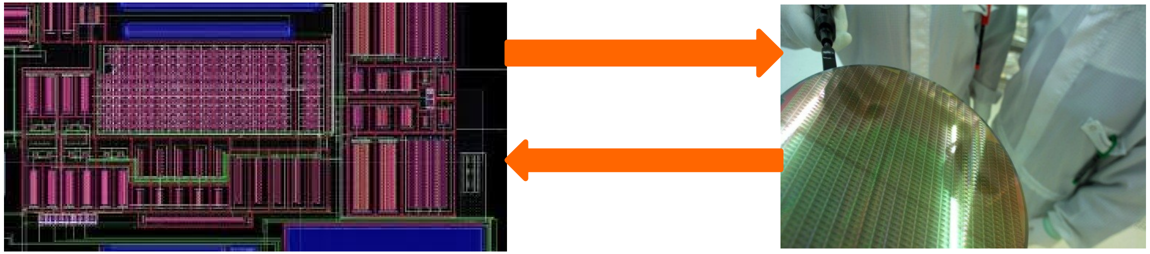
```
OR2X1 U1468 ( .IN1(n1317), .IN2(g45), .Q(n1347) );
NOR4X0 U1469 ( .IN1(g31), .IN2(g41), .IN3(g46), .IN4(n1449), .QN(n1319) );
INVX0 U1470 ( .INP(g45), .ZN(n1448) );
INVX0 U1471 ( .INP(g44), .ZN(n1351) );
NAND2X1 U1472 ( .IN1(n1319), .IN2(n1809), .QN(n1317) );
NAND2X1 U1473 ( .IN1(g44), .IN2(g43), .QN(n1336) );
OR2X1 U1474 ( .IN1(n1343), .IN2(n1448), .Q(n1335) );
NOR2X0 U1475 ( .IN1(g41), .IN2(n1320), .QN(n1447) );
INVX0 U1476 ( .INP(g46), .ZN(n1450) );
NAND2X1 U1477 ( .IN1(n1486), .IN2(n1485), .QN(n1494) );
NOR2X0 U1478 ( .IN1(g10720), .IN2(g10664), .QN(n1437) );
NOR2X0 U1479 ( .IN1(n1317), .IN2(n1322), .QN(n1325) );
OA21X1 U1480 ( .IN1(n1336), .IN2(n1335), .IN3(n1334), .Q(n1359) );
INVX0 U1481 ( .INP(g48), .ZN(n1449) );
NAND2X1 U1482 ( .IN1(g48), .IN2(n1806), .QN(n1324) );
NAND2X1 U1483 ( .IN1(g45), .IN2(n1807), .QN(n1322) );
NAND2X1 U1484 ( .IN1(n1327), .IN2(n1348), .QN(n1349) );
NOR2X0 U1485 ( .IN1(n1351), .IN2(n1347), .QN(n1327) );
INVX0 U1486 ( .INP(g43), .ZN(n1348) );
NAND2X1 U1487 ( .IN1(g47), .IN2(n1319), .QN(n1343) );
NOR2X0 U1488 ( .IN1(g1868), .IN2(n1591), .QN(n1646) );
INVX0 U1489 ( .INP(g1696), .ZN(n1460) );
NOR2X0 U1490 ( .IN1(n1705), .IN2(n1793), .QN(n1708) );
                                                    274,3        10%
```

- Foundry looks for its own profit
- Once mask is produced, IC fabrication relatively simple
- Lack of communication makes it difficult for owner to track chips

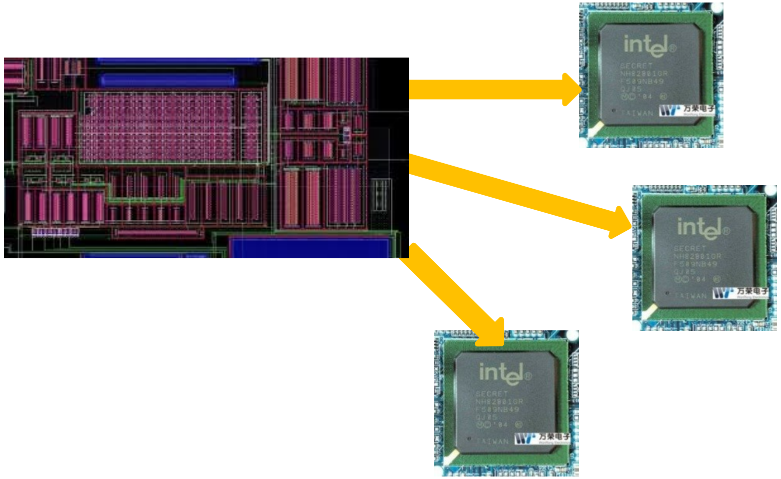
THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

Need for Hardware Metering

- Need better communication between IP owner & foundry



- Need for IP owner ability to *track produced chips*



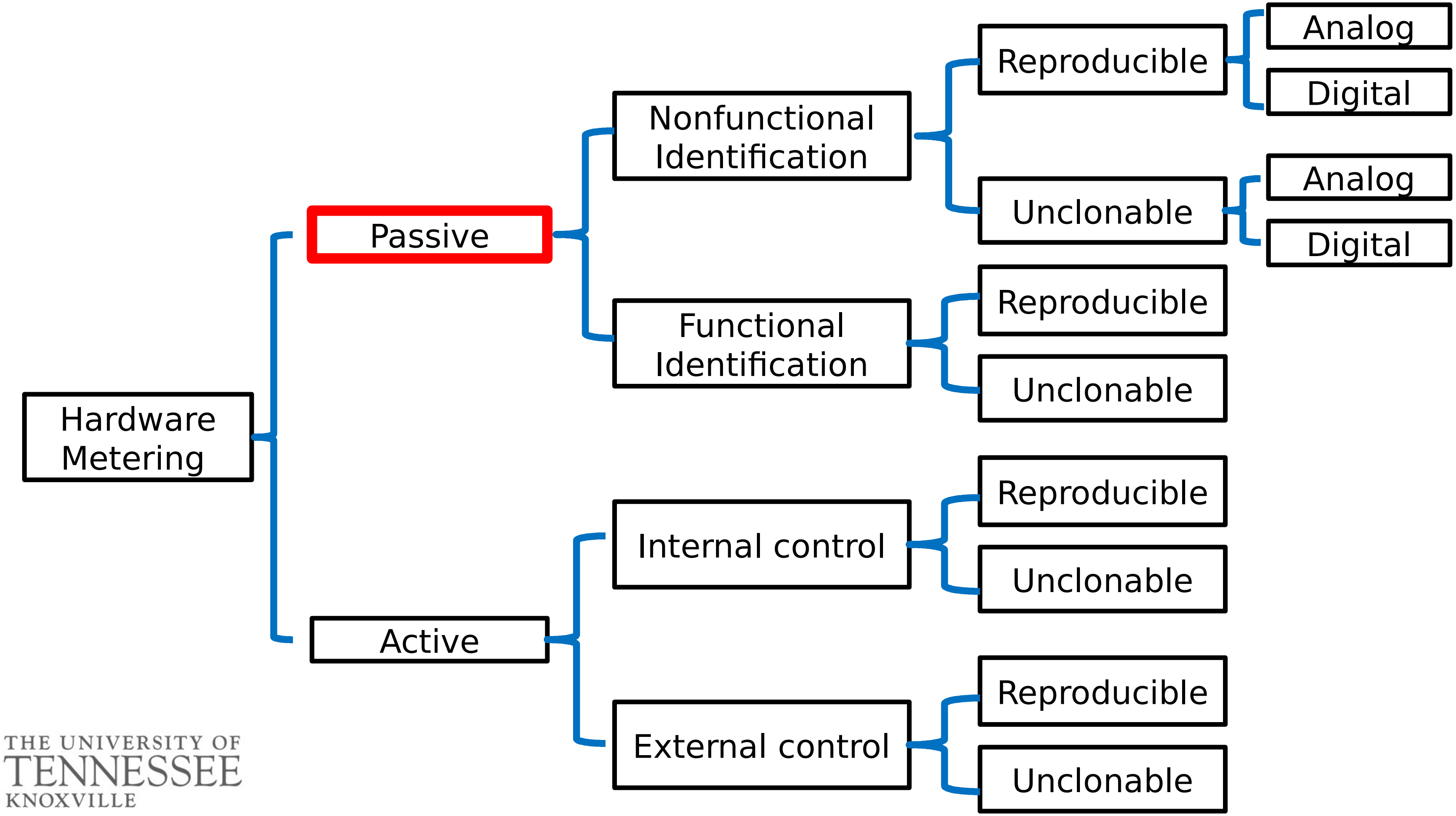
Hardware Metering

- Set of security protocols that enable IP owners to achieve post-fabrication control over their integrated circuits

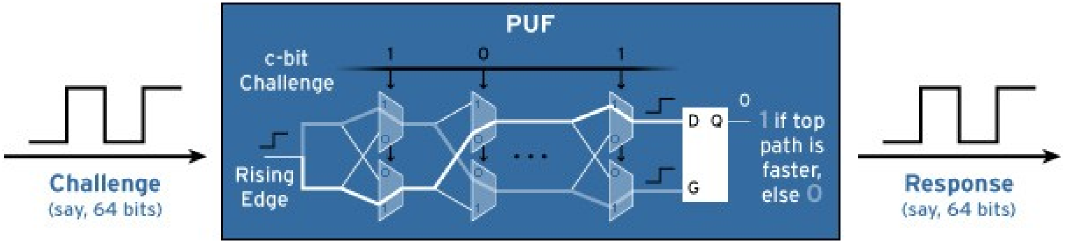
- Methods attempt to uniquely tag each chip to facilitate tracing them once in the market

- Two main categories: Active and Passive

Taxonomy of Metering Methods

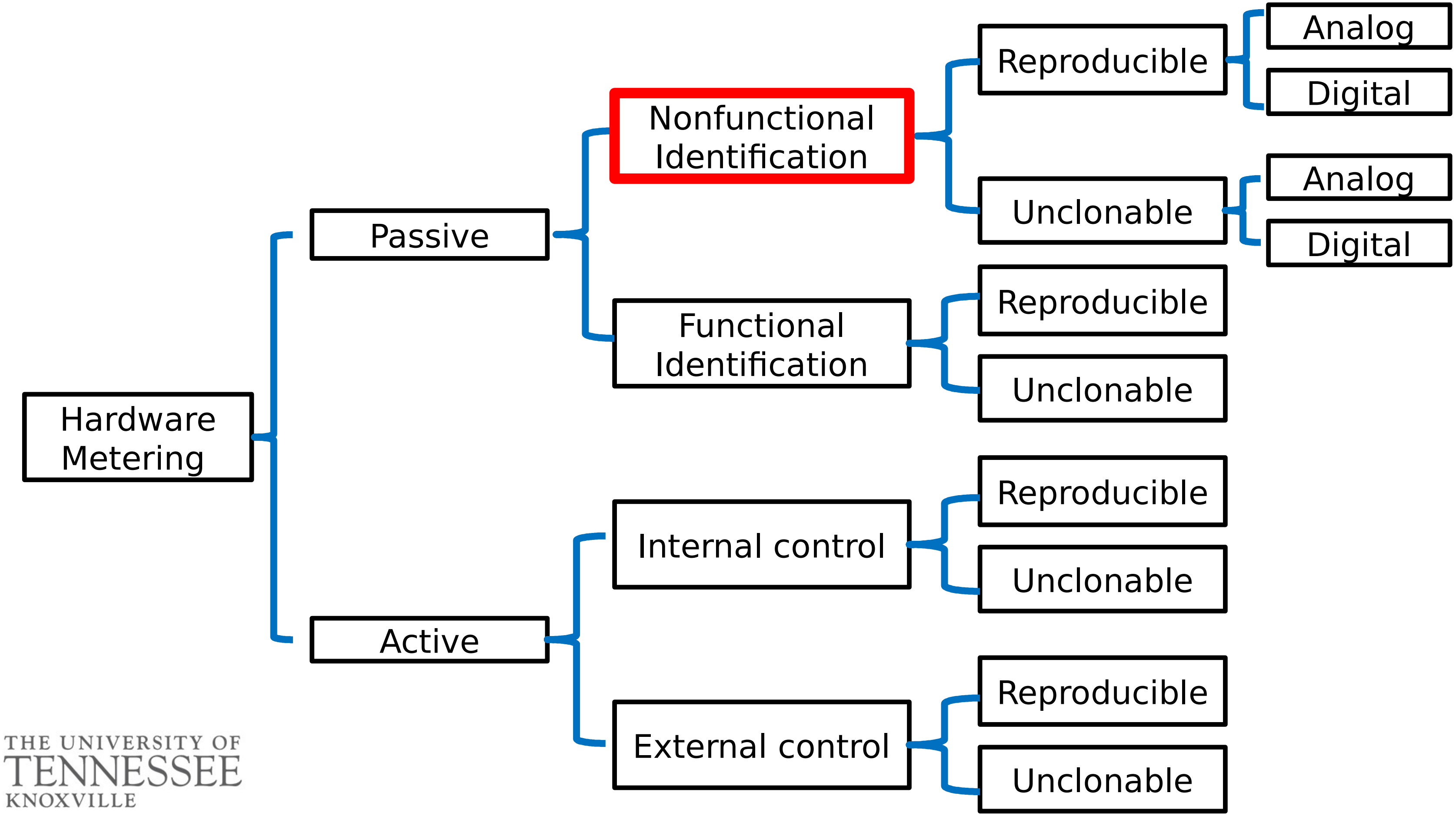


Passive Metering



- ICs can be passively monitored
- Can be achieved by physically identifying:
  - Serial numbers on chips
  - Strong unique identifiers in memory – nonfunctional identification
- Tagging an IC's functionality: Functional Identification

Taxonomy of Metering Methods

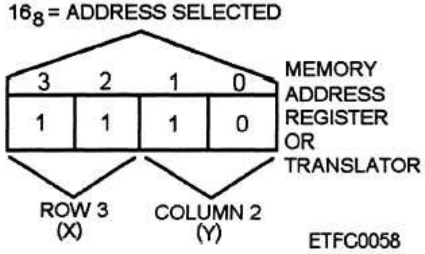


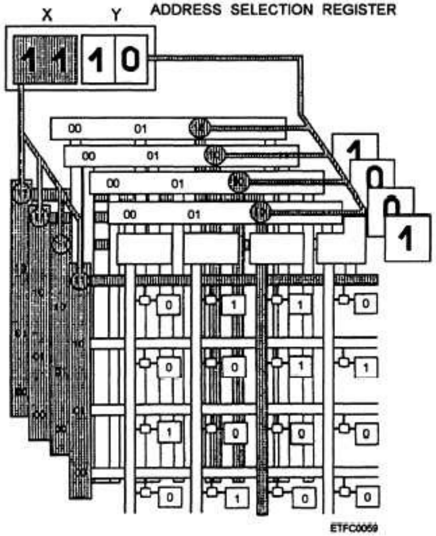
Nonfunctional Identification

- Unique ID is separate from the chip's functionality
- Vulnerable to cloning and/or removal
- Possible to overproduce
  - Foundry can produce multiple chips with same tag
  - Out of millions of chips, probability of finding two matching tags is small
- Two main types:
  - Reproducible
  - Unclonable

Nonfunctional Identification: Reproducible Identifiers

- Unique ID stored on package, on die, or in on-chip memory
- Examples:
  - Indented serial numbers
  - Digitally stored serial numbers
- Advantages:
  - Do not depend on randomness
  - Easy to track/identify
- Disadvantages:
  - Easy to clone/modify
  - Easy to overproduce


16_8 = ADDRESS SELECTED
3
2
1
0
1
1
1
0
MEMORY ADDRESS REGISTER OR TRANSLATOR
ROW 3 (X)
COLUMN 2 (Y)
ETFC0058



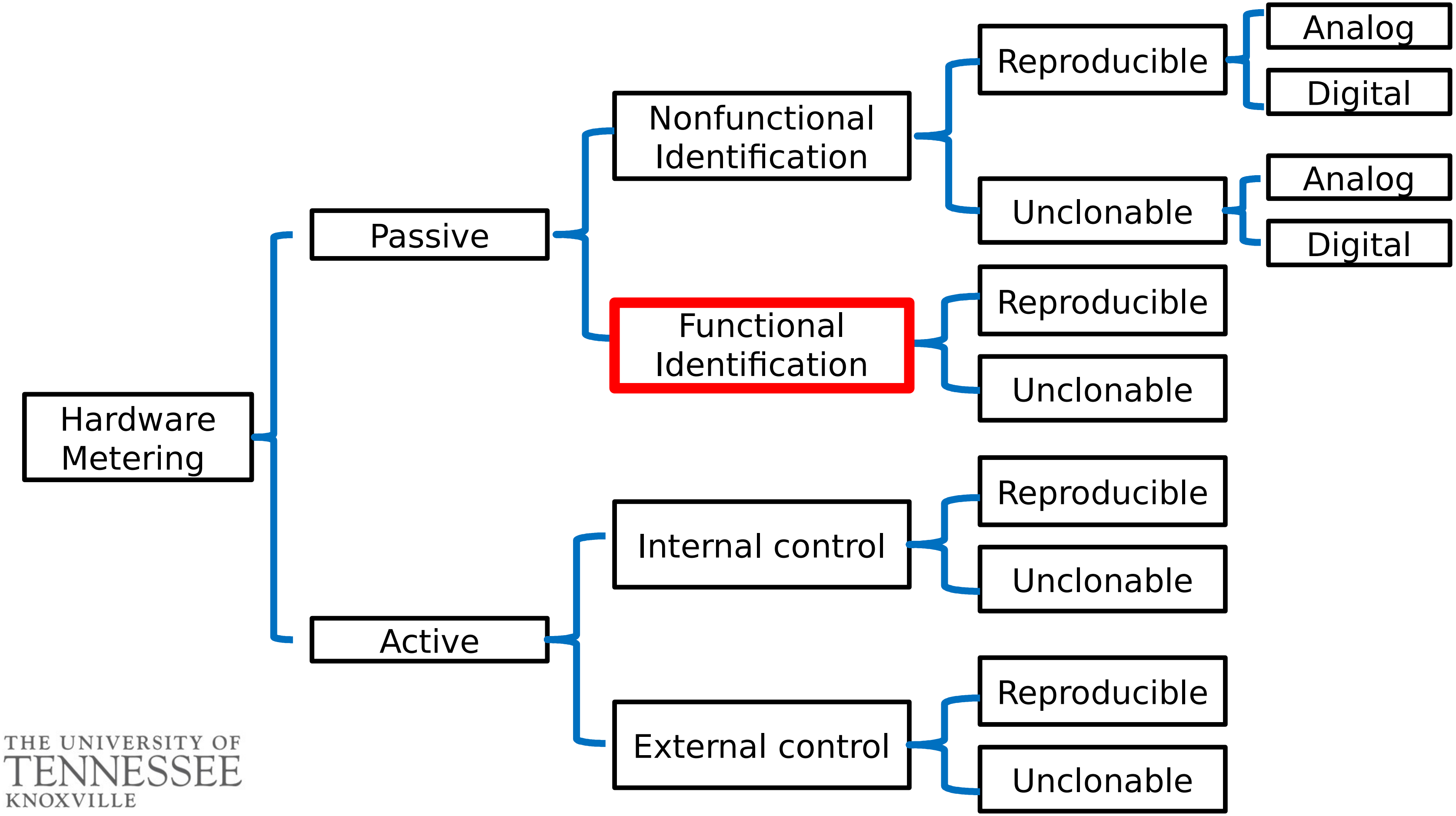
Nonfunctional Identification: Unclonable Identifiers

- Uses random process variations in silicon to generate random unique numbers or fingerprints
- If additional logic needed to generate these values, the method is said to be extrinsic
- If no additional logic needed, method is intrinsic
- Advantages:
  - Values cannot be reproduced due to randomness in process
- Disadvantages:
  - Foundry could overproduce ICs without knowledge of IP owner
  - Method *does not prevent* counterfeiting but owner can detect overproduced chip by comparing to fingerprint database

Unclonable Identifiers

- Extrinsic methods:
  - Require additional logic such as PUF or ICID
  - ICID – threshold mismatch in array of transistors incurs different currents and therefore random numbers
  - PUF (Physical Unclonable Function) – several types
    - Series of ring oscillators (ROs) generate random value due to differences in oscillator frequencies
    - PUF sensitive to power supply noise, temperature, delay, etc. – the values likely change often (unreliable)
- Intrinsic methods:
  - Unique identification if external test vectors applied
  - Use IC leakage, power, timing, and path signatures
  - Does not need additional logic and can be readily used with existing designs

Taxonomy of Metering Methods



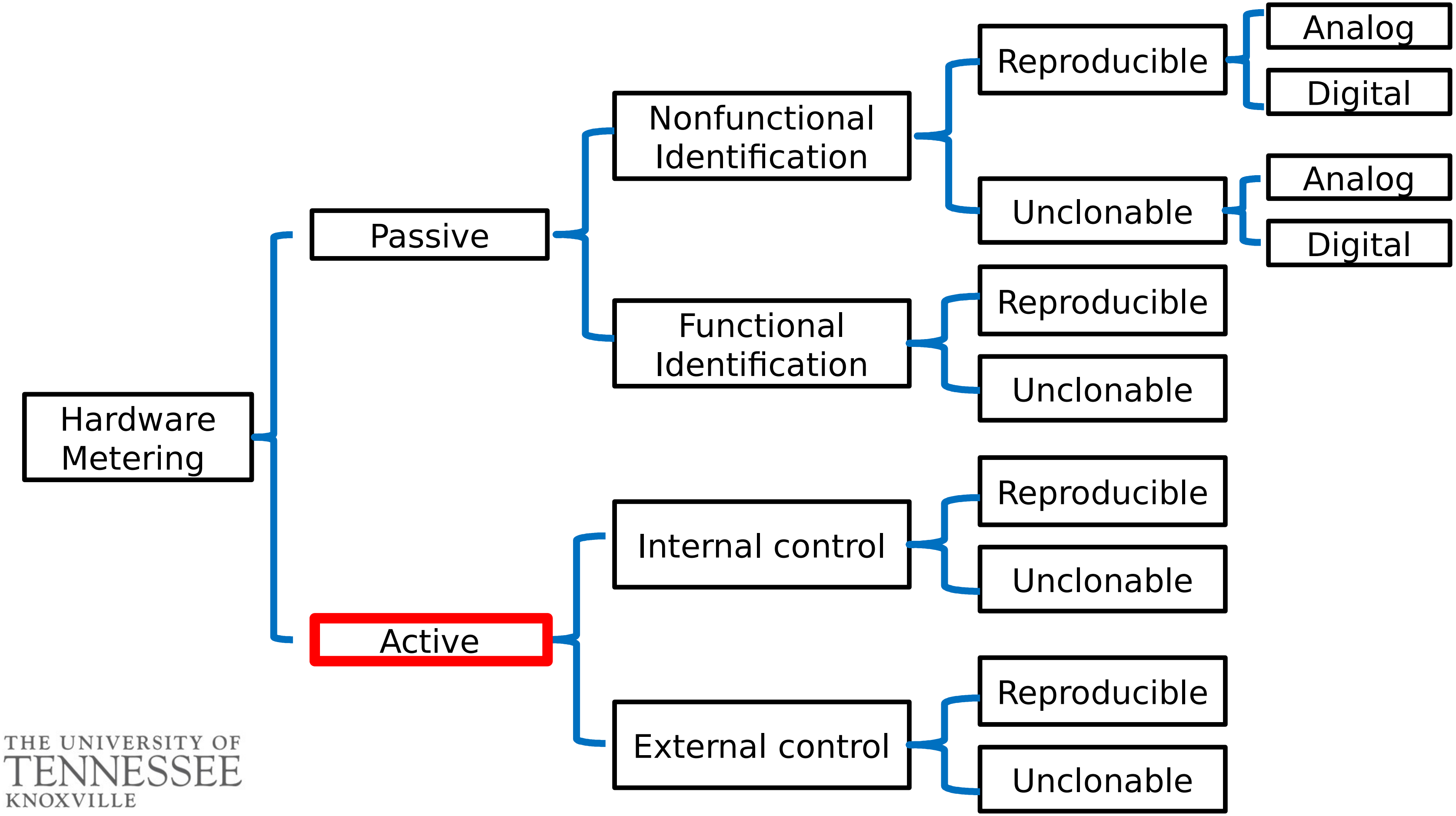
Functional Identification

- Identifiers linked to chip's internal functional details during synthesis
- Each chip's function gets a unique signature
    - Additional states added that generate same output
- Function unchanged from input to output
- Internal transactions unique to each chip
- Challenge in fabricating ICs with different paths from same mask

Functional Identification

- One method is fabricating chips from same mask and maintaining one programmable path
  – datapath programmed post-fabrication
  - IP owner provides correct input/key combination to foundry to program chip post-fabrication
- Additional work proposed adding redundant states
  – Programmable read logic enables selection of correct permutation for a control sequence
- Drawbacks
  - Testing such circuitry provides low coverage because functionality hidden during test by foundy & assembly
  - Requires additional circuitry that is useless after testing

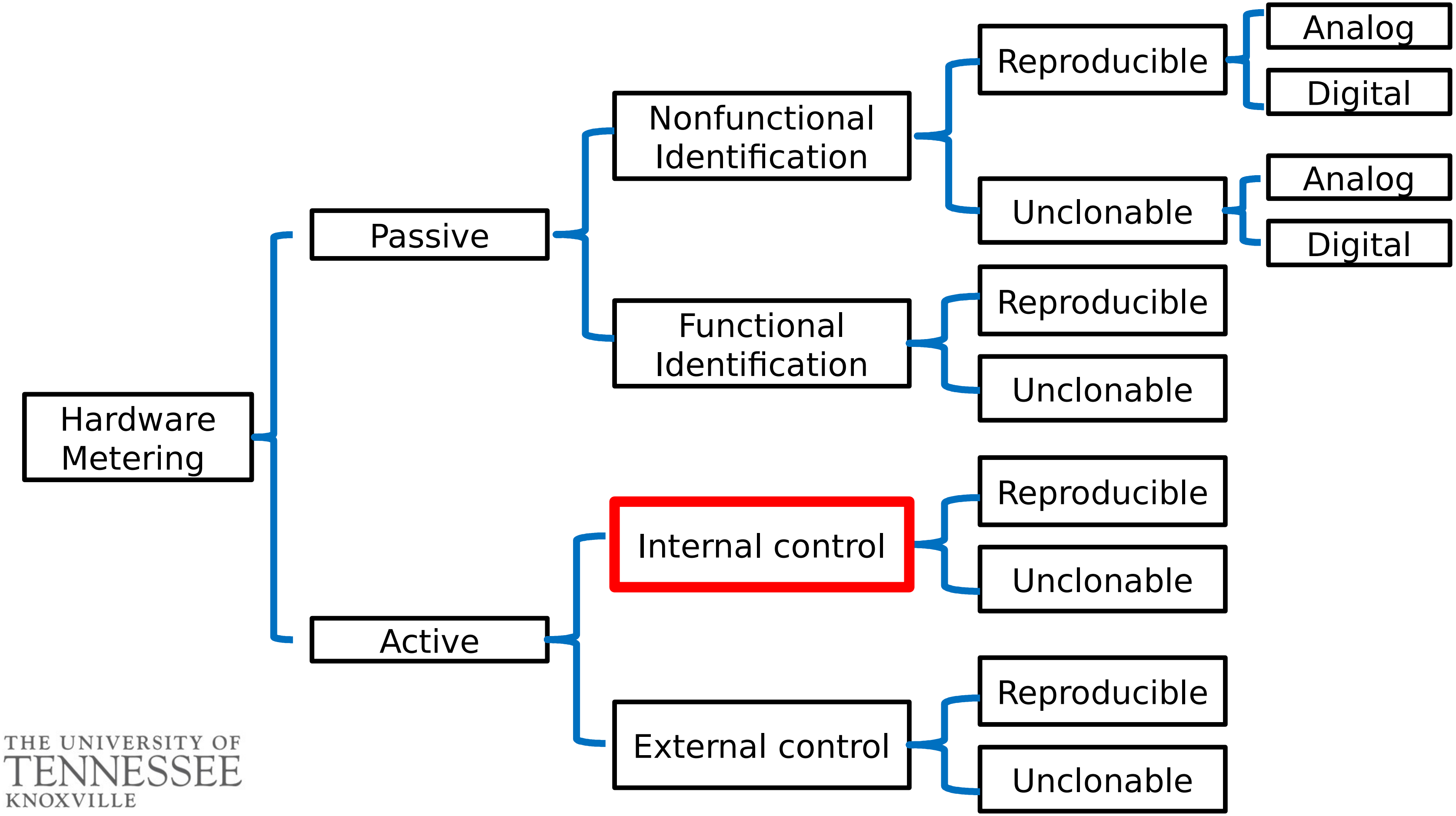
Taxonomy of Metering Methods



Active Metering

- Provides active method for designer to enable, control or disable IC
- Unlike passive metering, active metering requires communication between design house (IP owner) and foundry
- Two types: internal and external

Taxonomy of Metering Methods

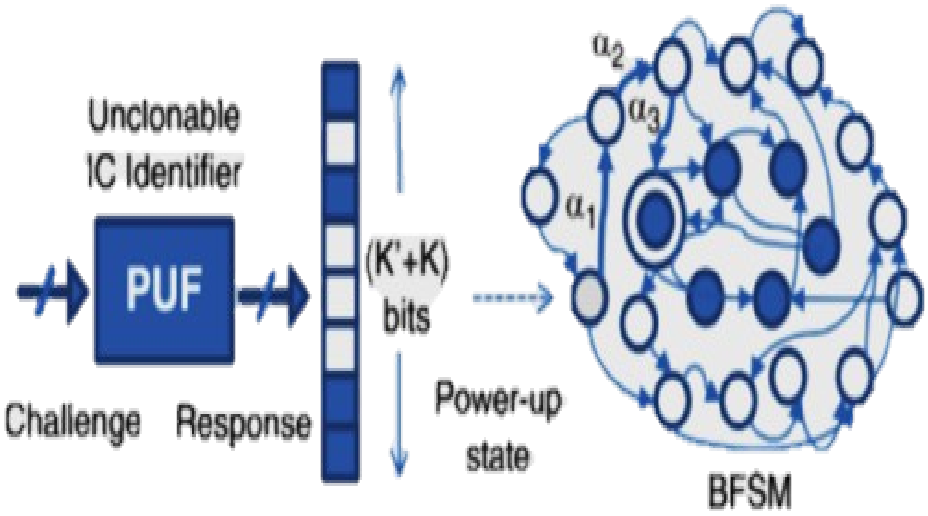


Internal (Integrated) Active Metering

- Hides states and transition in design that can only be accessed by designer
- Locks are embedded within structure of computation model in hardware design in form of FSM
- Adding additional states or duplicating certain states in FSM adds ability for designer to decide which datapath (sequence of states) to use post-fabrication
  - Since states are added, specific combinations are needed to bring FSM to correct output
  - Only IP owner knows the correct combinations

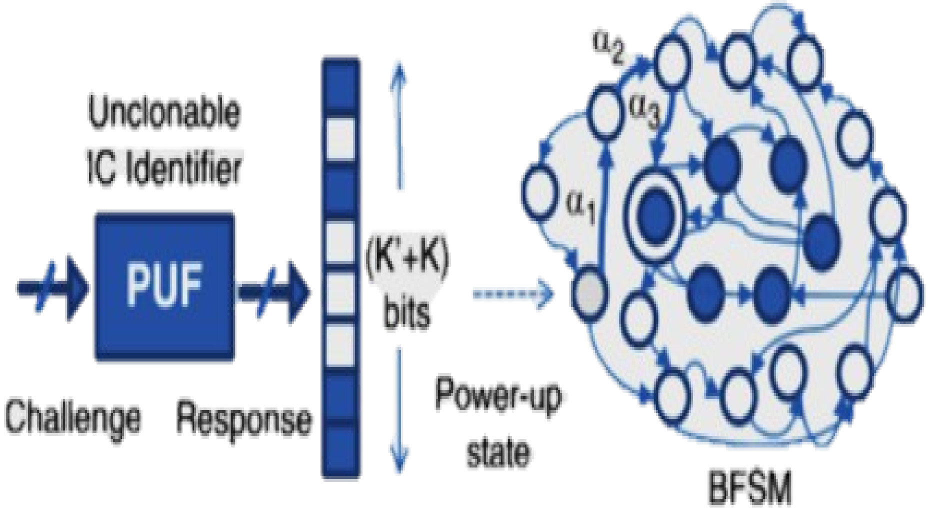
Internal (Integrated) Active Metering

- States and transitions for controlling chip integrated in functional specifications
- K = $\log_2(S)$ flops needed to implement S states
- Adding S1 states requires K1 = $\log_2(S1+S)$ flops
- Few additional flops can exponentially increase number of states

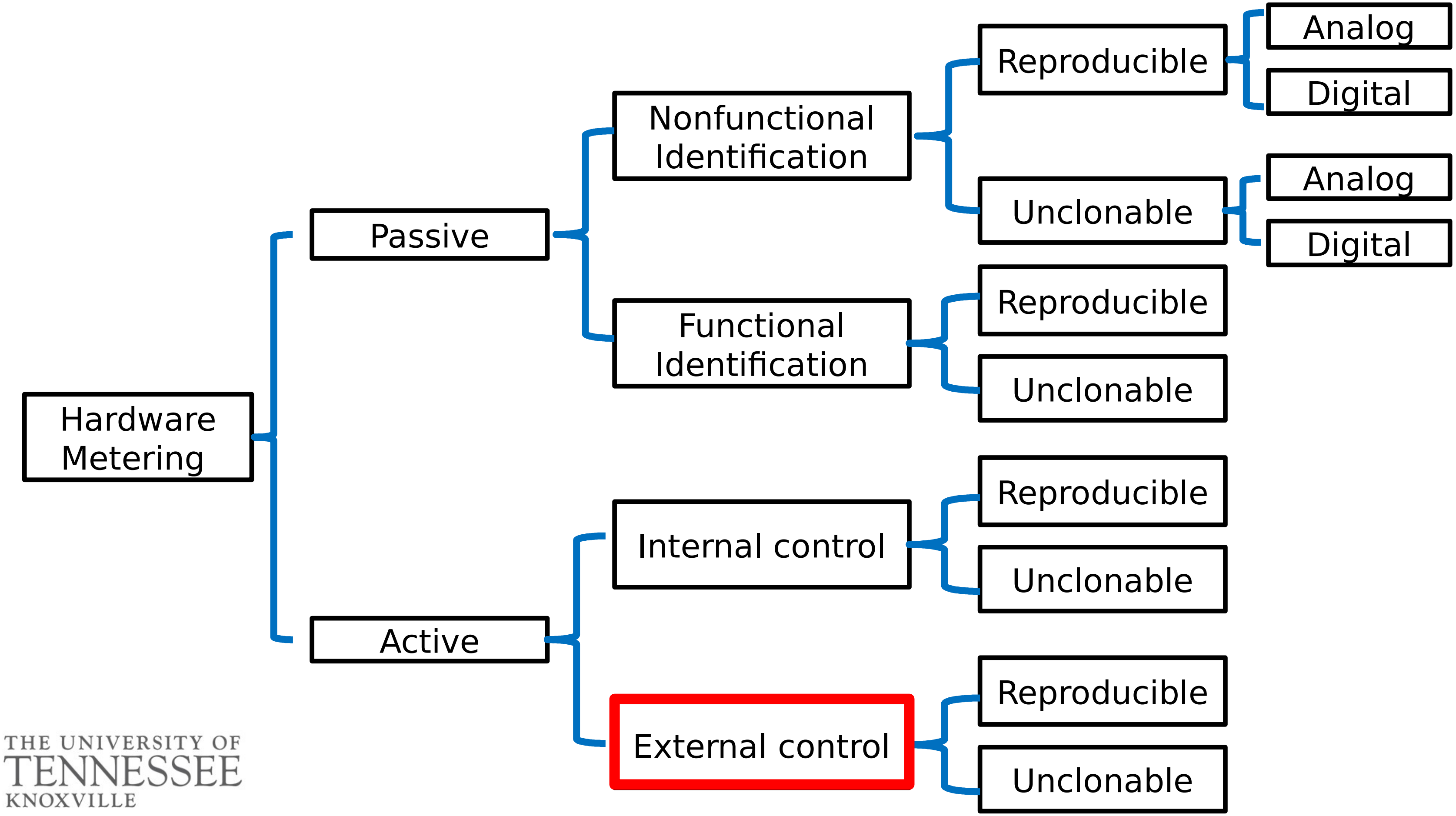


Internal (Integrated) Active Metering

- PUF generates random values, sends device into random FSM state
- Only IP owner with knowledge of FSM can find correct sequence to set FSM to RESET state
- Storing sequence on-chip requires additional logic and also requires long wait to shift in entire sequence



Taxonomy of Metering Methods



Hardware Metering

Passive

Nonfunctional Identification

Reproducible

Analog

Digital

Unclonable

Analog

Digital

Functional Identification

Reproducible

Unclonable

Active

Internal control

Reproducible

Unclonable

External control

Reproducible

Unclonable

THE UNIVERSITY OF TENNESSEE KNOXVILLE

External Active Metering

- External asymmetric cryptographic techniques lock IC
- Cryptographic circuits rely on public and private keys to give IP owner control over activation/correct function
- Only IP owner knows private key to unlock IC functionality or testability