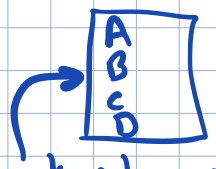
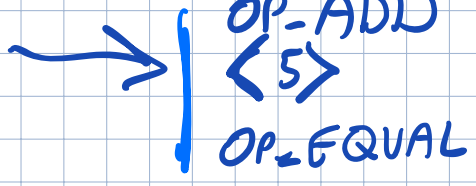
Transactions

- Theory of operation
  - Execute satisfier script



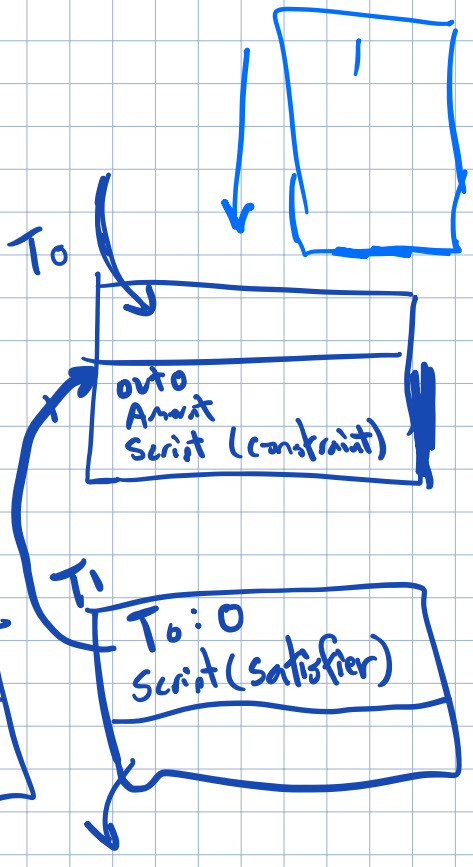
  - Execute constraint script on Stack!
    - Check Top Stack item
    - Examples

- Constraint



- Satisfier
  - <1>
  - <4>

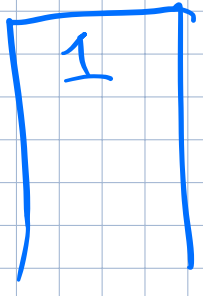
<Foo>
<Bar>



P2PK:

Constraint:

- <pubkey>
  CHECK_SIG



Satisfier:

- <signature>

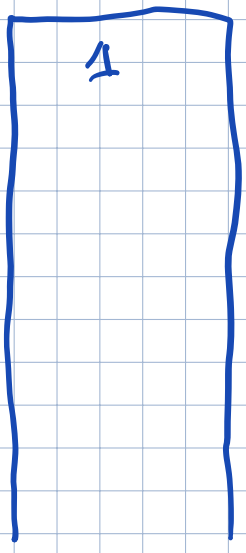
operation:

① execute satisfier
② Execute constraint

<sig>
<pubkey>
> CHECKSIG ⇒

- P2PKH (Address)

→ DUP
→ HASH160
→ <address (PKH)>
→ EQUAL_VERIFY
→ CHECKSIG



- Satisfier

<signature>
→ <Pubkey>

- operation

P2SH (CHEATS!)

Constraint:

HASH_160
DATA (redeem script hash)
OP_EQUAL

OP_EQUAL

Satisfier:

DATA (scriptArgs)

⋮

DATA (redeem script)

How is the redeem script executed?
(this is the cheat)

Multisig

Constraint:

[Min number of signers]

DATA (Public keys)

⋮

[Total # of signers - # of pub keys pushed]

OP_CHECKMULTISIG

SATISFIER:

0 (bug added a field)

DATA (sig)

⋮

CLAUSE ACTIVATION

- Pick a random preimage, $h = H(s)$

IF

HASH160

DATA(h)

EQUALVERIFY

(code to Activate)
if s is disclosed

ELSE

(ALWAYS Activated Clause)

ENDIF

⇓

Cross chain atomic swap

SPOIL
if code is
"pay to counterparty"
OR
"pay to escrow"
can be used to
"spoil" the whole
TX!

in the "SPOIL"
scenario, this would
pay to the hash
secret h

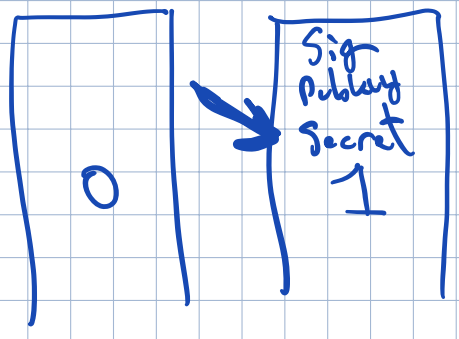
- Satisfier

activated script args
DATA(s)
1

HTLC

IF
  DUP
  HASH160
  <addressA>
  EQUAL_VERIFY
  CHECKSIG
ELSE
  <time in epoch seconds>
  CHECKLOCKTIMEVERIFY
  DUP
  HASH160
  <RefundAddress>
  EQUAL_VERIFY
  CHECKSIG
ENDIF

HASH160
<hash of secret>
EQUAL_VERIFY



Spend

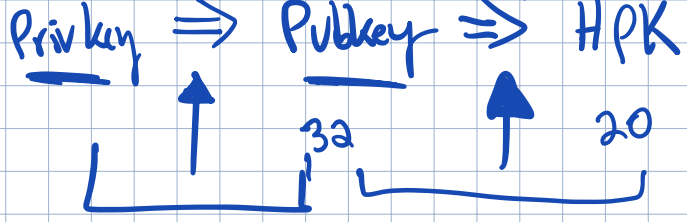
<sig>
<refund pubkey>
0

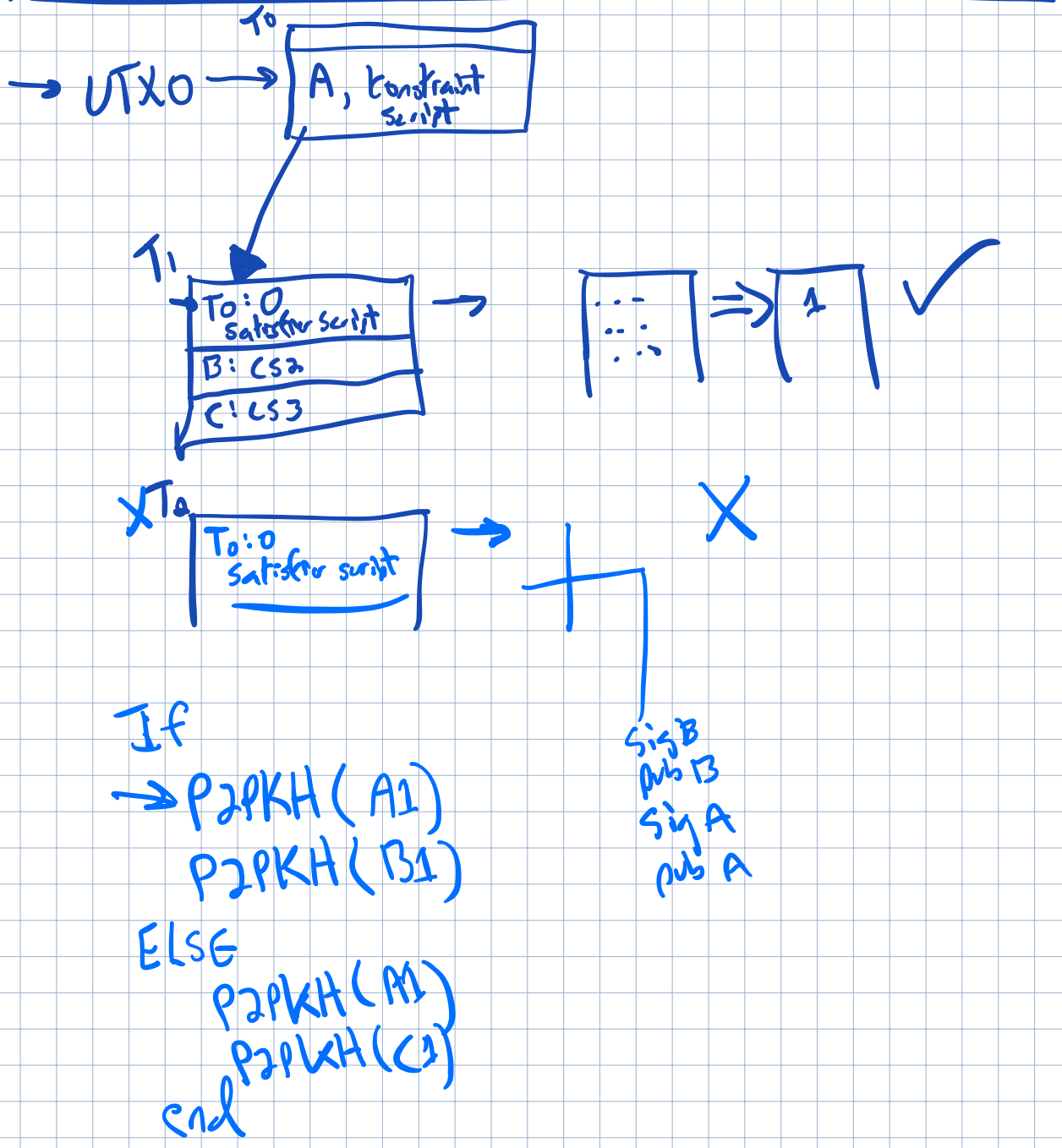
<sig>
<pubkey>
1

Preimage

ECC

Hash160


Privkey
Pubkey
HPK
32
20


T0
UTXO
A, Constraint script
T1
T0:0 satisfier script
B: CS2
C: CS3
T0:0 satisfier script
If
P2PKH(A1)
P2PKH(B1)
ELSE
P2PKH(A1)
P2PKH(C1)
end
SigB
PubB
SigA
PubA
1