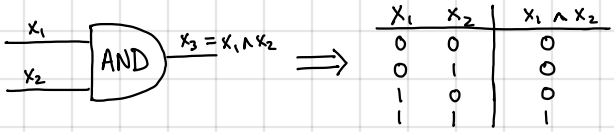
Yao's Protocol (2PC):

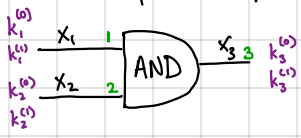
Key ingredient: "garbling" protocol (garbled circuits)



truth table:

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

1) Associate a pair of keys $(k_i^{(0)}, k_i^{(1)})$ with each wire i in the circuit



$k_i^{(b)}$: key associated with wire value b for wire i [symmetric encryption key]

2) Prepare <u>garbled truth table</u> for the gate
- ↳ Replace each entry of truth table with corresponding key
- ↳ Encrypt output key with each of the input keys

x_1		x_2		$x_3 = x_1 \wedge x_2$	
0	$k_1^{(0)}$	0	$k_2^{(0)}$	0	$k_3^{(0)}$
0	$k_1^{(0)}$	1	$k_2^{(1)}$	0	$k_3^{(0)}$
1	$k_1^{(1)}$	0	$k_2^{(0)}$	0	$k_3^{(0)}$
1	$k_1^{(1)}$	1	$k_2^{(1)}$	1	$k_3^{(1)}$

$\Longrightarrow$

$ct_{00} \leftarrow \text{Encrypt}(k_1^{(0)}, \text{Encrypt}(k_2^{(0)}, k_3^{(0)}))$
$ct_{01} \leftarrow \text{Encrypt}(k_1^{(0)}, \text{Encrypt}(k_2^{(1)}, k_3^{(0)}))$
$ct_{10} \leftarrow \text{Encrypt}(k_1^{(1)}, \text{Encrypt}(k_2^{(0)}, k_3^{(0)}))$
$ct_{11} \leftarrow \text{Encrypt}(k_1^{(1)}, \text{Encrypt}(k_2^{(1)}, k_3^{(1)}))$

} randomly shuffle ciphertexts

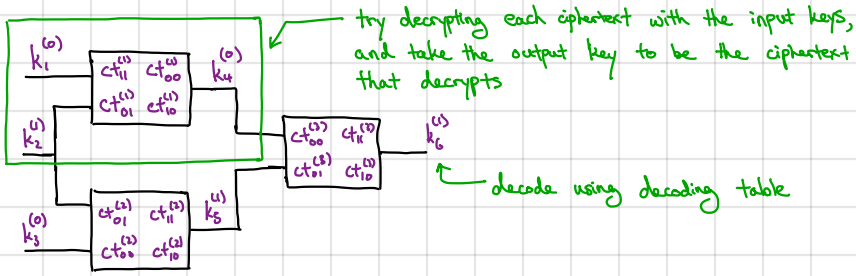
3) Construct decoding table for output values

$k_3^{(0)} \mapsto 0$
$k_3^{(1)} \mapsto 1$

} Alternatively, can just encrypt output values instead of keys for output wires

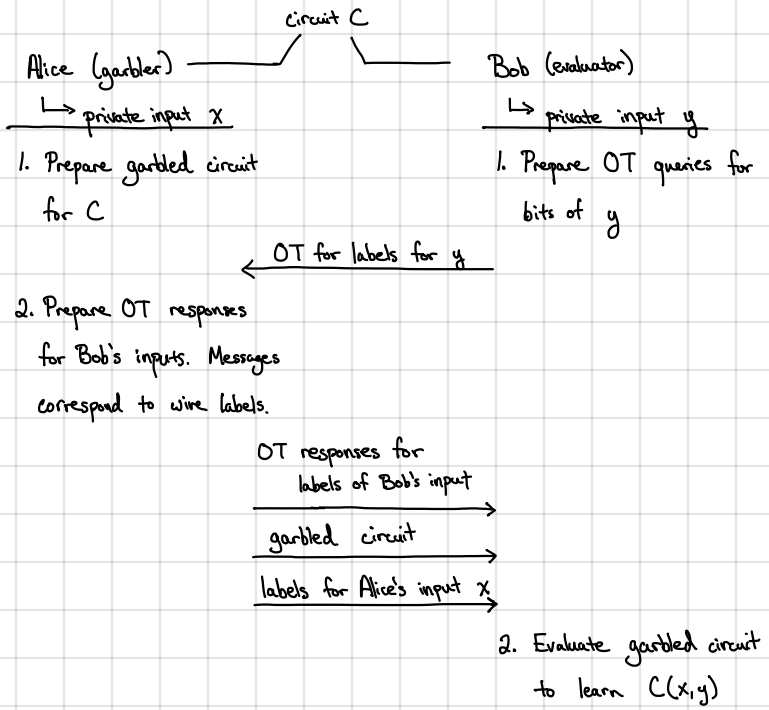
General garbling transformation: construct garbled table for each gate in the circuit, prepare decoding table for each output wire in the circuit

Evaluating a garbled circuit:



Invariant: given keys for input wires of a gate, can derive key corresponding to output wire $\Longrightarrow$ enables gate-by-gate evaluation of garbled circuit
- ↳ <u>Requirement</u>: Evaluator needs to obtain keys (labels) for its inputs (but without revealing which set of labels it requested)

Yao's garbled circuit protocol:



Correctness: Follows by correctness of OT and of the garbling construction

Security: Relies on security of OT and garbling transformation
- relies on OT simulator to simulate OT responses
- ↳ Simulate Bob's view given output of computation (using the garbled circuit simulator)
- ↳ Simulate Alice's view using OT simulator

Variants:
1. If both parties should learn output, Bob can send it to Alice.
2. If Alice and Bob should learn distinct outputs, Alice can have the functionality output a blinded / encrypted version of her output.
3. Can extend to malicious security (need additional rounds and some modifications).

Many optimizations possible:
1. free XOR – no need to send garbled tables for XOR gates in circuit
2. half gates – only need two ciphertexts for each AND gate (not 4)

} AND and XOR are universal
↳ standard basis for garbled circuits

3. no need to double encrypt – can "encrypt" once using key derived from input keys