# Bounded Model Checking

Testing & Verification

Dept. of Computer Science & Engg, IIT Kharagpur

Pallab Dasgupta

Professor, Dept. of Computer Science & Engg.,
Professor-in-charge, AVLSI Design Lab,
Indian Institute of Technology Kharagpur

# Bounded Model Checking (BMC)

- **Broad Methodology**
  - We construct a Boolean formula that is satisfiable iff the underlying state transition system can realize a finite sequence of state transitions that satisfy the temporal property we are trying to validate
  - We use powerful SAT solvers to determine the satisfiability of the Boolean formula
  - The bound may be increased incrementally until we reach the diameter of the state transition graph

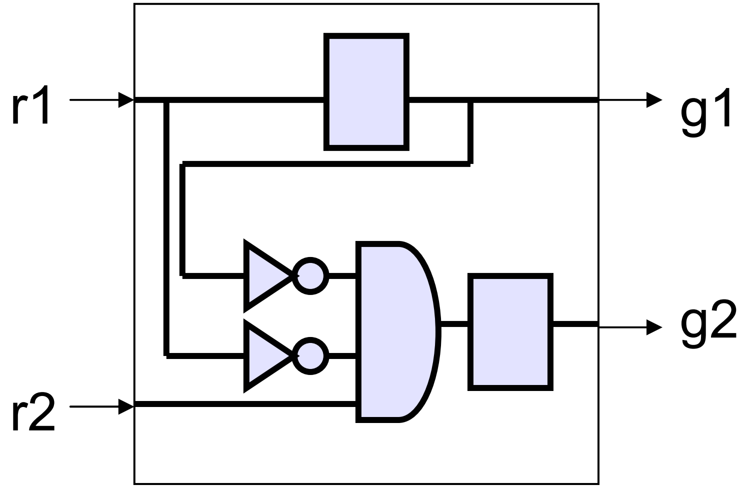
# Requirements

- Specification in temporal logic.
- System as a finite state machine.
- Bound, k, on path length.
  - In bounded model checking, only paths of bounded length k or less are considered.

# BMC: Translation to SAT

- We unfold the property into Boolean clauses over different time steps
- We unfold the state machine into Boolean clauses over the same number of time steps
- We check whether the clauses are together satisfiable

# Example: *Priority Arbiter*




***Implementation***

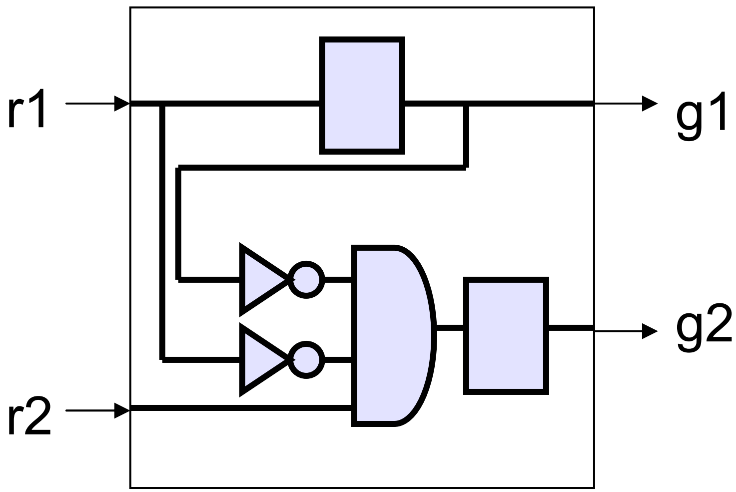
**Initial state: g1=0, g2=1**

***Specification***

**Property:**

- ***When r1 is high, g1 must be asserted for the next two cycles***
- ***In Linear Temporal Logic:*** **$G(\ r1 \Rightarrow Xg1 \wedge XXg1\ )$**

# Example: *Priority Arbiter*




**Transition Relation**:

$g2' \leftarrow r2 \wedge \neg\, r1 \wedge \neg\, g1$

$g1' \leftarrow r1$

**Initial state: g1=0, g2=1**

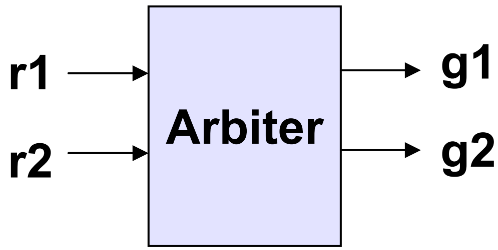
***Strategy:***

***Property:*** $G(\, r1 \Rightarrow Xg1 \wedge XXg1\, )$

***Negate property:*** $F(\, r1 \wedge (\neg Xg1 \vee \neg XXg1)\, )$

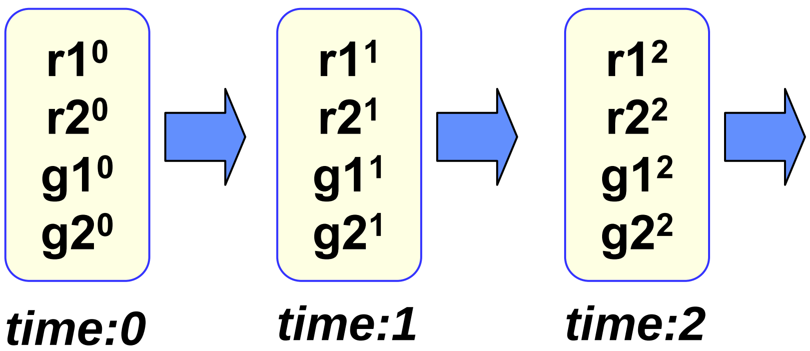
**Unfold transition relation one step at a time and check whether a witness for the negated property exists**

# Variables in Temporal Worlds



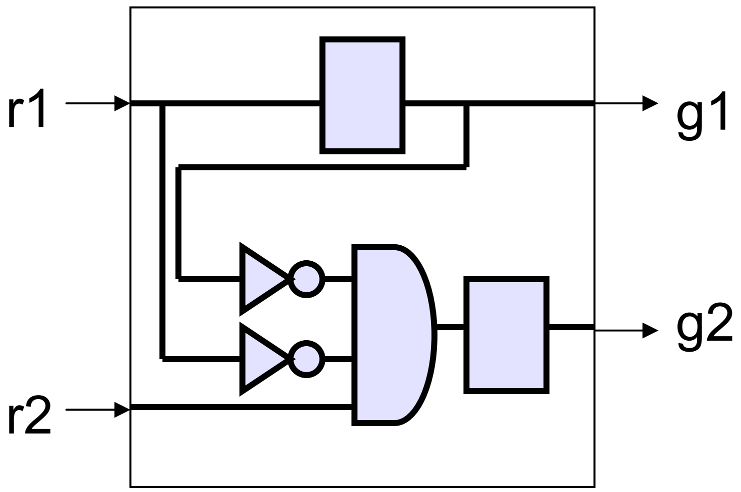

***If r1 is true in a cycle then g1 has to be true for the next two cycles***

<u>Temporal worlds</u>




$$\forall t \, [\, r1^t \Rightarrow g1^{t+1} \wedge g1^{t+2} \,]$$

# Example: *Bound=2*




Is there a witness of length=2?

**Clauses from Transition Relation:**

$C_1^1$: $r2^0 \wedge \neg r1^0 \wedge \neg g1^0 \Rightarrow g2^1$

$C_2^1$: $r1^0 \Rightarrow g1^1$

**Clauses from Initial State:**

I: $g2^0 \wedge \neg g1^0$

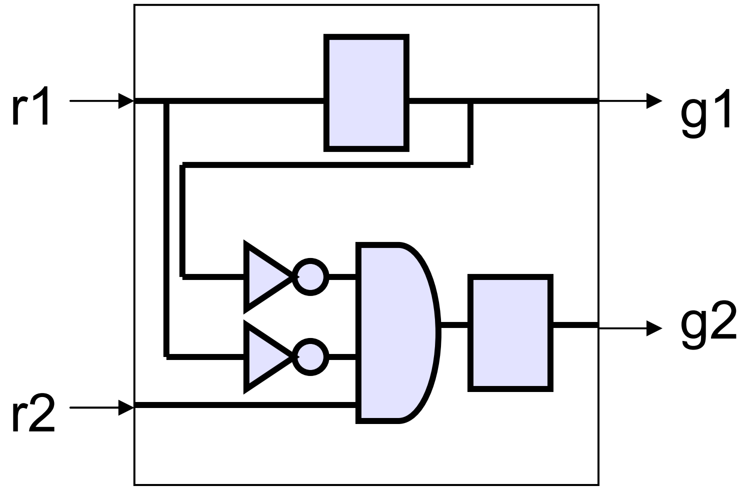
**Clauses from Property:** $F( r1 \wedge (\neg Xg1 \vee \neg XXg1) )$

$Z^1$: $r1^0 \wedge \neg g1^1$

**SAT Check:** Is $Z^1 \wedge I \wedge C_1^1 \wedge C_2^1$ satisfiable?

Answer: No, since $Z^1$ conflicts with $C_2^1$

# Example: *Bound=3*




**Is there a witness of length=3?**

**Clauses from Transition Relation:**

**$C_1^1$, $C_2^1$: from previous iteration**

**$C_1^2$: $r2^1 \wedge \neg r1^1 \wedge \neg g1^1 \Rightarrow g2^2$**

**$C_2^2$: $r1^1 \Rightarrow g1^2$**

**Clauses from Initial State:**

**I: $g2^0 \wedge \neg g1^0$**

**Clauses from Property: $F( r1 \wedge (\neg Xg1 \vee \neg XXg1) )$**

**$Z^2$: $(r1^0 \wedge ( \neg g1^1 \vee \neg g1^2 )) \vee (r1^1 \wedge \neg g_1{}^2)$**

**SAT Check: Is $Z^2 \wedge I \wedge C_1^1 \wedge C_2^1 \wedge C_1^2 \wedge C_2^2$ satisfiable?**

**Yes: Witness: $r1^0 = 1$, $r1^1 = 0$, $g1^1 = 1$, $g1^2 = 0$, rest are don't cares**

**Conclusion: We have found a bug!!**

# Formal Methodology

- Bound on path length k
- Clauses describing the system M :
  - Initial state : $I(s_0)$
  - Unrolled transition relation : $\bigwedge_{i=0..k-1} \rho(s_i , s_{i+1})$
- Loop clause $loop_k = \bigvee_{i=0..k} \rho(s_k , s_i)$
- $[f]_{i,k}$ means that temporal property f is true at state $s_i$ .
- For the property f to hold on the system $M \wedge [f]_{i,k}$ must be satisfiable.

# Translation of LTL to SAT

$[\ X\ f\ ]_{i,k} = (i < k\ ) \wedge [\ f\ ]_{i+1,k}$

$[\ F\ f\ ]_{ik} = \bigvee_{j=i..k} [\ f\ ]_{j,k}$

$[\ G\ f\ ]_{i,k} = \bigwedge_{j=i..k} [\ f\ ]_{j,k} \wedge \text{loop}_k$

$[\ f\ U\ g\ ]_{i,k} = \bigvee_{j=i..k}(\ [\ g\ ]_{j,k} \wedge \bigwedge_{n=i..j-1}[\ f\ ]_{n,k})$

# Advantages

- Able to handle larger state spaces  as compared to BDD’s.
- Takes advantage of several decades of research on efficient SAT solvers.
- The witness/counterexample produced are usually of minimum possible length, making them easier to understand and analyze.

# Limitations of BMC

- **Sound but not complete**
  - **Works for a bounded depth**
  - **In order to have a complete procedure, we need to run it at least up to the diameter (unknown) of the transition system**

- **For larger depths the number of clauses can grow rapidly, thereby raising capacity issues**

- **Nevertheless, SAT-based FPV tools can handle much larger designs as compared to BDD-based tools**