Variants of Turing Machine

Alternative Definitions

- There are many **alternative definitions** of Turing machine.
- Every **reasonable alternative** has **identical computing power** i.e. if some language is **acceptable** but **not decidable** in one version, then it remains so in all its variants etc.
- But the **number of moves** required in **one varient** may be **order of magnitude** different in **another varient**.

A Few Important Variants

- Multitape Turing machine.
- Nondeterministic Turing machine.
- Alternating Turing machine.

Multitape Turing Machine

There are **more than one tapes** and each tape has a **read/write head** that can **move** to left or to right **independently**. The transition function for a n**-tape** machine is defined as follows.

$$\delta : \mathbf{Q} \times \mathbf{\Gamma^n} \longrightarrow \mathbf{Q} \times \mathbf{\Gamma^n} \times \{\mathbf{L}, \mathbf{S}, \mathbf{R}\}^\mathbf{n}.$$

$$\delta(\mathbf{q}, \gamma_\mathbf{1}, \cdots, \gamma_\mathbf{n}) = (\mathbf{p}, \gamma'_\mathbf{1}, \cdots, \gamma'_\mathbf{n}, \mathbf{M_1}, \cdots, \mathbf{M_n}),$$

where $\mathbf{M_i} \in \{\mathbf{L}, \mathbf{S}, \mathbf{R}\}$.

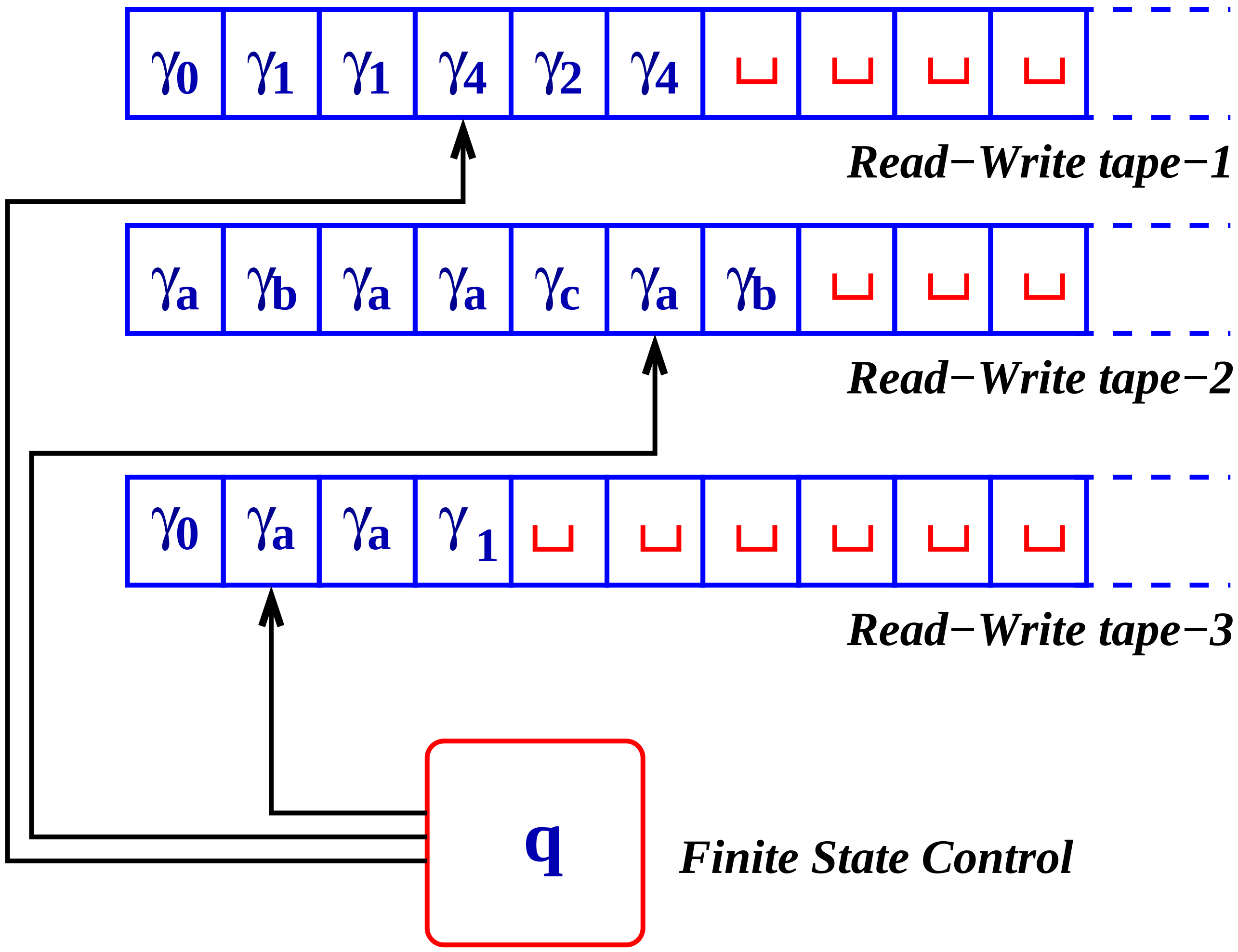


Figure 1: **3-tape TM**

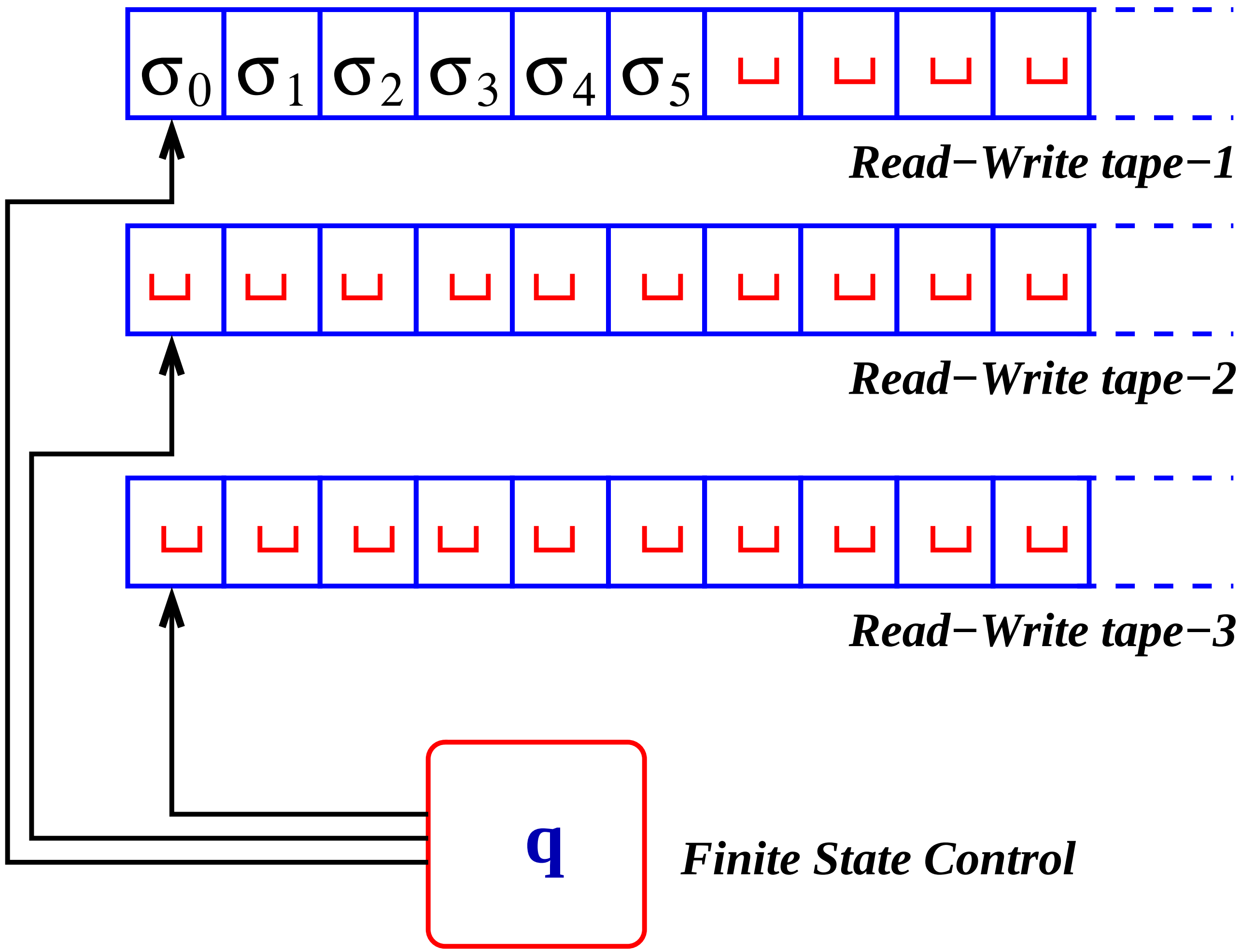


Figure 2: **3-tape TM : Start Configuration**

2-tape TM : An Example

Let the language **L** be our old **palindrome**.

$$\mathbf{L = \{x \in \{a, b\}^* : x = x^{reverse}\}}$$

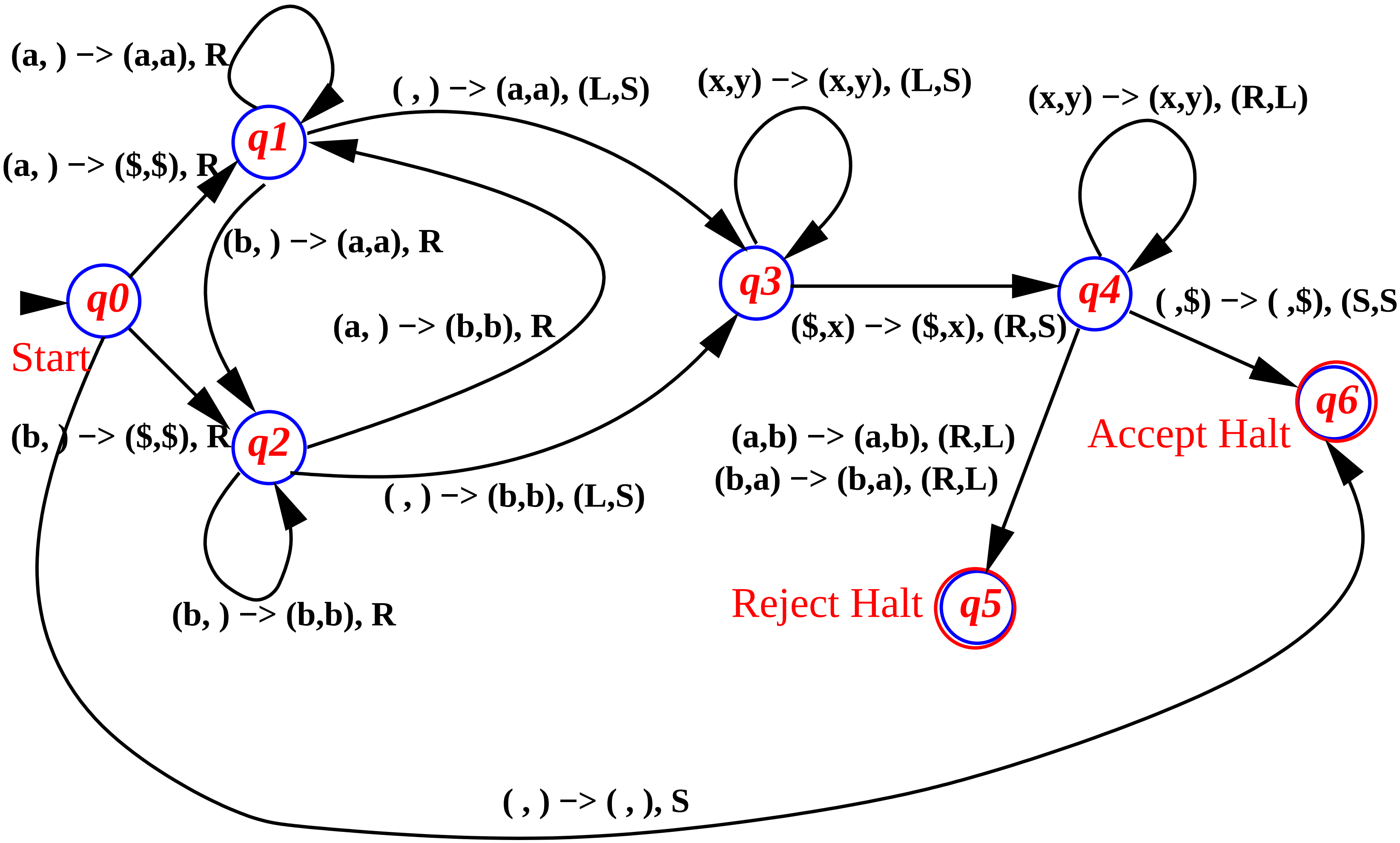


Figure 3: **2-tape TM : An Example**

Simuation on Single Tape TM

- A **multitape Turing machine** can be simulated on a **single tape Turing machine**.
- Let $\mathbf{M} = (\mathbf{Q}, \mathbf{\Sigma}, \mathbf{\Gamma}, \delta, \mathbf{q_0}, \mathbf{q_a}, \mathbf{q_r})$ be an n**-tape** Turing machine.
- The **content of** n**-tapes** are to be written on the single tape. We use '$\#$' as a **delimiter** of the contents of each tape. We assume that $\# \notin \mathbf{\Gamma}$.

Simuation on Single Tape TM

- We remember the **head positions** on each tape by introducing a **new symbol** in the **tape alphabet** of the single tape machine. If the **read/write head** of the ith tape is scanning a symbol γ, we write $\dot{\gamma}$ in the corresponding position of the single tape machine.
- The **tape alphabet** of the single tape machine contains $\mathbf{\Gamma}$, $\dot{\mathbf{\Gamma}} = \{\dot{\gamma} : \gamma \in \mathbf{\Gamma}\}$ and '#'.

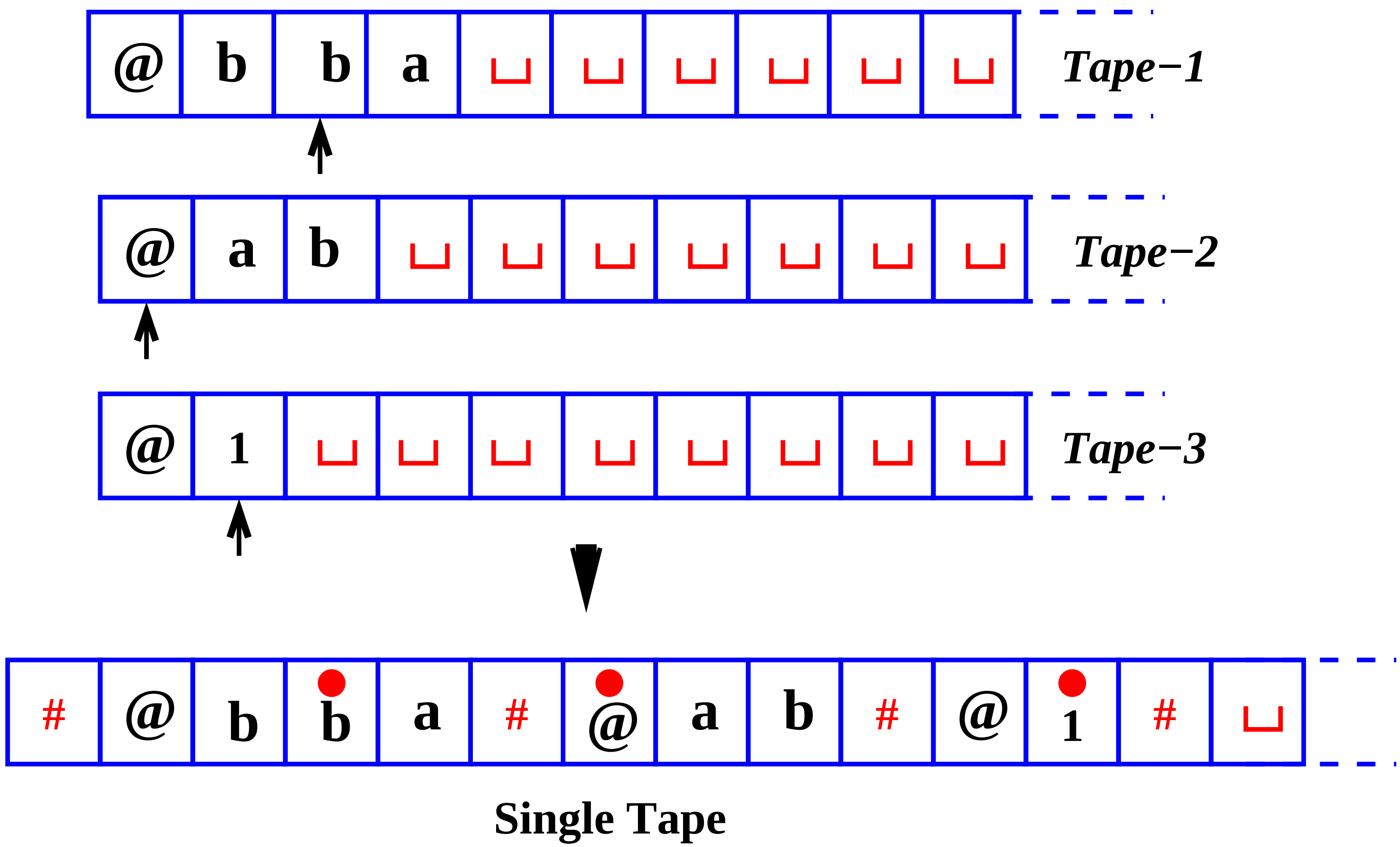


Figure 4: **3-tape to 1-tape**

Simuation on Single Tape TM

- The simulation starts from the **start configuration**

$$\# \dot{\sigma}_{\mathbf{0}}\ \sigma_{\mathbf{1}} \cdots \sigma_{\mathbf{n-1}} \# \dot{\sqcup} \# \cdots \# \dot{\sqcup} \#$$

- To go from one configuration to the next, the **single tape Turing machine** first **scans** the **whole tape** to detect the **head positions** on n different tapes (this number is fixed) of the **n-tape** machine.
- In the second pass it simulates the action of different heads. In principle it can be done but will be very cumbersome.

Simuation on Single Tape TM

- If the ***n*-tape** machine tries to write a **nonblank** symbol in a **new blank cell**, the content of the single tape

Nondeterministic Turing Machine

- Definition of a **Nondeterministic Turing machine (NTM)** - the transition function δ is as defined.

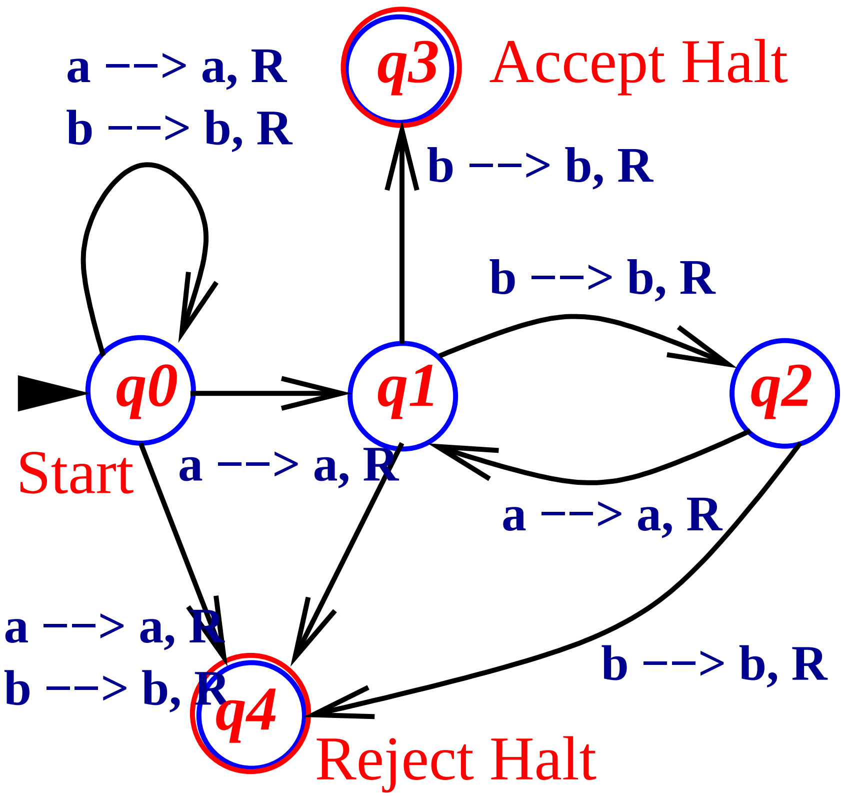
$$\delta : \mathbf{Q} \times \mathbf{\Gamma} \longrightarrow \mathcal{P}(\mathbf{Q} \times \mathbf{\Gamma} \times \{\mathbf{L}, \mathbf{R}\}).$$

- There may be **different computation** of an NTM on an input.
- An input is **accepted** if **there is a computation** that reaches to an **accept halt** state.

Nondeterministic Turing Machine : An Example

Design a **nondeterministic Turing machine** for the following language.

$$\mathbf{L = \{x \in \{a, b\}^* \ : \ x = yz, \ z = (ab)^n, n > 0\}.}$$



a --> a, R
b --> b, R
q3
Accept Halt
b --> b, R
b --> b, R
q0
q1
q2
Start
a --> a, R
a --> a, R
a --> a, R
b --> b, R
q4
b --> b, R
Reject Halt

Figure 5: **NTM**

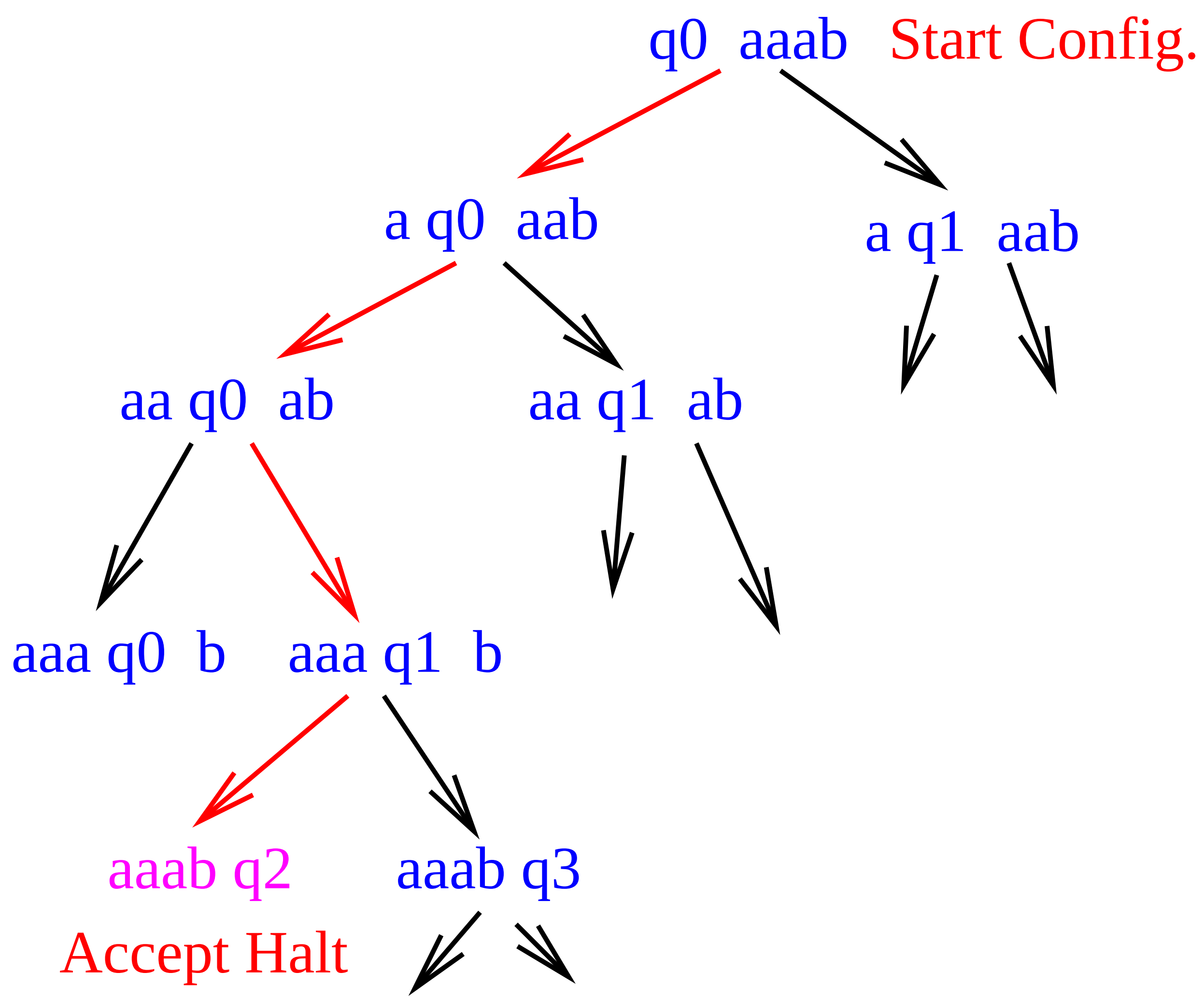


Figure 6: **Computation on NTM**

NTM and DTM are Equivalent

- A **nondeterministic Turing machine** can be simulated on a **3-tape deterministic Turing machine**.
- A **3-tape deterministic Turing machine** can be simulated on a **single tape Turing machine**.
- Hence a **nondeterministic Turing machine** can be simulated on a **single tape Turing machine**.

Simulation of NTM on a 3-Tape DTM

- The **3-tape** Turing machine will simulate the **nondeterministic computation** of the NTM in a **breadth-first order**.
- The **tape-I** always contains the input.
- The **tape-II** actually does the simulation starting from the **start configuration** upto the coniguration of a particular depth.

Simulation of NTM on a 3-Tape DTM

- If the **maximum degree of nondeterminism** for a NTM is **k**, then the computation of an NTM starting from the **start configuration** to a **particular configuration** can be **encoded** as a string over an **alphabet** of size **k**.

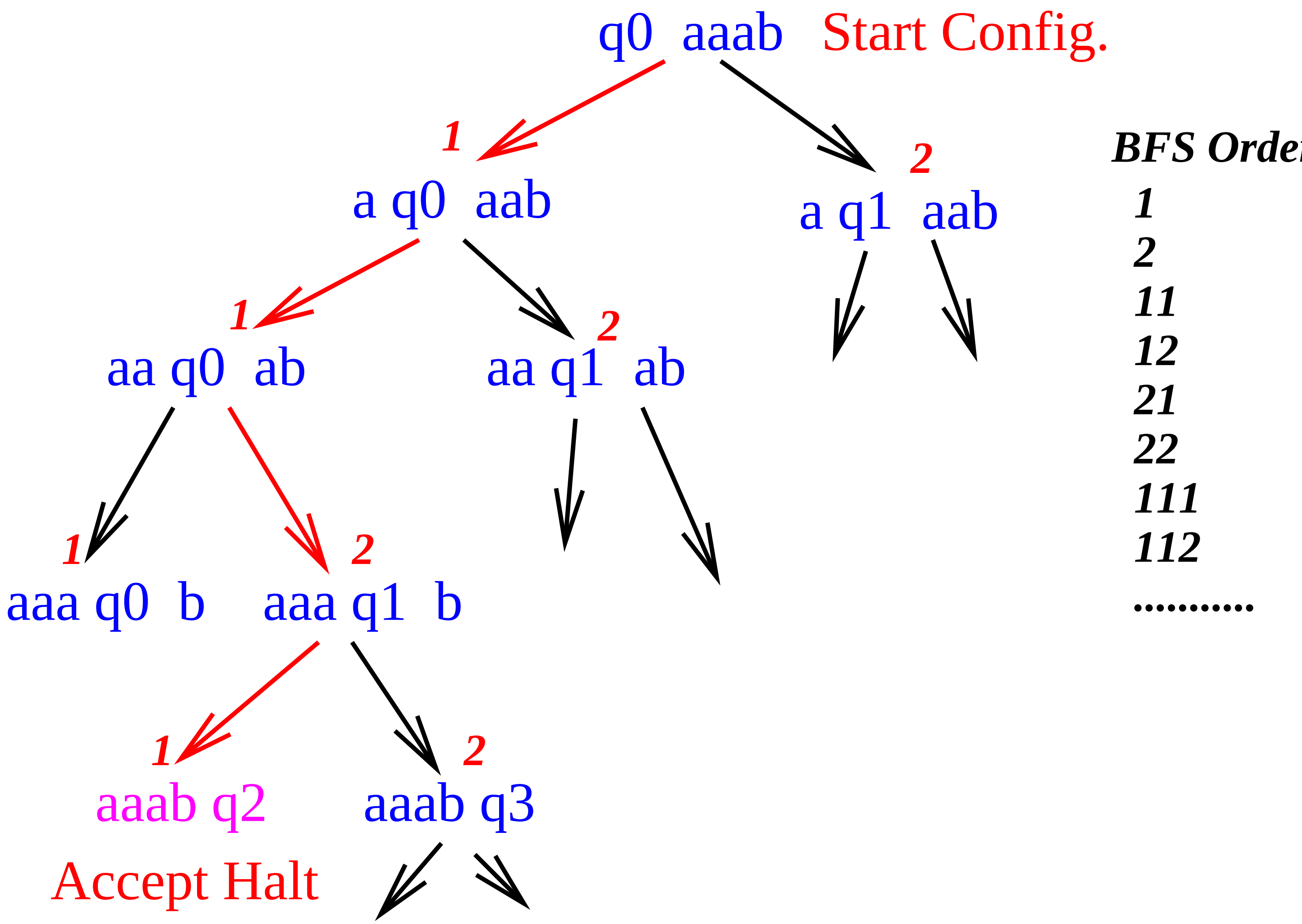


Figure 7: **Encoding of Computation**

Simulation of NTM on a 3-Tape DTM

- The **tape-III** is used to keep track of the computation of the NTM.
- The **encoding string** for different paths of computation is enumerated in **BFS order** and the simulation starting from the **start configuration** is done upto that step.
- The **3-tape** machine **halts** if in the process it reaches an **accept halt** configuration of the NTM.