Mächtigkeit/Churchsche These

Gliederung

- Warum verwenden wir das Modell Turingmaschine?
- Wie funktioniert die Turingmaschine?
- Churchsche These
- Gegenseitige Simulation (TM -> Programmiersprachen & RM -> TM)

Modell Turingmaschine

- Erlaubt mathematisches Formalisieren von Algorithmus und Berechenbarkeit

- Kann mit mathematischen Methoden untersucht werden

- Die Berechenbarkeit eines Algorithmus nachweisbar über die
  - Turingmaschine oder
  - Programm in gleichmächtiger Programmiersprache

- Wird nicht als produktives Rechnermodell eingesetzt

- Berechenbarkeitsklasse:
  Turing-berechenbare Algorithmen/Funktionen

Die deterministische 1-Band-Turingmaschine

Formal: $M = (Q, \Sigma, \Gamma, \delta, q_0, b, F)$

Q : Nicht-leere Menge von Zuständen

Γ : Band-Alphabet

$\Sigma \subset \Gamma$: Eingabe-Alphabet

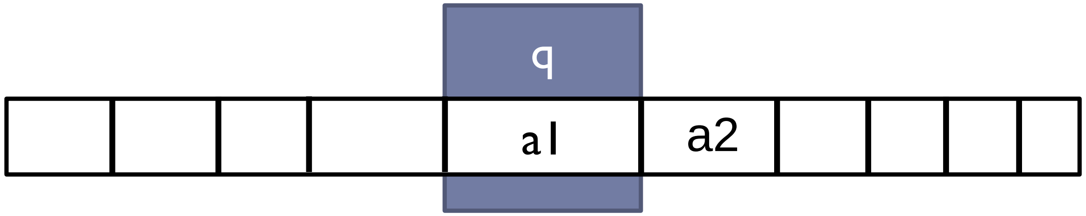
δ : Übergangsfunktion $(Q \backslash \mathrm{F}) \times \Gamma \rightarrow Q \times \Gamma \times \{L, 0, R\}$

$q_0 \in Q$: Startzustand

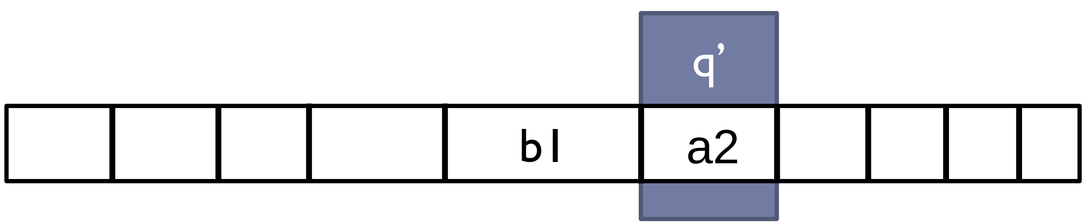
$b \in \Gamma$: Symbol für Leeres Feld

$F \subset Q$: Menge der akzeptierenden Zustände / Endzustände

Ein Schritt einer 1-Band-Turingmaschine



$$\delta(q, a1) = (q', b1, R)$$



Churchsche These

„Die Klasse der turing-berechenbaren Funktionen stimmt mit der Klasse der intuitiv berechenbaren Funktionen überein.“

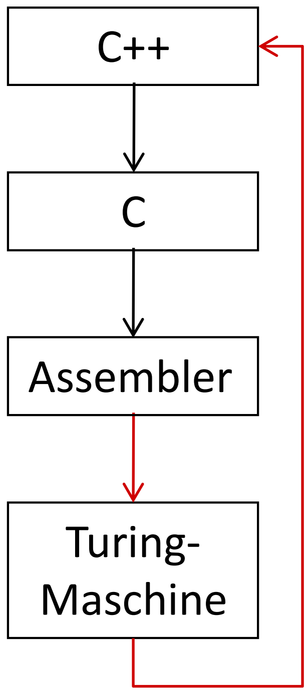
- Kein umfassenderer Berechenbarkeitsbegriff als die Turing-Berechenbarkeit, der nicht außerhalb unserer Intuition liegt
- Kann nicht bewiesen werden, höchstens falsifiziert (unwahrscheinlich)

$\Rightarrow$ Wenn das so wäre, dann sind folgende Fragen interessant:

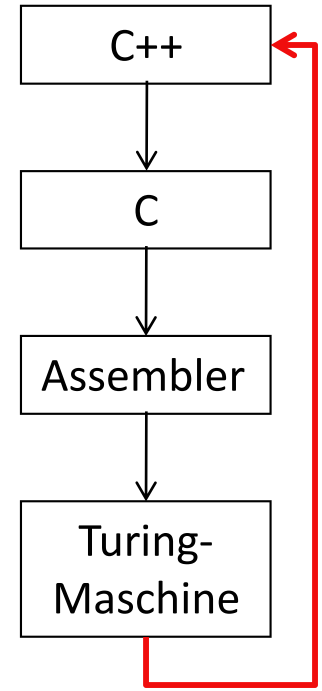
- Kann die Turingmaschine “mehr” als moderne Programmiersprachen?
- Können moderne Programmiersprachen “mehr” als die Turingmaschine?

Voller Funktionsumfang = alle berechenbare Funktionen??

- Idee
  - Gegenseitig Simulierbar
- Übersetzer
  - C++ → C
  - C → Assembler
- Theoriebeweis
  - Fleißarbeit
    - Assembler → Turingmaschine
  - Einfach
    - Turingmaschine → C++



Simulation einer Turing-Maschine in C++



- Modellierung
  - Band: Array Band [-n,n], Zeiger Pos
  - Zustand: Sprungadresse der Zeile

- Schritt einer Turingmaschine mit GoTos
  - Turing Zeile mit Alphabet = {B,0,1}

    q: q´ X´ R´ q´´ X´´ R´´ q´´´ X´´´ R´´´
  - C++ Zeile:

    q: case Band(Pos)

    B: Band(Pos)=X´, Pos := Pos+R´, goto q´

    0: Band(Pos)=X´´, Pos := Pos+R´´, goto q´´

    1: Band(Pos)=X´´´, Pos := Pos+R´´´, goto q´´´

Simulation einer Turing-Maschine in C++

- **Modellierung**
  - Band: Array Band [-n,n], Zeiger Pos
  - Zustand: Sprungadresse der Zeile

- **Schritt einer Turingmaschine mit while**
  - Turing Zeile mit Alphabet = {B,0,1}

    q: q´ X´ R´   q´´ X´´ R´´   q´´´ X´´´ R´´´
  - While Zustand ≠ Finalzustand do

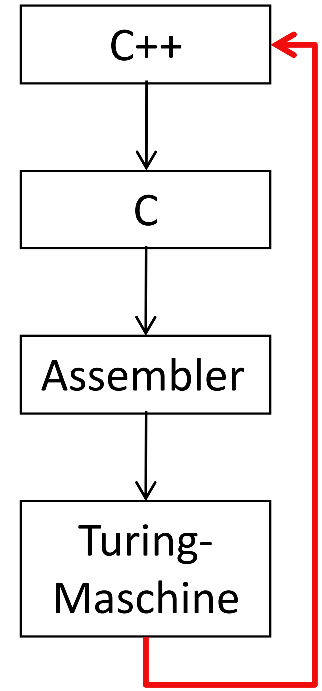
    Case Zustand of

    q: case Band(Pos)

    B: Band(Pos)=X´, Pos := Pos+R´, Zustand := q´

    0: Band(Pos)=X´´, Pos := Pos+R´´, Zustand := q´´

    1: Band(Pos)=X´´´, Pos := Pos+R´´´, Zustand := q´´´

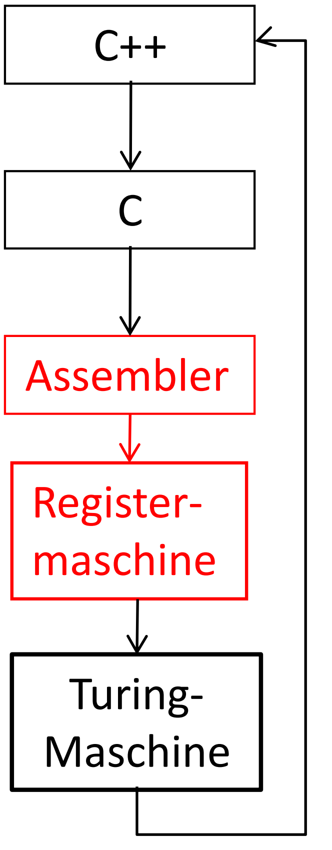


Simulation eines Assemblers mit einer Turing Maschine

- Assembler haben verschiedene Befehlssätze und Syntax
- Modell für unsere Betrachtung:

  <u>Registermaschine</u>

  - Einfache Grundoperationen
  - Endliche Zahl festgelegter Register
  - Indirekte Adressierung möglich (dadurch auch Zugriff auf mehr Register als festgelegt)



Befehlssatz RM (Auszug)

Befehl	Beschreibung	Aktion
LOAD i	Lädt den Wert aus Register c_i in den Akku	a:= c_i , b:=b+1
CLOAD i	Lädt den Wert i in den Akkumulator	a:=i , b:=b+1
INDLOAD i	Lädt den Wert des Registers, auf das c_i zeigt in den Ak.	a:=$c_{(c_i)}$ b:=b+1
STORE i	Lädt den Inhalt des Akkus in Register c_i	c_i:=a , b:=b+1
INDSTORE i	Lädt den Akku in das Register, auf das c_i zeigt	$c_{(c_i)}$:=a , b:=b+1
ADD i	Addiert c_i zum Akku und speichert Ergebnis im Akku	a:=a+c_i , b:=b+1
CADD i	Addiert i zum Akku und speichert Ergebnis im Akku	a:=a+i, b:=b+1
INDADD i	Addiert Register auf das c_i zeigt zu Akku und speichert Ergebnis im Akku	a:=a+$c_{(c_i)}$, b:=b+1

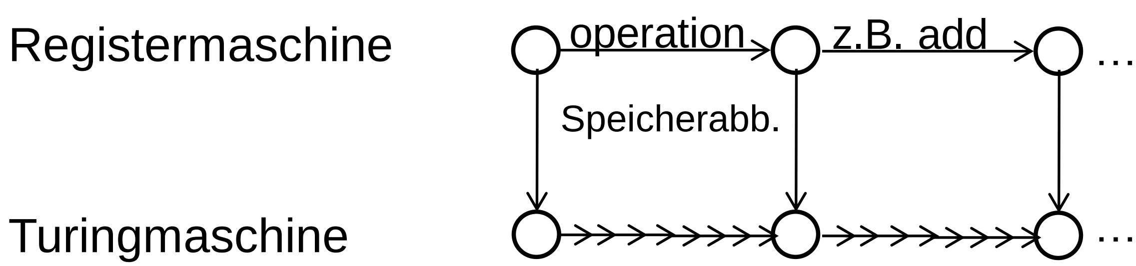
- Diese weiteren Befehle existieren mit ähnlichem Schema
- Subtraktion, Multiplikation, Division, GOTO, IF … GOTO, END

Einfaches Beispiel

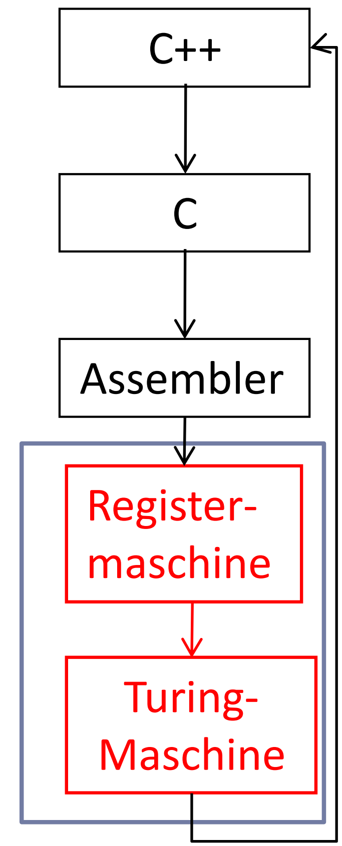
- Programm soll Register c_1 und c_2 addieren und das Ergebnis in Register c_3 speichern.

Programm P		Status der Registermaschine
i	$p_{[b]}$	$K_{RM}(b, R[0], R[1] \ldots R[n])$
0	LOAD 1	$K_{RM}(0, [0], [3][9][0])$
1	ADD 2	$K_{RM}(1, [3], [3][9][0])$
2	STORE 3	$K_{RM}(2, [12], [3][9][0])$
3	END	$K_{RM}(3, [12], [3][9][12])$

- Speicherabbildung der Registermaschine auf das Band der Turingmaschine

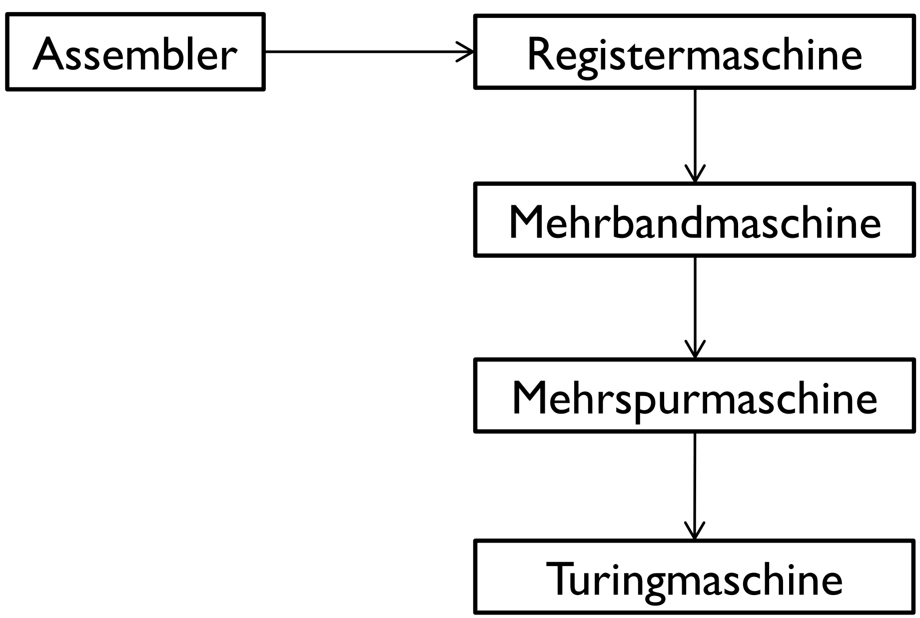


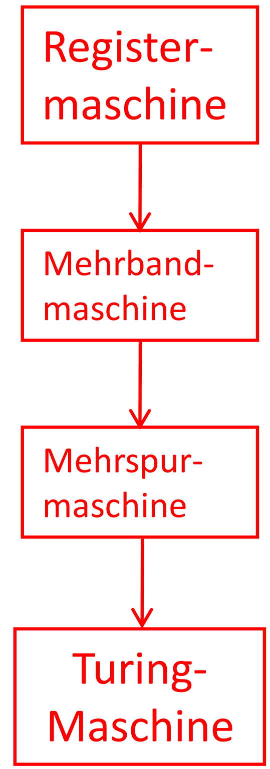
- Unterschiedlicher Aufwand
  - $S_{TM}(n)$ = Speicherverbrauch der TM für Darst. einer Registermaschine mit Eingabe der Länge n
  - $t_{TM}(n)$ = Anzahl Schritte der TM



Simulation eines Assemblers mit einer Turing Maschine

- In einem Schritt nicht übersichtlich
- Schrittweise Simulation



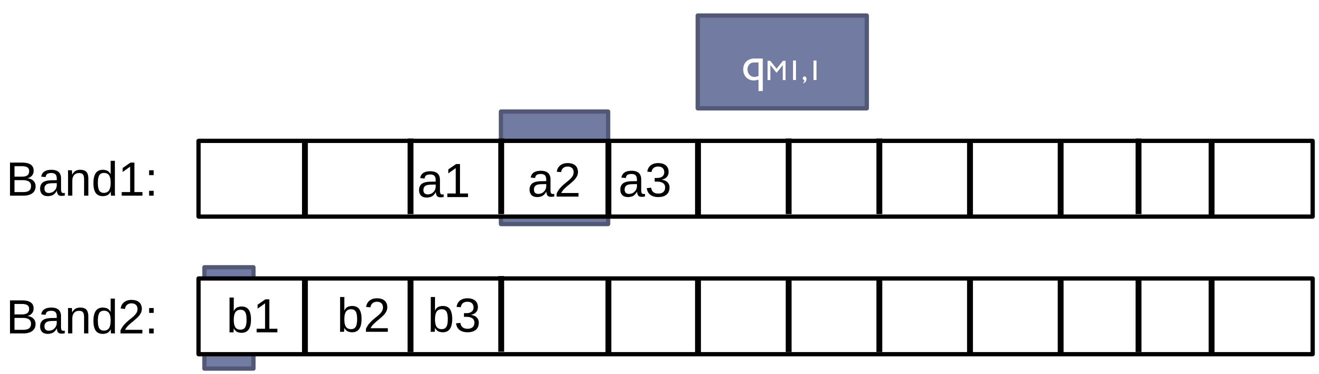


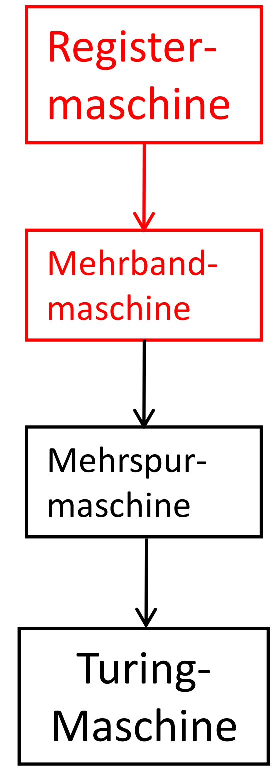
Simulation einer Registermaschine mit einer Mehrbandmaschine

Die k-Band-Turingmaschine:

- k Bänder mit je einem Kopf
- Zustandsübergangsfunktion:

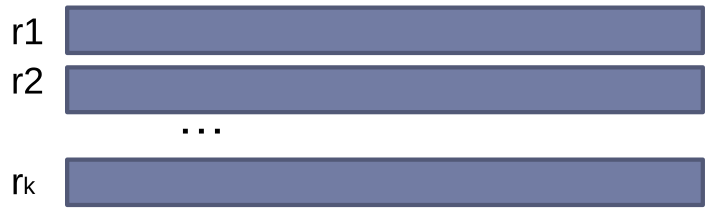
$$\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, N\}^k$$

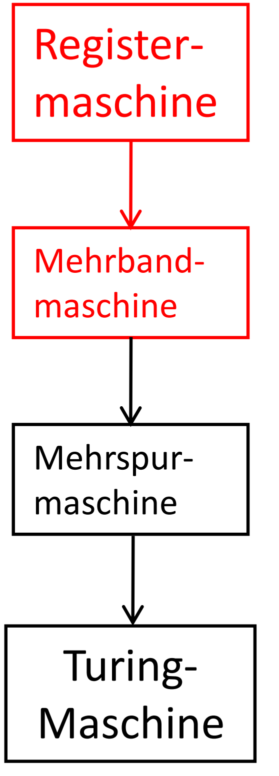




Simulation einer Registermaschine mit einer Mehrbandmaschine

Erster Ansatz: Ein Band pro Register (Anzahl: k)





Funktioniert nicht, da indirekte Adressierung der Registermaschine eine theoretisch unendliche Anzahl von Registern erlaubt.

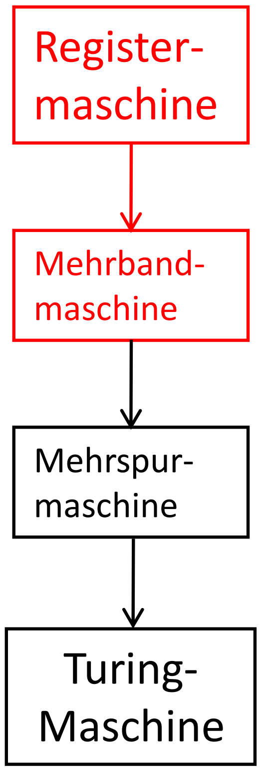
Simulation einer Registermaschine mit einer Mehrbandmaschine

Mehrbandmaschine mit zwei Bändern und p+2 (p = Anzahl Befehle) Unterprogrammen.

Bänder:
1. Band: Akkumulator
2. Band: Speichersimulation

Unterprogramme:
M0 => Initialisierung
M1 bis Mp => für die p Programmzeilen
Mp+1 => Ausgabe des Ergebnisses

Unterprogramme können auf freiem Bereich des Eingabebandes oder auf einem dritten Band ausgeführt werden.



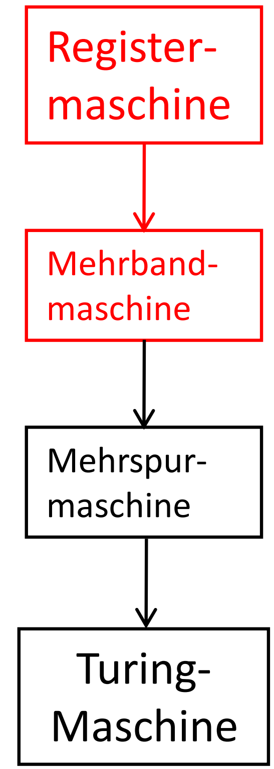
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

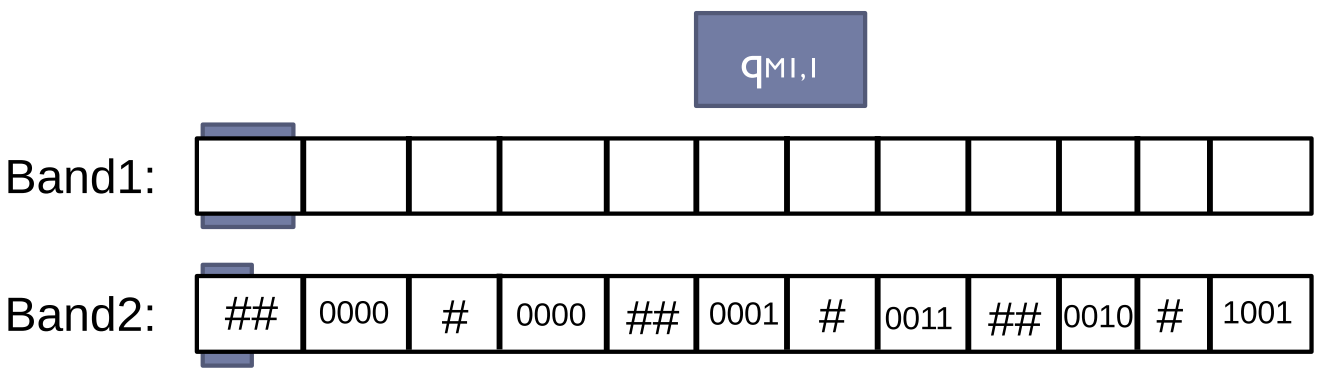
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

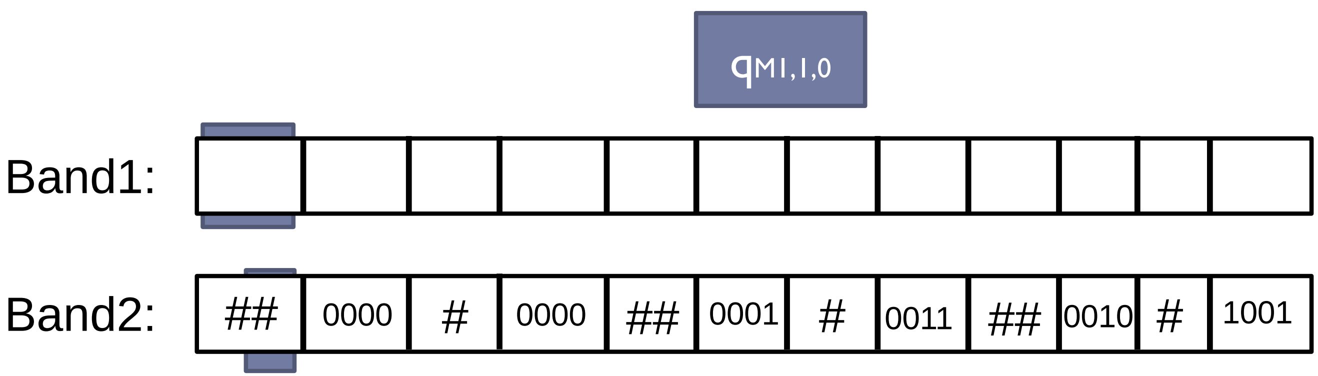
Beispiel: Addition 3 + 9 (0011 + 1001)

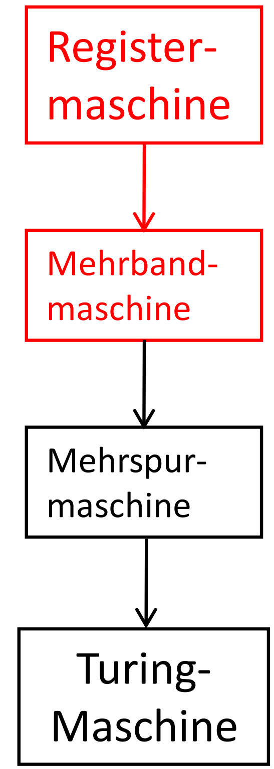
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
- 0: 0000
- 1: 0011
- 2: 1001

Mehrbandmaschine:





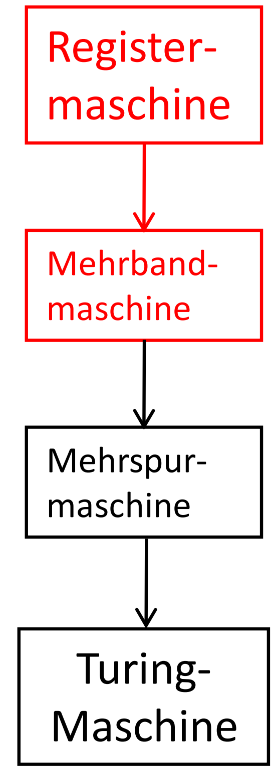
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

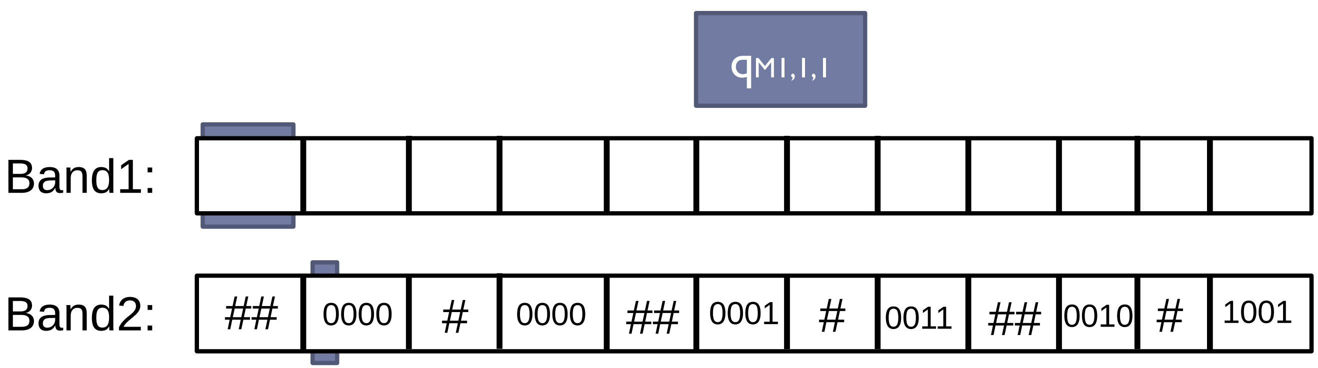
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



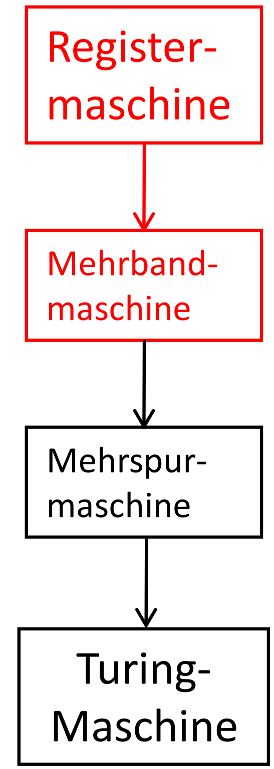
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

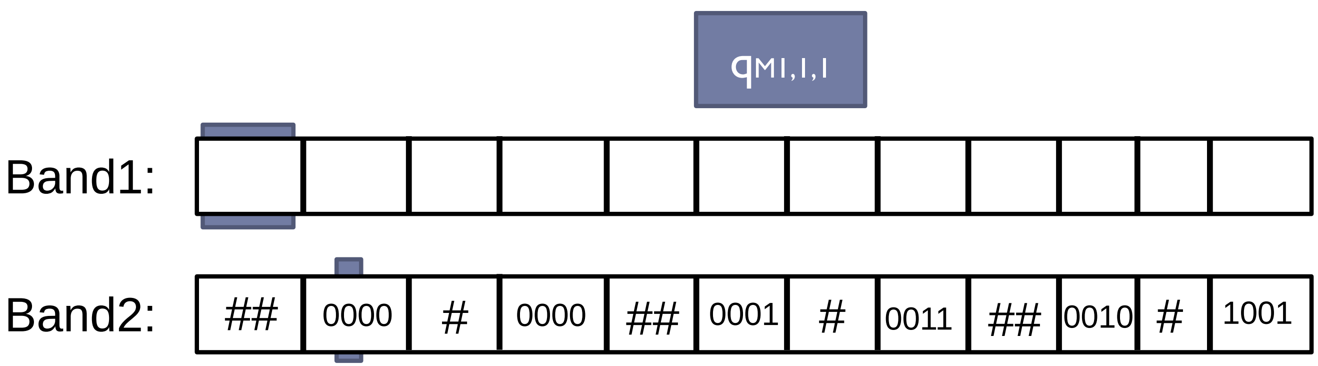
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



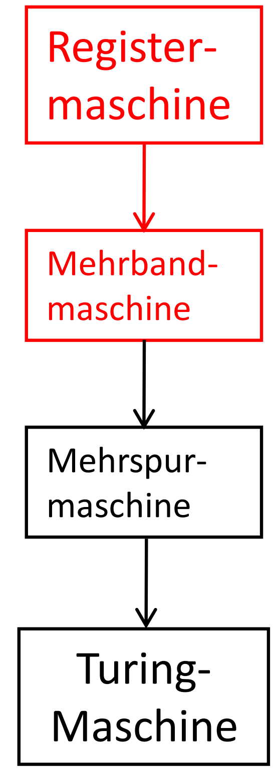
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

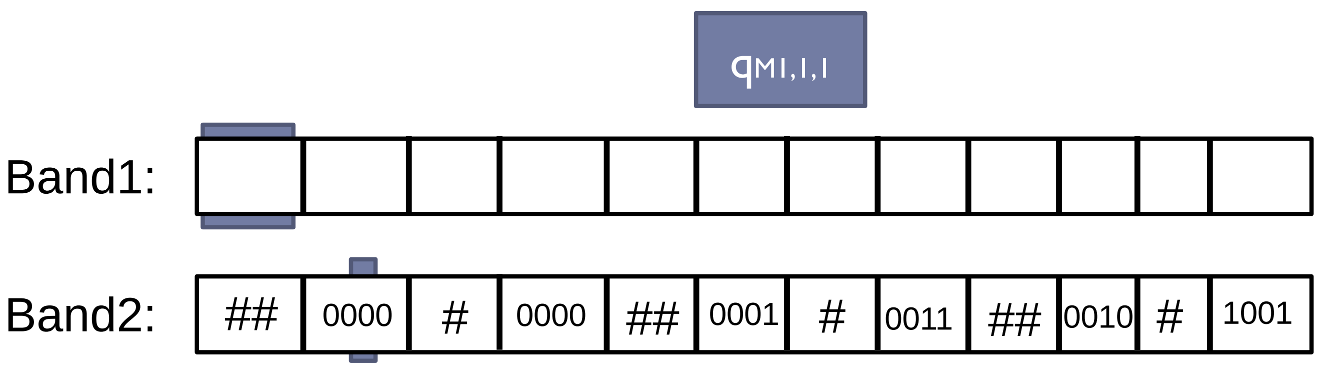
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

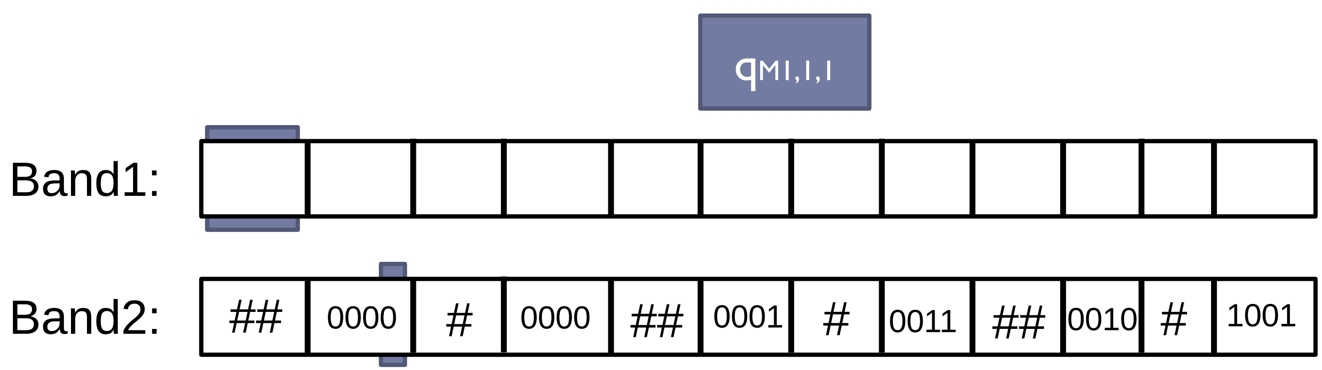
Beispiel: Addition 3 + 9 (0011 + 1001)

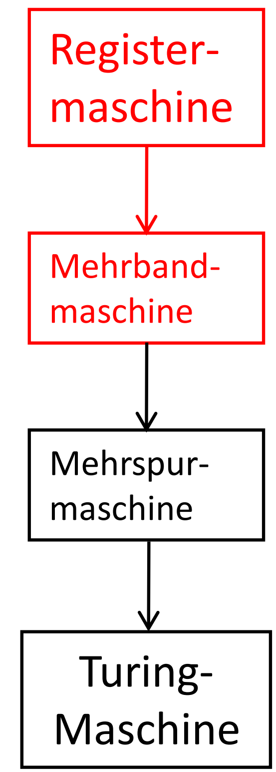
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





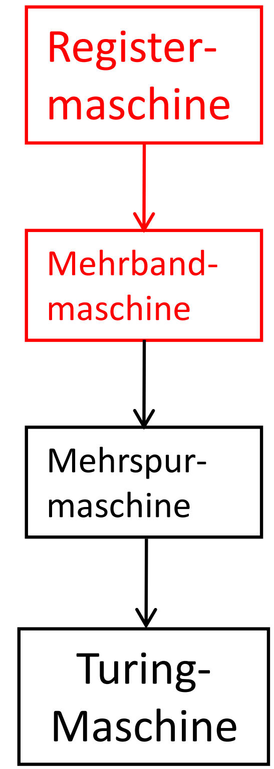
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

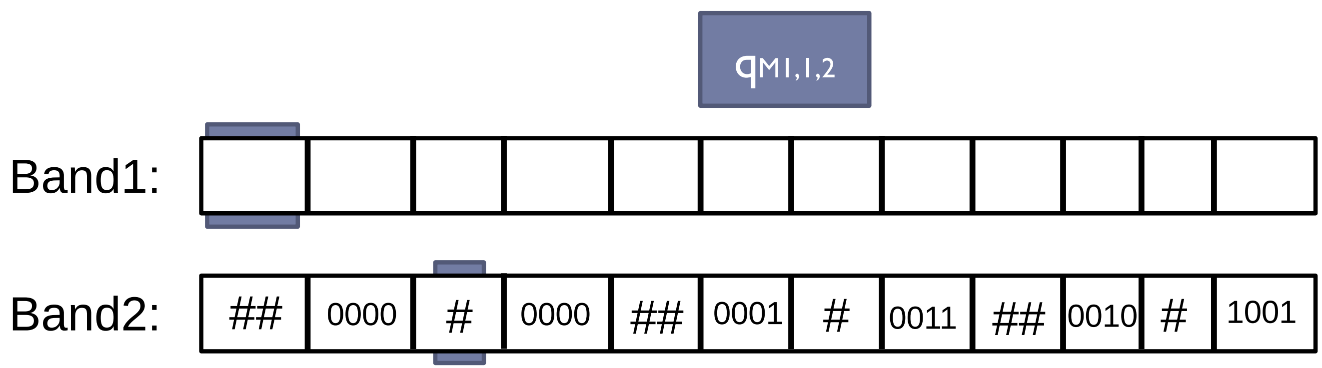
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



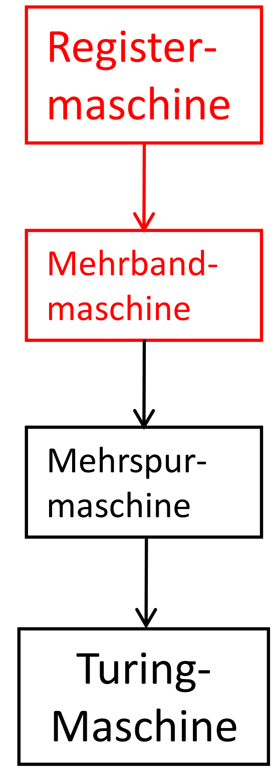
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

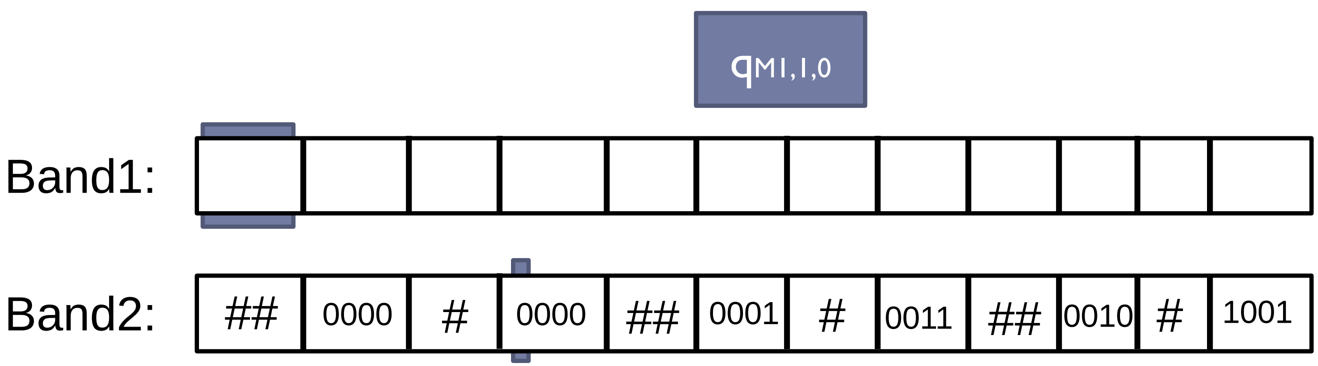
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

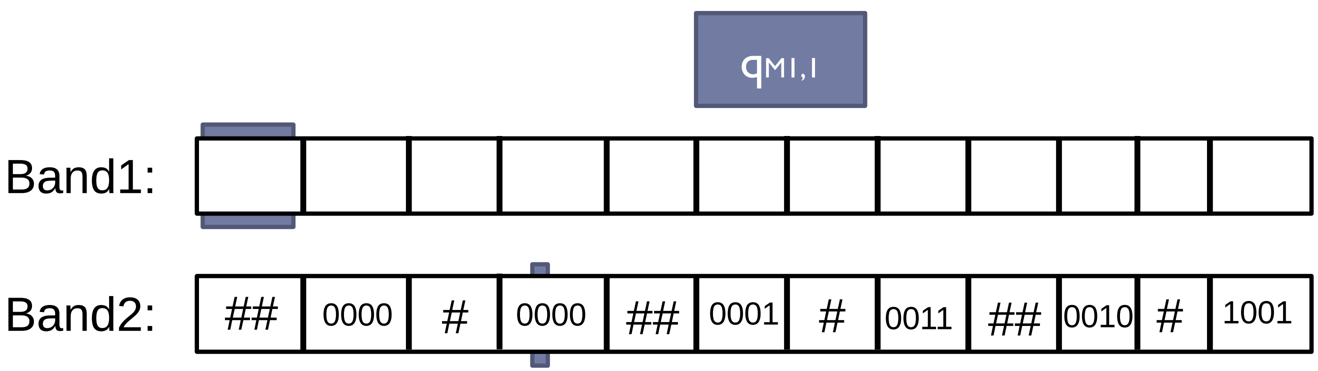
Beispiel: Addition 3 + 9 (0011 + 1001)

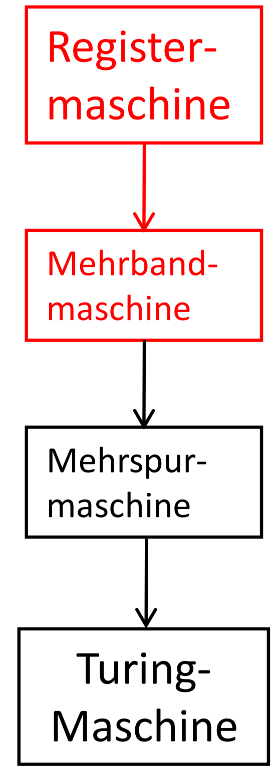
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

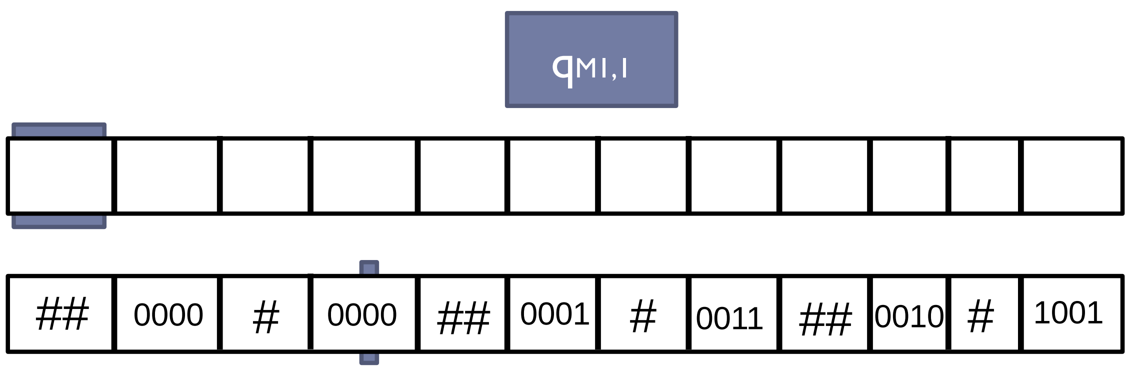
Registermaschine:

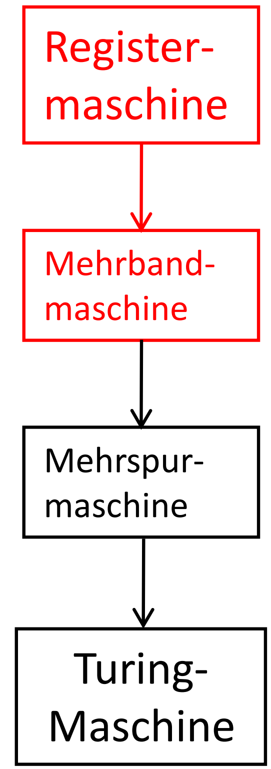
Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:

Band1:
Band2:





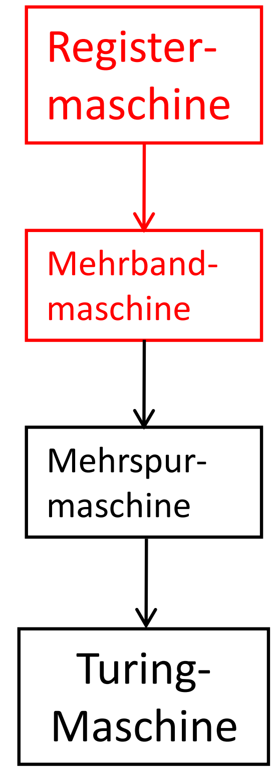
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

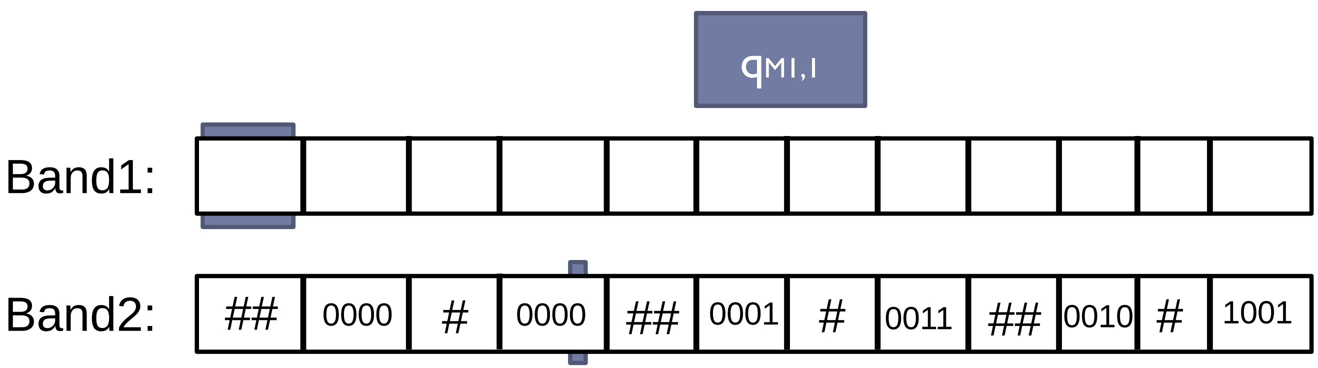
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



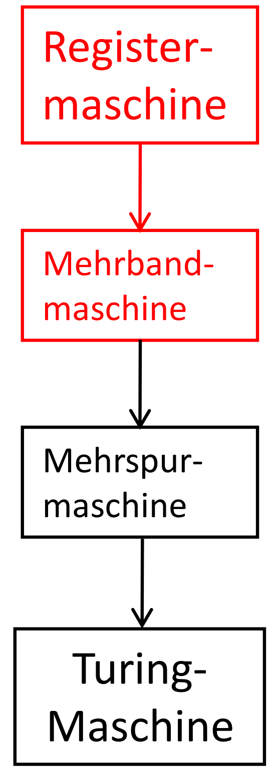
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

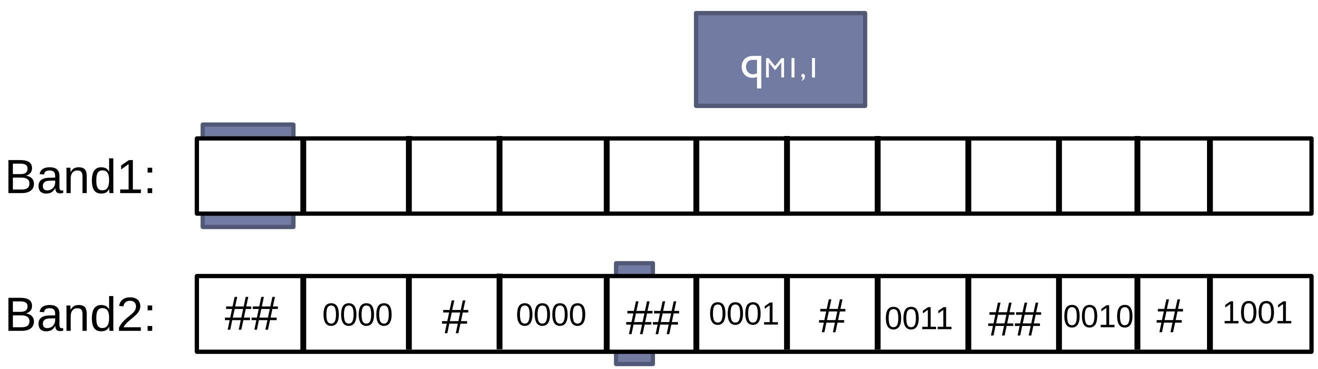
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



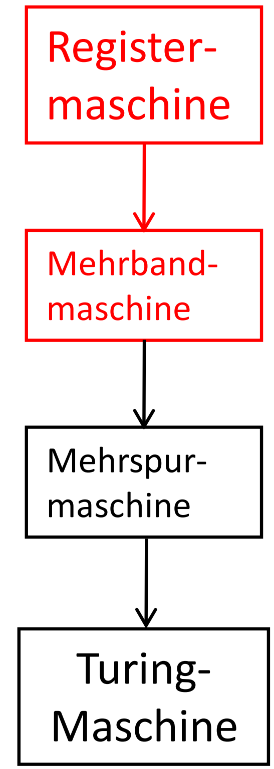
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

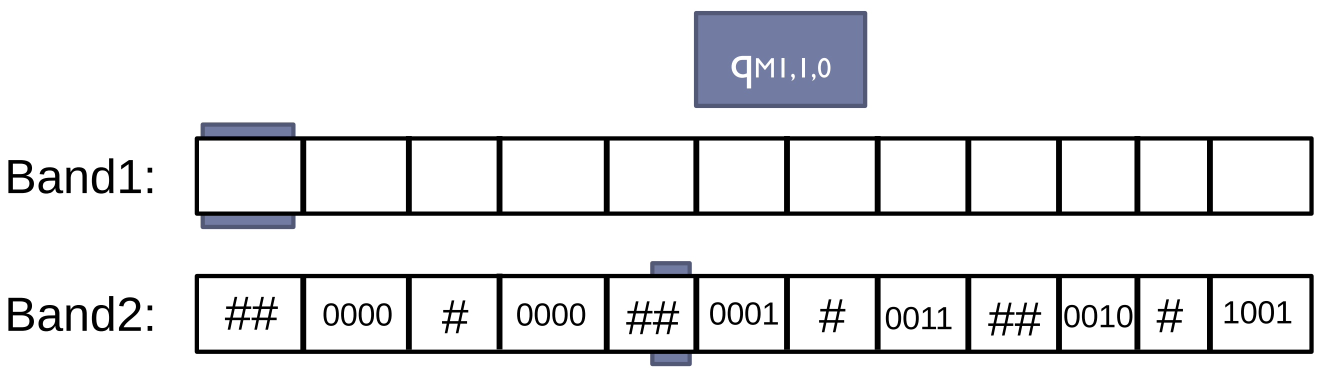
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



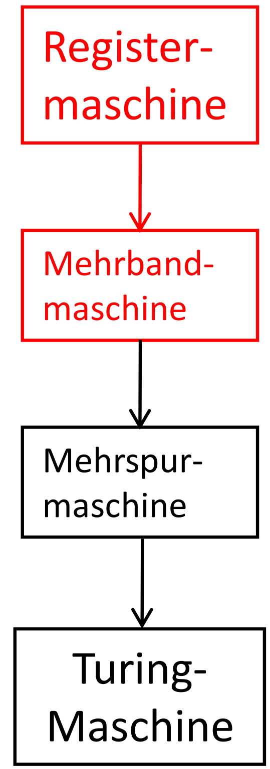
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

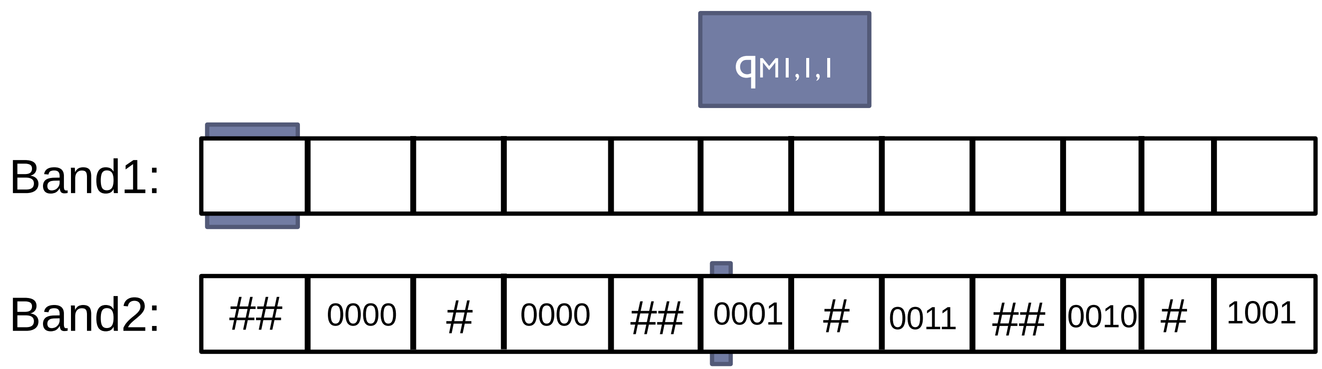
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



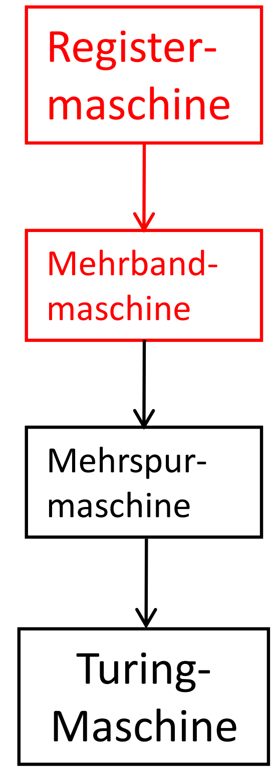
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

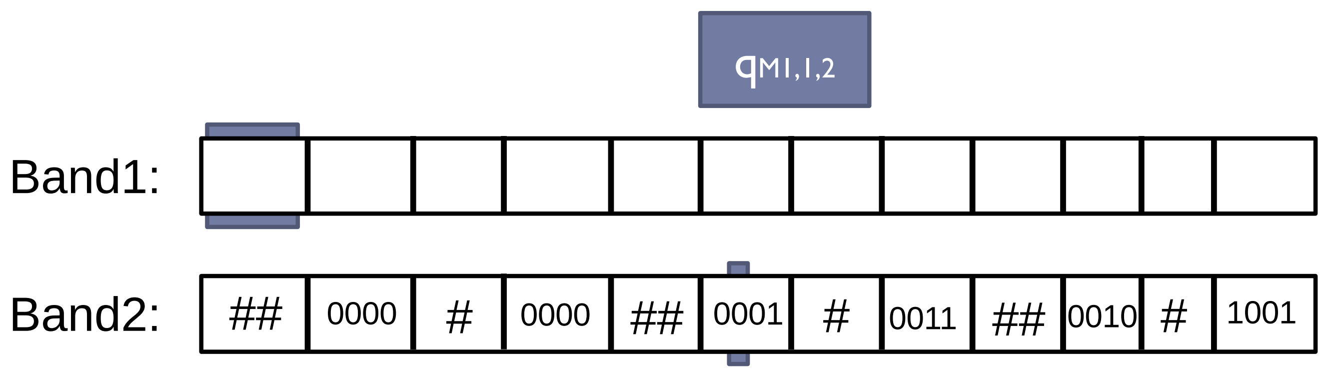
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



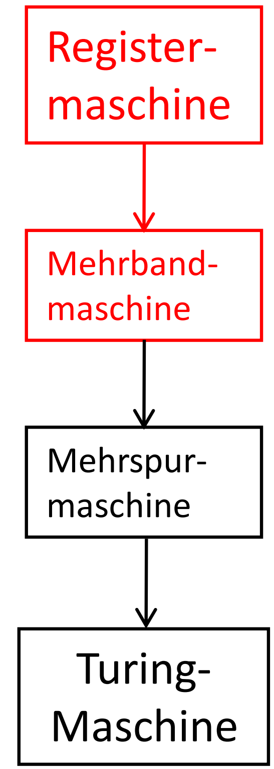
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

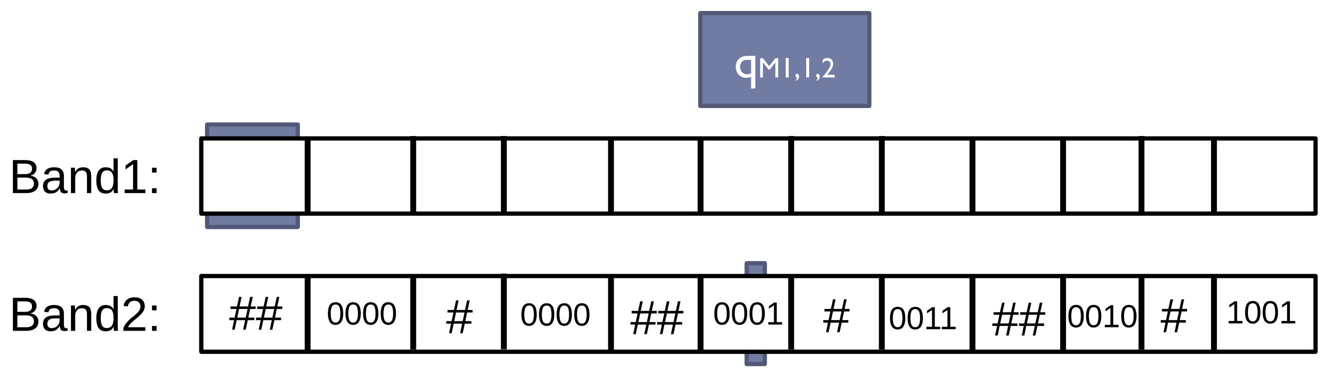
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



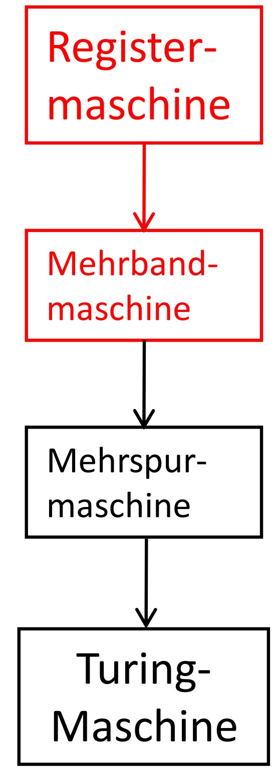
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

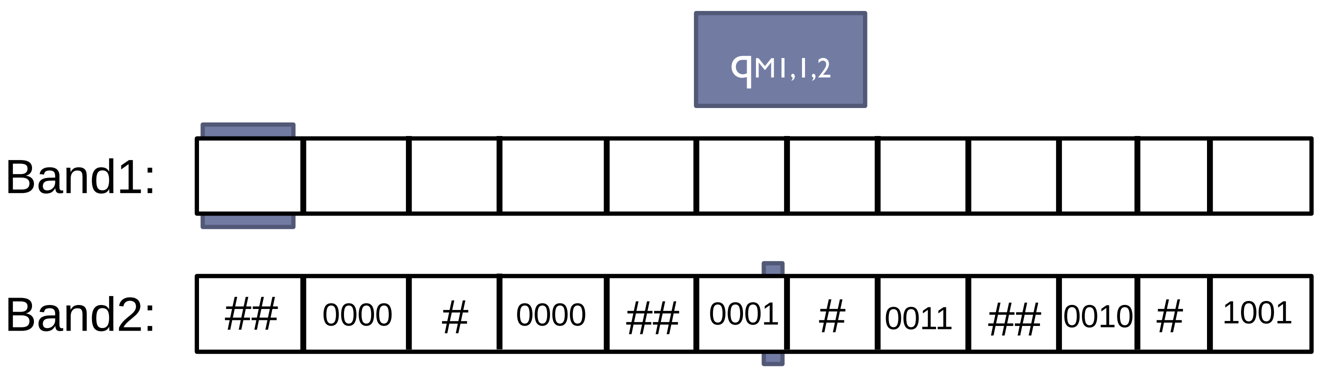
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



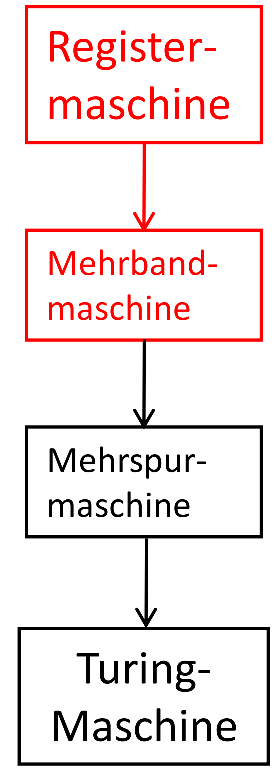
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

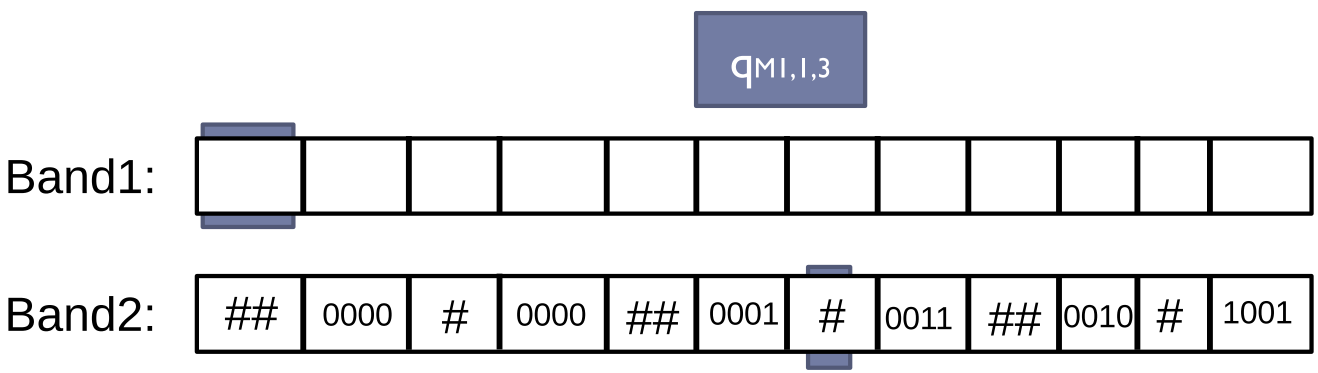
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

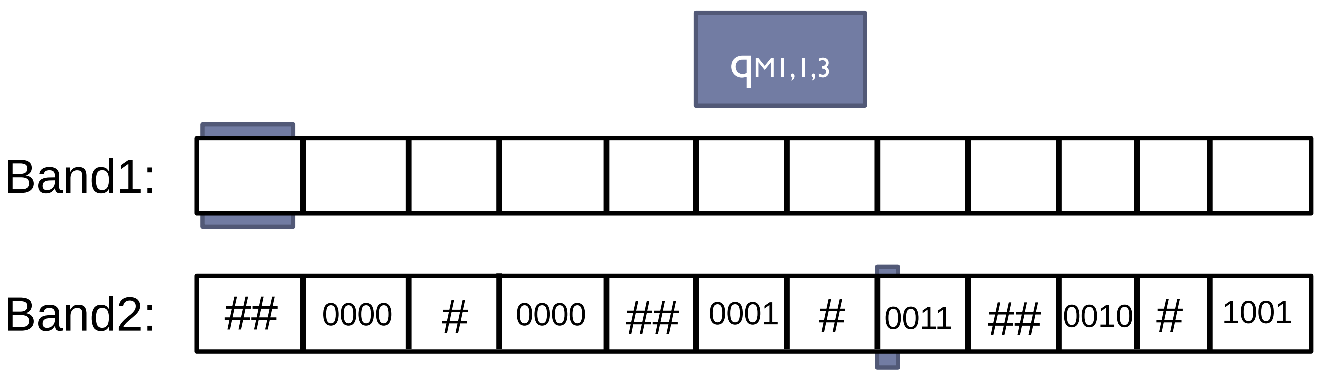
Beispiel: Addition 3 + 9 (0011 + 1001)

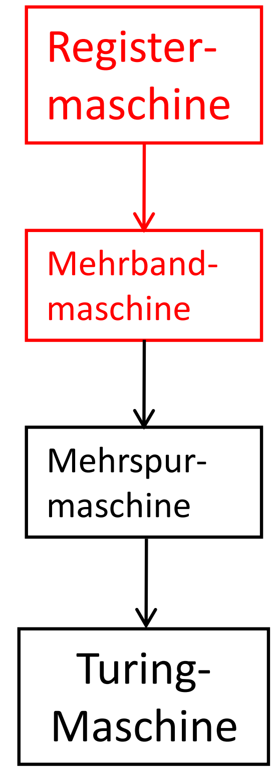
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





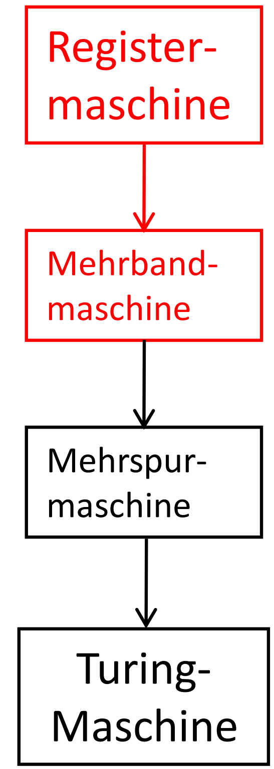
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

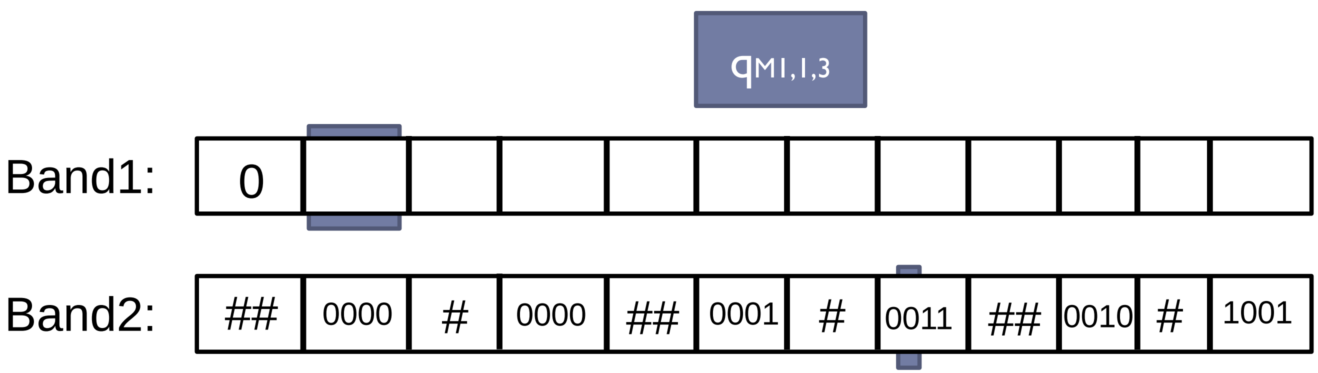
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

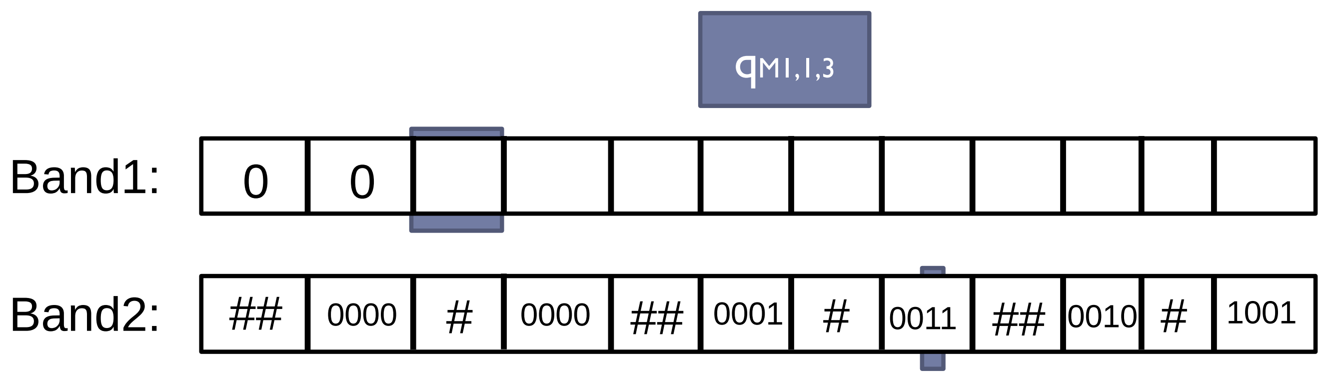
Beispiel: Addition 3 + 9 (0011 + 1001)

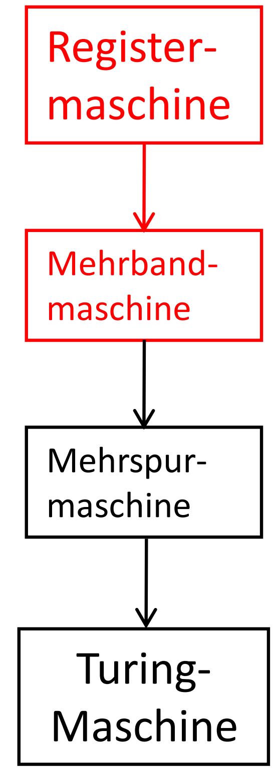
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

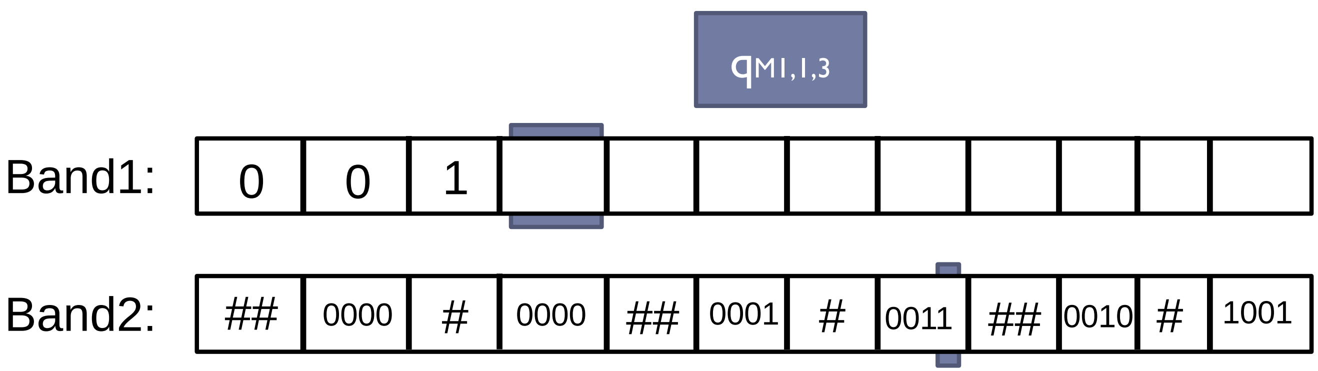
Beispiel: Addition 3 + 9 (0011 + 1001)

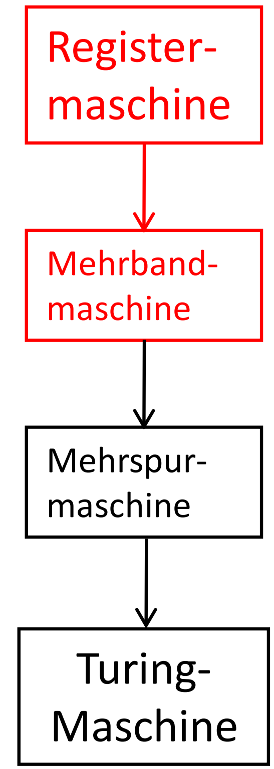
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





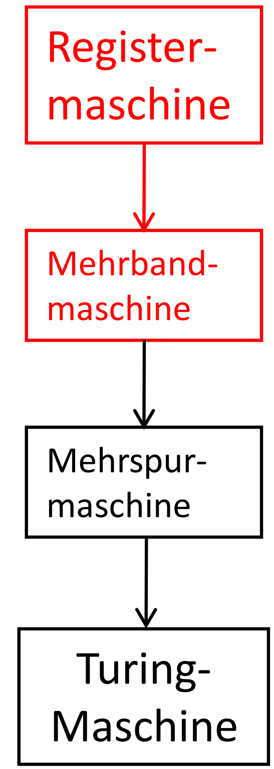
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

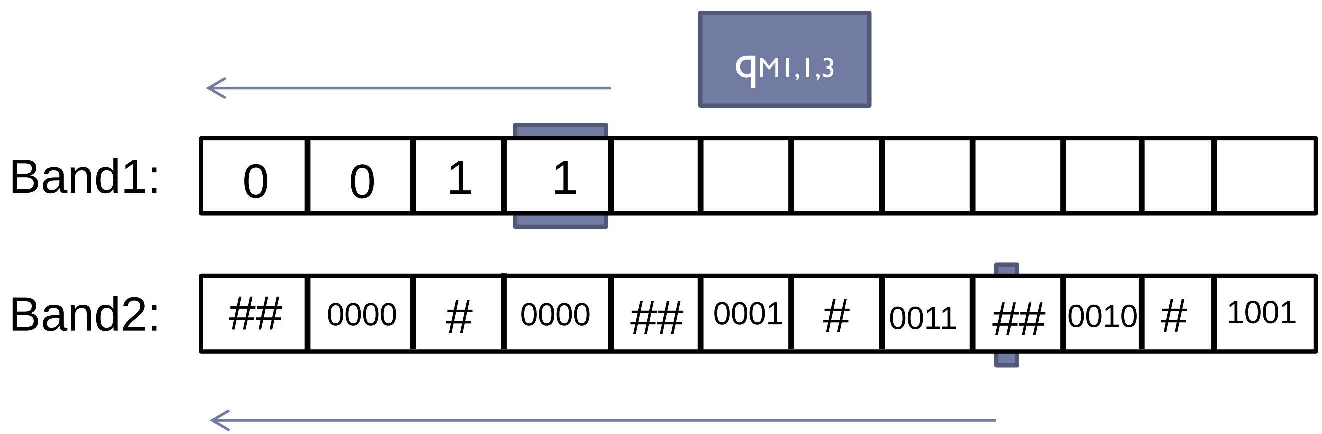
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

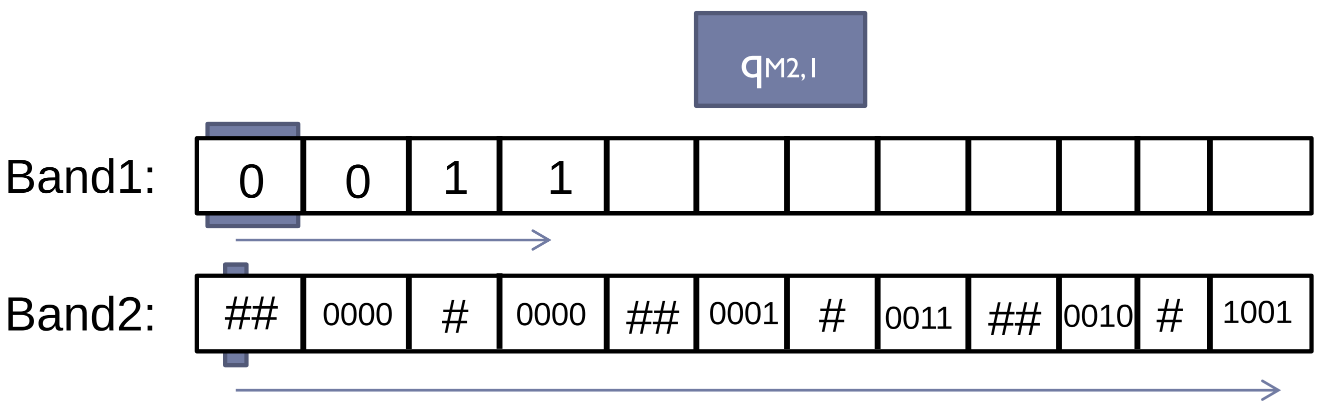
Beispiel: Addition 3 + 9 (0011 + 1001)

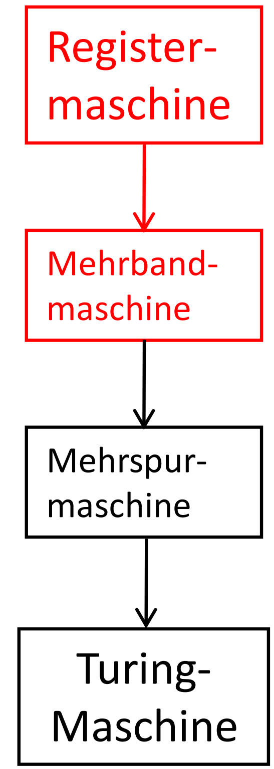
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

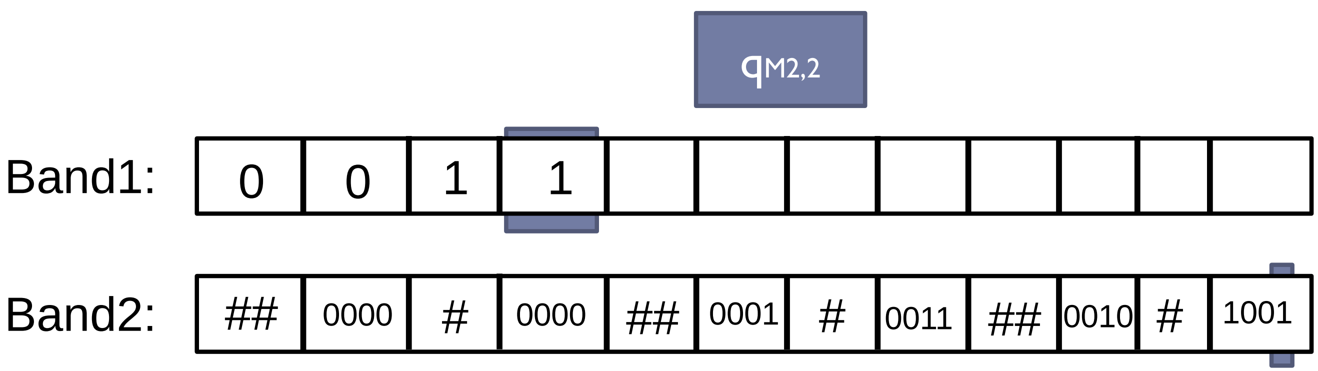
Beispiel: Addition 3 + 9 (0011 + 1001)

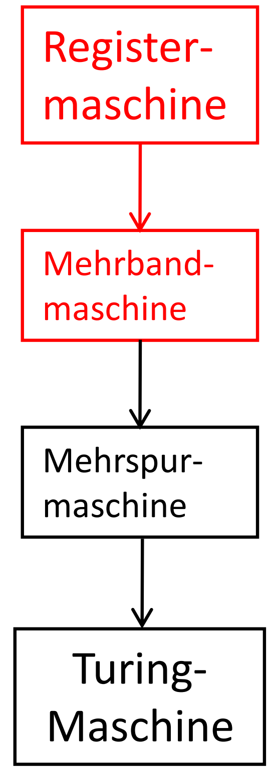
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

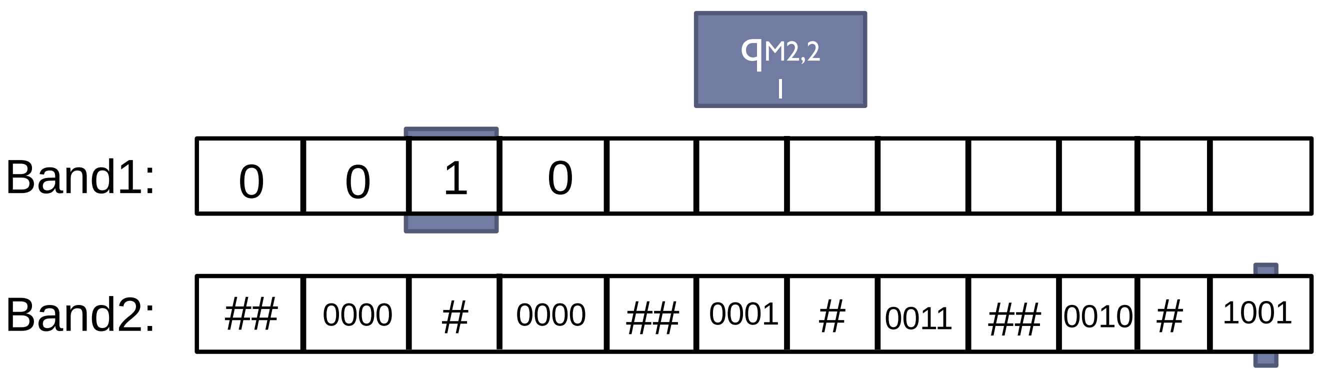
Beispiel: Addition 3 + 9 (0011 + 1001)

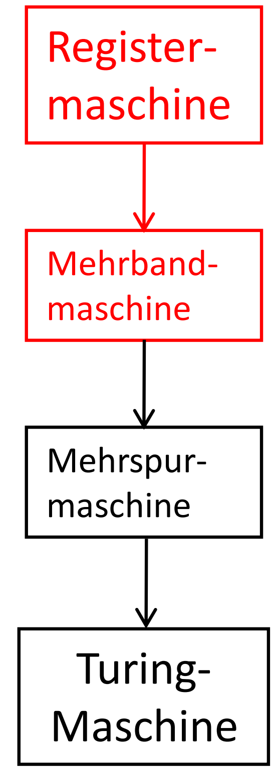
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

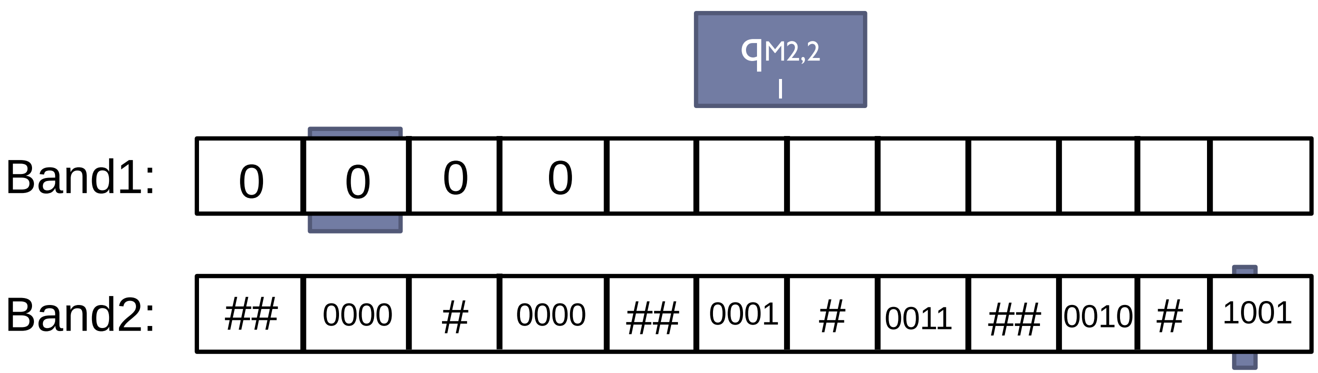
Beispiel: Addition 3 + 9 (0011 + 1001)

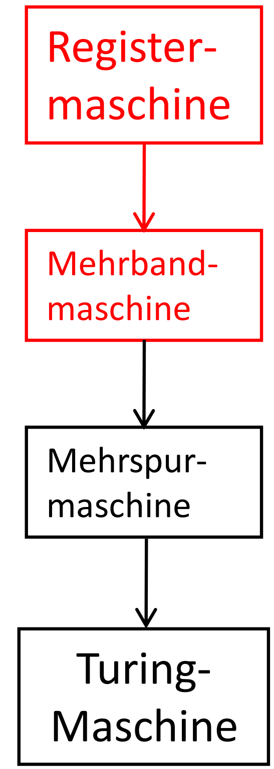
Registermaschine:

Befehle:
- -LOAD 1
- → -ADD 2
- -STORE 1
- END

Register:
- 0: 0000
- 1: 0011
- 2: 1001

Mehrbandmaschine:





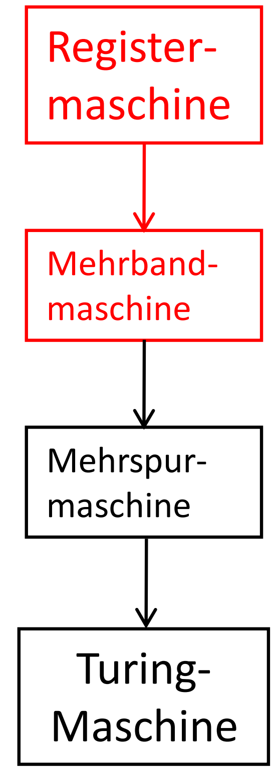
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

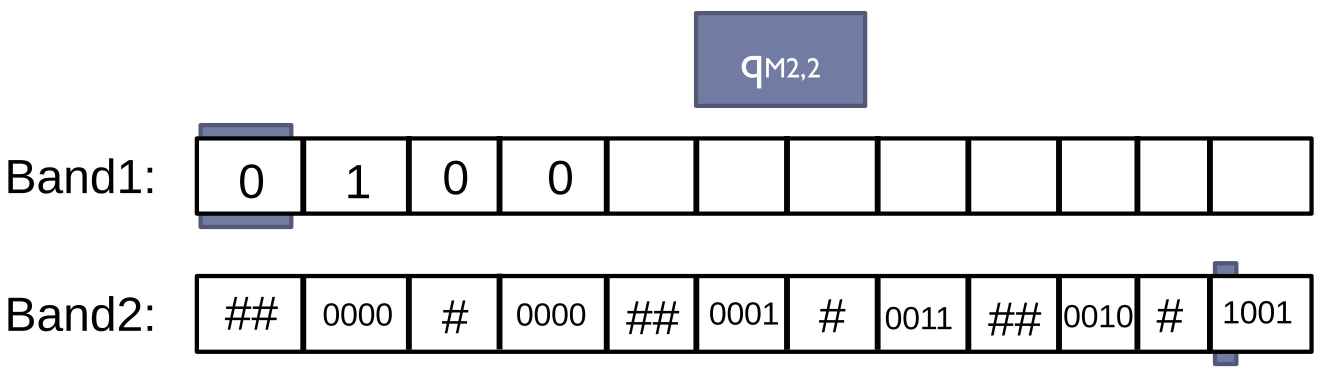
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

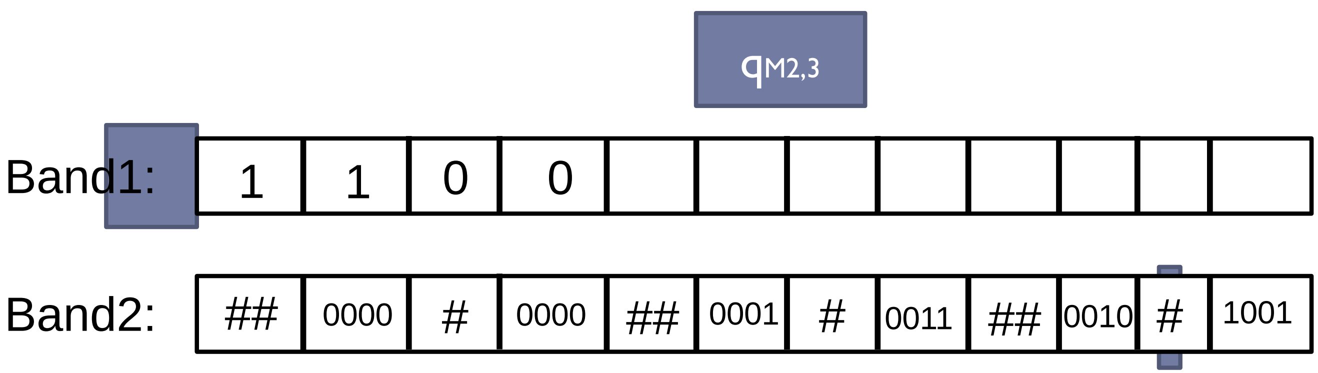
Beispiel: Addition 3 + 9 (0011 + 1001)

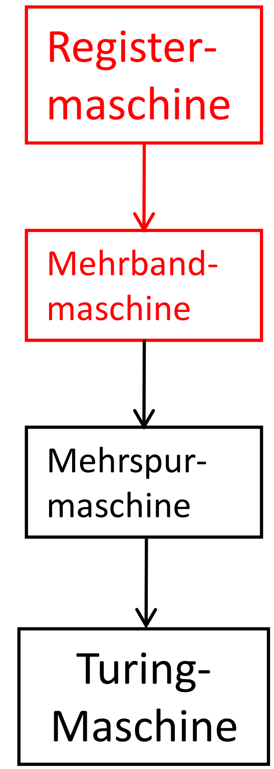
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

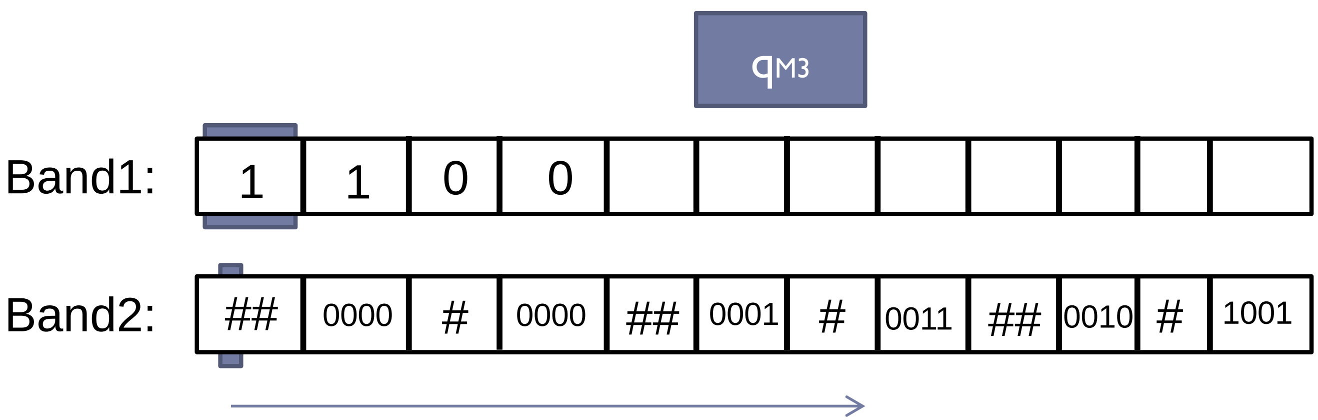
Beispiel: Addition 3 + 9 (0011 + 1001)

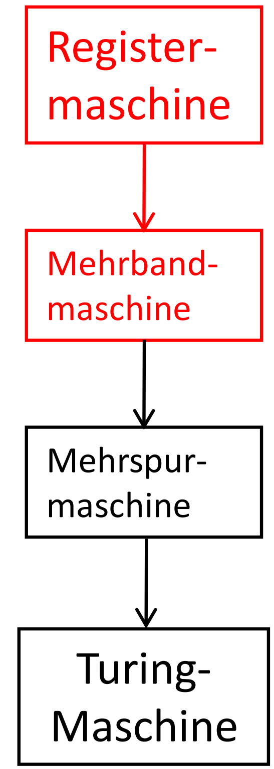
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

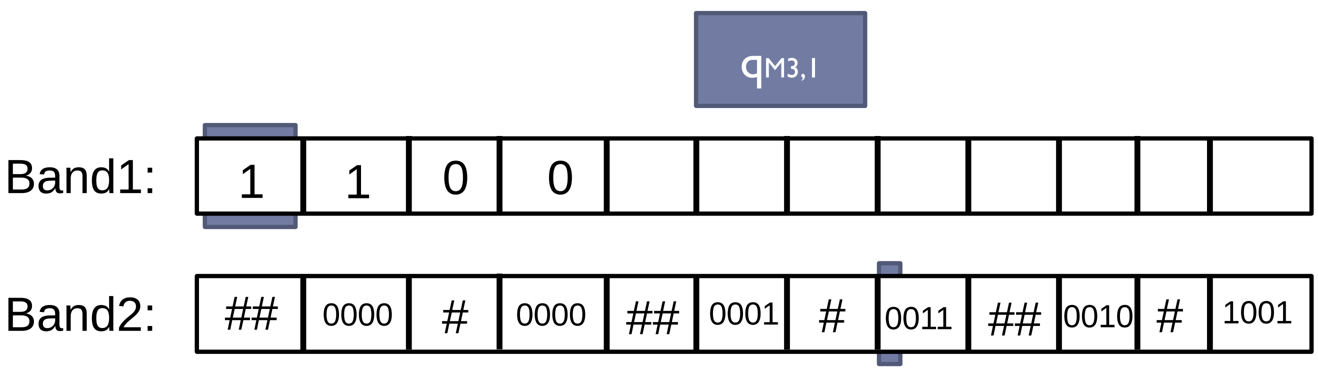
Beispiel: Addition 3 + 9 (0011 + 1001)

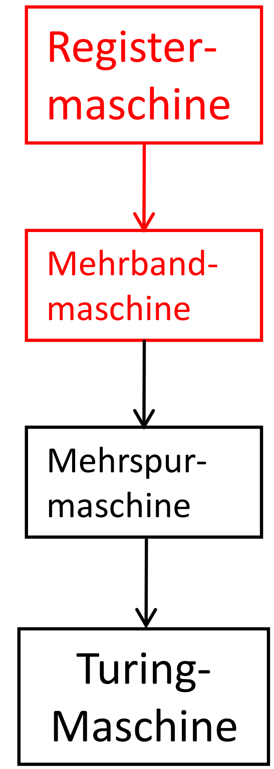
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





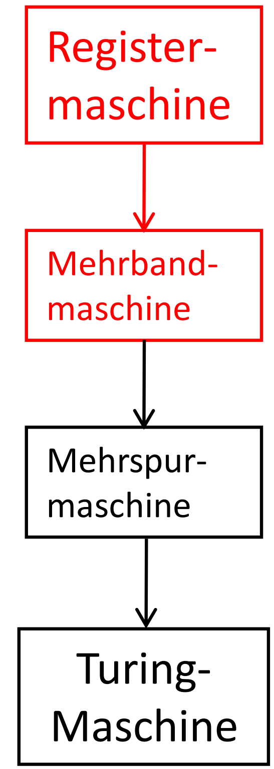
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

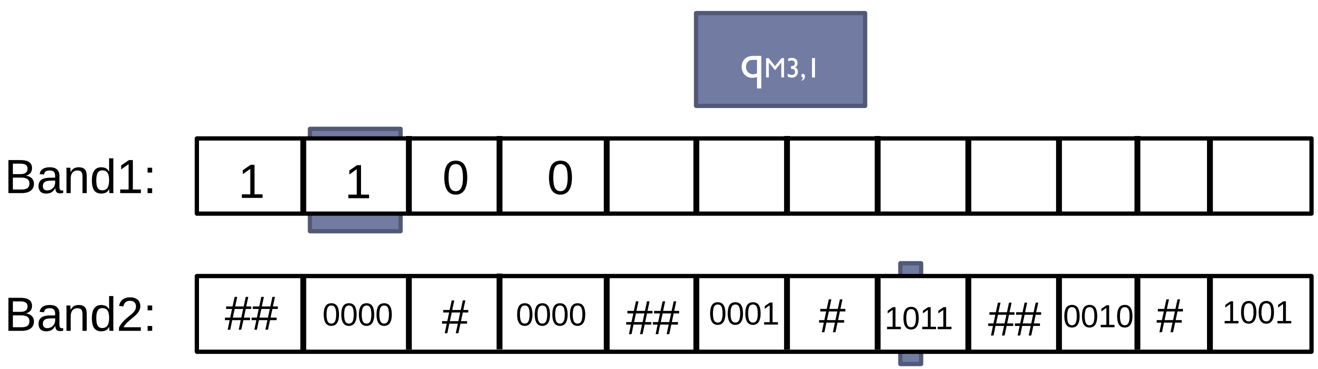
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
- 0: 0000
- 1: 0011
- 2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

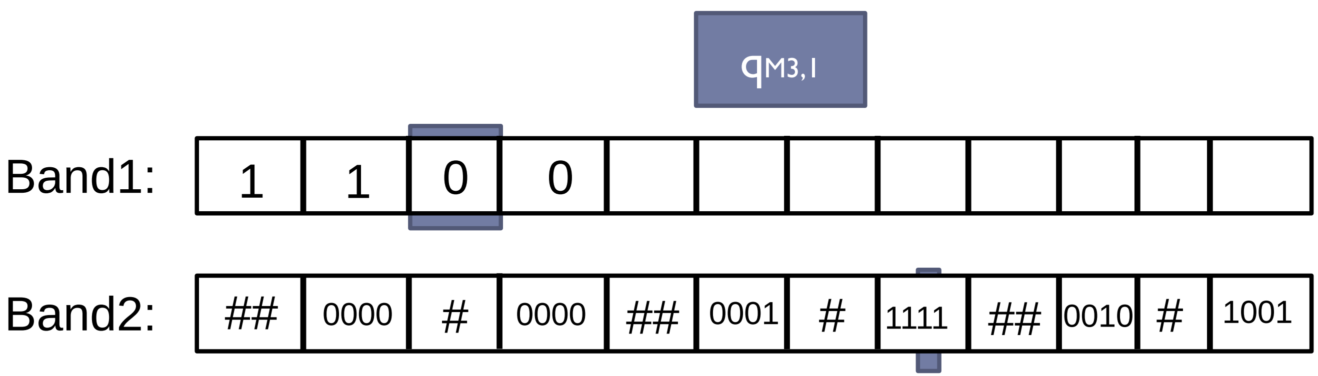
Beispiel: Addition 3 + 9 (0011 + 1001)

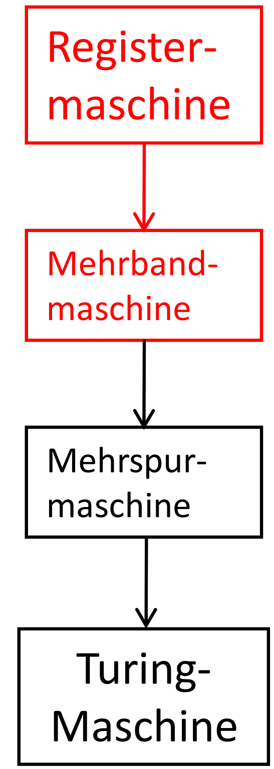
Registermaschine:

Befehle:
- -LOAD 1
- -ADD 2
- -STORE 1
- END

Register:
- 0: 0000
- 1: 0011
- 2: 1001

Mehrbandmaschine:





Simulation einer Registermaschine mit einer Mehrbandmaschine

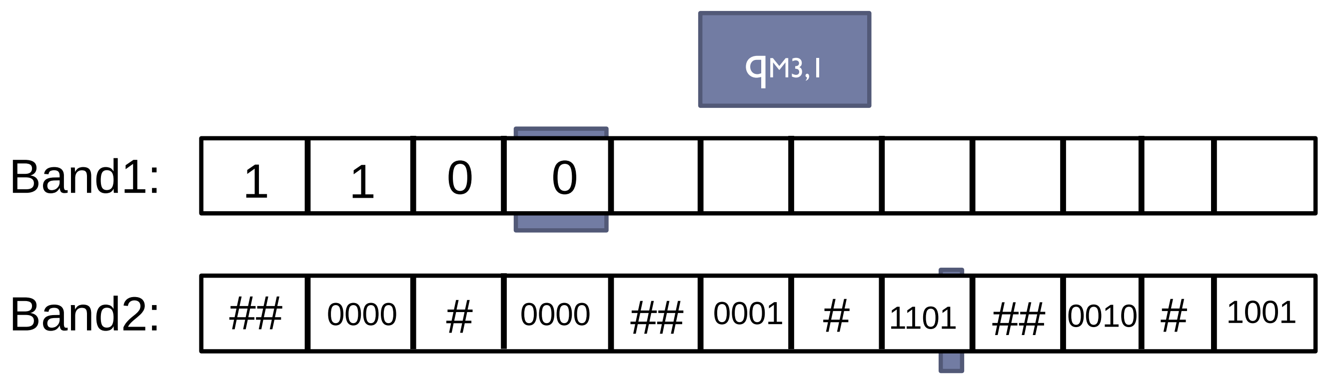
Beispiel: Addition 3 + 9 (0011 + 1001)

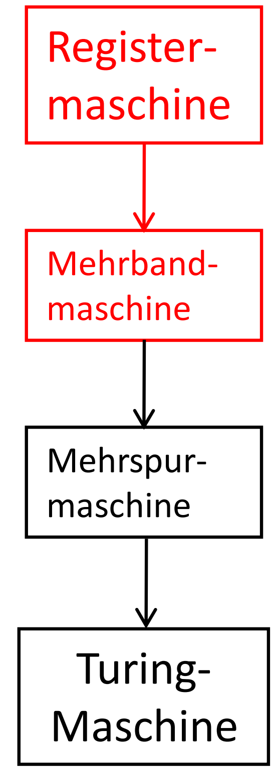
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:





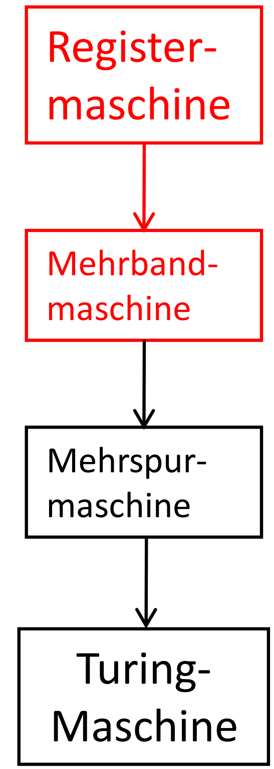
Simulation einer Registermaschine mit einer Mehrbandmaschine

Beispiel: Addition 3 + 9 (0011 + 1001)

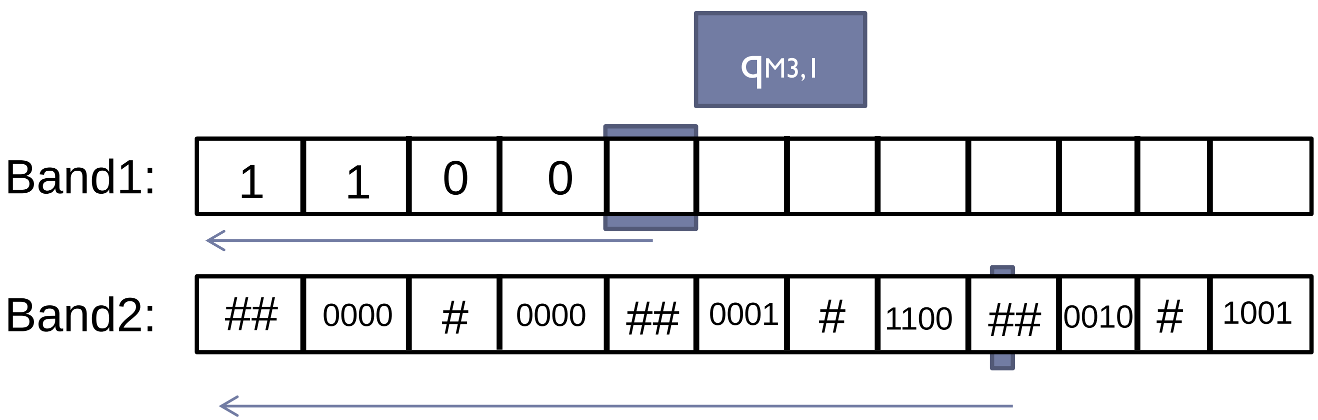
Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

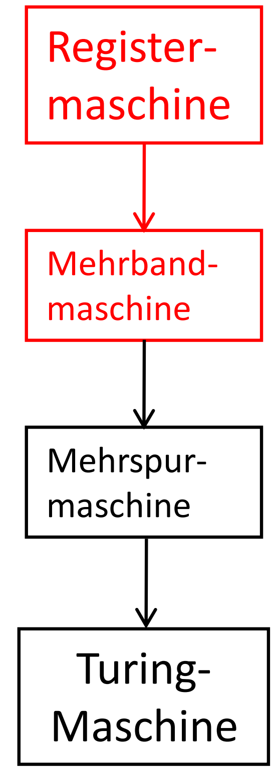
Register:
0: 0000
1: 0011
2: 1001



Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine



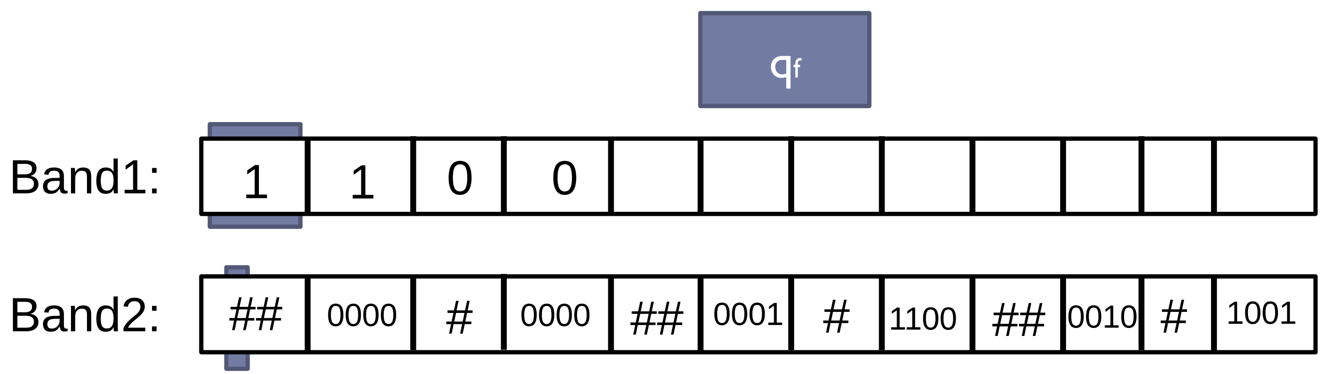
Beispiel: Addition 3 + 9 (0011 + 1001)

Registermaschine:

Befehle:
-LOAD 1
-ADD 2
-STORE 1
END

Register:
0: 0000
1: 0011
2: 1001

Mehrbandmaschine:



Simulation einer Registermaschine mit einer Mehrbandmaschine

Zeit- und Speicheraufwand:

- Speicherbedarf Registermaschine: $S(n)$
- Anzahl Schritte Registermaschine: $t(n)$
- Komplexitätsparameter/Eingabegröße: n

- Speicherbedarf $S'(n)$:

  $S'(n) = c_1 * S(n)$ mit $S(n) \leq c_2 * t(n)$

  $S'(n) \leq c_1 * c_2 * t(n)$

  $S'(n) \leq c_S * t(n)$ c -> Konstanten

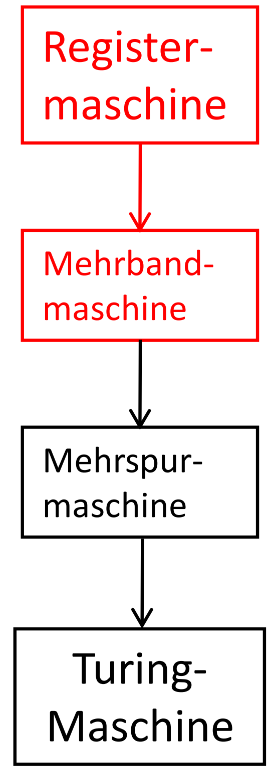
- Gesamtlaufzeit Turingmaschine $t'(n)$:

  1 Schritt Registermaschine: $c_3 * S'(n)$

  $t'(n) = t(n) * c_3 * S'(n)$

  $t'(n) = t(n) * c_3 * c_1 * c_2 * t(n)$

  $t'(n) = c * t(n)^2$

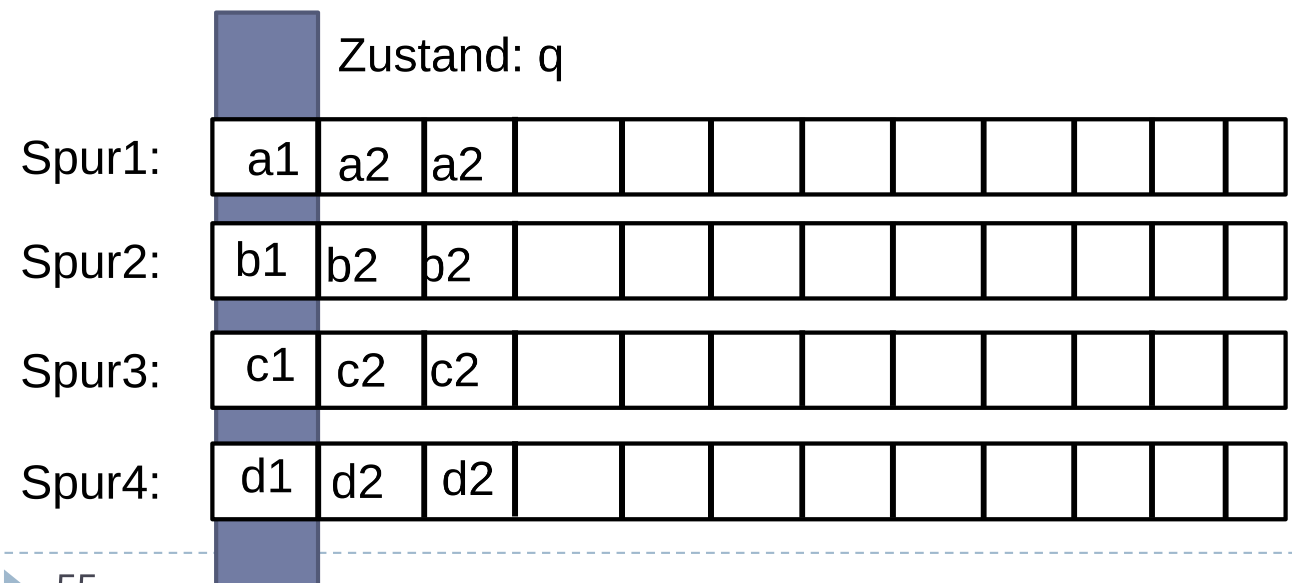


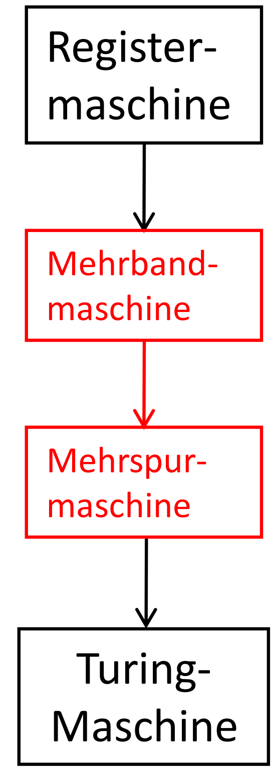
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

Die Mehrspurmaschine hat eine festgelegte Anzahl von Spuren, jedoch nur einen globalen Schreiblesekopf (Unterschied zur Mehrbandmaschine)

Übergangsfunktion:

$$\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, N\}$$

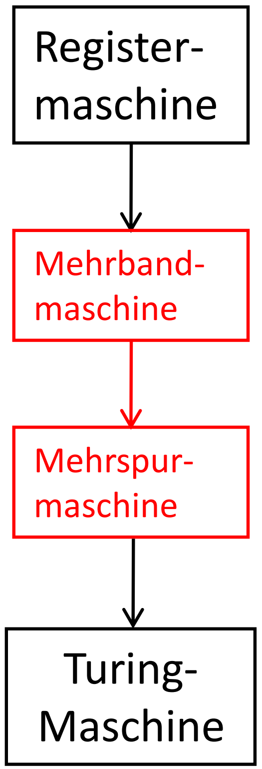




Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

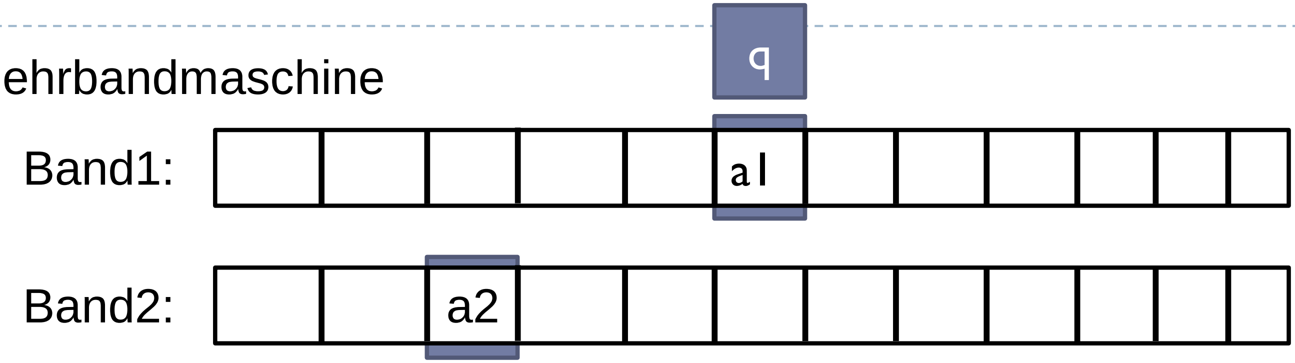
<u>Idee:</u>

- Für jedes Band der Mehrbandmaschine werden zwei Spuren bei der Mehrspurmaschine eingerichtet.
- Je eine Spur enthält den Inhalt des Bandes und die anderen den Zeiger, der auf das gerade zu bearbeitende Band zeigt.
- Nun durchläuft die Mehrspurmaschine den Bereich von erstem gesetzten Zeiger bis zum letzten und arbeitet die Zustandsübergänge ab, setzt die Zeiger um und springt wieder zum ersten Zeiger (mit kleinstem Spurindex)



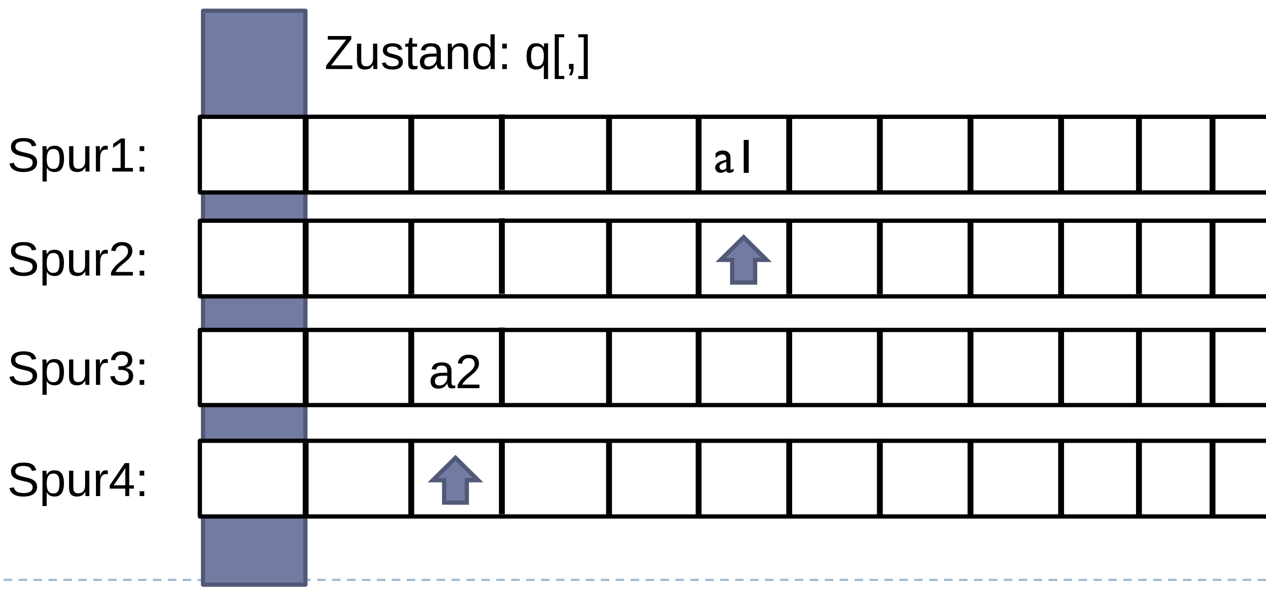
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

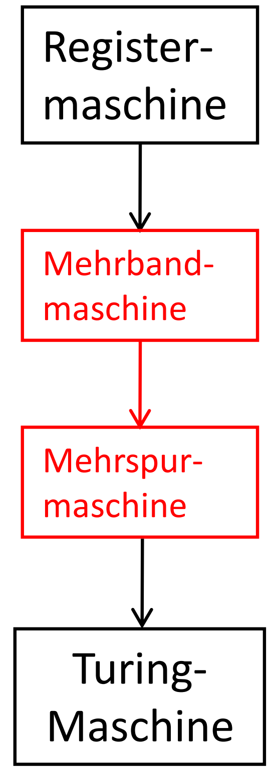
Mehrbandmaschine



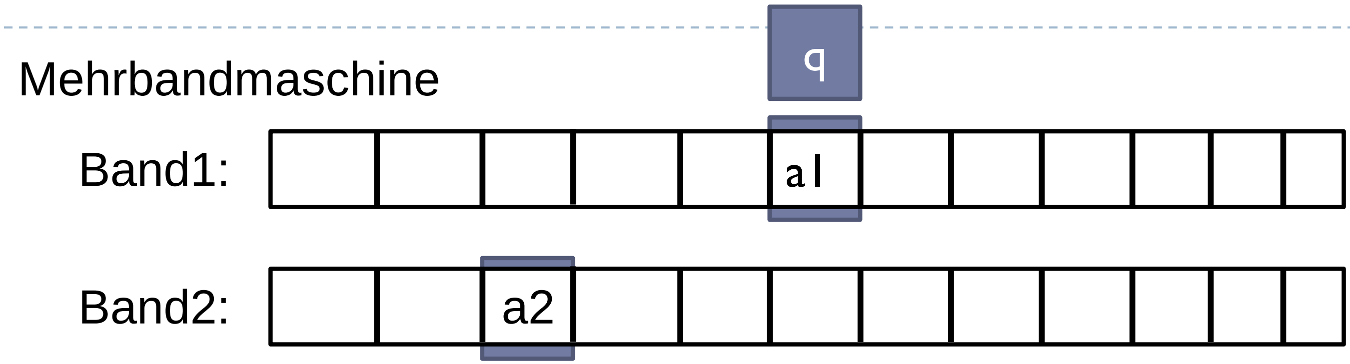
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

Mehrspurmaschine

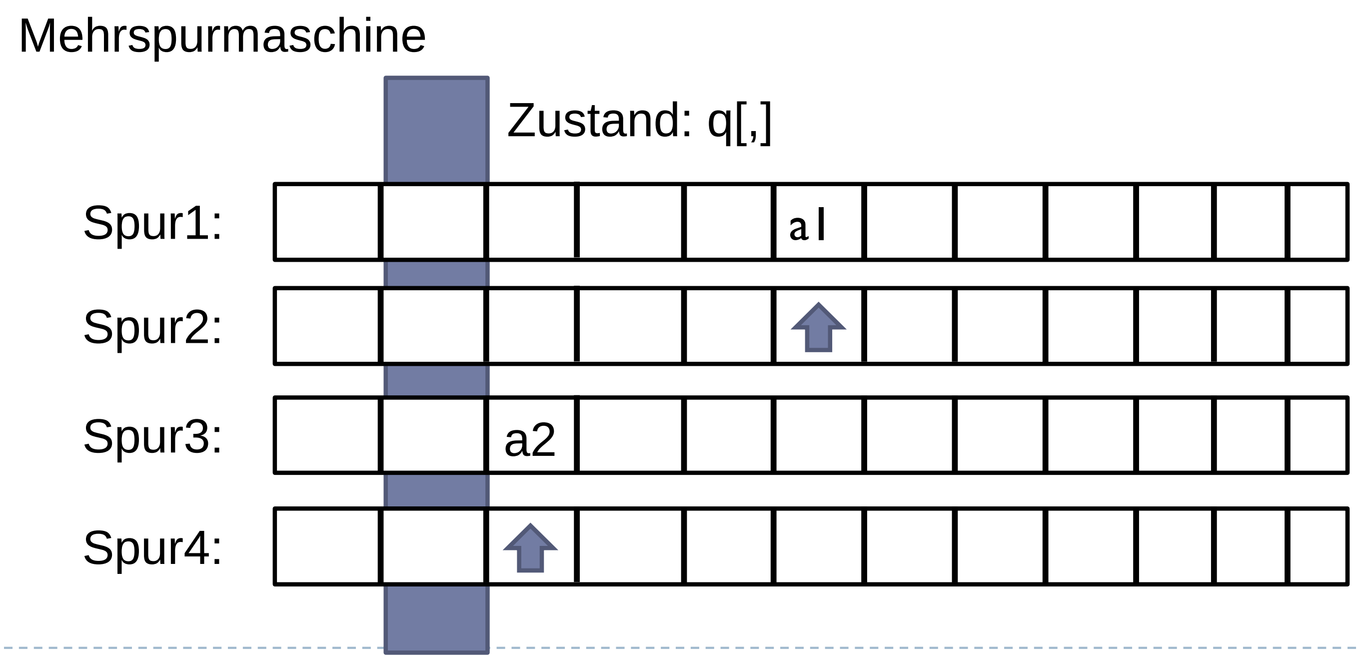


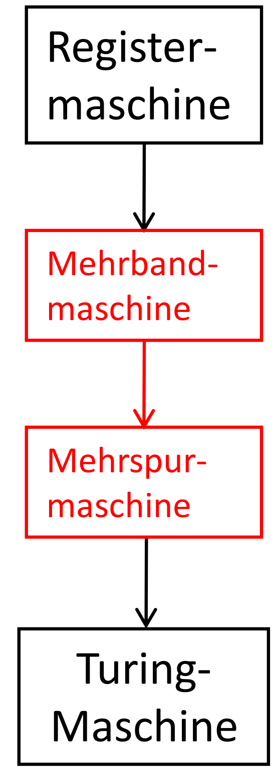


Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine



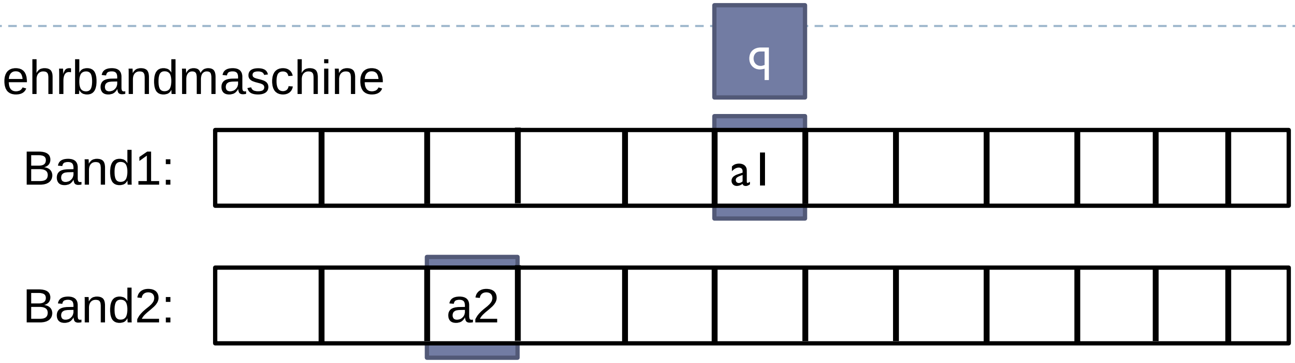
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$





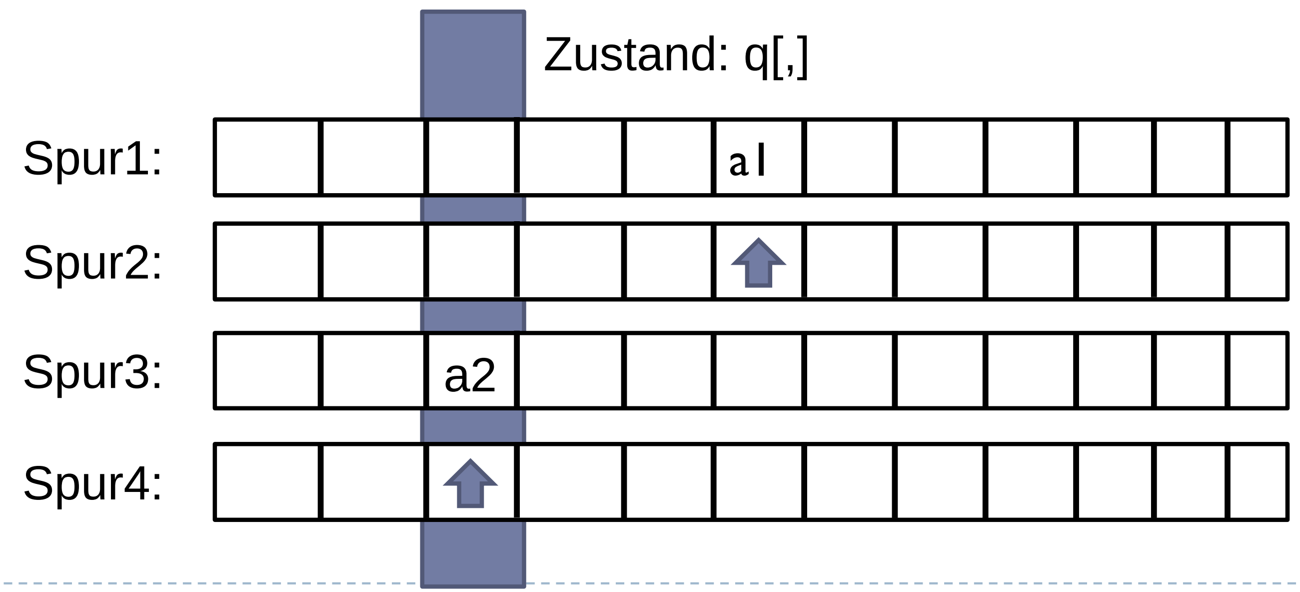
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

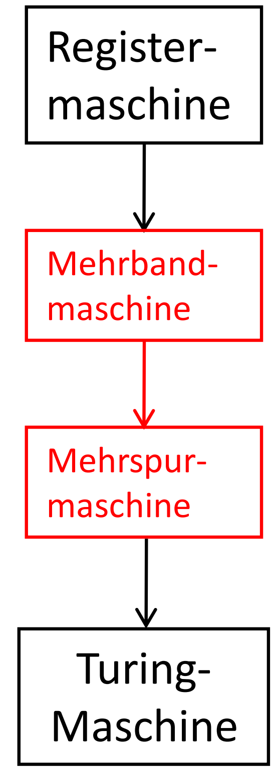
Mehrbandmaschine



$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

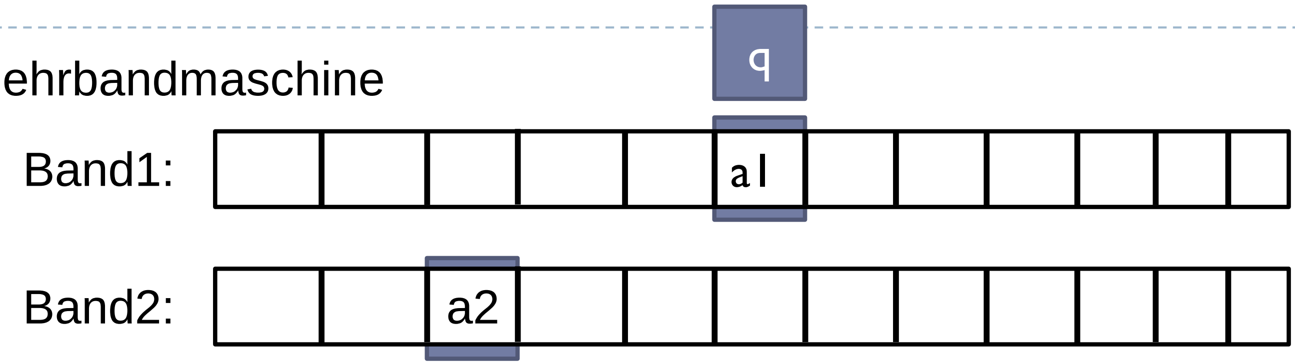
Mehrspurmaschine





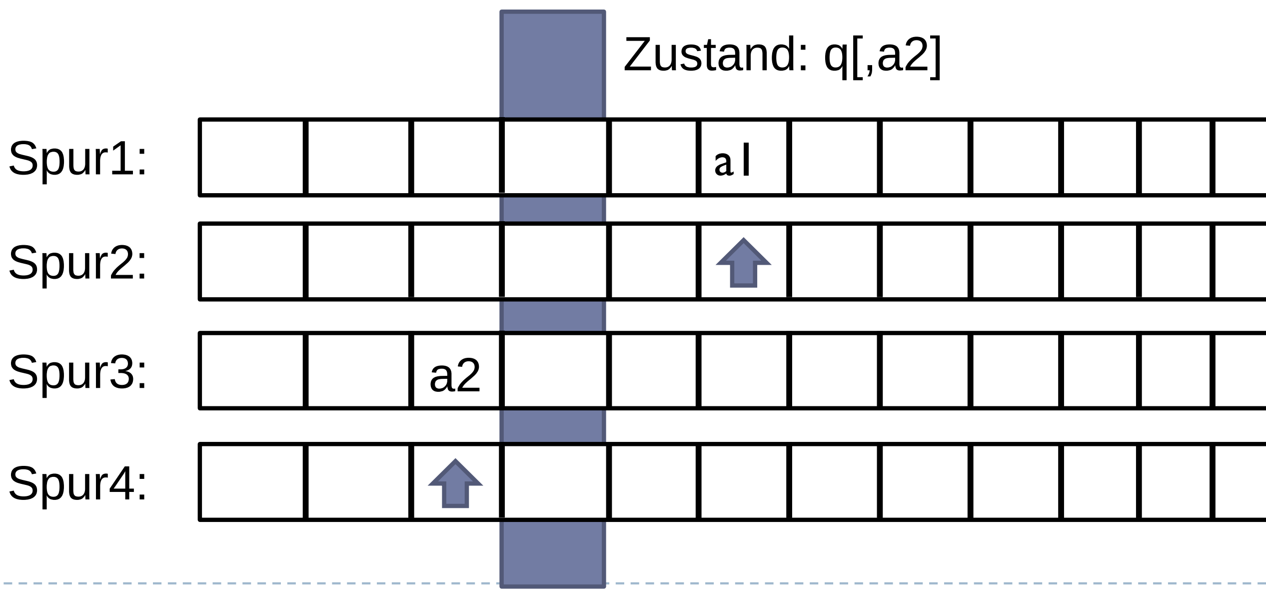
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

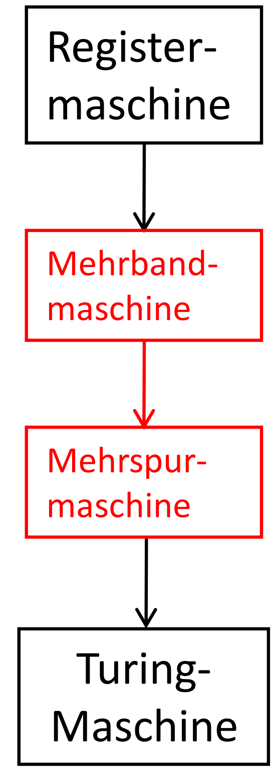
Mehrbandmaschine



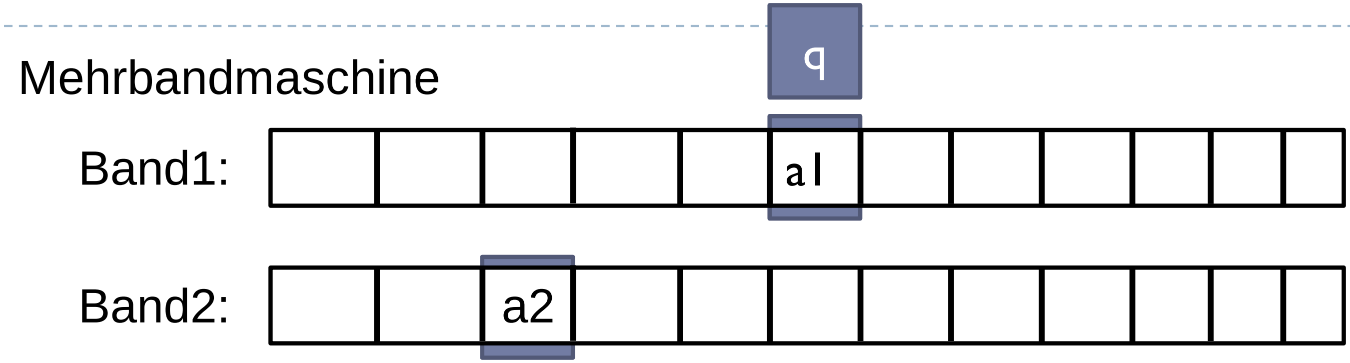
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

Mehrspurmaschine

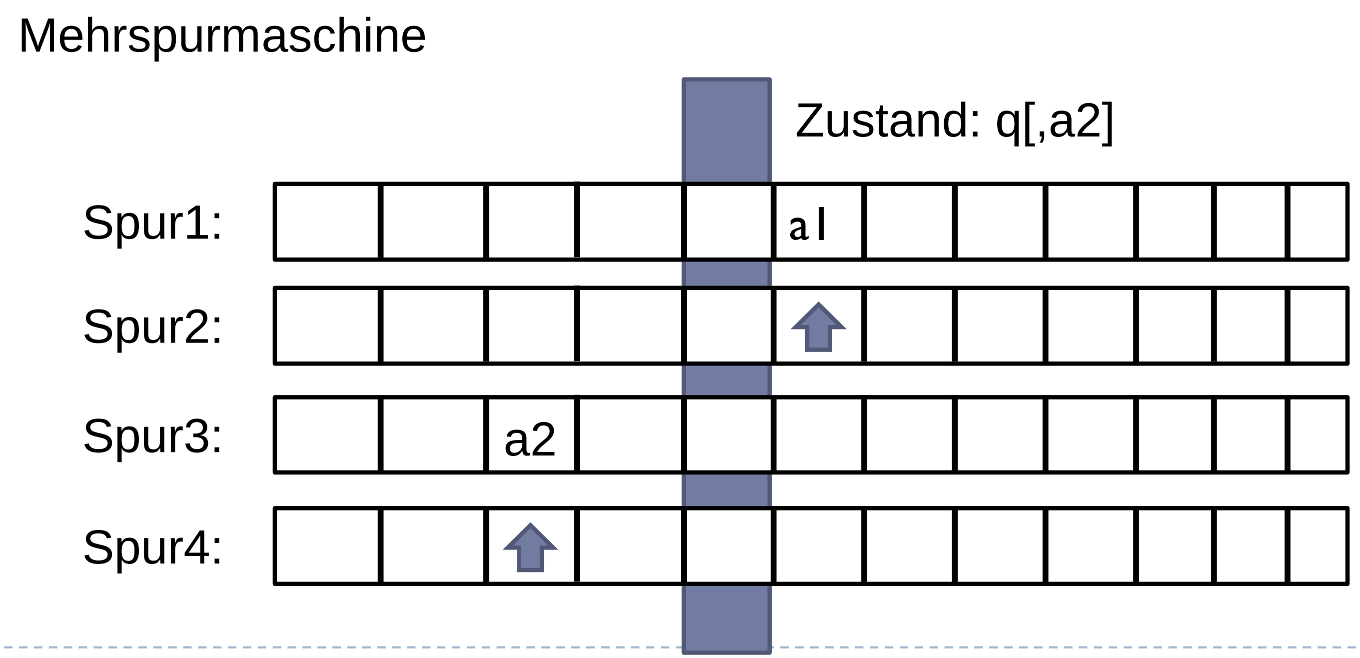


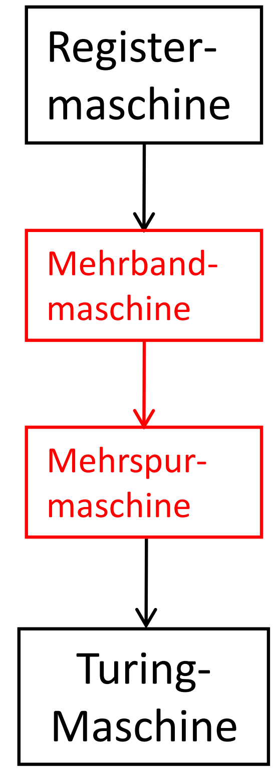


Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine



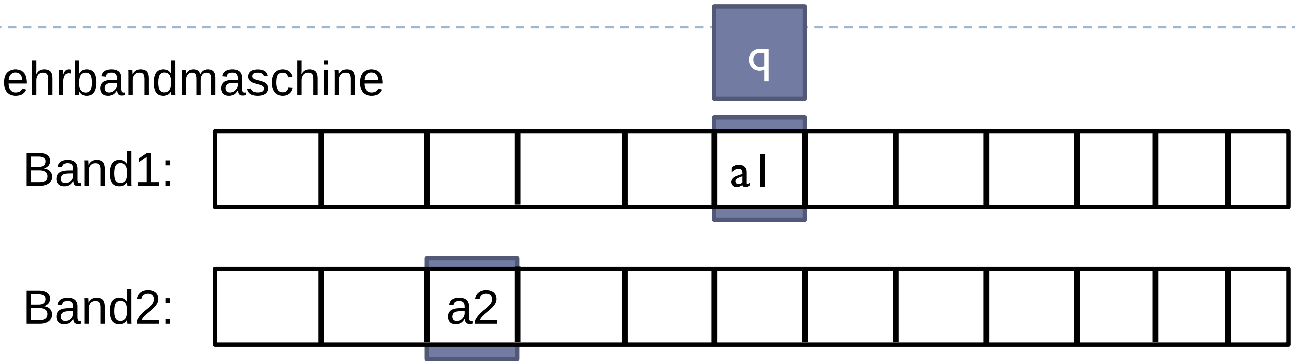
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

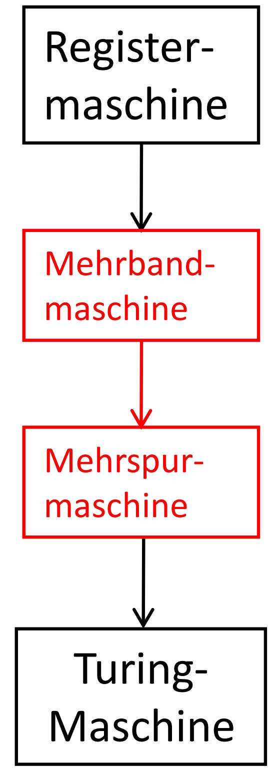




Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

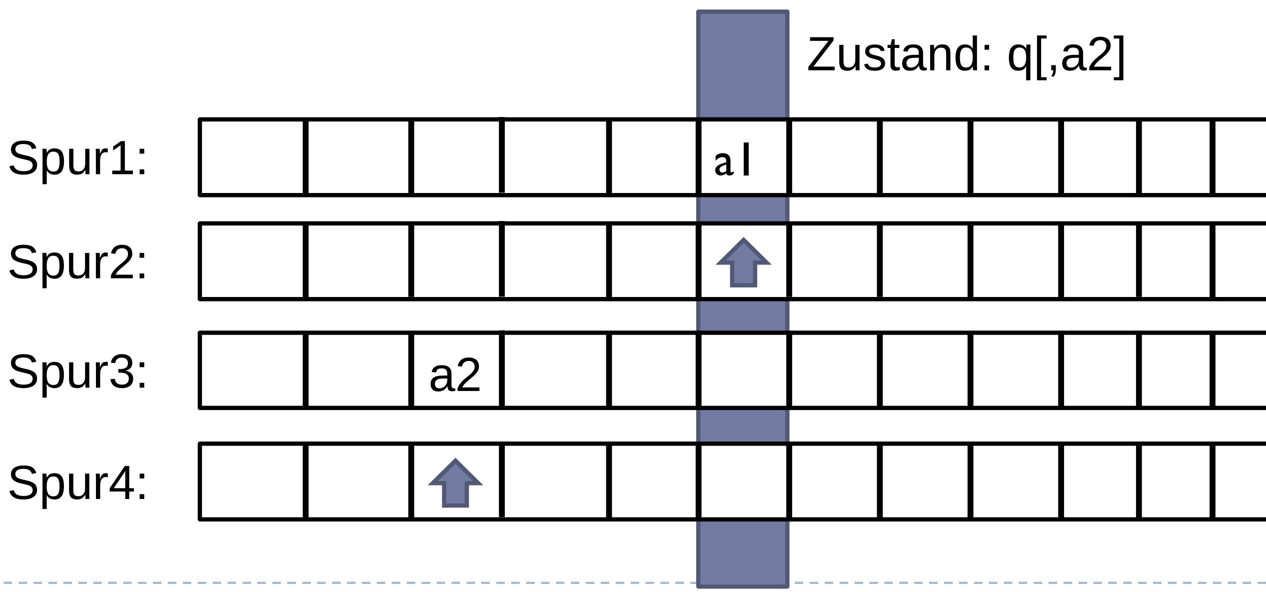
Mehrbandmaschine





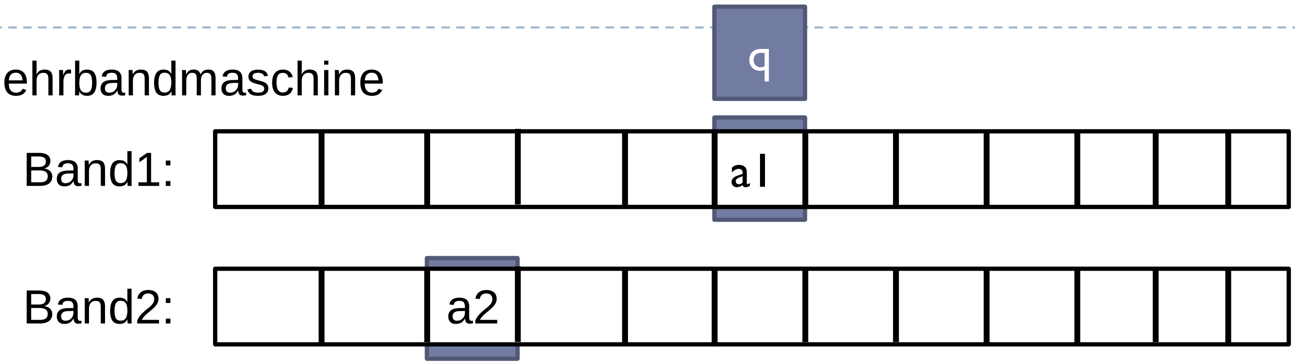
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

Mehrspurmaschine



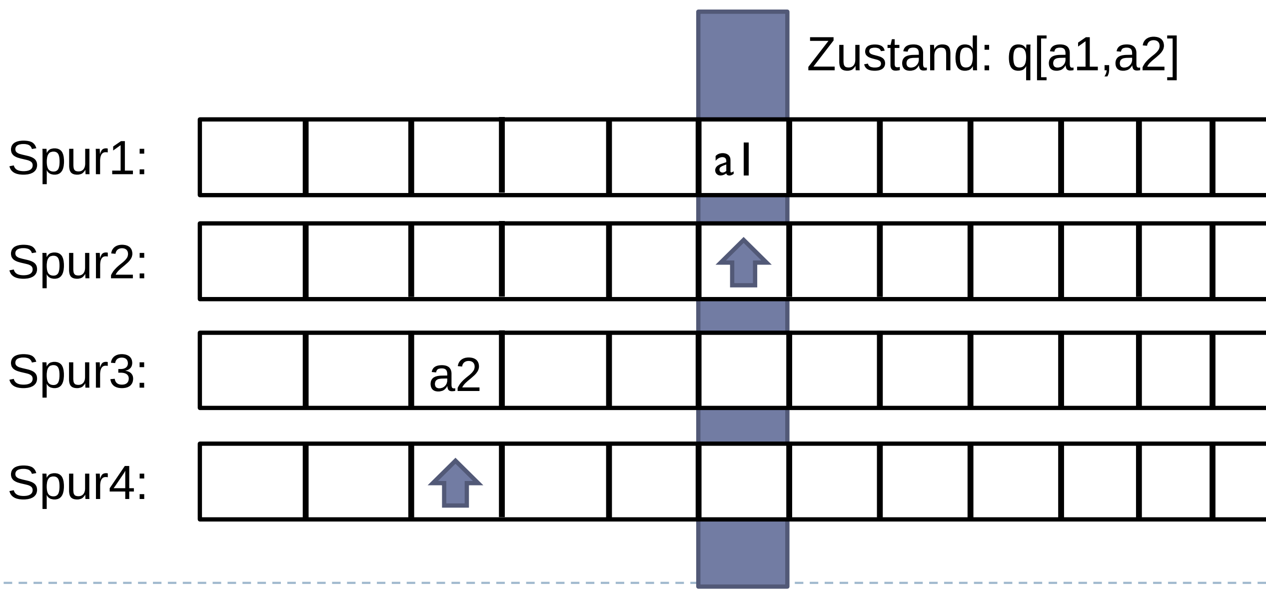
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

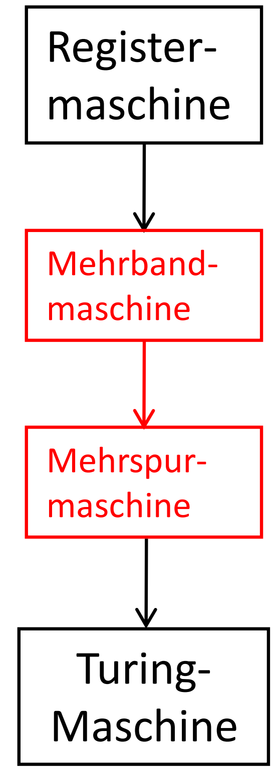
Mehrbandmaschine



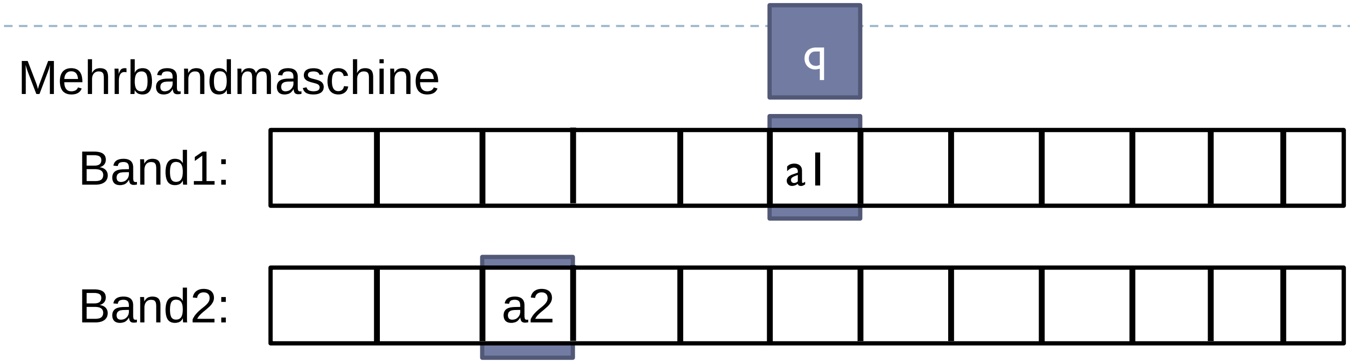
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

Mehrspurmaschine

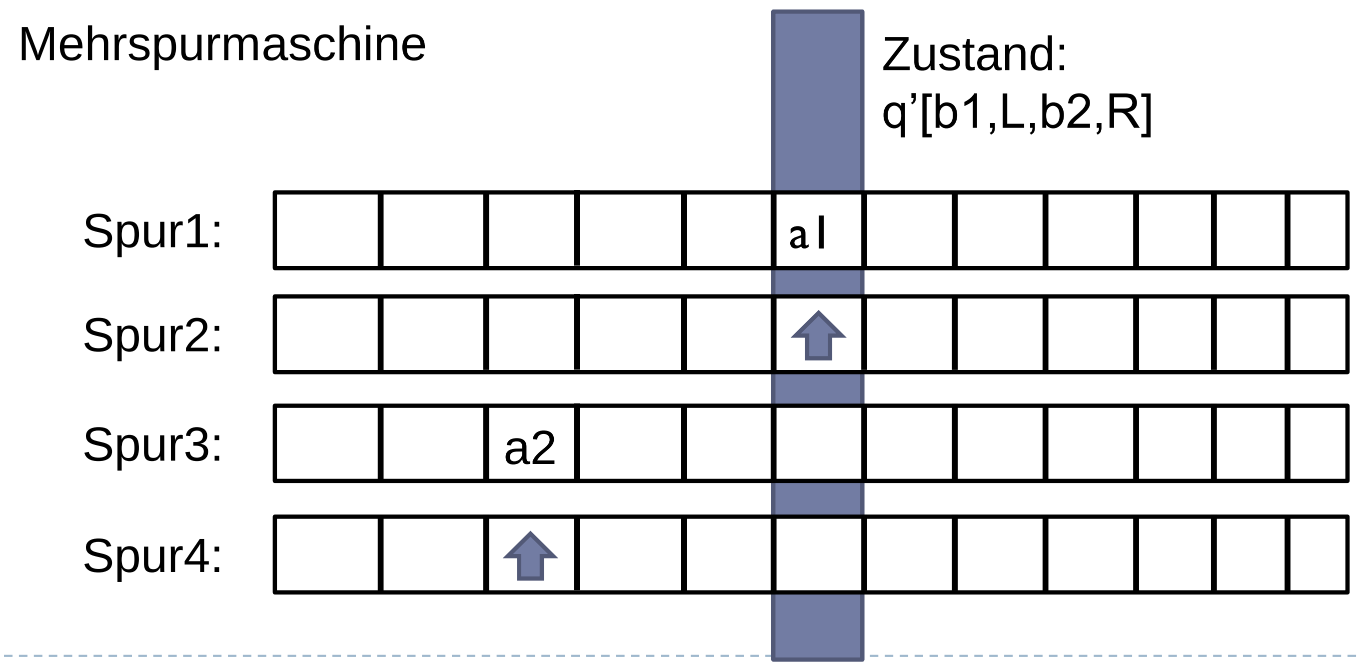


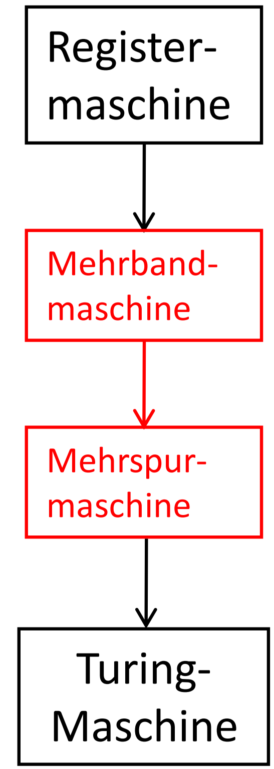


Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

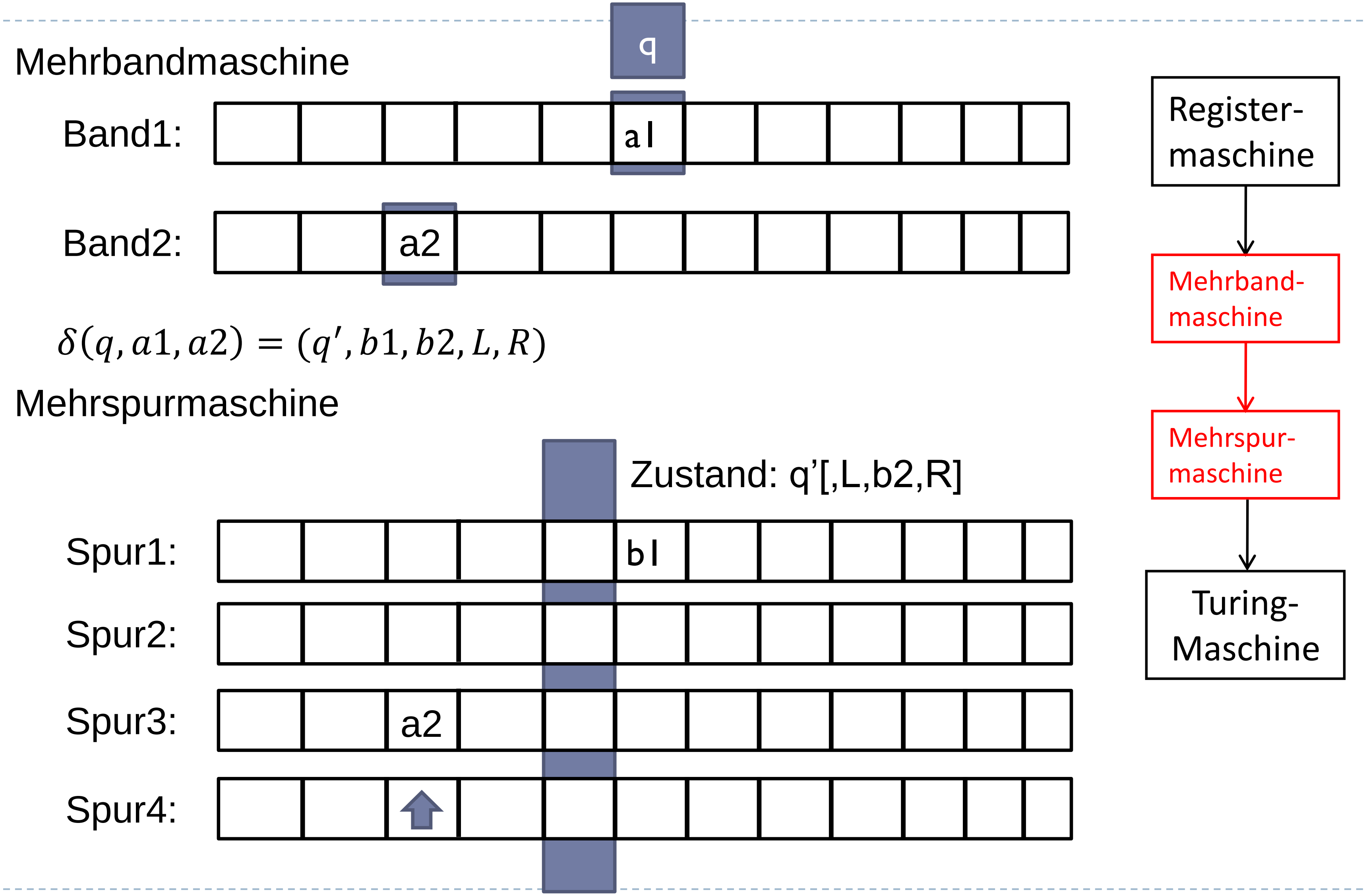


$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$



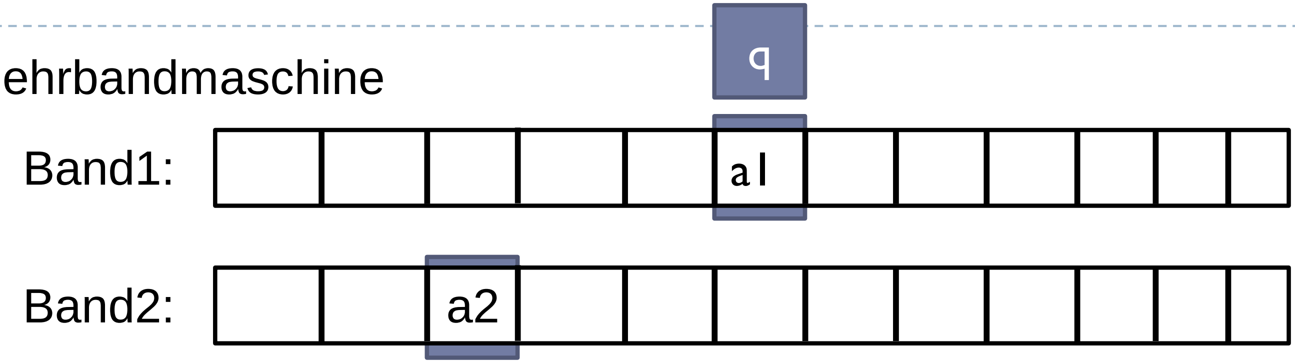


Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine



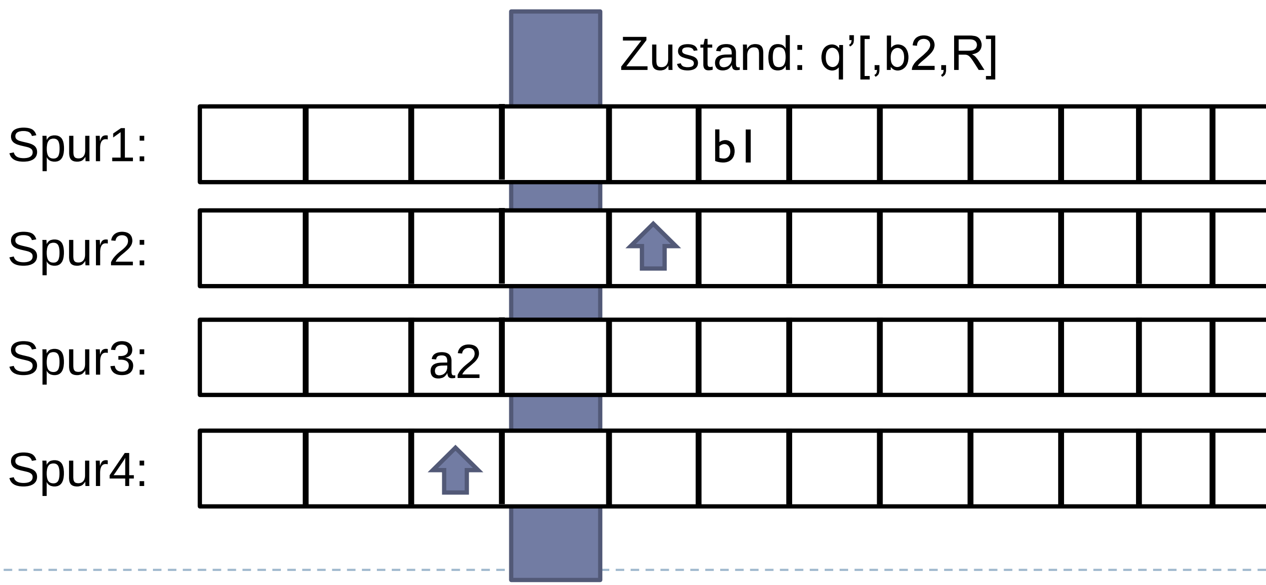
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

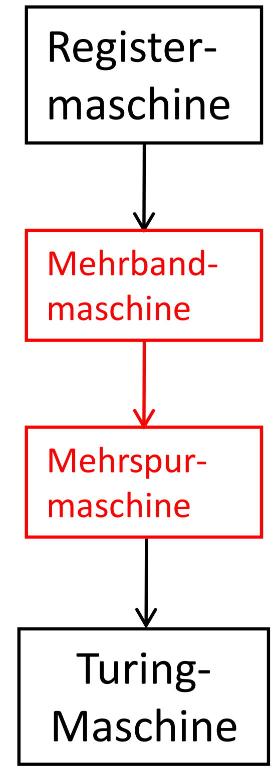
Mehrbandmaschine



$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

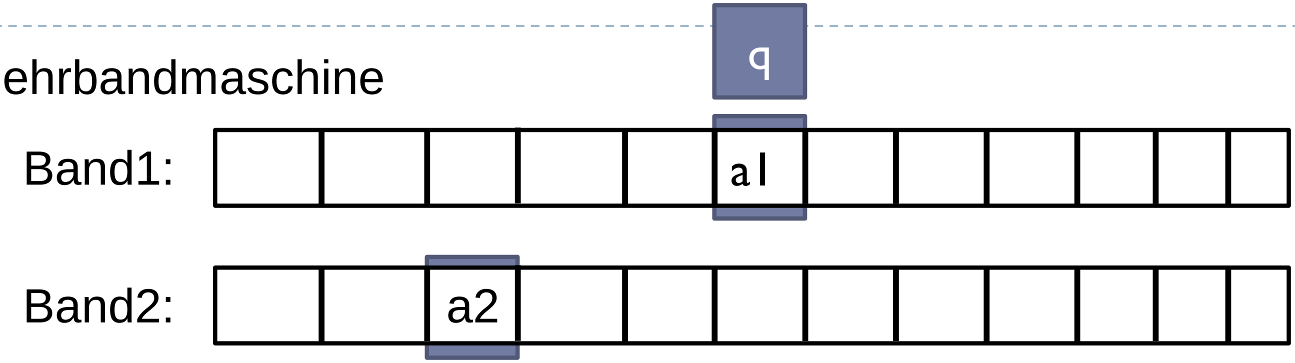
Mehrspurmaschine





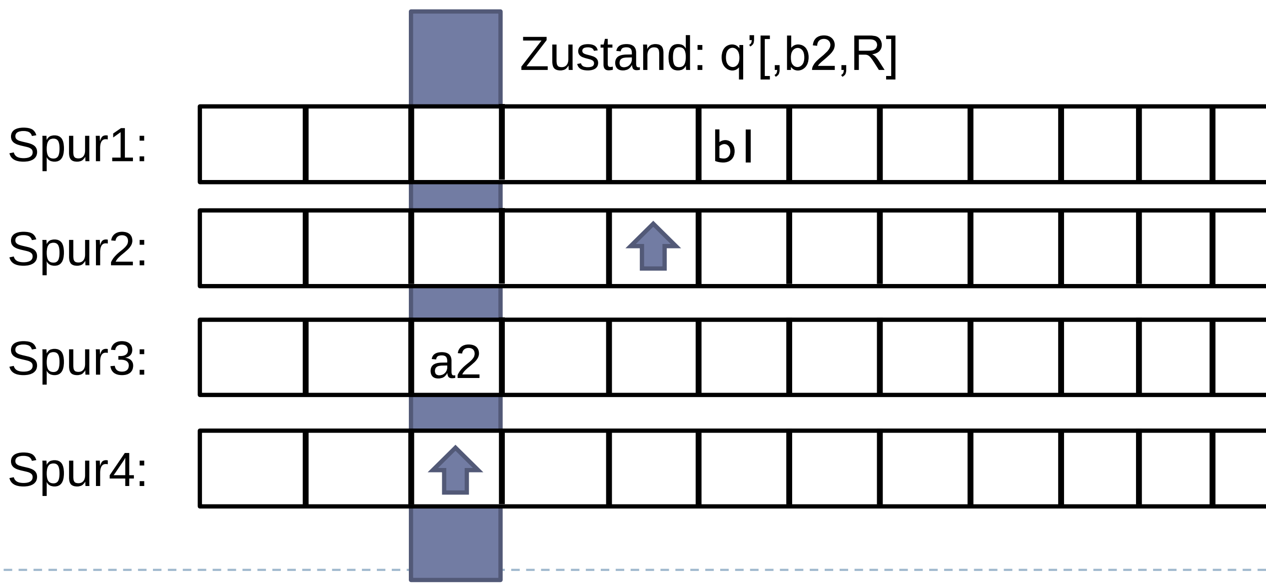
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

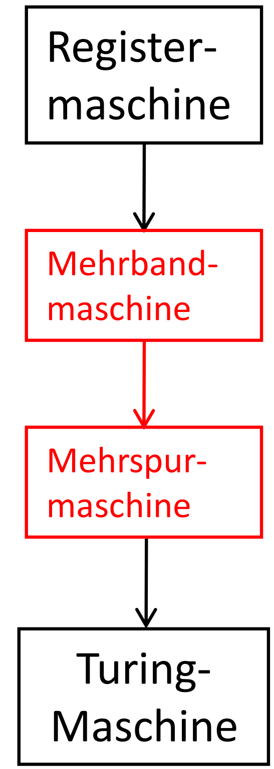
Mehrbandmaschine



$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

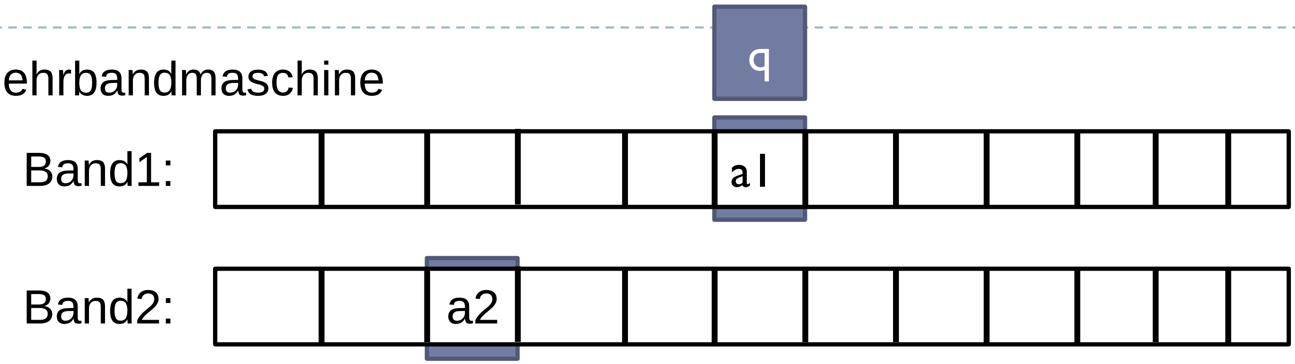
Mehrspurmaschine





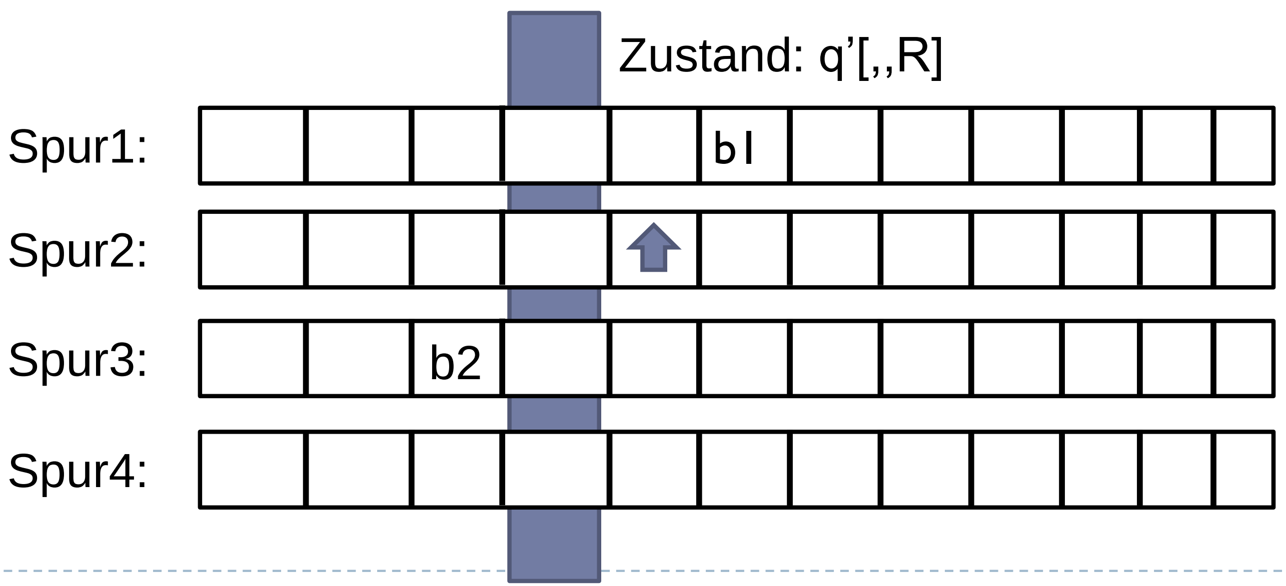
Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

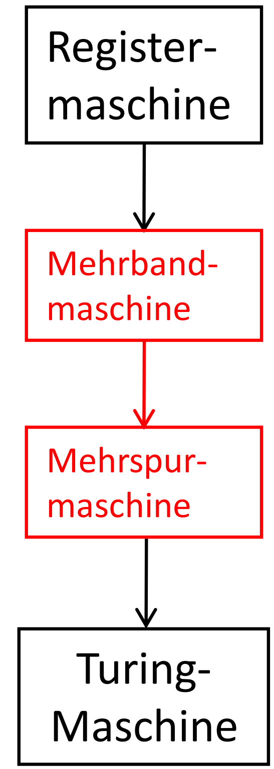
Mehrbandmaschine



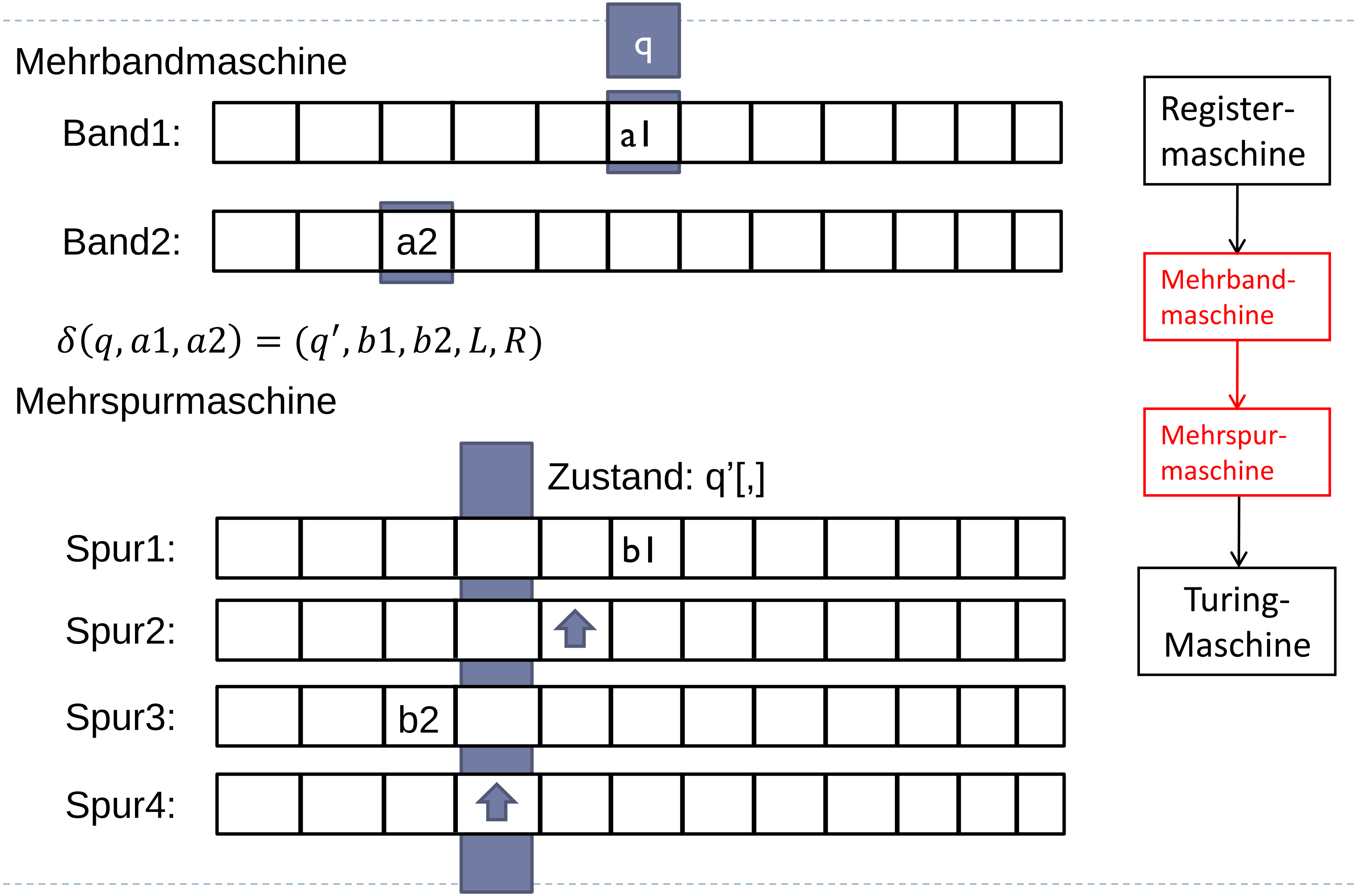
$$\delta(q, a1, a2) = (q', b1, b2, L, R)$$

Mehrspurmaschine



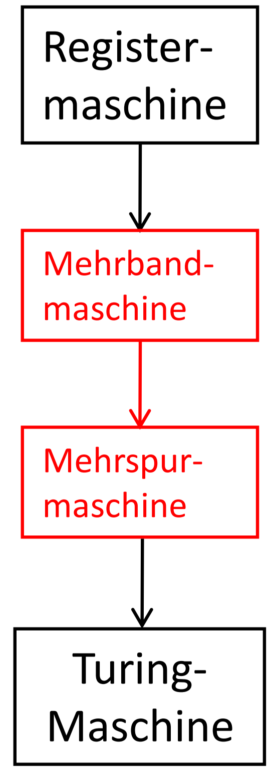


Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine



Simulation einer Mehrbandmaschine mit einer Mehrspurmaschine

Zeit- und Speicheraufwand:



- Speicherbedarf Mehrbandmaschine: S'(n)
- Anzahl Schritte Mehrbandmaschine: t'(n)
- Komplexitätsparameter/Eingabegröße: n

- Speicherbedarf S''(n):

  $S''(n) = c_1 * S'(n)$ mit $S'(n) \leq c_2 * t'(n)$

  $S''(n) \leq c_1 * c_2 * t'(n)$

  $S''(n) \leq c_S * t'(n)$

  c -> Konstanten

- Gesamtlaufzeit t''(n):

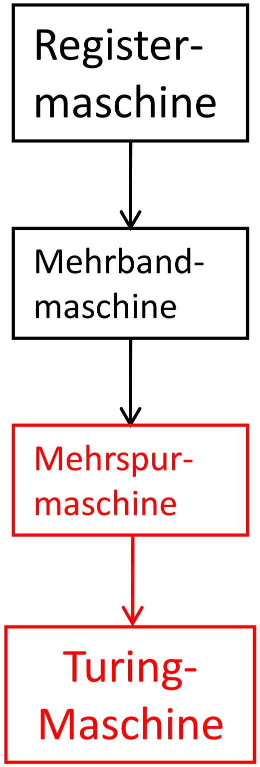
  1 Schritt Mehrbandmaschine: $c_3 * S''(n)$

  $t''(n) = t'(n) * c_2 * S''(n)$

  $t''(n) = t'(n) * c_1 * c_2 * c_3 * t'(n)$

  $t''(n) = c * t'(n)^2$

Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine



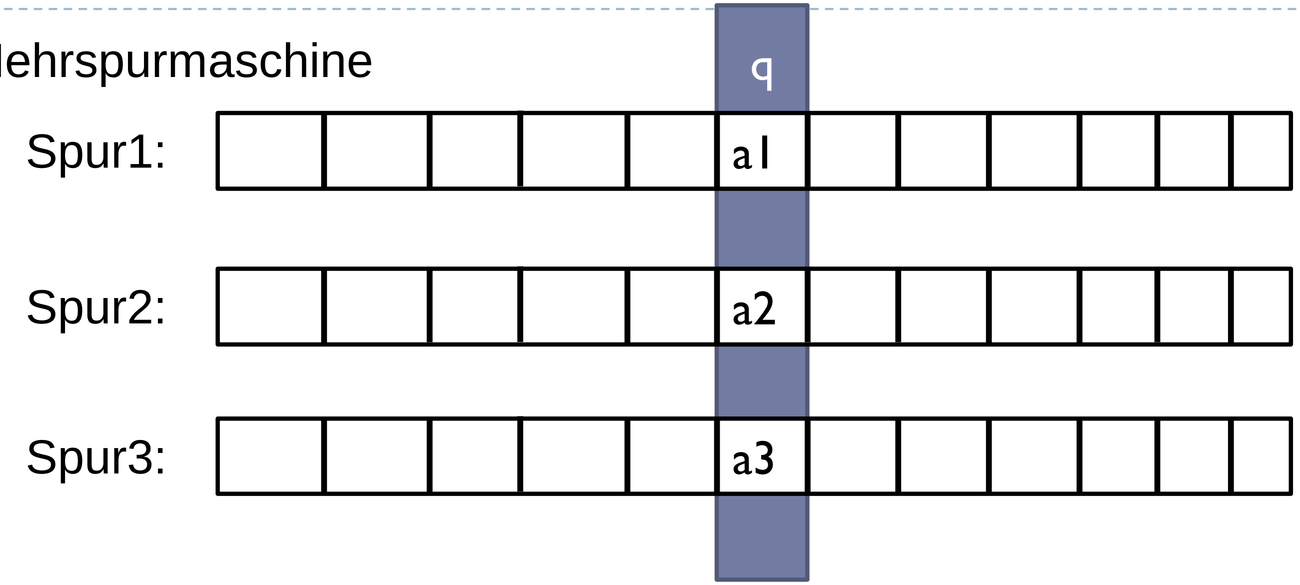
<u>Idee:</u> Jede mögliche gelesene Kombination von Symbolen auf den Spuren der Mehrspurmaschine ist ein Symbol der Einband-Turingmaschine

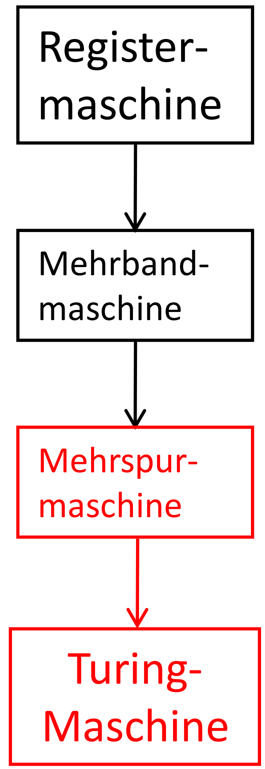
Übergangsfunktion:

$$\delta(q, a1a2a3) = (q', b1b2b3, L/R)$$

Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine

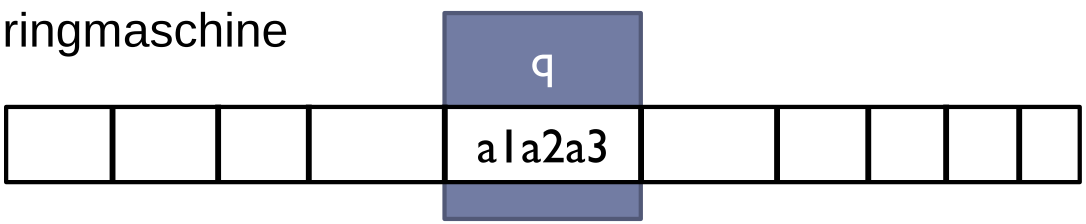
Mehrspurmaschine





$$\delta(q, a1, a2, a3) = (q', b1, b2, b3, R)$$

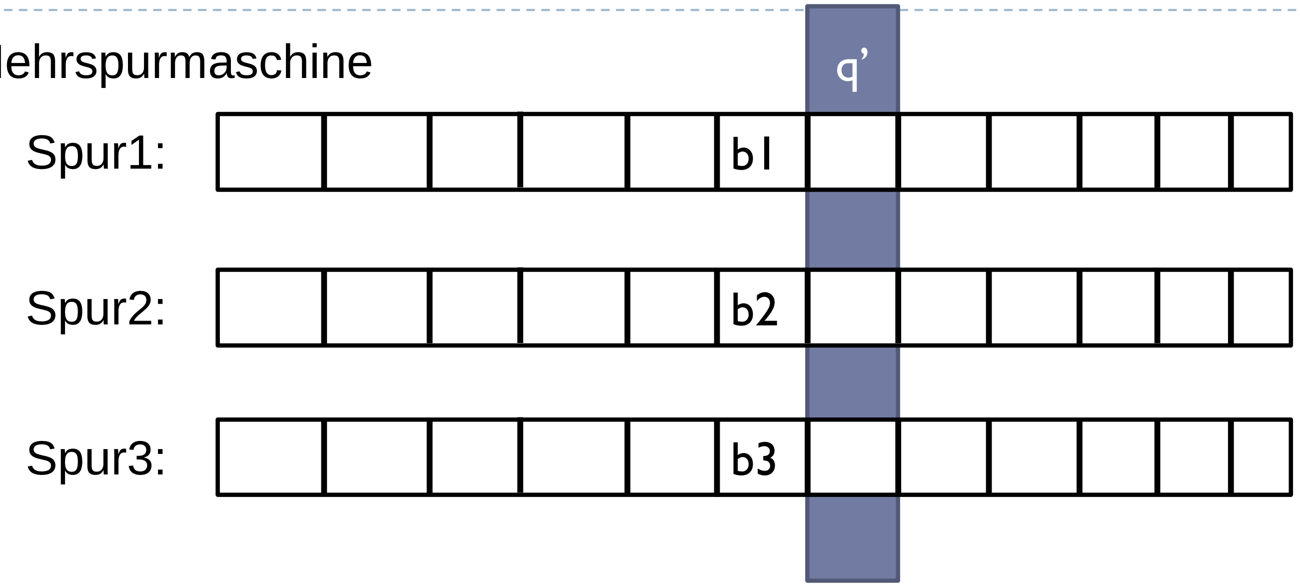
Einband-Turingmaschine

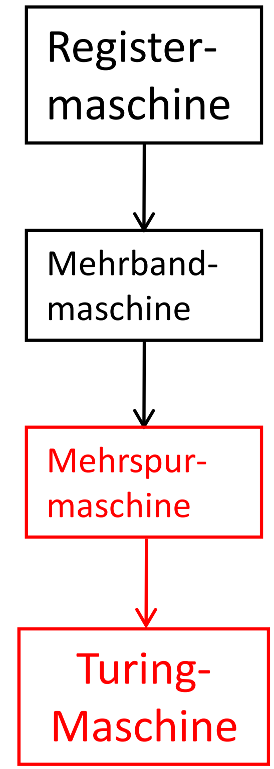


$$\delta(q, a1a2a3) = (q', b1b2b3, R)$$

Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine

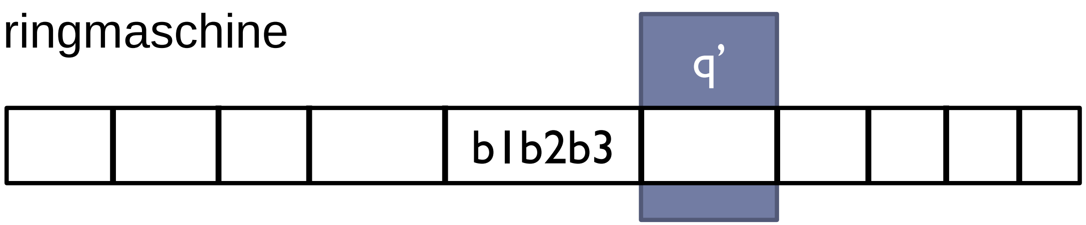
Mehrspurmaschine





$$\delta(q, a1, a2, a3) = (q', b1, b2, b3, R)$$

Einband-Turingmaschine



$$\delta(q, a1a2a3) = (q', b1b2b3, R)$$

Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine

Zeit- und Speicheraufwand:

- Speicherbedarf Mehrspurmaschine: $S''(n)$
- Anzahl Schritte Mehrspurmaschine: $t''(n)$
- Komplexitätsparameter/Eingabegröße: n

- Speicherbedarf $S'''(n)$:

  $S'''(n) = c_1 * S''(n)$ mit $S'(n) \leq c_2 * t''(n)$

  $S'''(n) \leq c_1 * c_2 * t''(n)$

  $S'''(n) \leq c_S * t''(n)$

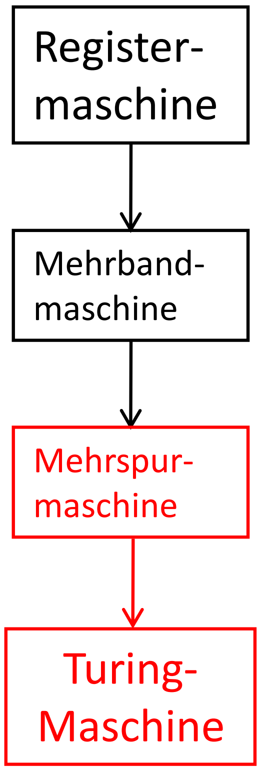
  c -> Konstanten

- Gesamtlaufzeit $t'''(n)$:

  1 Schritt Mehrspurmaschine: 1

  $t''(n) = t'(n) * 1$

  $t''(n) = t'(n)$



Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine

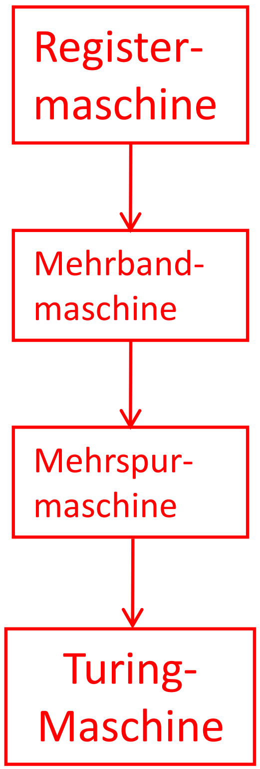
Zeit- und Speicheraufwand insgesamt (O-Kalkül):

- Speicherbedarf:

  $O(S'''(n)) = O(S(n))$

- Gesamtlaufzeit:

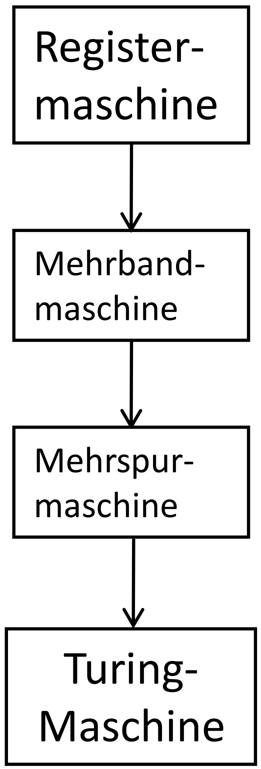
  $O(t'''(n)) = O(t(n)^4)$



Simulation einer Mehrspurmaschine mit einer Einband-Turingmaschine

Fazit - Laufzeit:

Die Registermaschine kann mit polynomiellem Aufwand durch eine Einband-Turingmaschine simuliert werden. Der Speicheraufwand verändert sich dabei sogar nur um den Faktor c.



Korrespondiert mit der erweiterten Churchen These

Diese besagt:

"Vernünftige" Rechnermodelle sind mit maximal polynomiellen Zeitverlust untereinander simulierbar.

Quellen

- Wegener, Ingo: Theoretische Informatik : - eine Algorithmenorientierte Einführung. Berlin: Springer DE, 2005.
- Hoffmann, Dirk W.: Theoretische Informatik. München, Wien: Hanser Verlag, 2009.
- Hromkovic, Juraj: Theoretische Informatik. Berlin: Springer DE, 2010.
- Vocking, Berthold: Skript zur Vorlesung Berechenbarkeit und Komplexität an der RWTH Aachen, Lehrstuhl Informatik 1, Oktober 2009