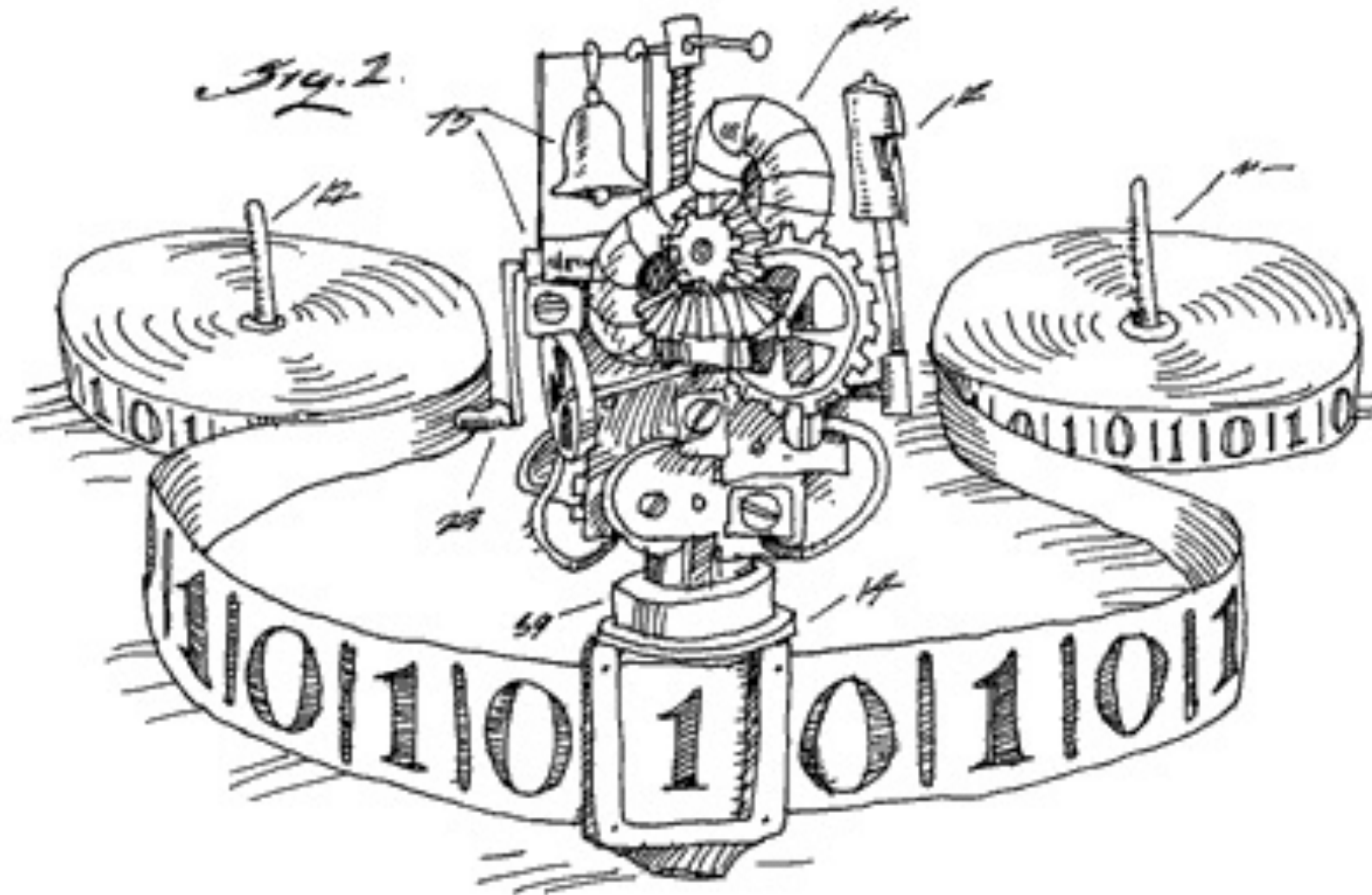


Introduction à l'informatique CM10

Antonio E. Porreca

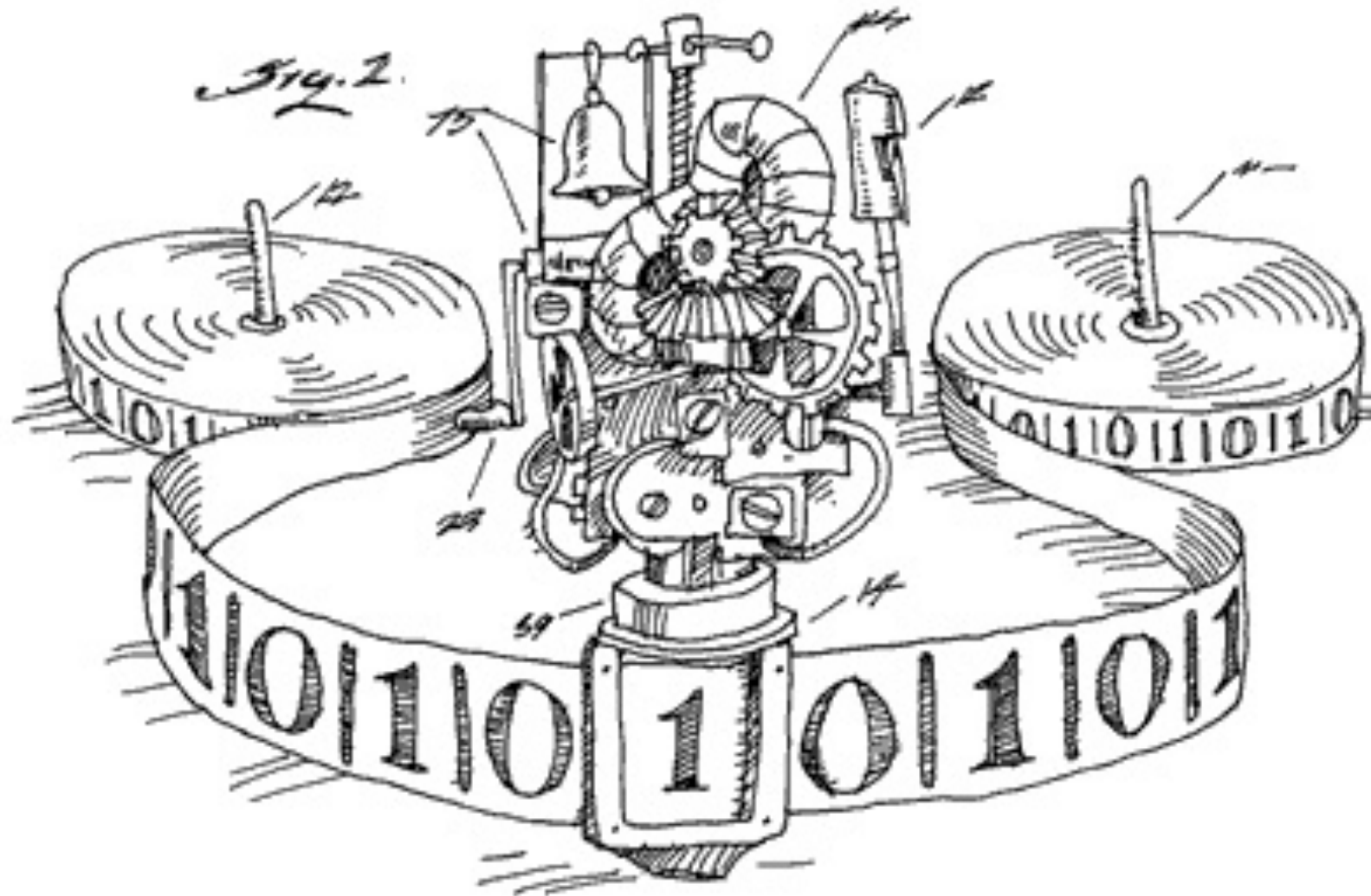
<https://aeporreca.org/teaching>

Machines de Turing



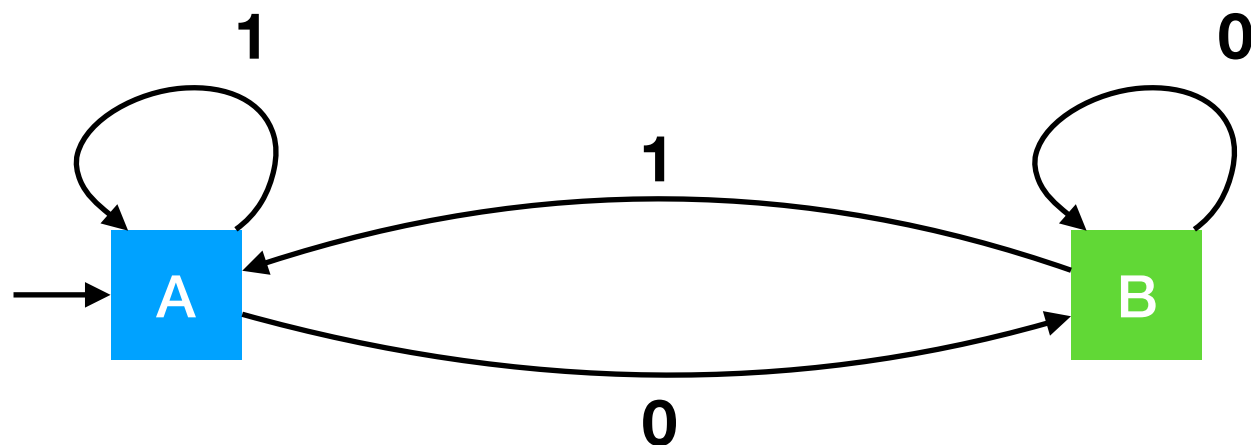
Machines de Turing

Alan Turing (1912-1954)

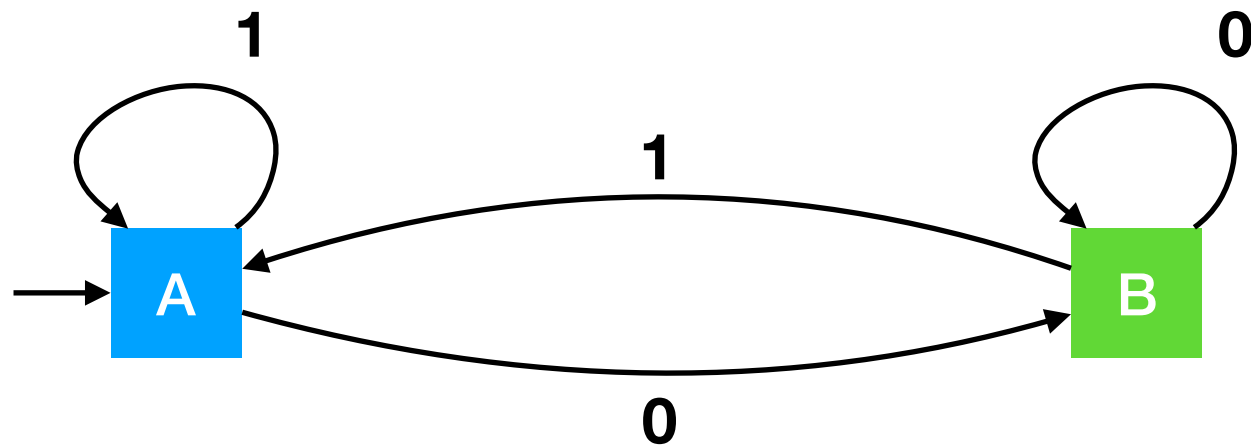
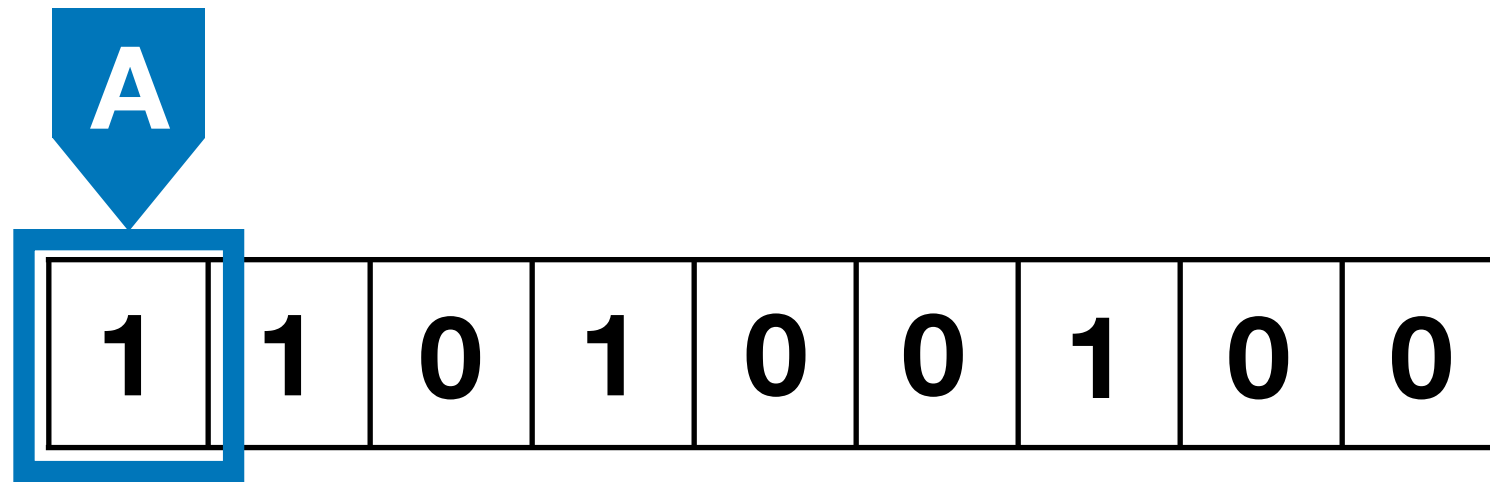


Des automates...

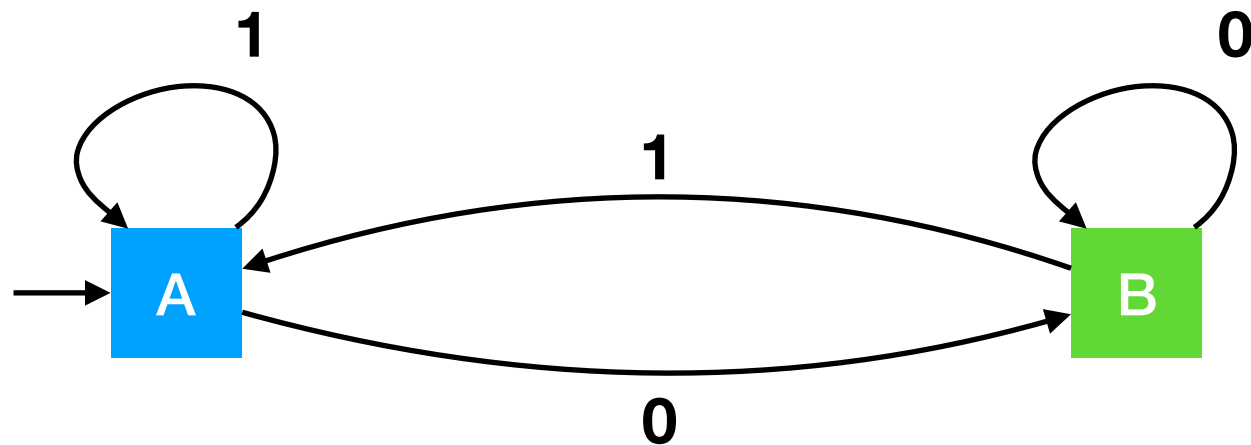
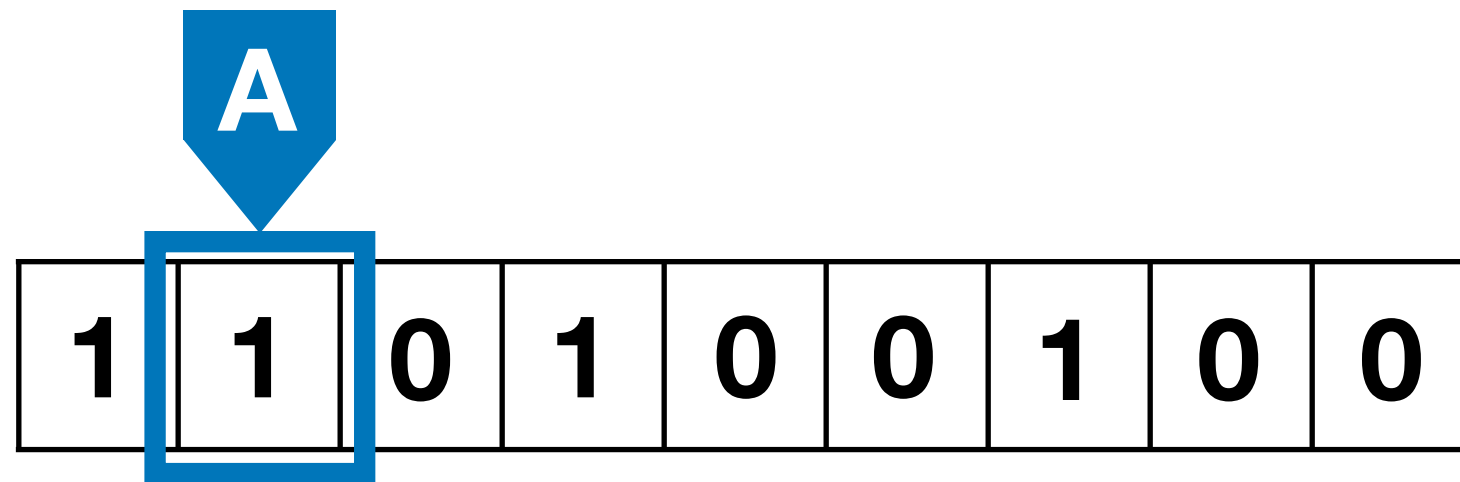
1	1	0	1	0	0	1	0	0
---	---	---	---	---	---	---	---	---



Des automates...

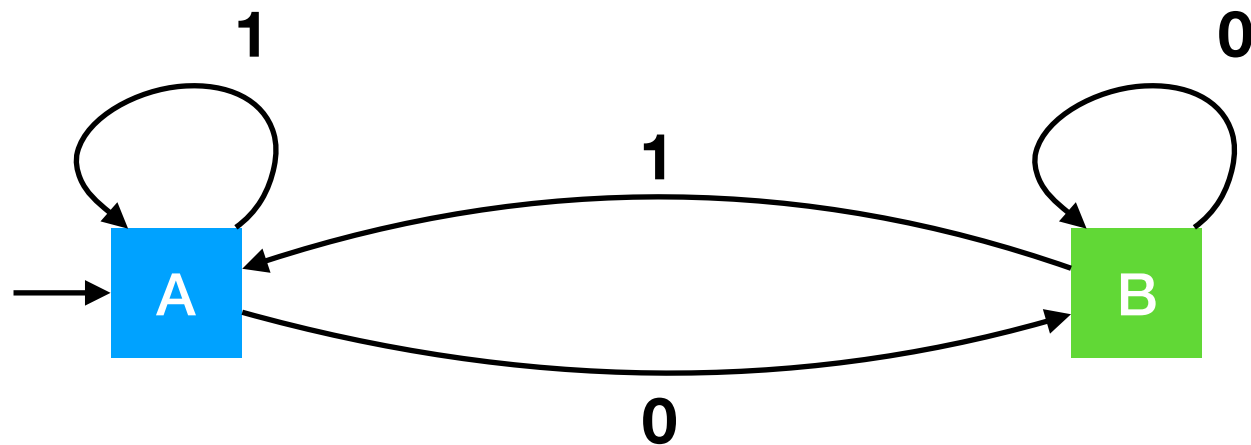
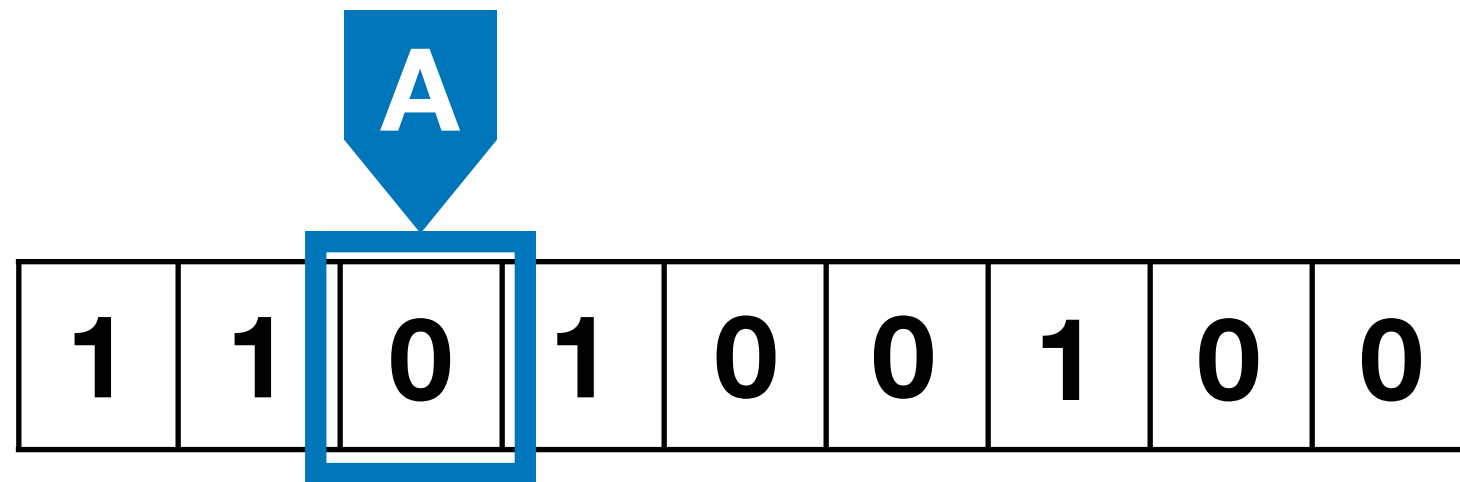


Des automates...



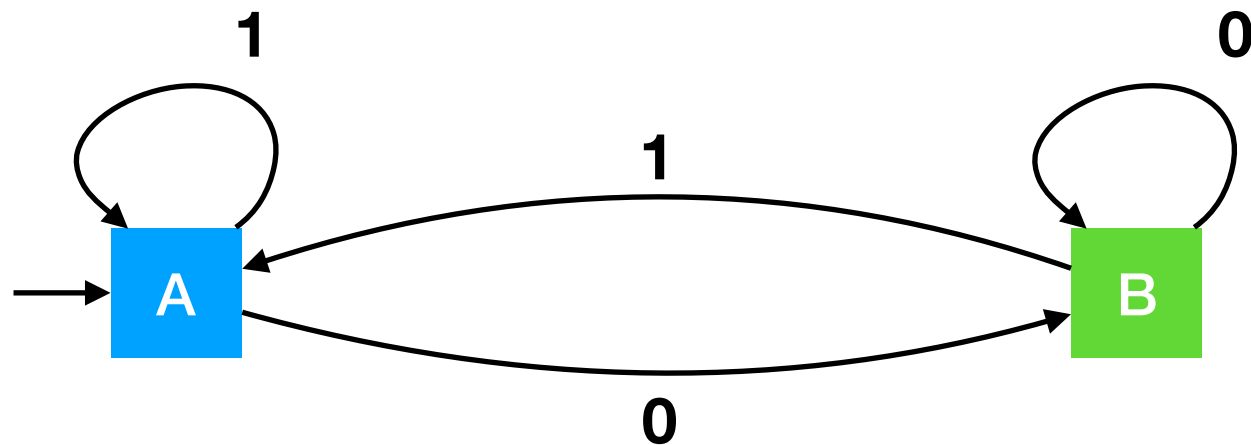
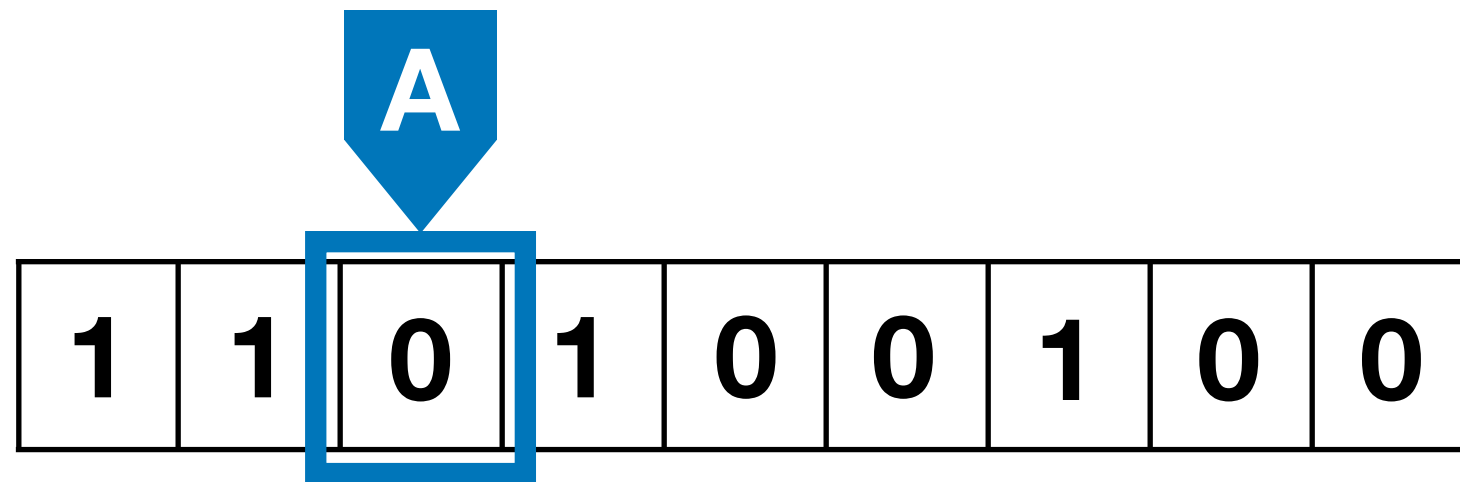
Etat	Symbole	Nouvel état
A	1	A

Des automates...



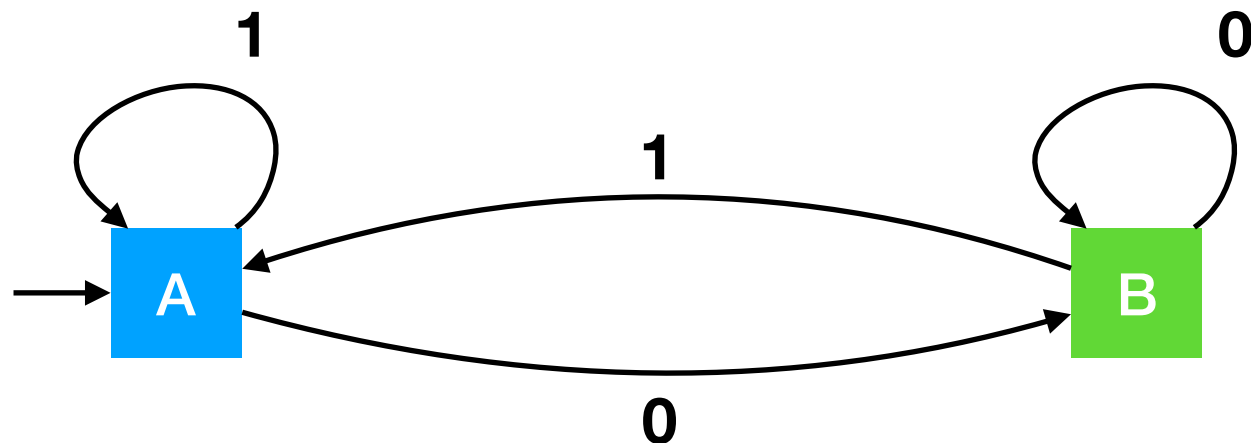
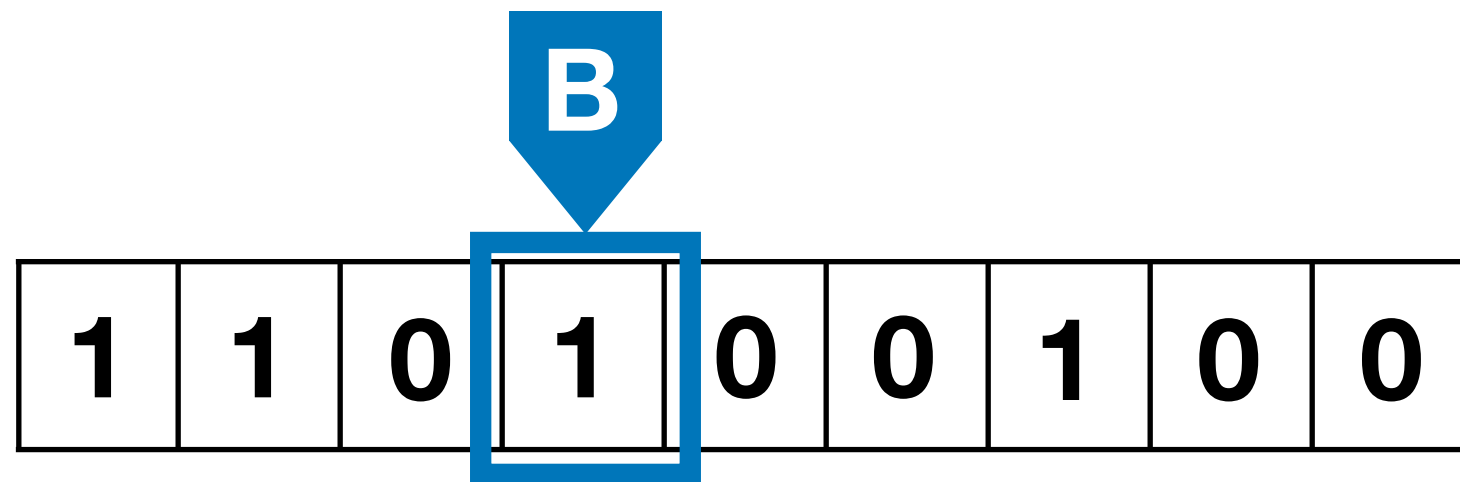
Etat	Symbole	Nouvel état
A	1	A

Des automates...



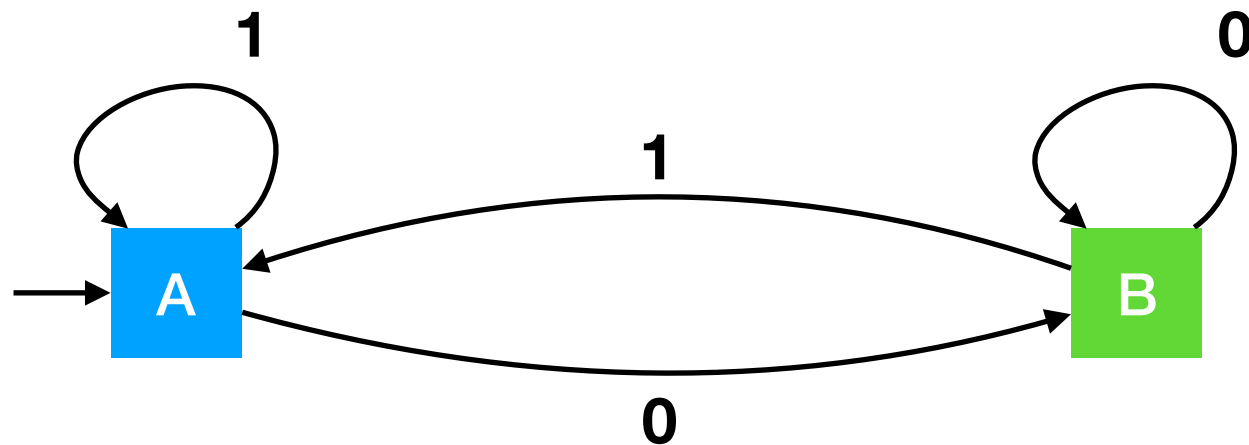
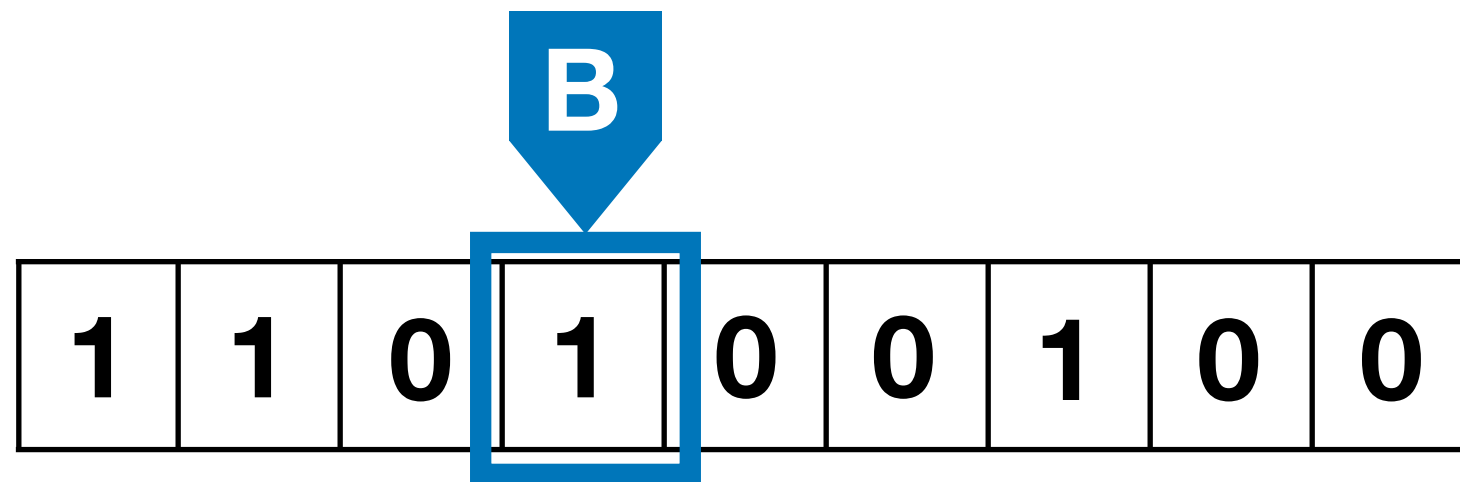
Etat	Symbole	Nouvel état
A	1	A
A	0	B

Des automates...



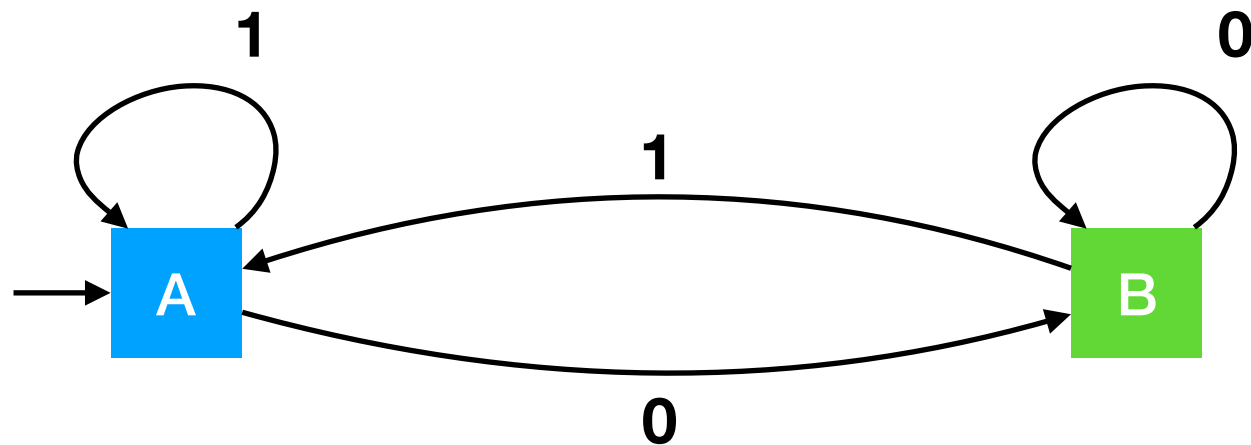
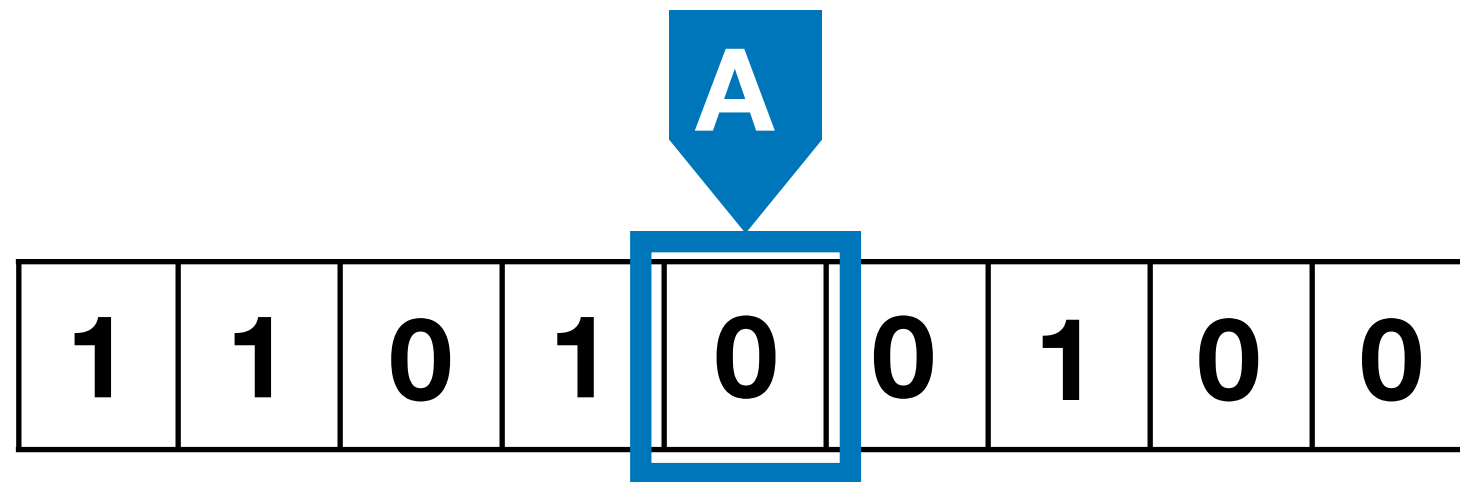
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



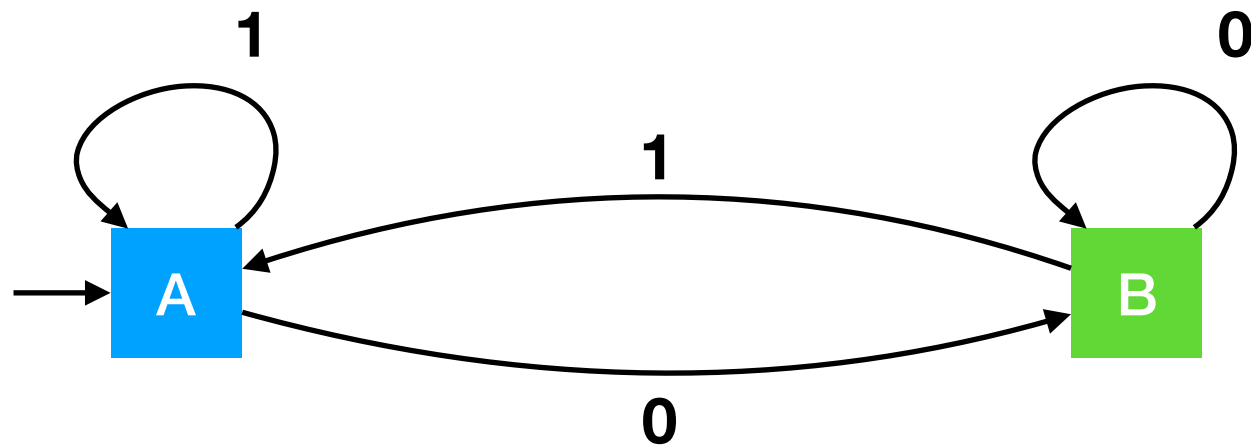
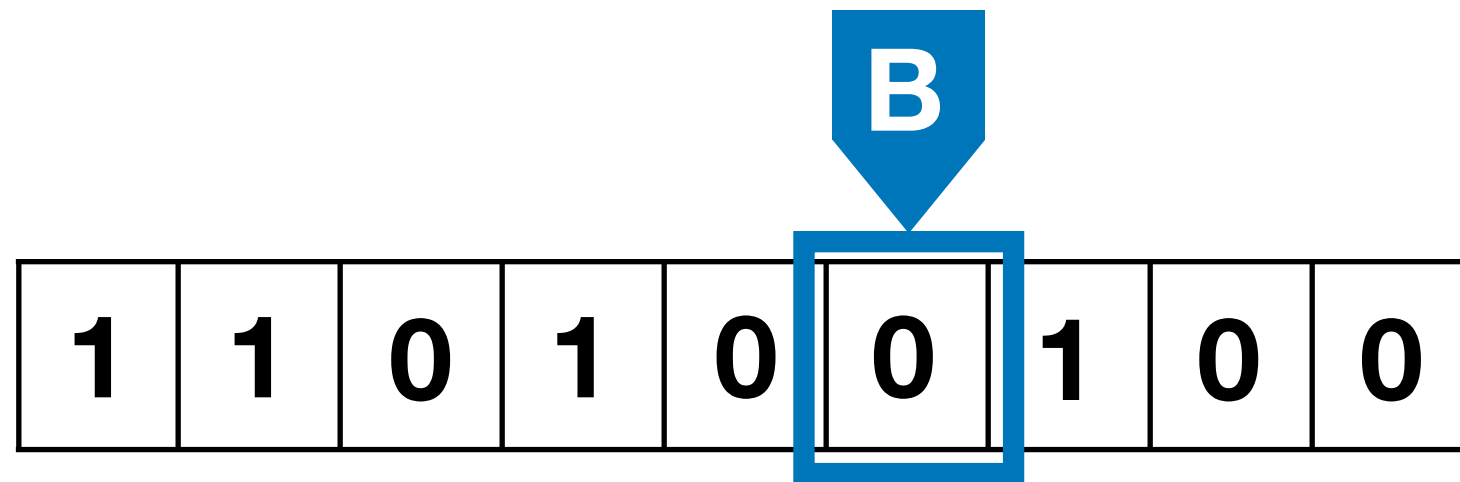
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A

Des automates...



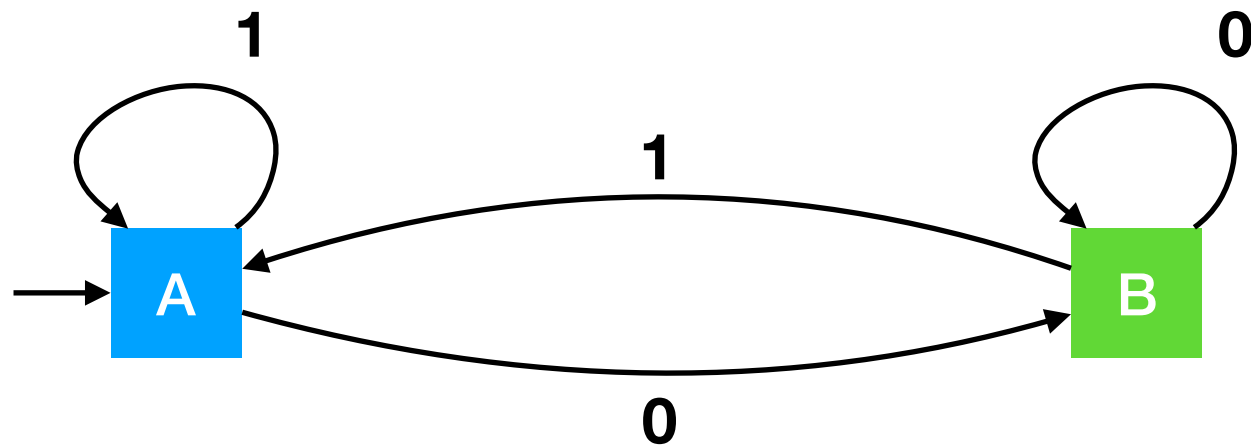
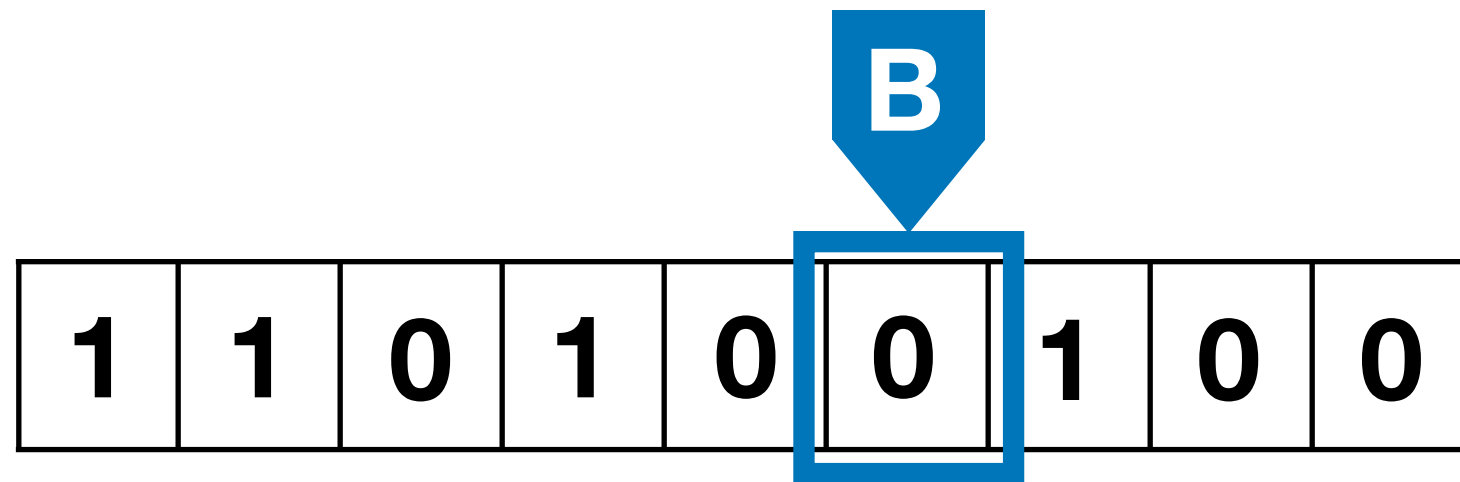
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A

Des automates...



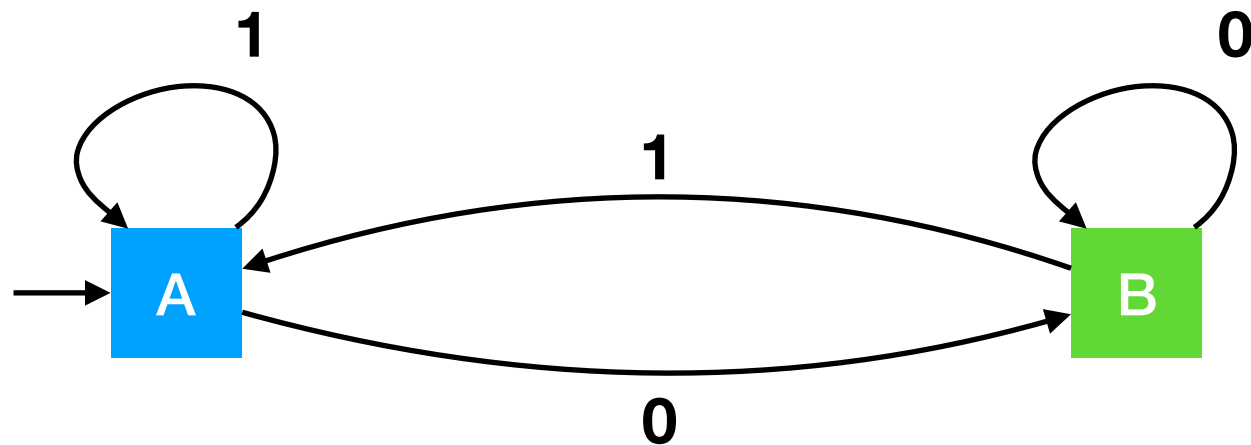
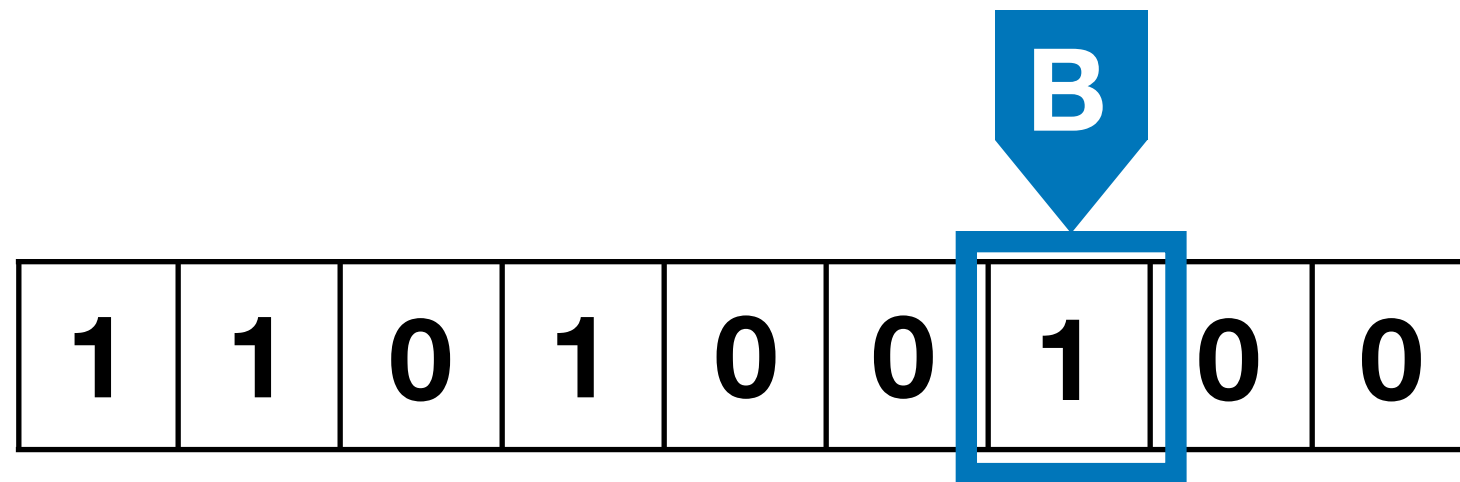
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A

Des automates...



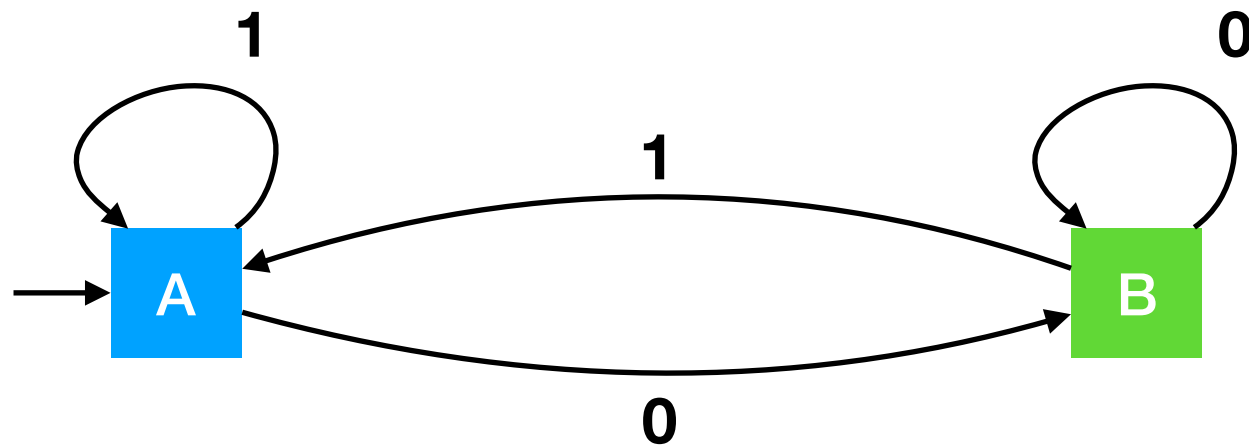
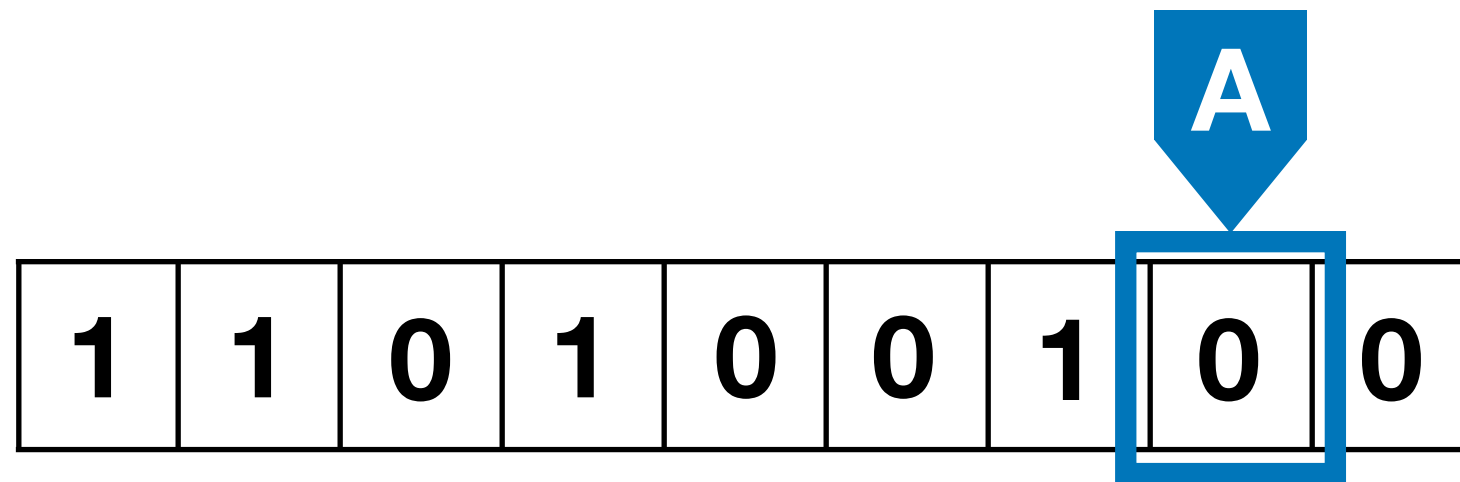
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



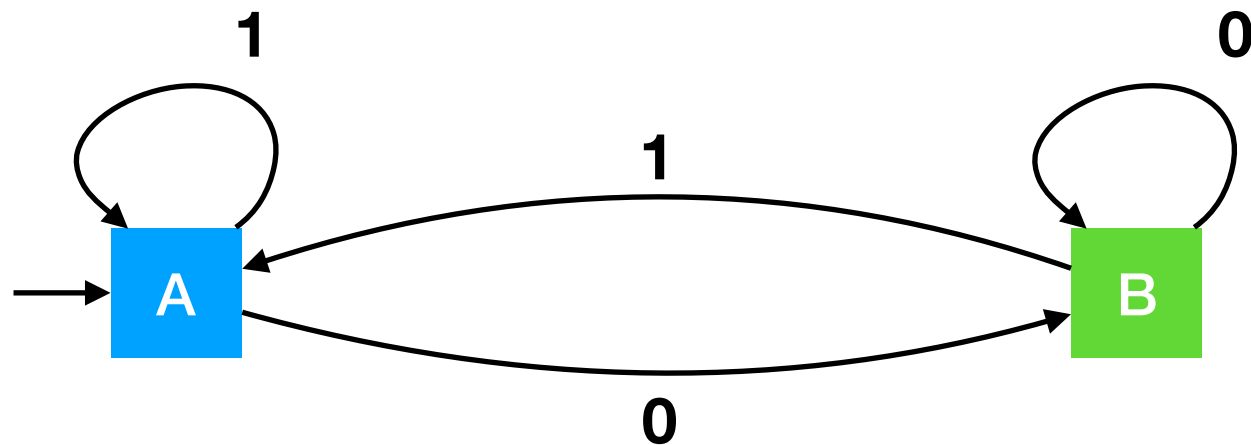
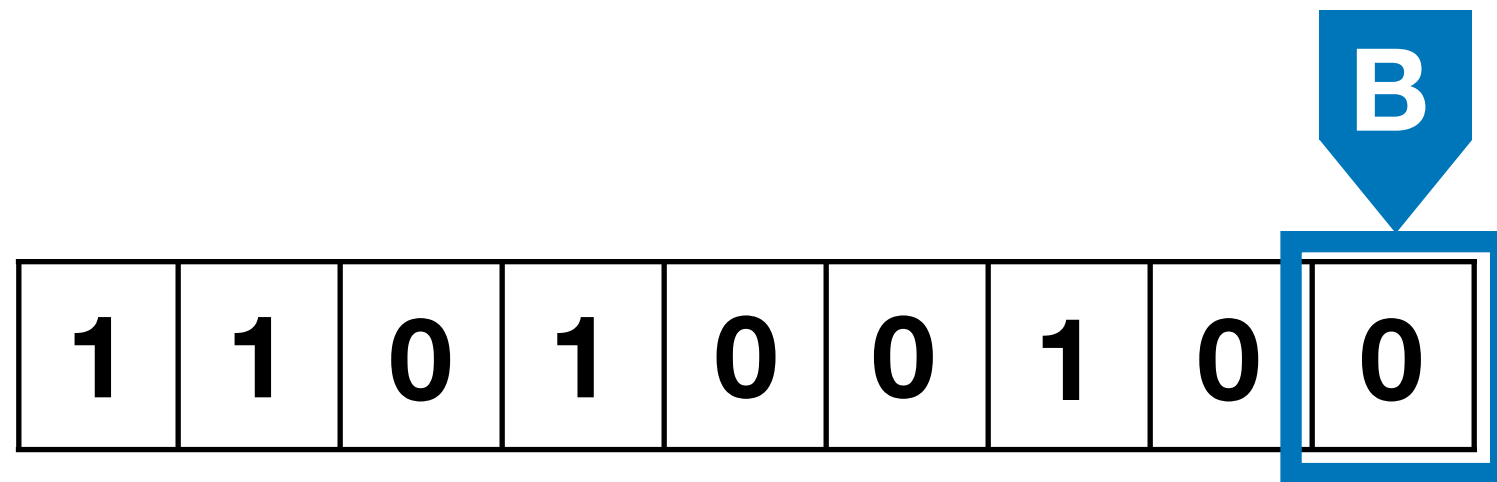
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



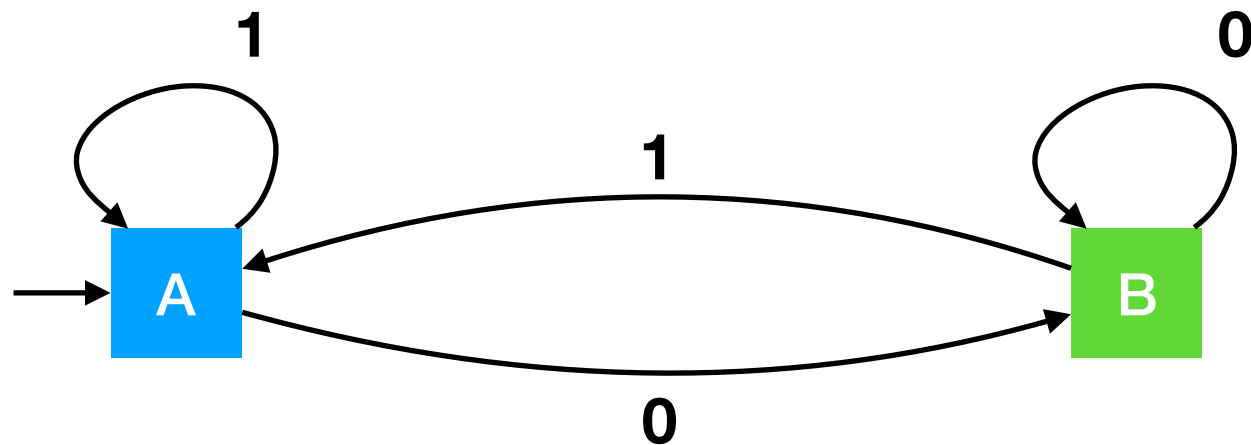
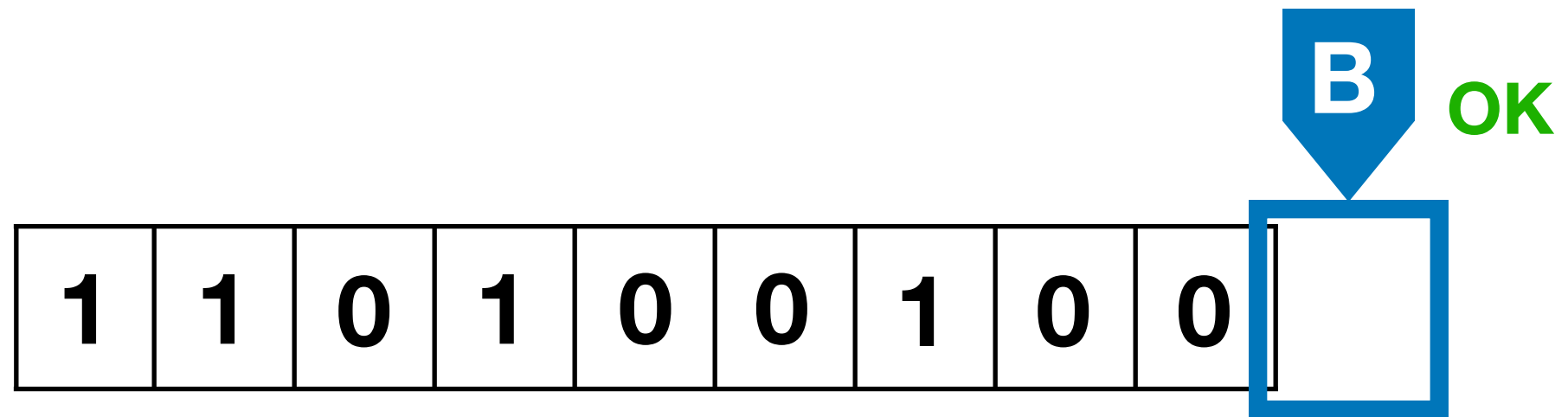
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



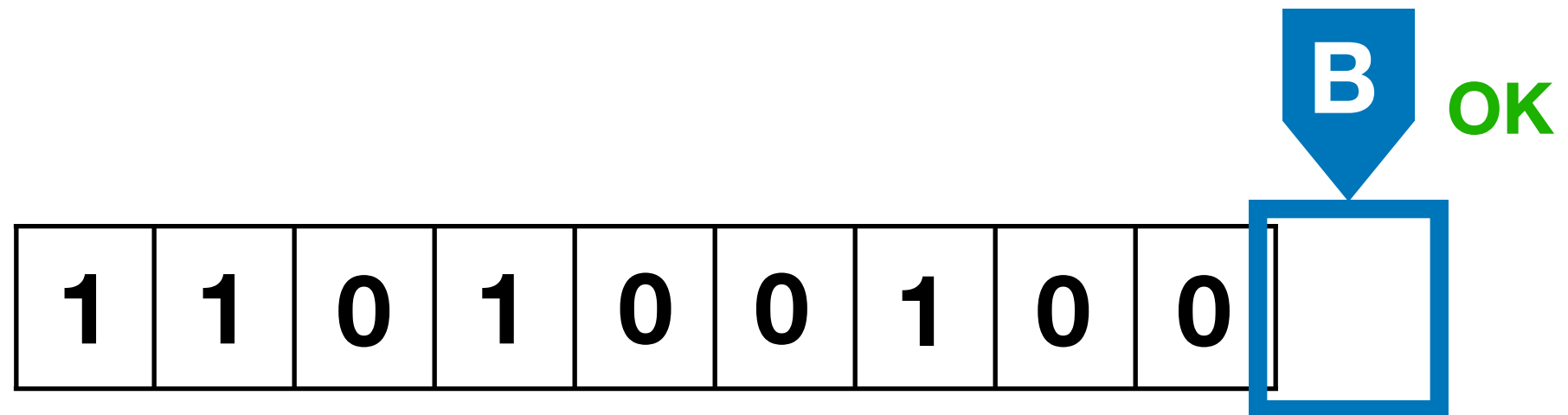
Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

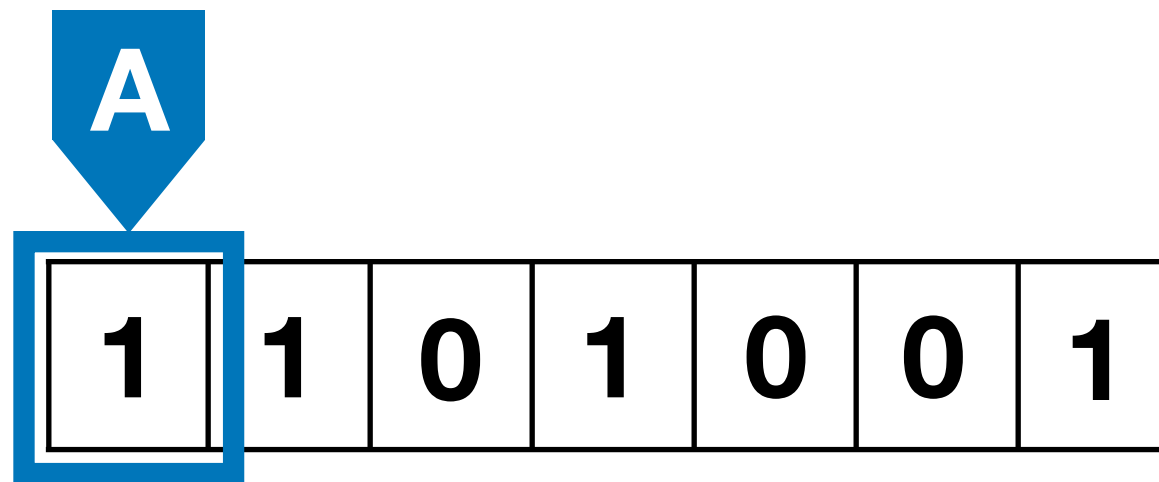
Des automates...

1	1	0	1	0	0	1
---	---	---	---	---	---	---

```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

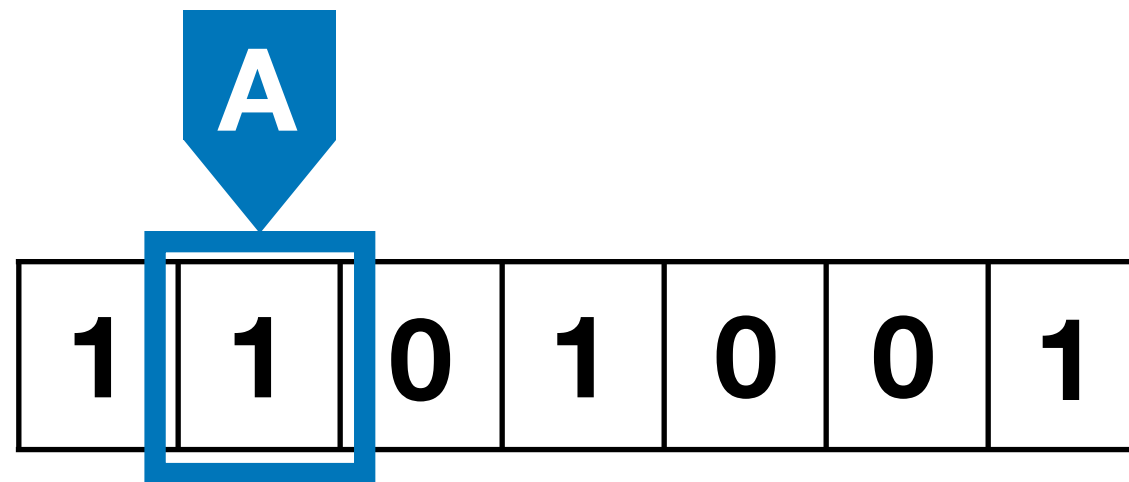
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

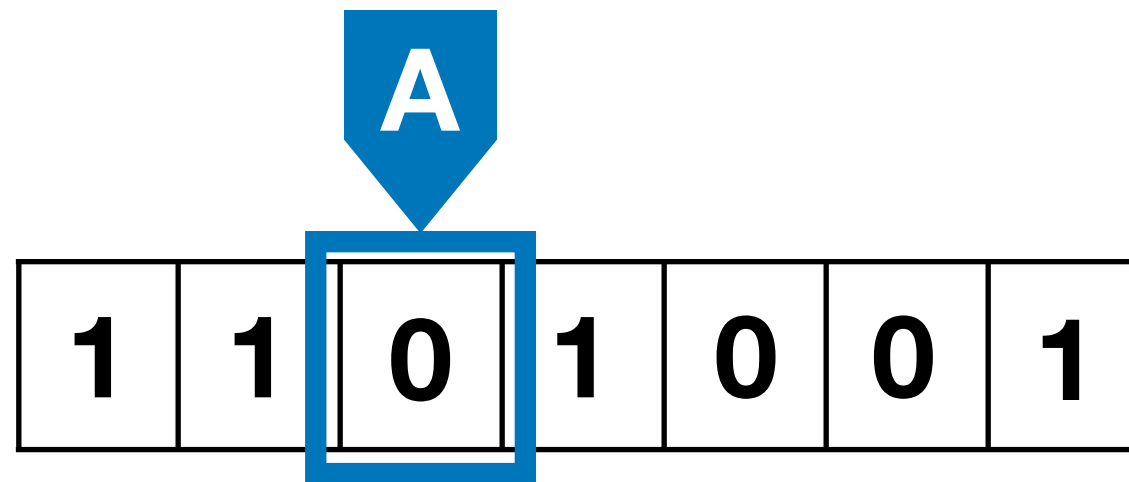
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

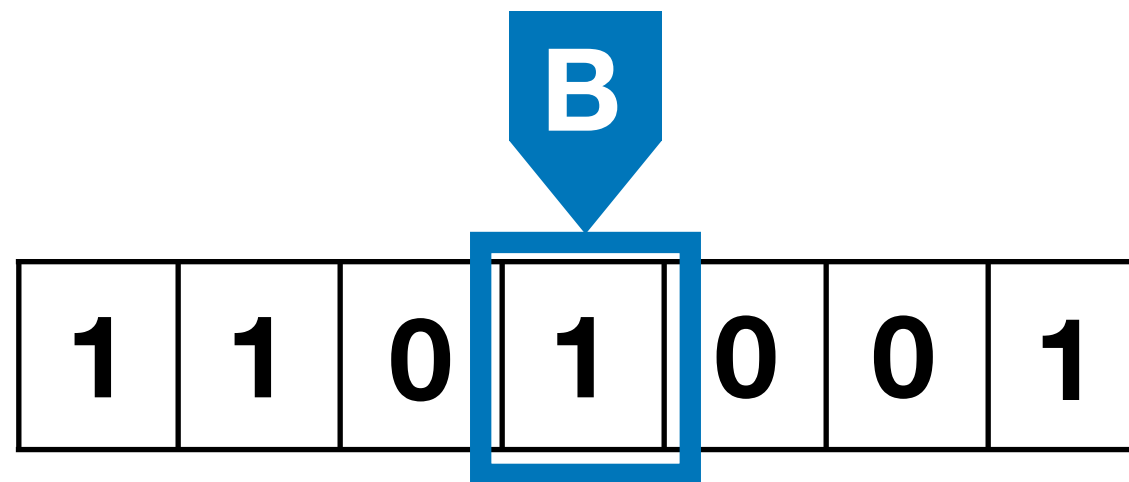
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

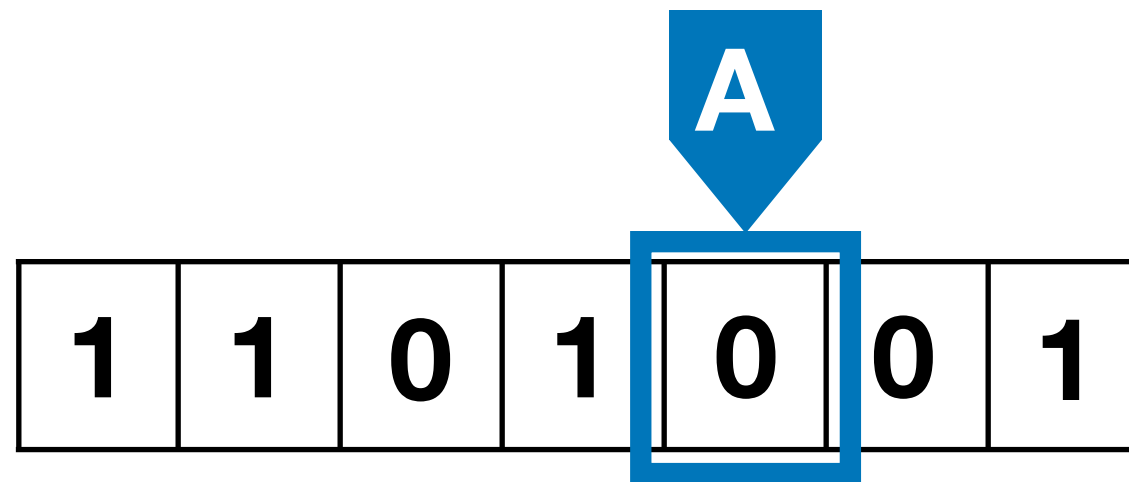
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

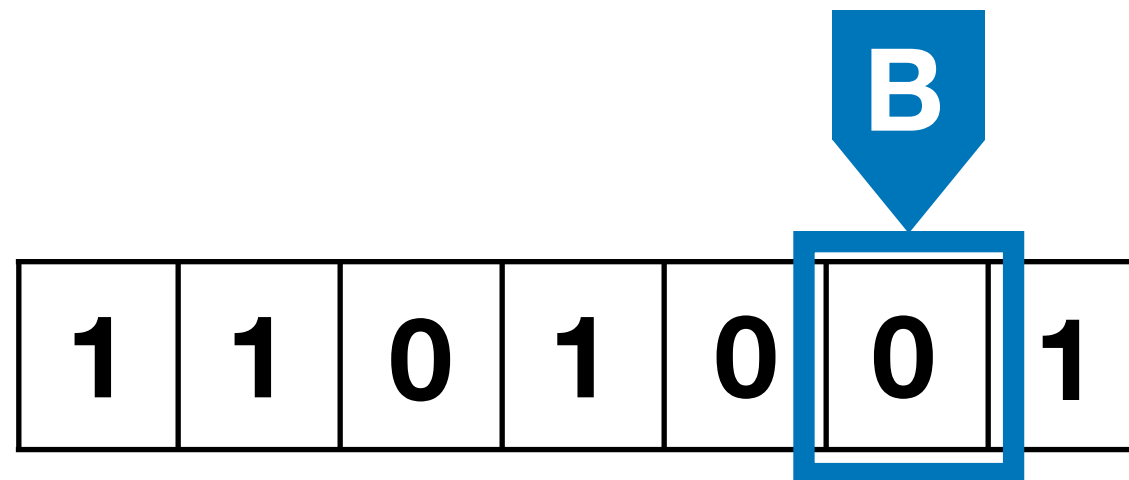
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

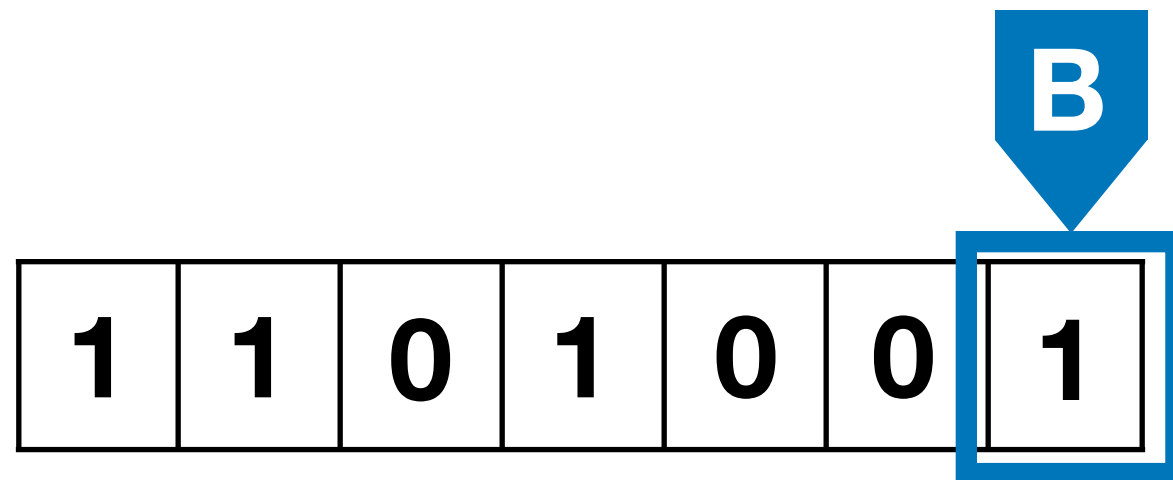
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

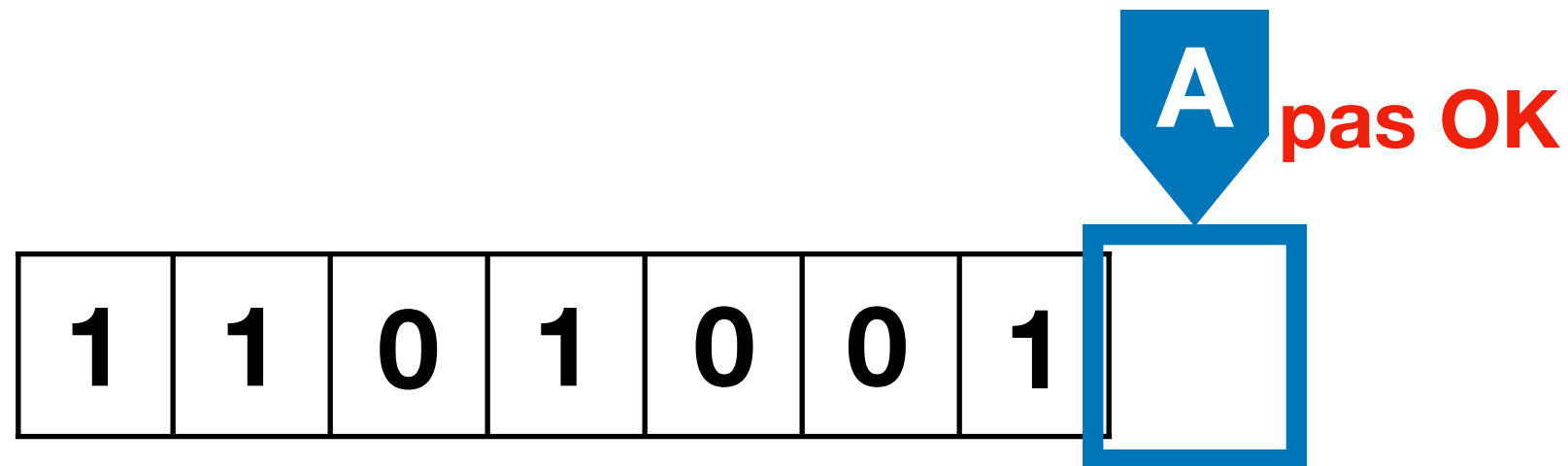
Des automates...



```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

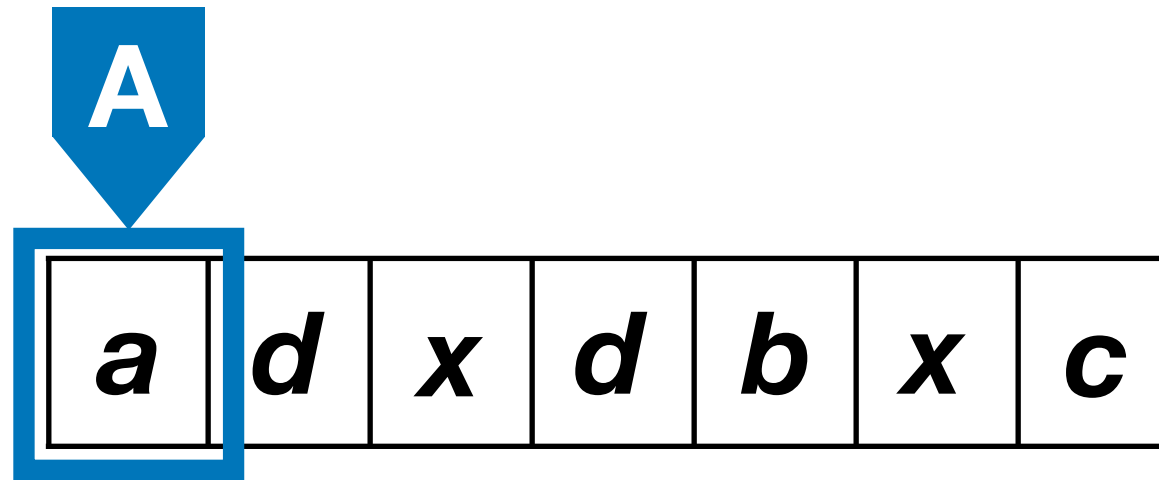
Des automates...



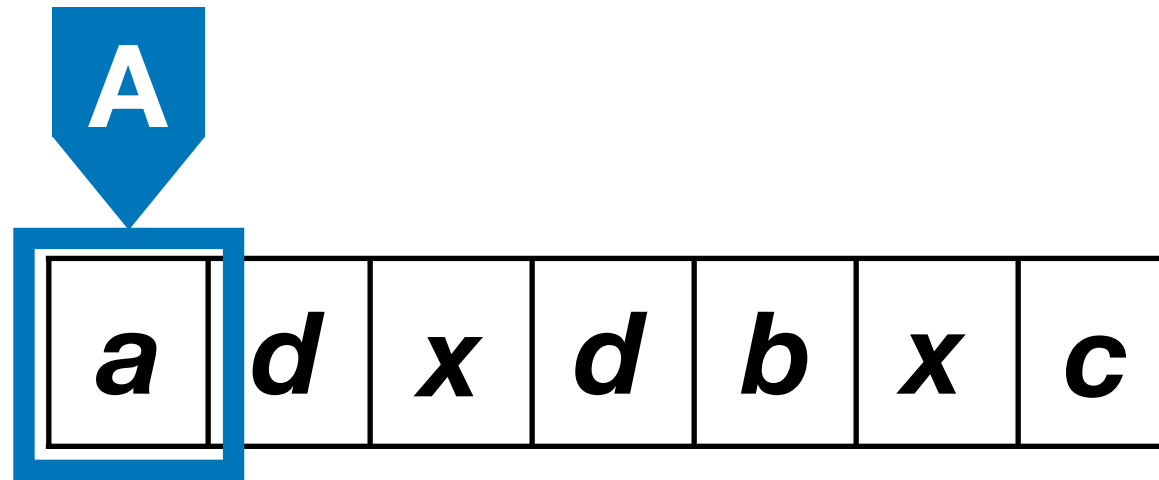
```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

... qui peuvent changer de sens ...

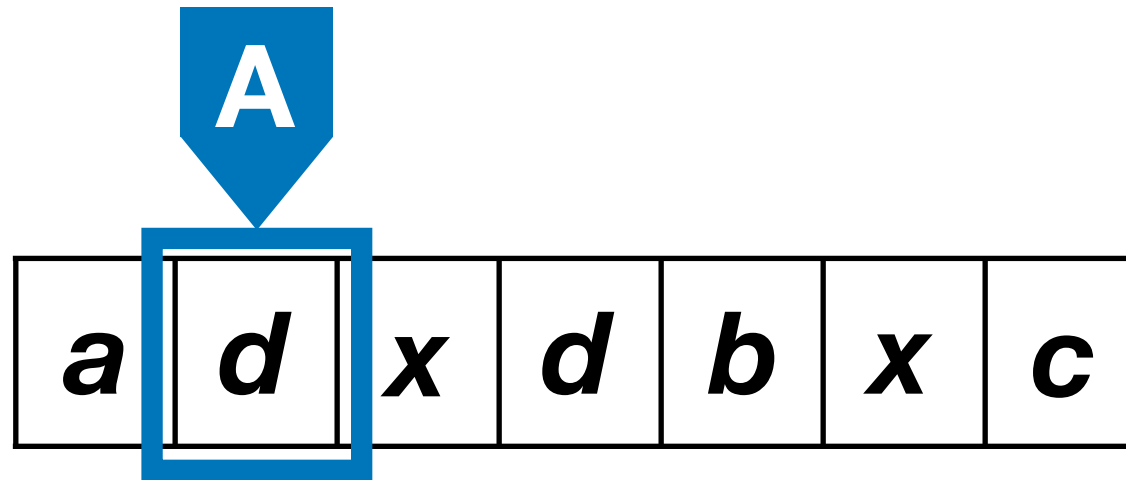


... qui peuvent changer de sens ...



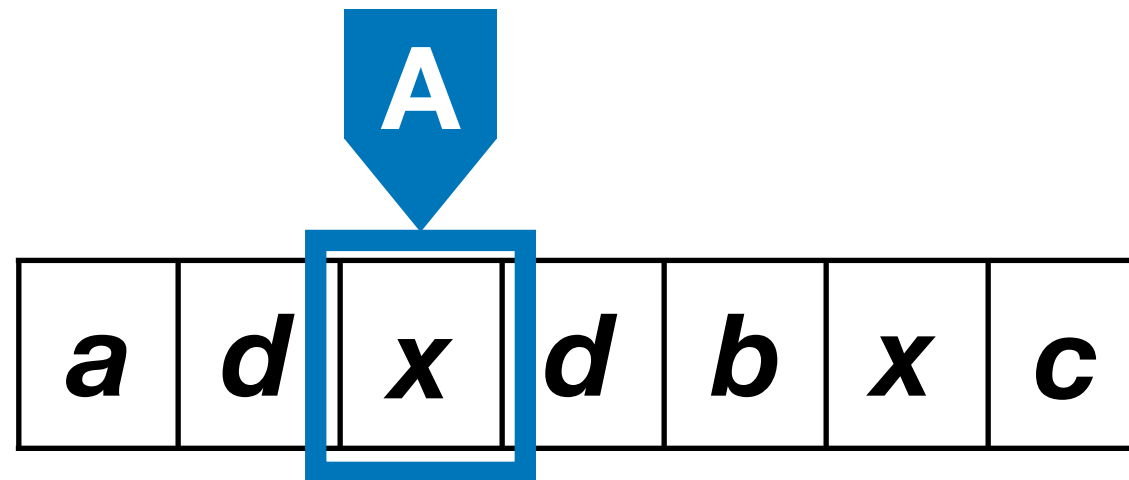
Etat	Symbole	Sens	Nouvel état
<i>A</i>	<i>a,b,c,...,z</i>	→	<i>A</i>

... qui peuvent changer de sens ...



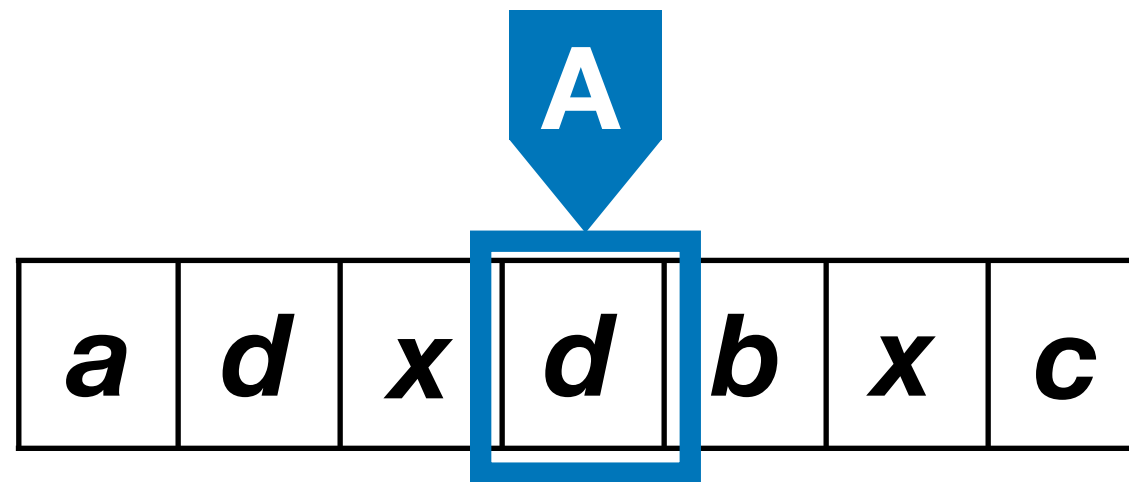
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A

... qui peuvent changer de sens ...



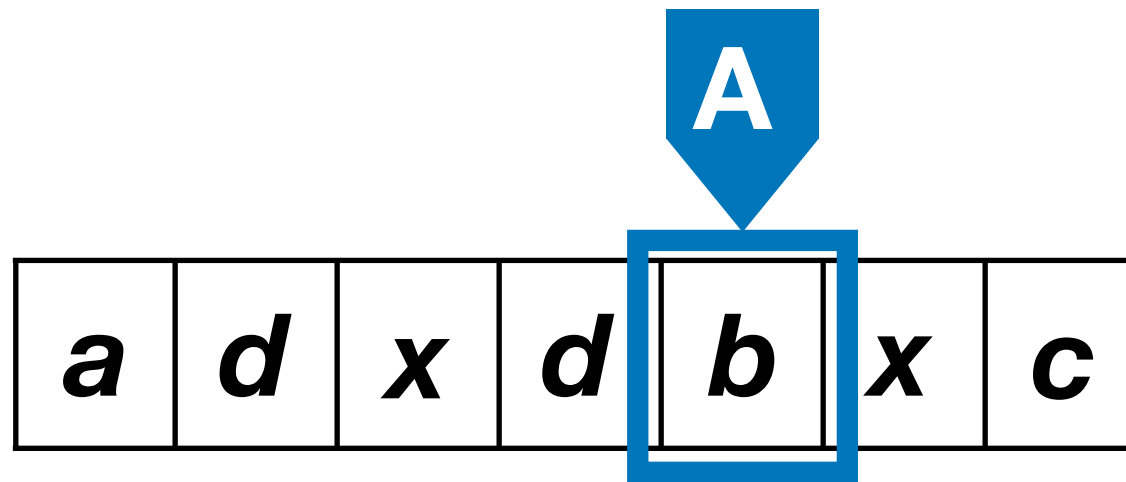
Etat	Symbole	Sens	Nouvel état
<i>A</i>	<i>a,b,c,...,z</i>	→	<i>A</i>

... qui peuvent changer de sens ...



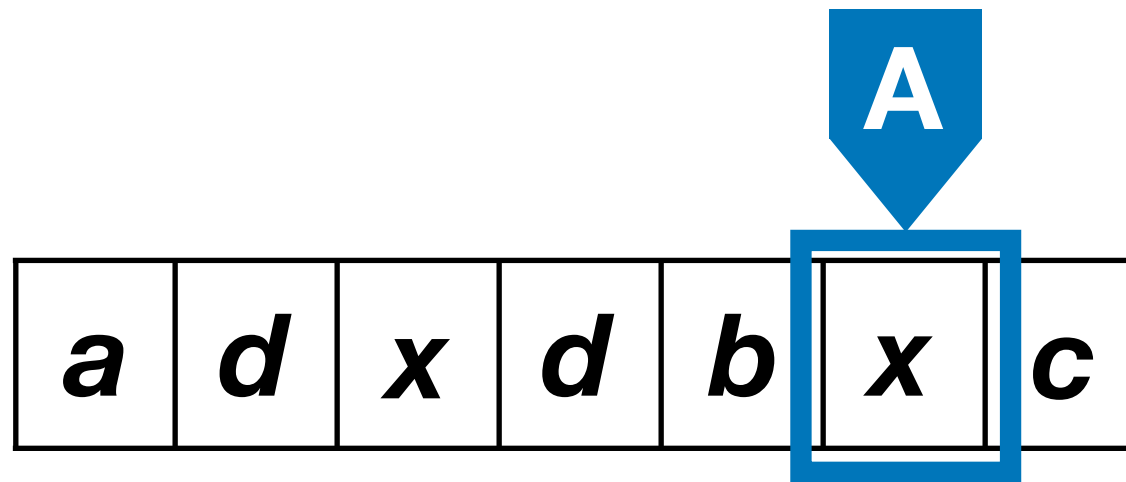
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A

... qui peuvent changer de sens ...



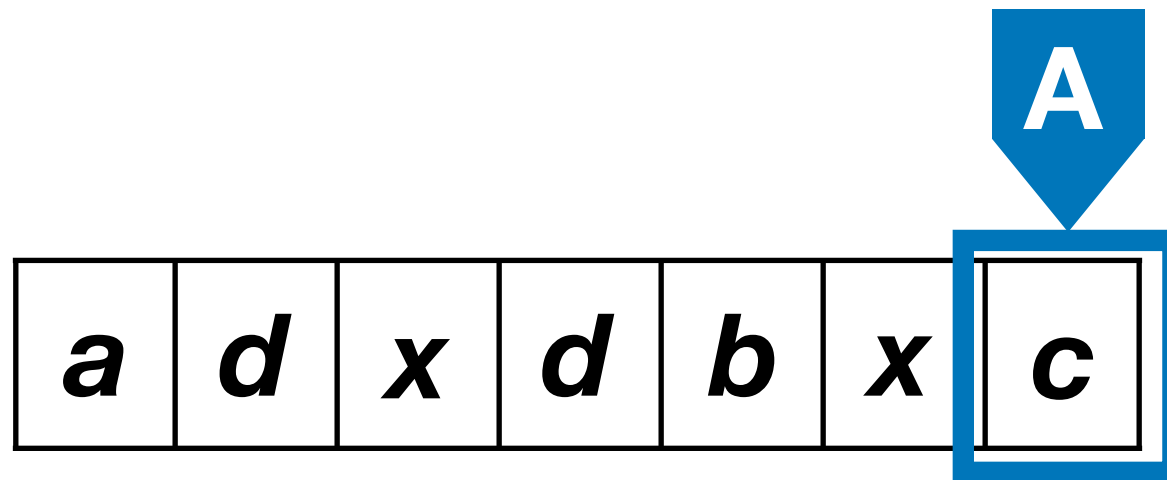
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A

... qui peuvent changer de sens ...



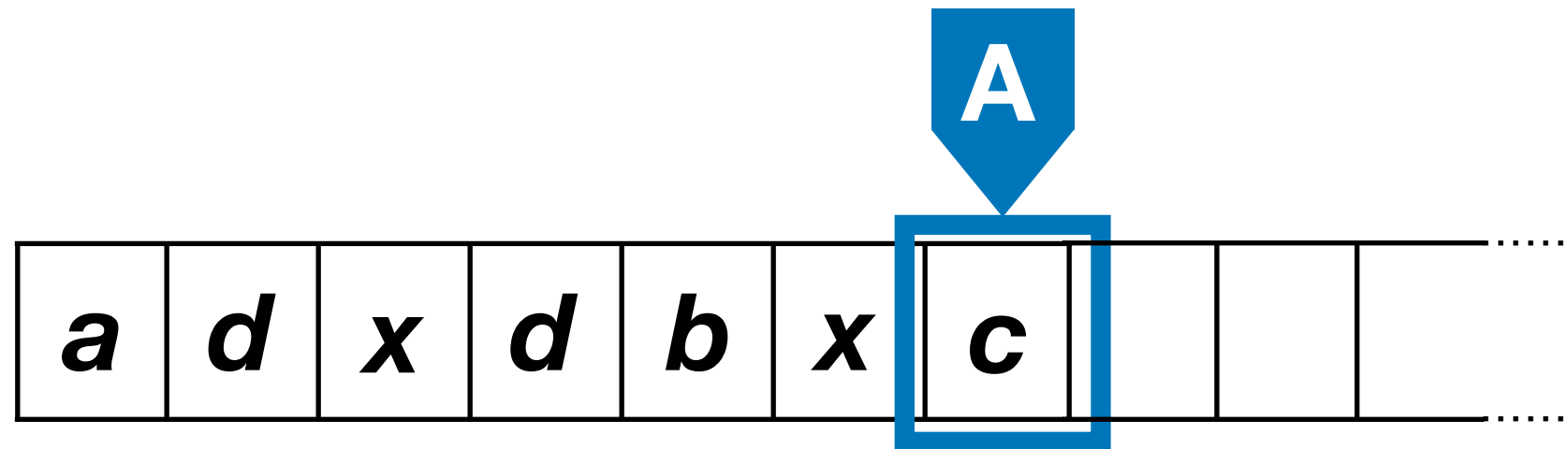
Etat	Symbole	Sens	Nouvel état
<i>A</i>	<i>a,b,c,...,z</i>	→	<i>A</i>

... qui peuvent changer de sens ...



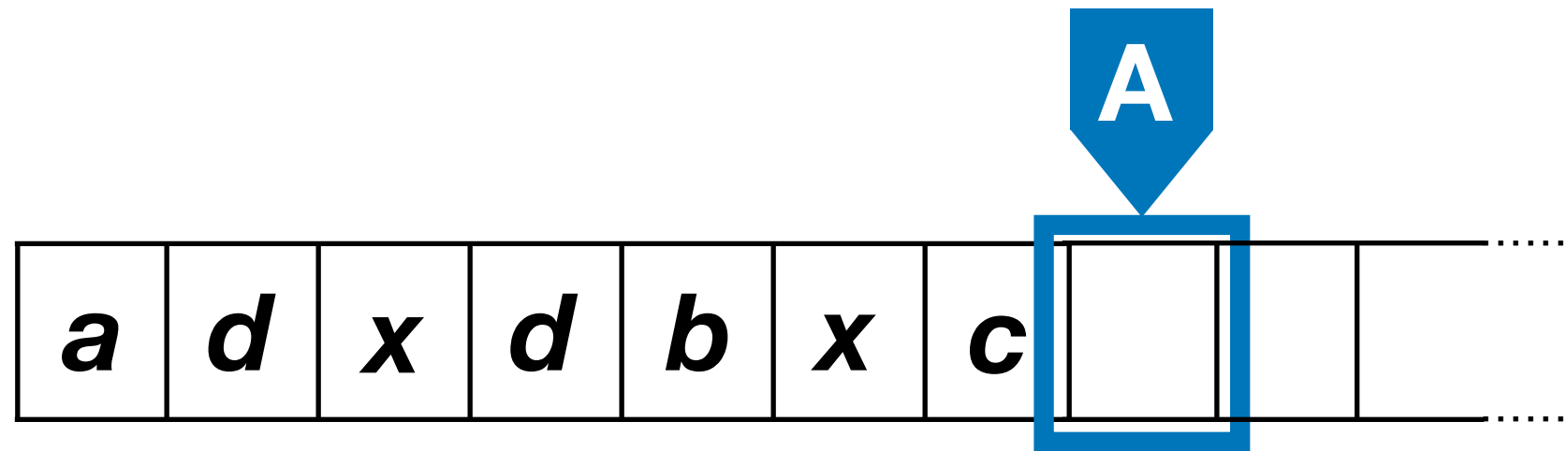
Etat	Symbole	Sens	Nouvel état
<i>A</i>	<i>a,b,c,...,z</i>	→	<i>A</i>

... qui peuvent changer de sens ...



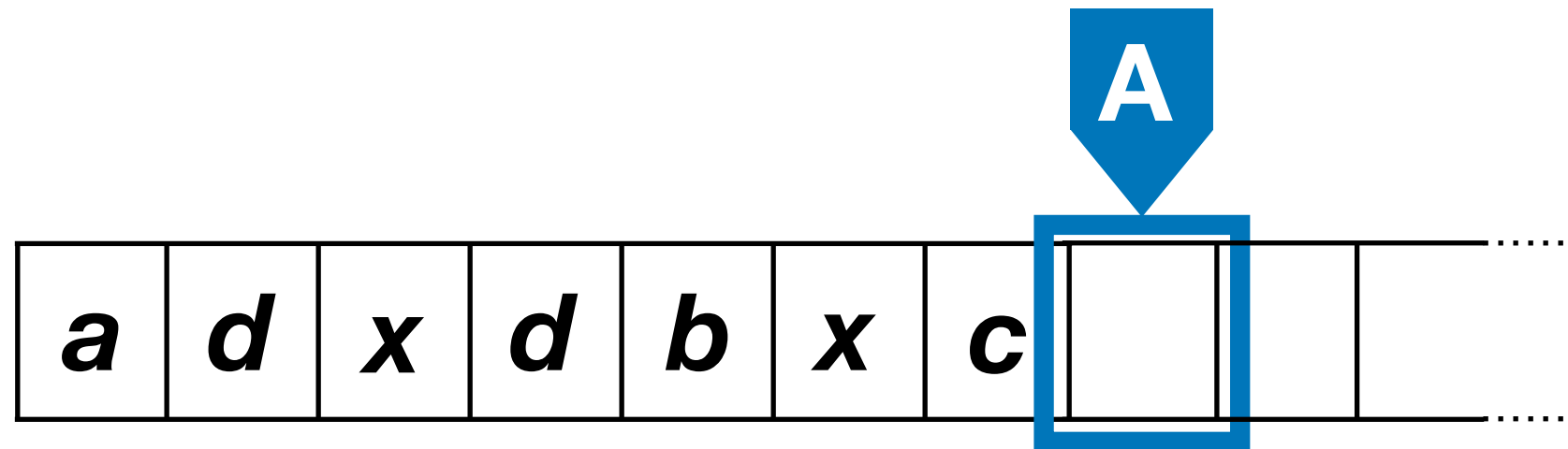
Etat	Symbole	Sens	Nouvel état
<i>A</i>	<i>a,b,c,...,z</i>	→	<i>A</i>

... qui peuvent changer de sens ...



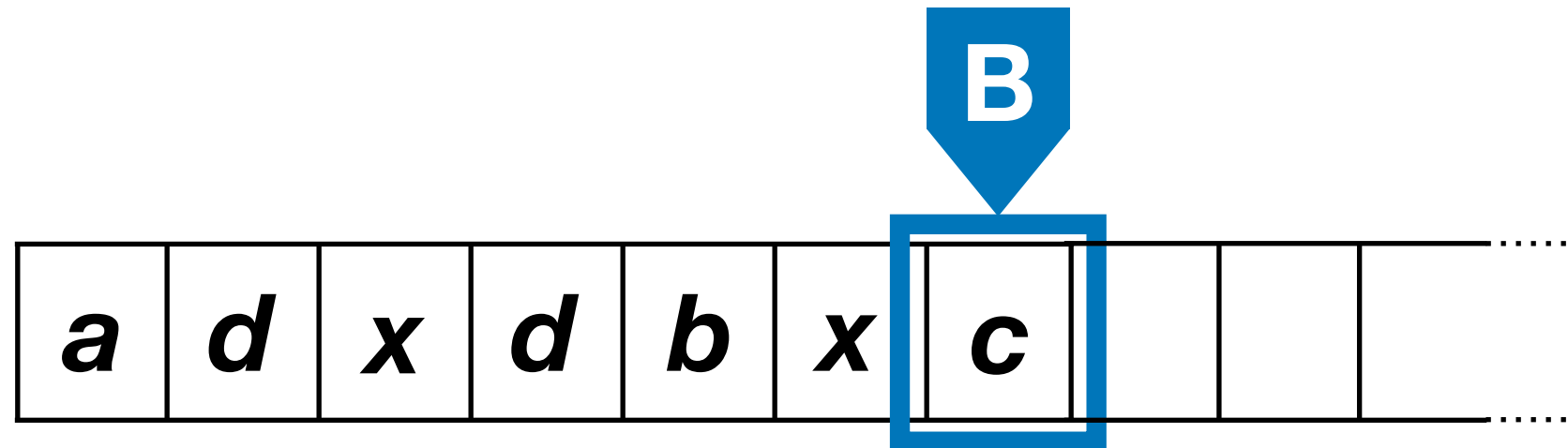
Etat	Symbole	Sens	Nouvel état
A	<i>a, b, c, ..., z</i>	→	A

... qui peuvent changer de sens ...



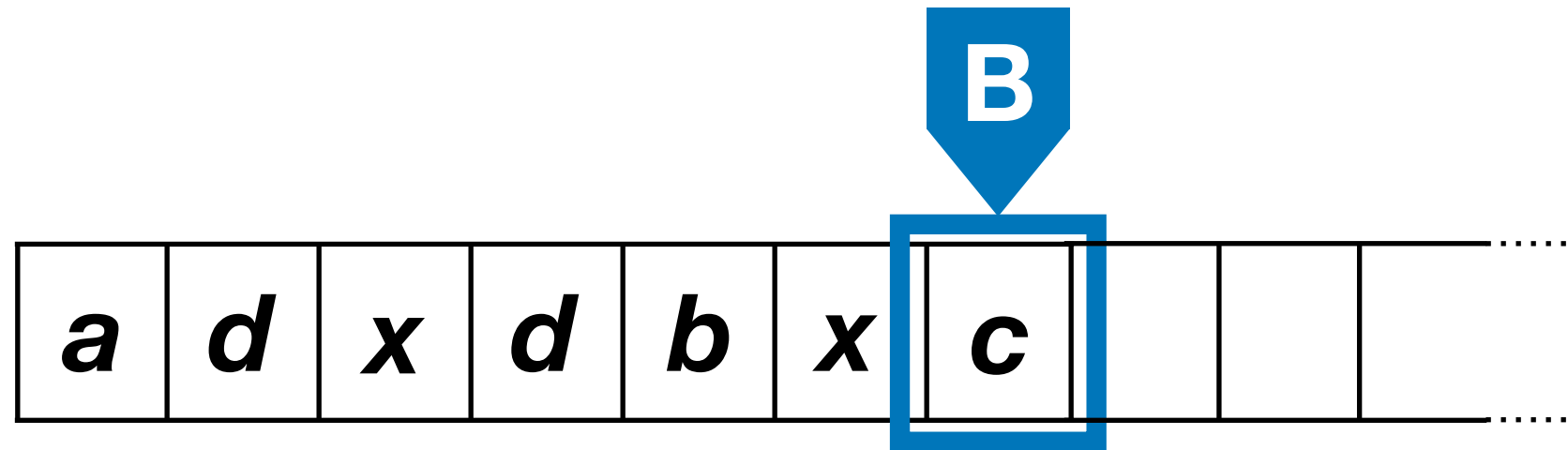
Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B

... qui peuvent changer de sens ...



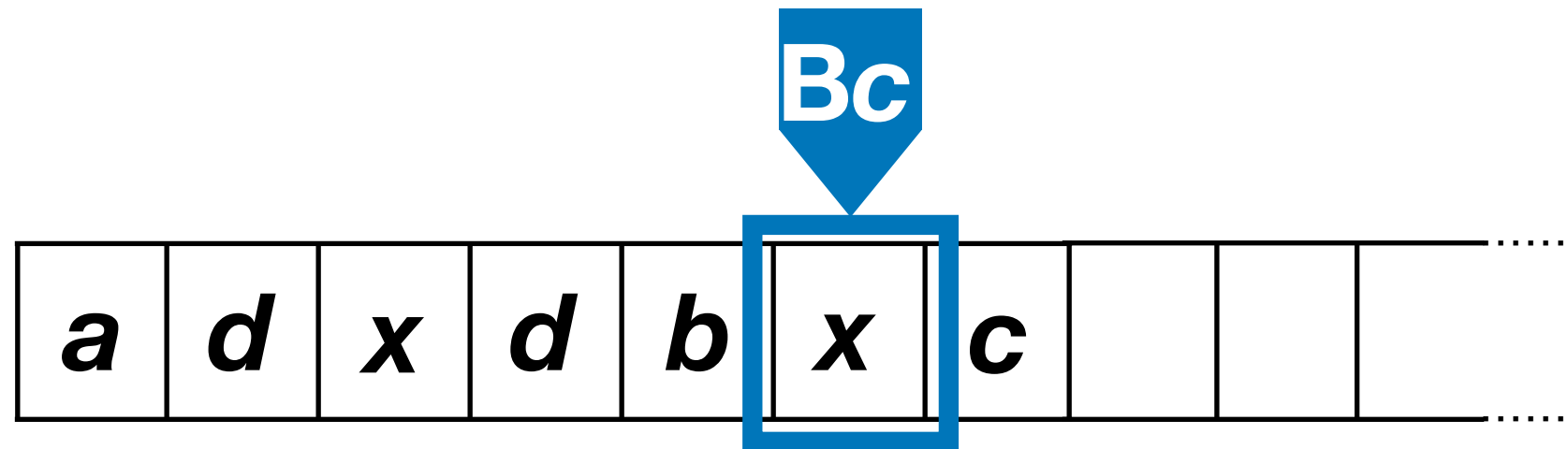
Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B

... qui peuvent changer de sens ...



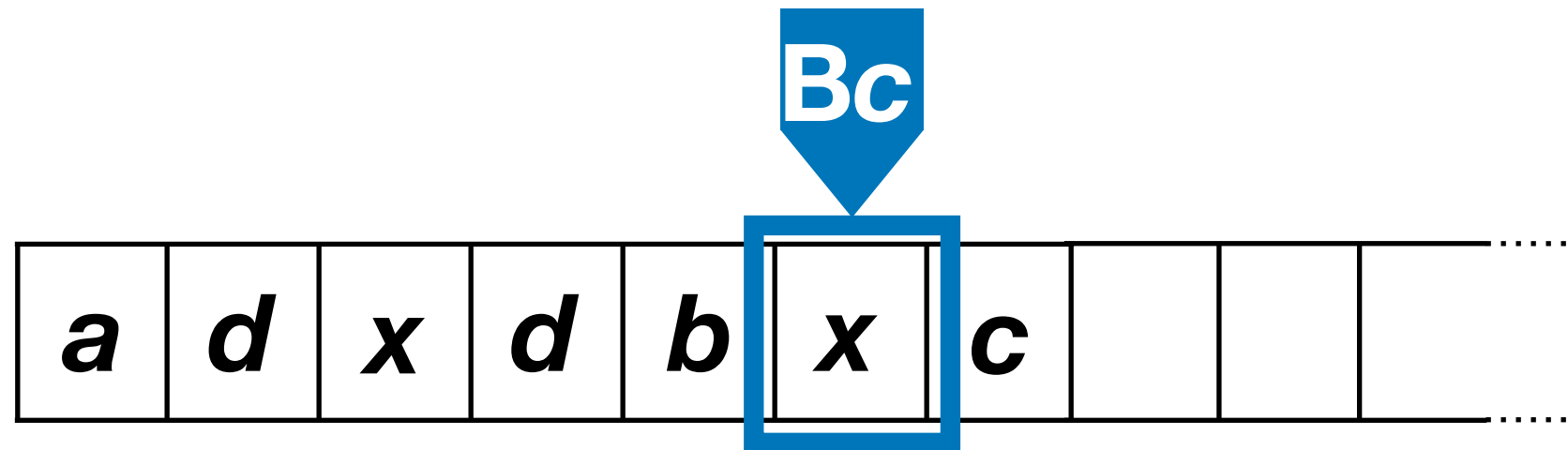
Etat	Symbole	Sens	Nouvel état
A	<i>a, b, c, ..., z</i>	→	A
A		←	B
B	<i>a</i>	←	B <i>a</i>
B	<i>b</i>	←	B <i>b</i>
B	<i>c</i>	←	B <i>c</i>
...
B	<i>z</i>	←	B <i>z</i>

... qui peuvent changer de sens ...



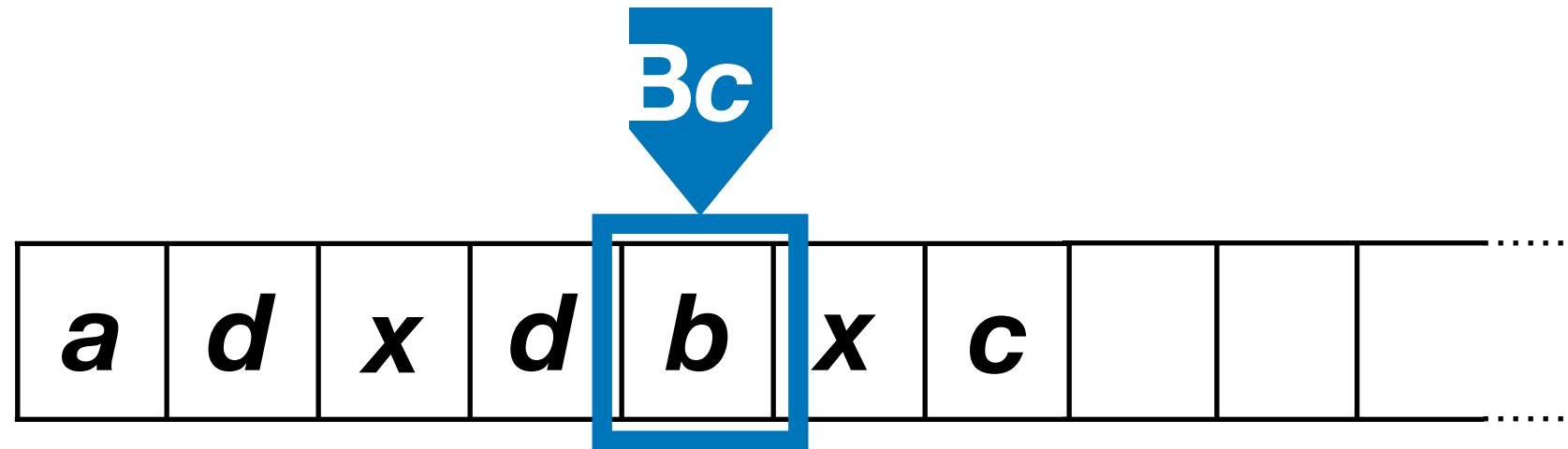
Etat	Symbole	Sens	Nouvel état
A	<i>a, b, c, ..., z</i>	→	A
A		←	B
B	<i>a</i>	←	Ba
B	<i>b</i>	←	Bb
B	<i>c</i>	←	Bc
...
B	<i>z</i>	←	Bz

... qui peuvent changer de sens ...



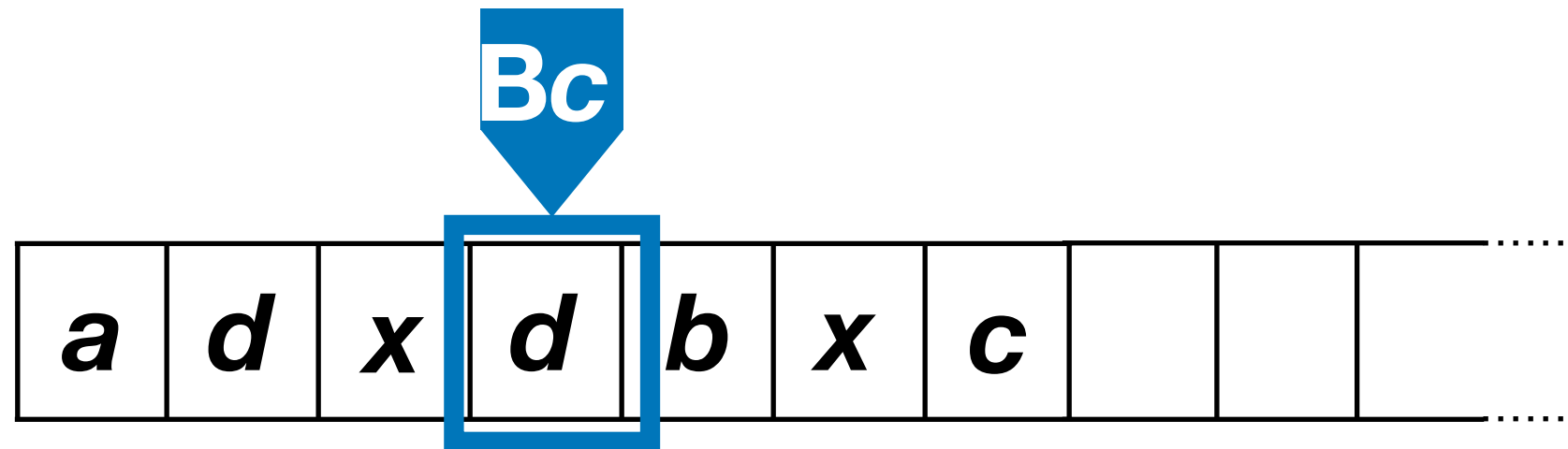
Etat	Symbole	Sens	Nouvel état
A	<i>a, b, c, ..., z</i>	→	A
A		←	B
B	<i>a</i>	←	Ba
B	<i>b</i>	←	Bb
B	<i>c</i>	←	Bc
...
B	<i>z</i>	←	Bz
Bc	<i>a, b, d, ..., z</i>	←	Bc

... qui peuvent changer de sens ...



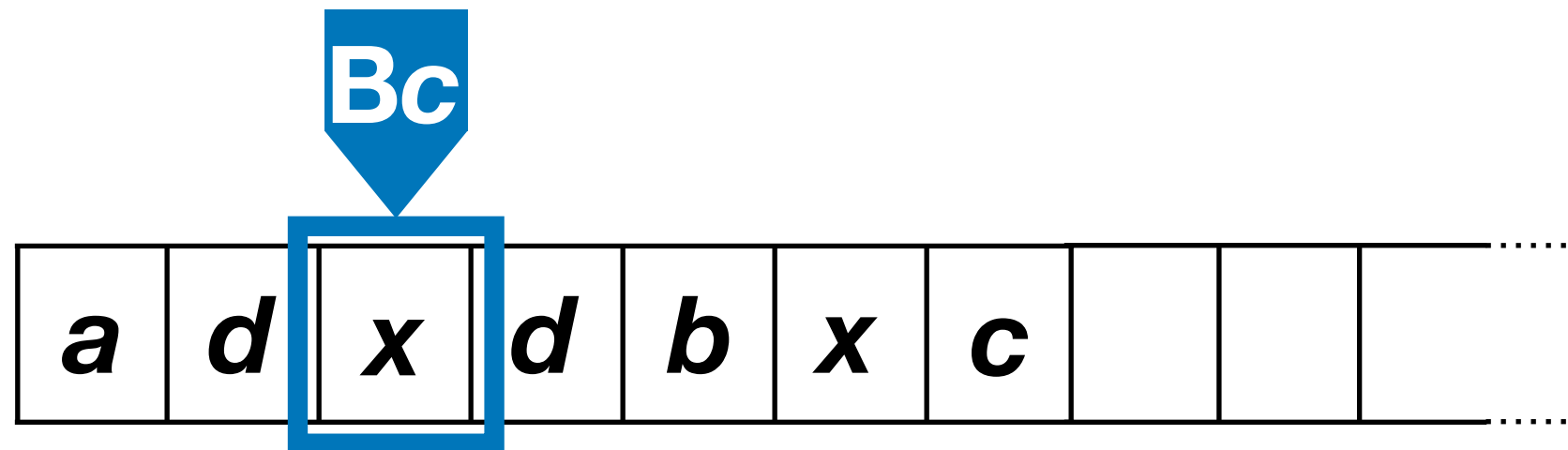
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
B	<i>a</i>	←	B <i>a</i>
B	<i>b</i>	←	B <i>b</i>
B	<i>c</i>	←	B <i>c</i>
...
B	<i>z</i>	←	B <i>z</i>
B <i>c</i>	a, b, d, \dots, z	←	B <i>c</i>

... qui peuvent changer de sens ...



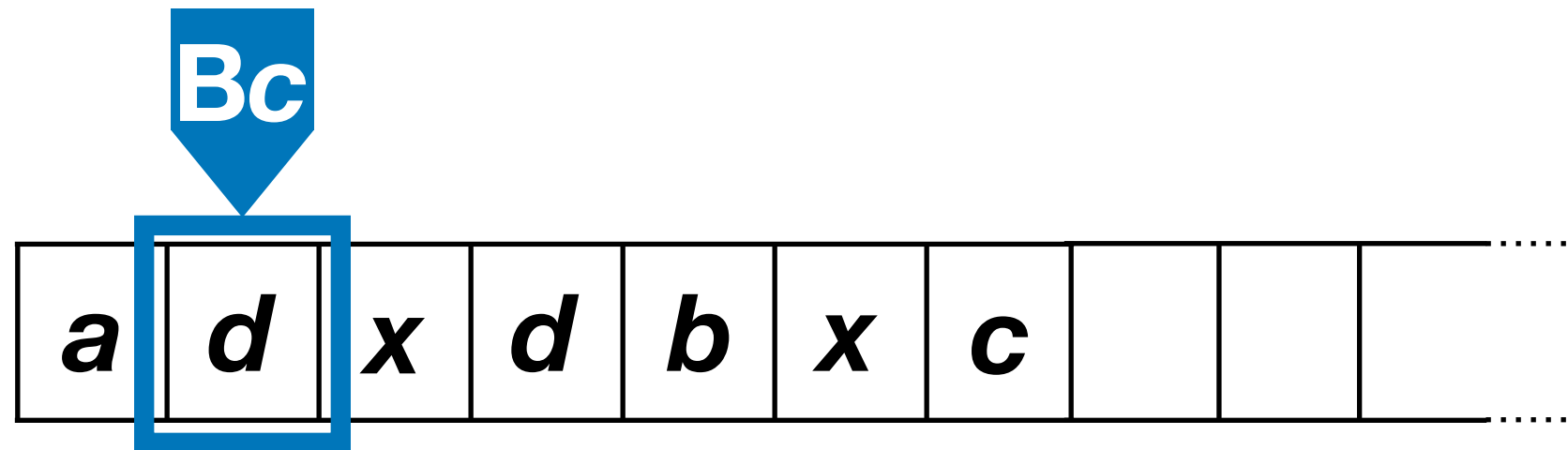
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	\rightarrow	A
A		\leftarrow	B
B	a	\leftarrow	Ba
B	b	\leftarrow	Bb
B	c	\leftarrow	Bc
...
B	z	\leftarrow	Bz
Bc	a, b, d, \dots, z	\leftarrow	Bc

... qui peuvent changer de sens ...



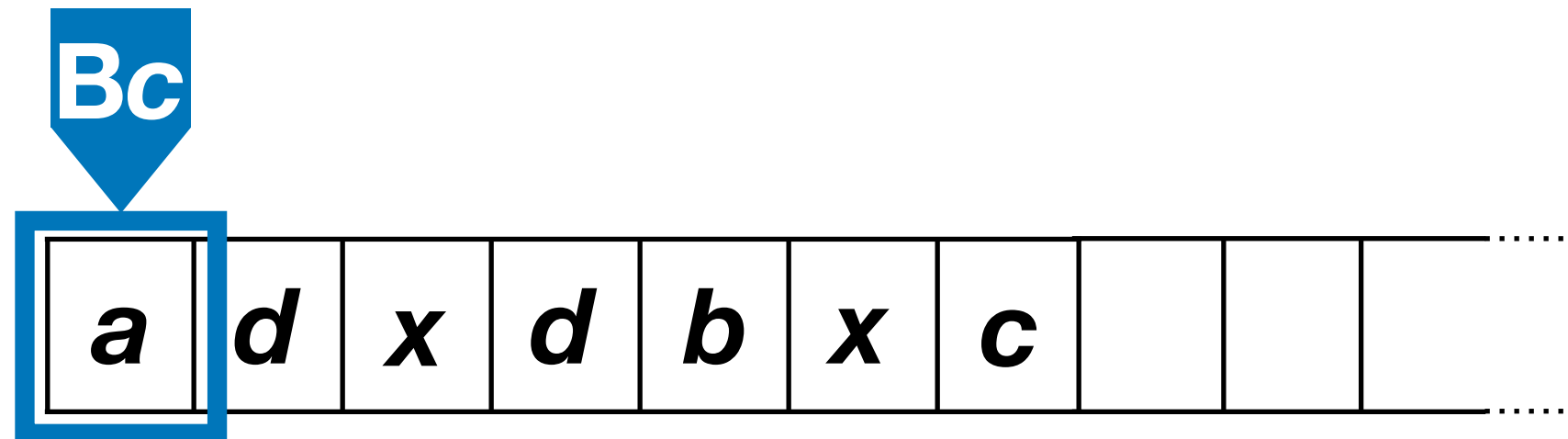
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	\rightarrow	A
A		\leftarrow	B
B	a	\leftarrow	Ba
B	b	\leftarrow	Bb
B	c	\leftarrow	Bc
...
B	z	\leftarrow	Bz
Bc	a, b, d, \dots, z	\leftarrow	Bc

... qui peuvent changer de sens ...



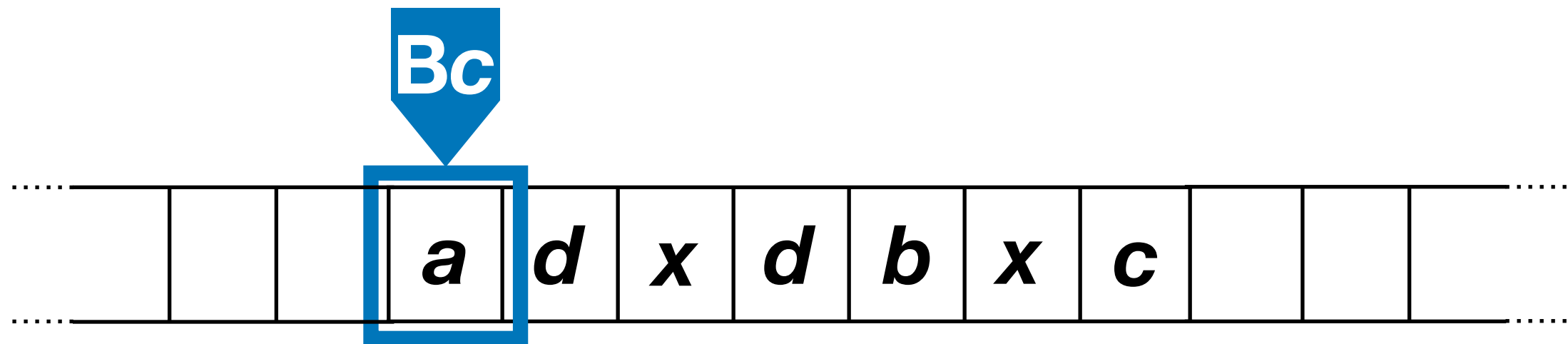
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
B	<i>a</i>	←	B <i>a</i>
B	<i>b</i>	←	B <i>b</i>
B	<i>c</i>	←	B <i>c</i>
...
B	<i>z</i>	←	B <i>z</i>
B <i>c</i>	a, b, d, \dots, z	←	B <i>c</i>

... qui peuvent changer de sens ...



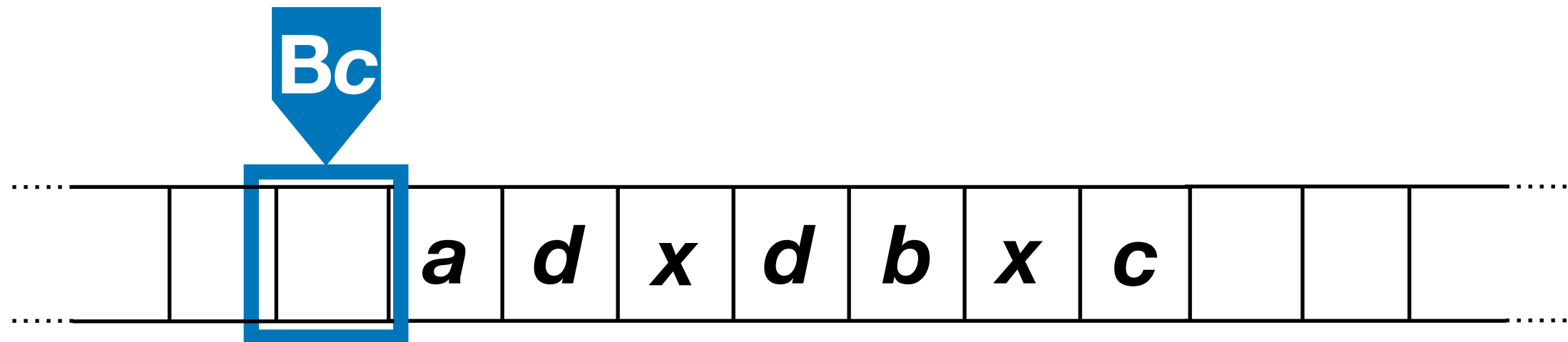
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
B	a	←	Ba
B	b	←	Bb
B	c	←	Bc
...
B	z	←	Bz
Bc	a, b, d, \dots, z	←	Bc

... qui peuvent changer de sens ...



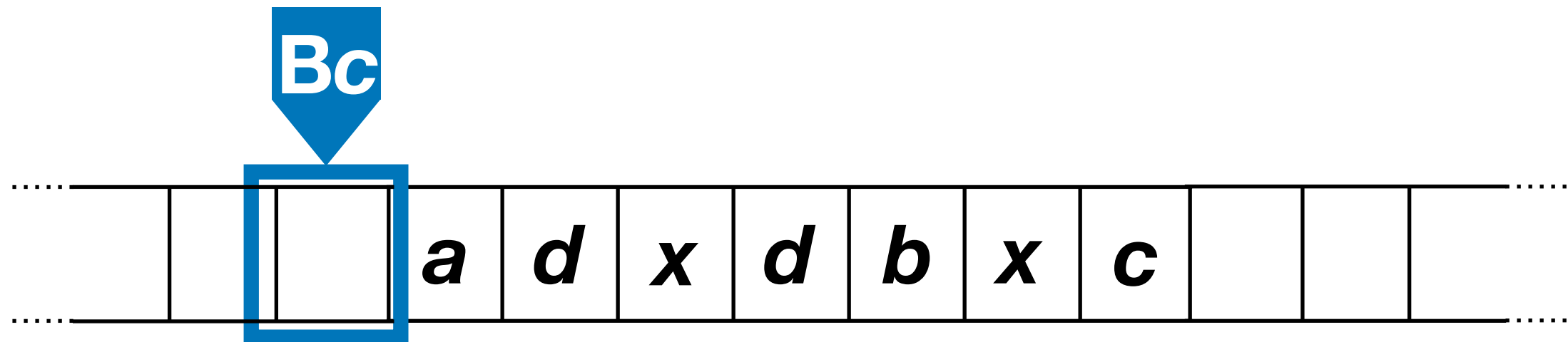
Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
B	<i>a</i>	←	B <i>a</i>
B	<i>b</i>	←	B <i>b</i>
B	<i>c</i>	←	B <i>c</i>
...
B	<i>z</i>	←	B <i>z</i>
B <i>c</i>	a, b, d, \dots, z	←	B <i>c</i>

... qui peuvent changer de sens ...



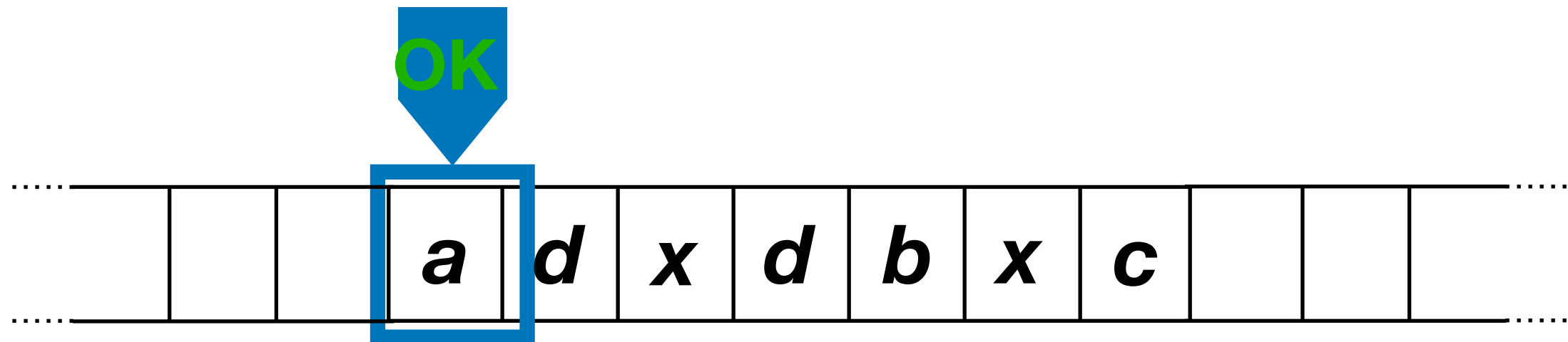
Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B
B	<i>a</i>	←	B <i>a</i>
B	<i>b</i>	←	B <i>b</i>
B	<i>c</i>	←	B <i>c</i>
...
B	<i>z</i>	←	B <i>z</i>
B <i>c</i>	a,b,d,\dots,z	←	B <i>c</i>

... qui peuvent changer de sens ...



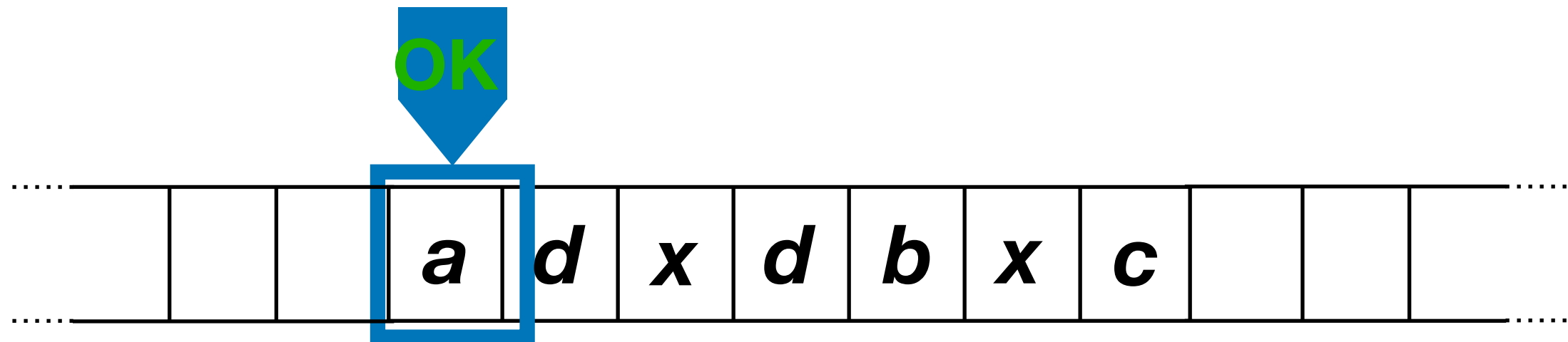
Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B
B	a	←	Ba
B	b	←	Bb
B	c	←	Bc
...
B	z	←	Bz
Bc	a,b,d,\dots,z	←	Bc
Bc		→	OK

... qui peuvent changer de sens ...



Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B
B	a	←	Ba
B	b	←	Bb
B	c	←	Bc
...
B	z	←	Bz
Bc	a,b,d,\dots,z	←	Bc
Bc		→	OK

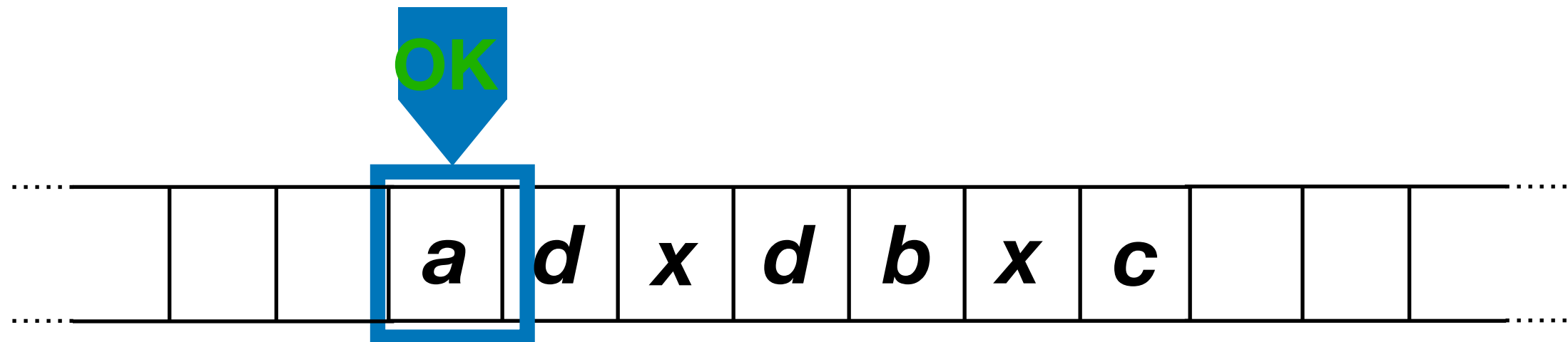
... qui peuvent changer de sens ...



Accepte si le
dernier symbole de
la séquence est
différent de tous les
autres symboles

Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
B	<i>a</i>	←	Ba
B	<i>b</i>	←	Bb
B	<i>c</i>	←	Bc
...
B	<i>z</i>	←	Bz
Bc	a, b, d, \dots, z	←	Bc
Bc		→	OK

... qui peuvent changer de sens ...

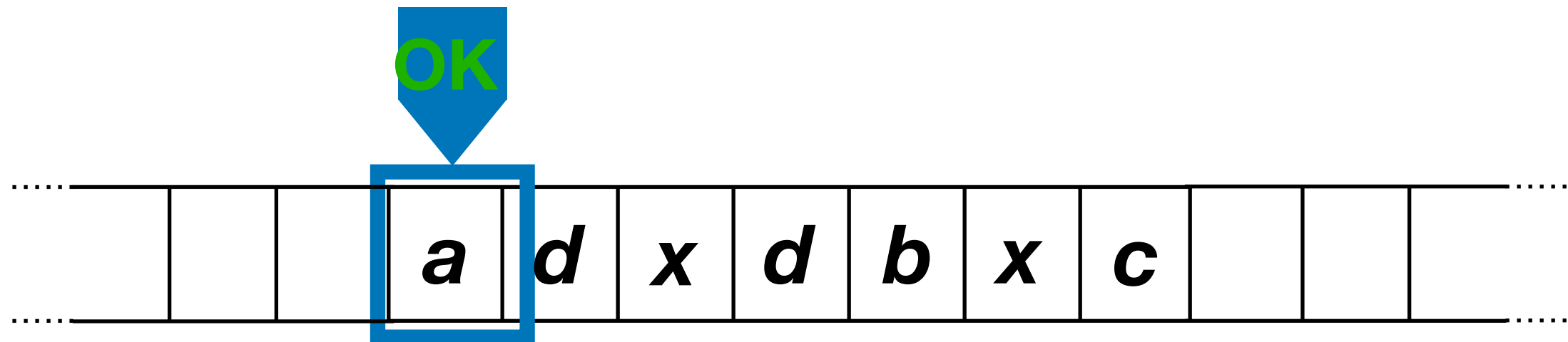


Accepte si le dernier symbole de la séquence est différent de tous les autres symboles

Faisable avec un très gros automate...

Etat	Symbole	Sens	Nouvel état
A	<i>a, b, c, ..., z</i>	→	A
A		←	B
B	<i>a</i>	←	Ba
B	<i>b</i>	←	Bb
B	<i>c</i>	←	Bc
...
B	<i>z</i>	←	Bz
Bc	<i>a, b, d, ..., z</i>	←	Bc
Bc		→	OK

... qui peuvent changer de sens ...



**Accepte si le
dernier symbole de
la séquence est
différent de tous les
autres symboles**

*Faisable avec un très
gros automate...*

Etat	Symbole	Sens	Nouvel état
A	a,b,c,\dots,z	→	A
A		←	B
	a	←	Ba
	b		Bb

Si état = A et on lit a **alors**

état := A

se déplacer à droite

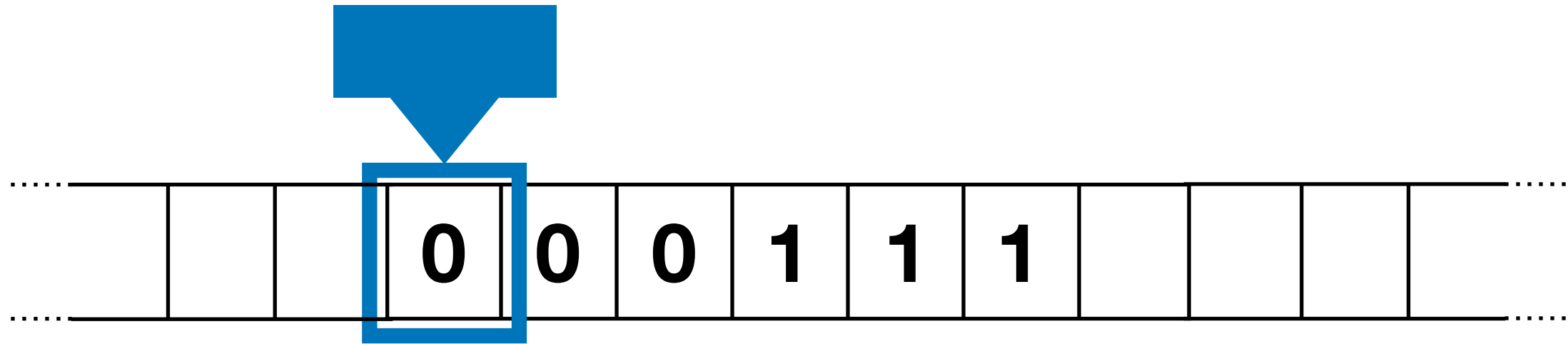
Sinon Si état = A et on ne lit rien **alors**

état := B

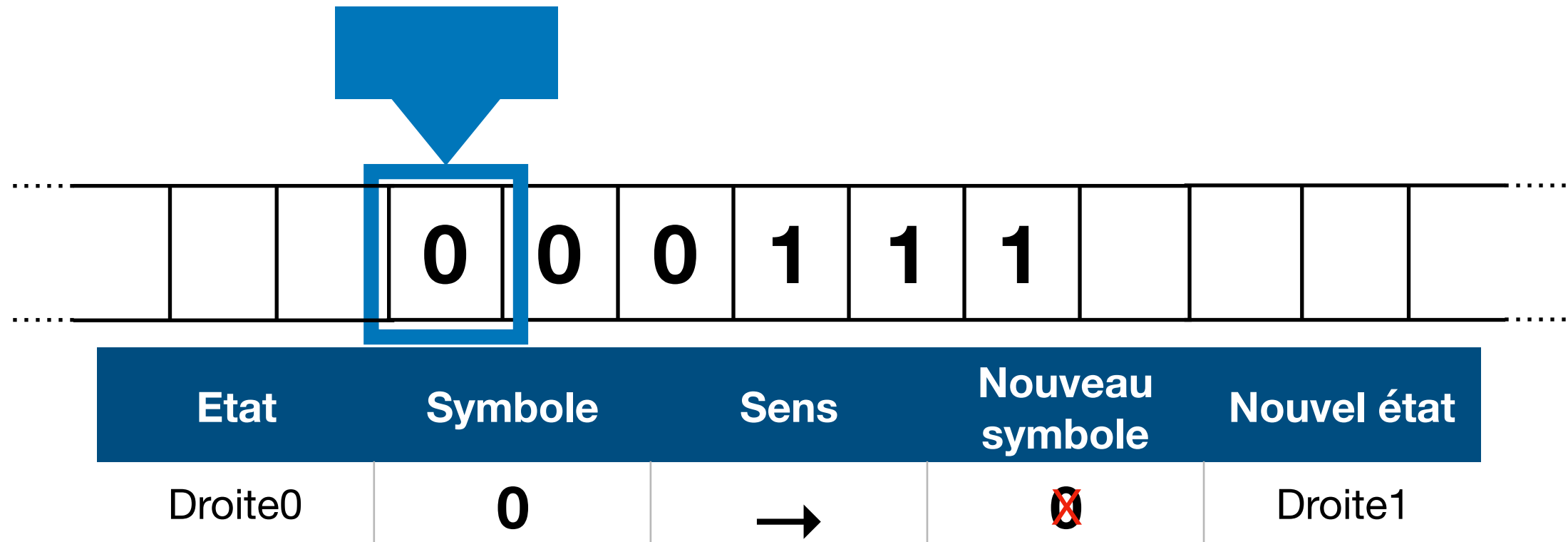
se déplacer à gauche

...

... et écrire sur la bande

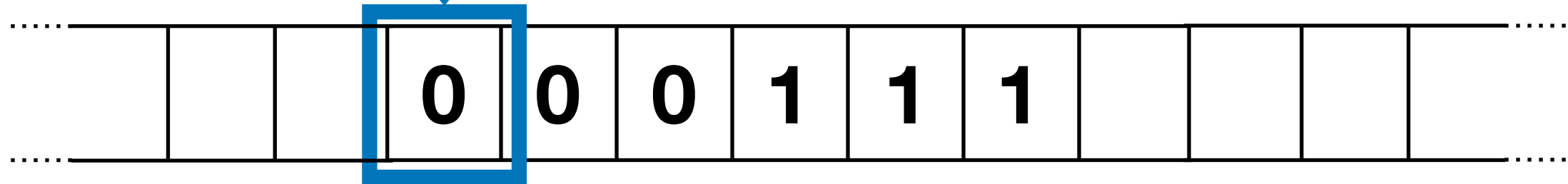


... et écrire sur la bande



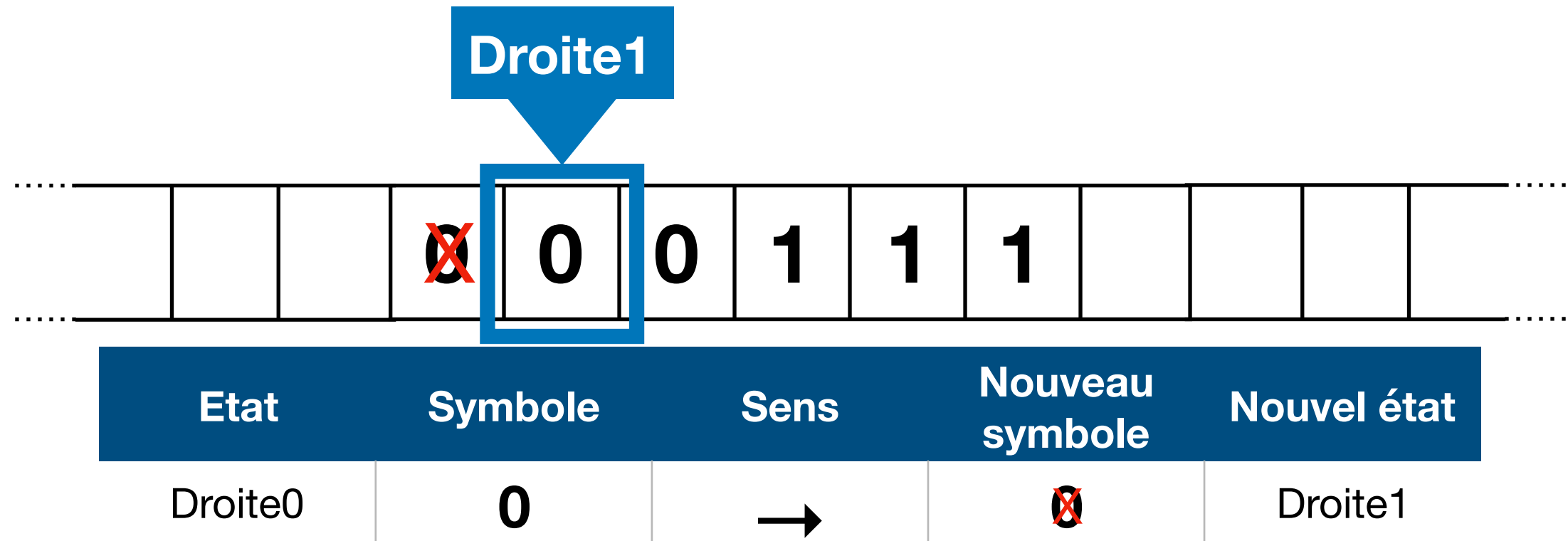
... et écrire sur la bande

Droite0

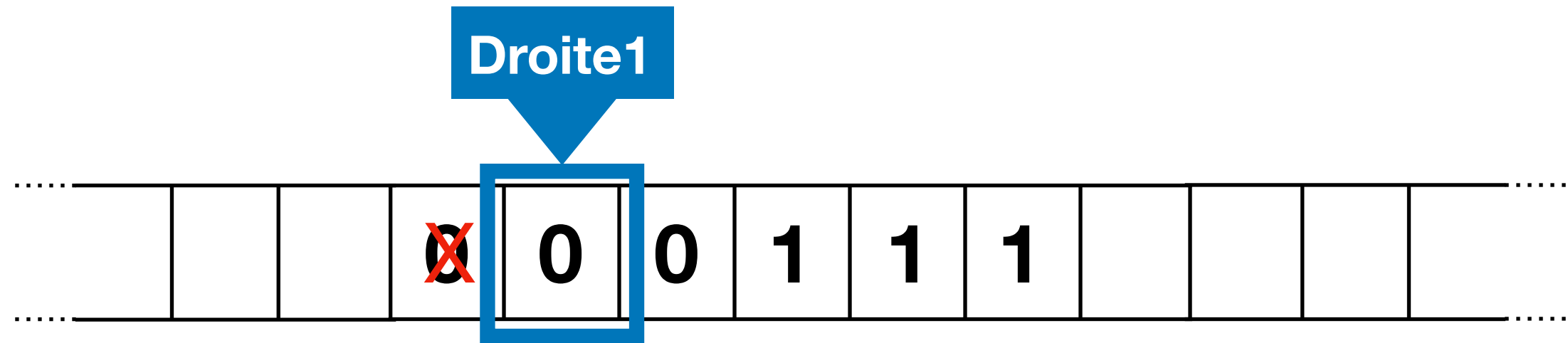


Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1

... et écrire sur la bande

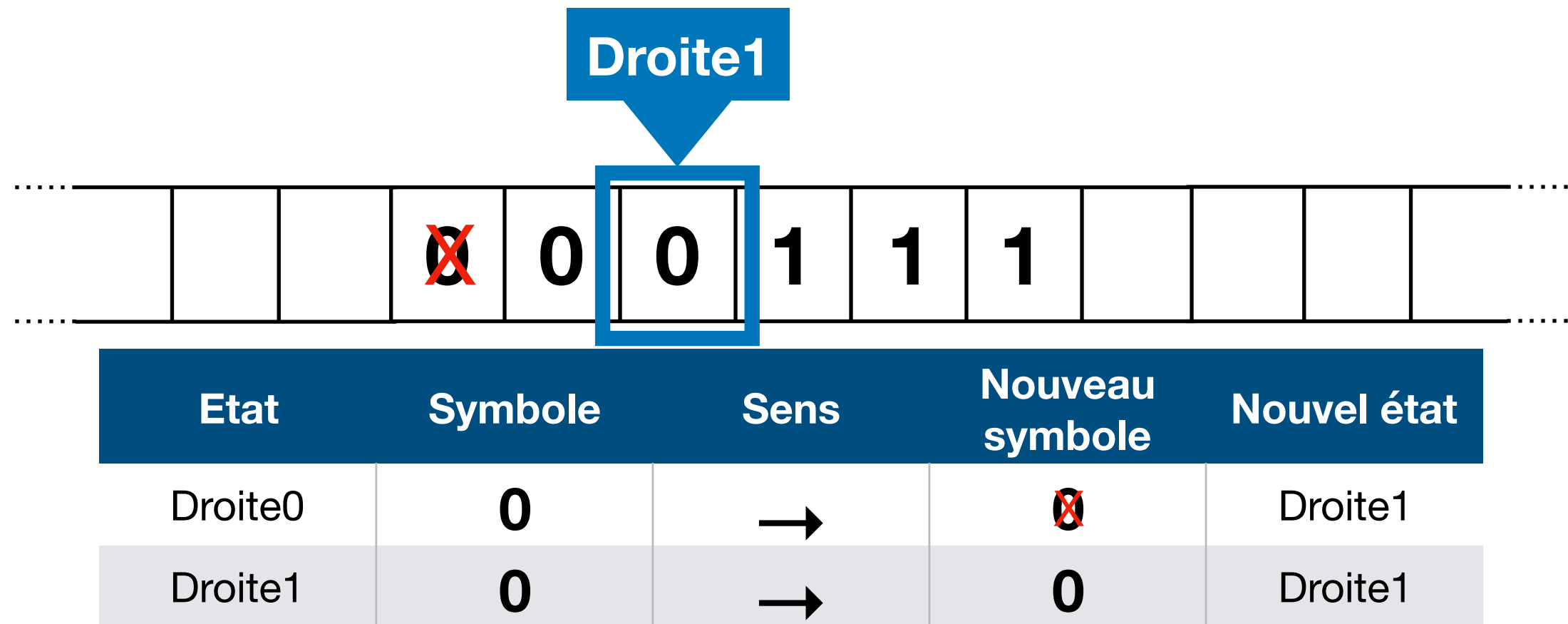


... et écrire sur la bande

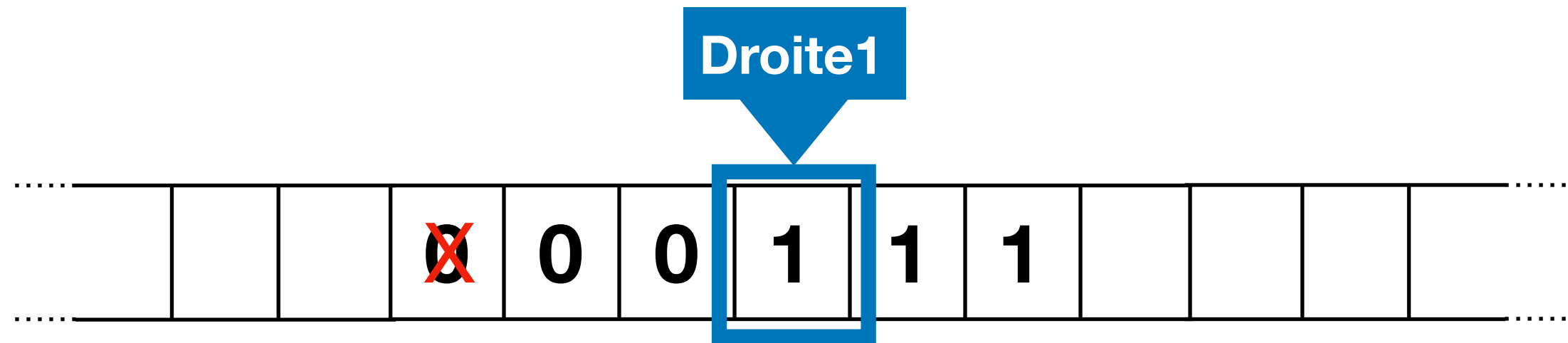


Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1

... et écrire sur la bande

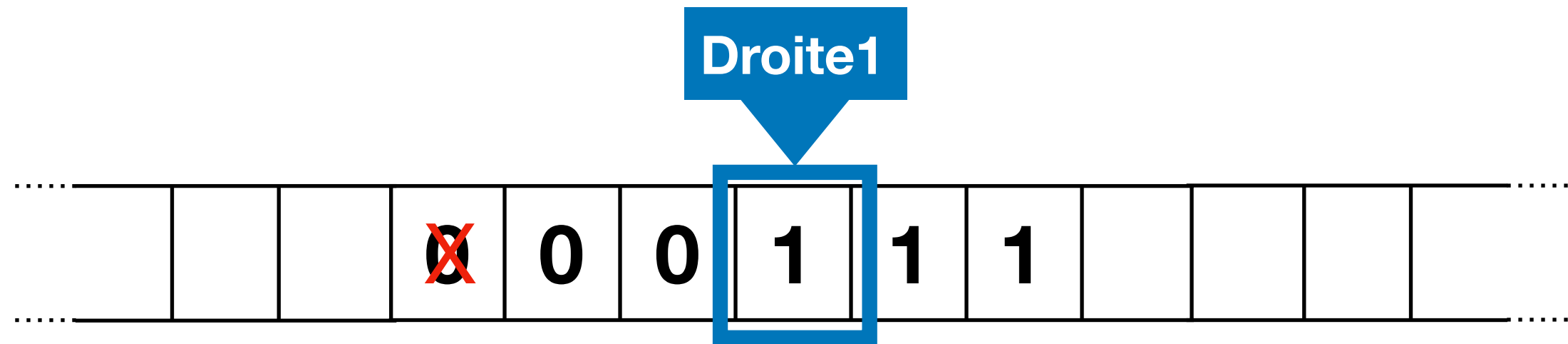


... et écrire sur la bande



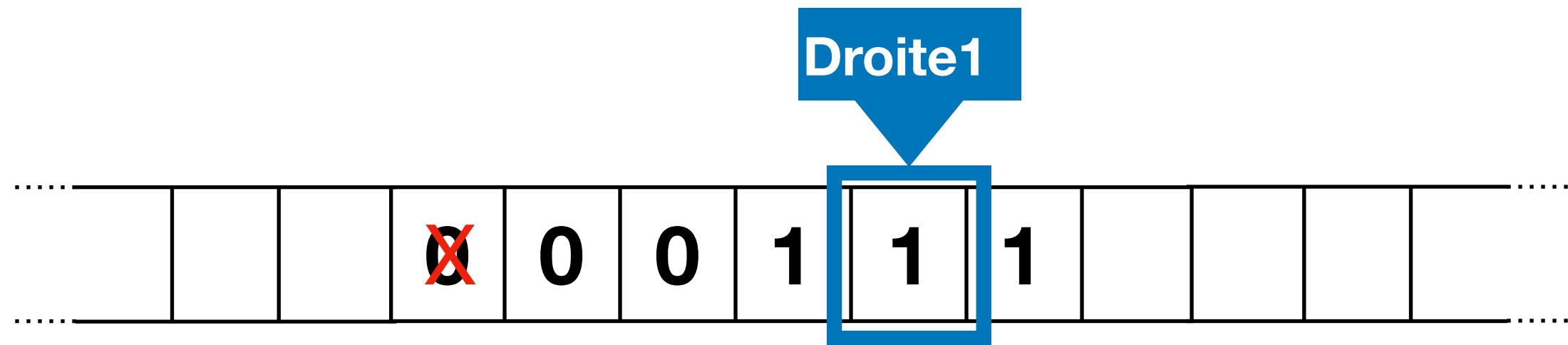
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1

... et écrire sur la bande



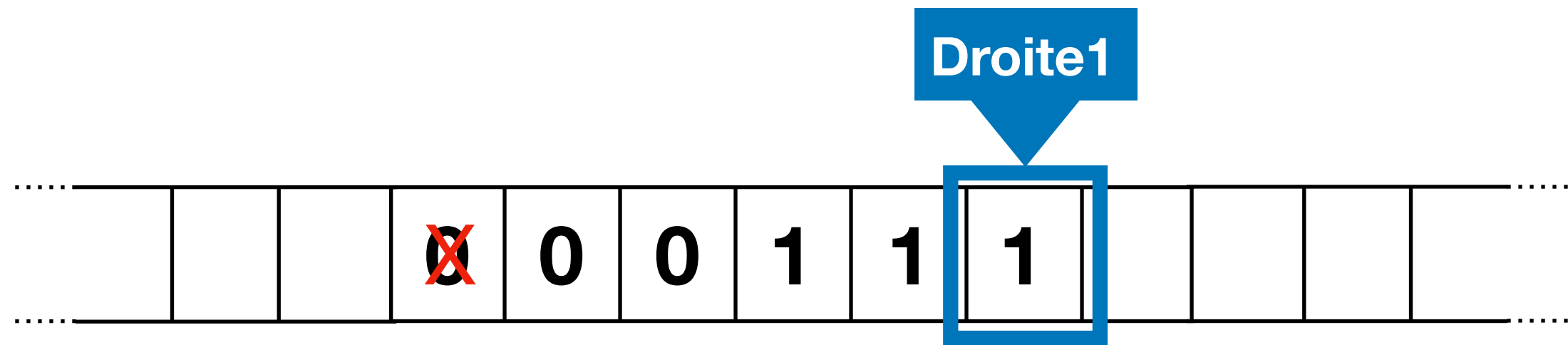
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1

... et écrire sur la bande



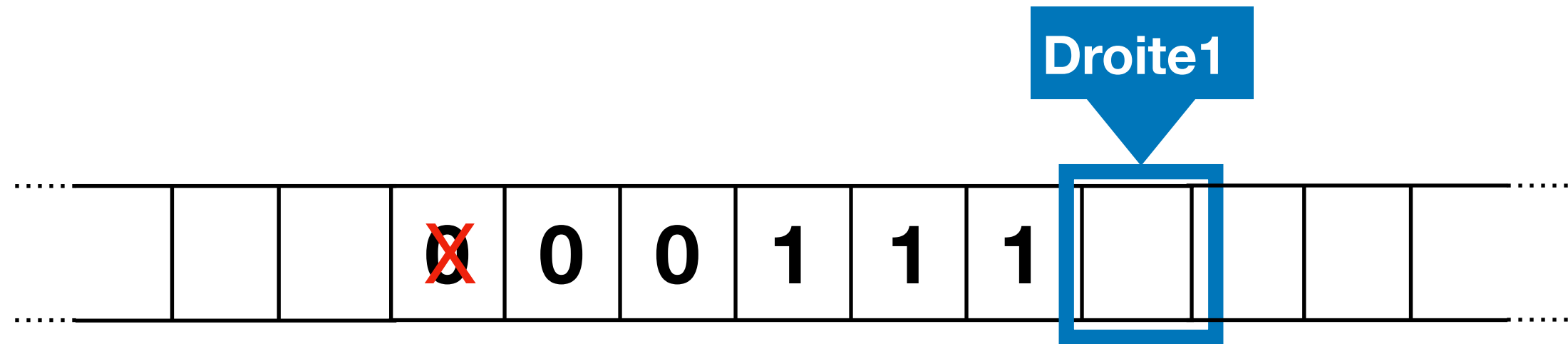
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1

... et écrire sur la bande



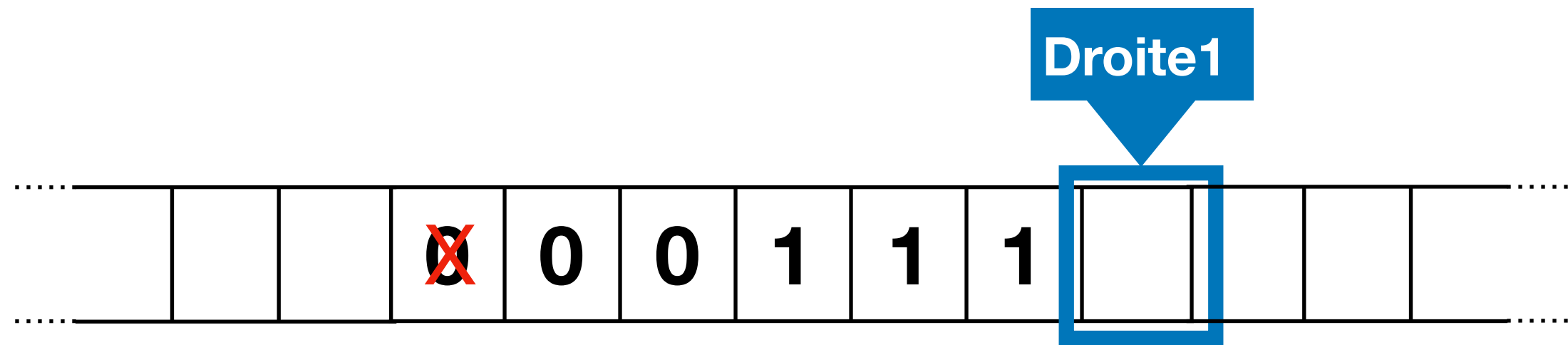
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1

... et écrire sur la bande



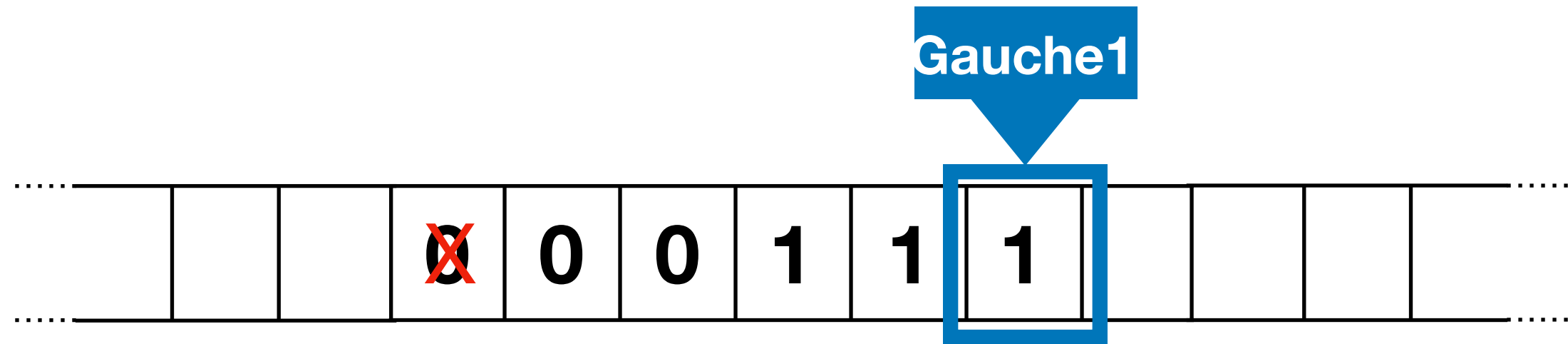
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1

... et écrire sur la bande



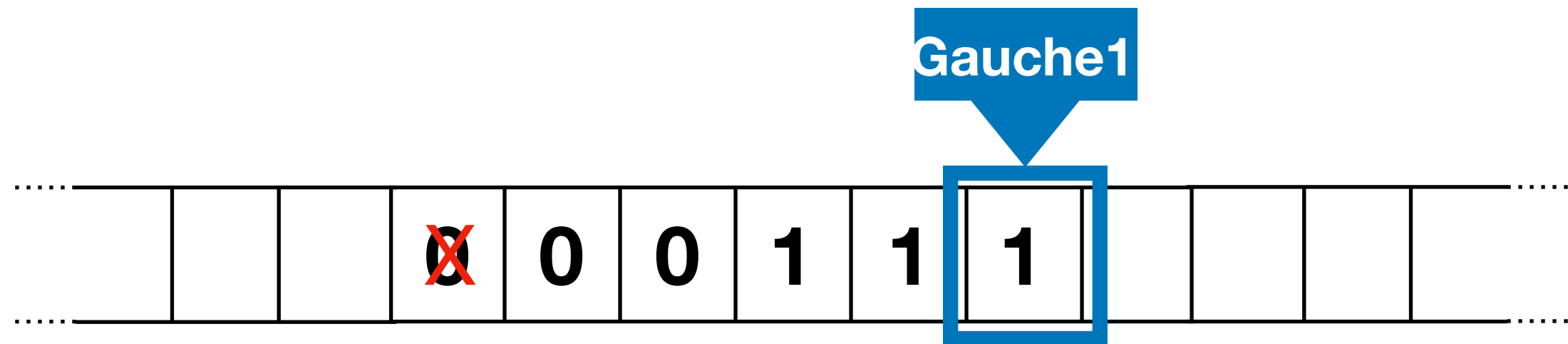
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1

... et écrire sur la bande



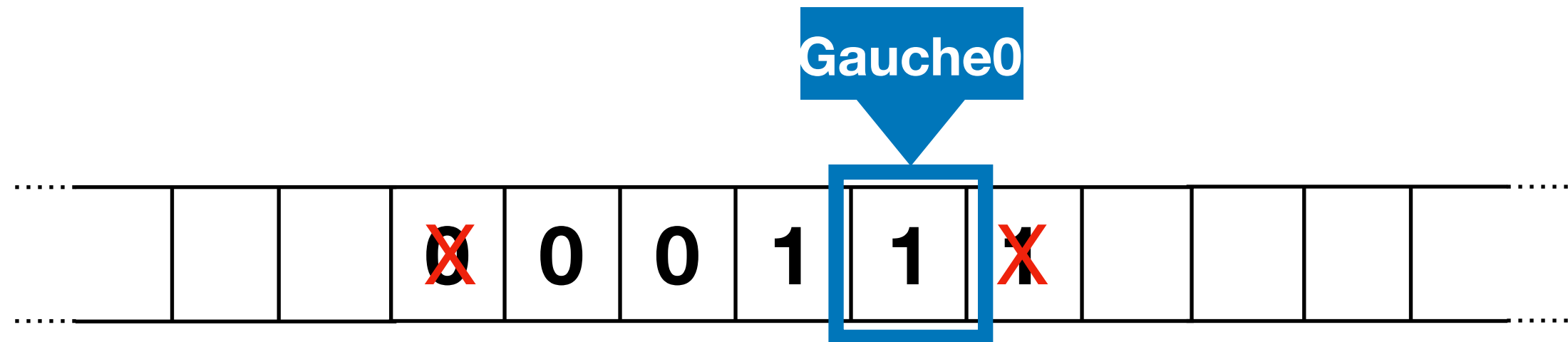
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1

... et écrire sur la bande



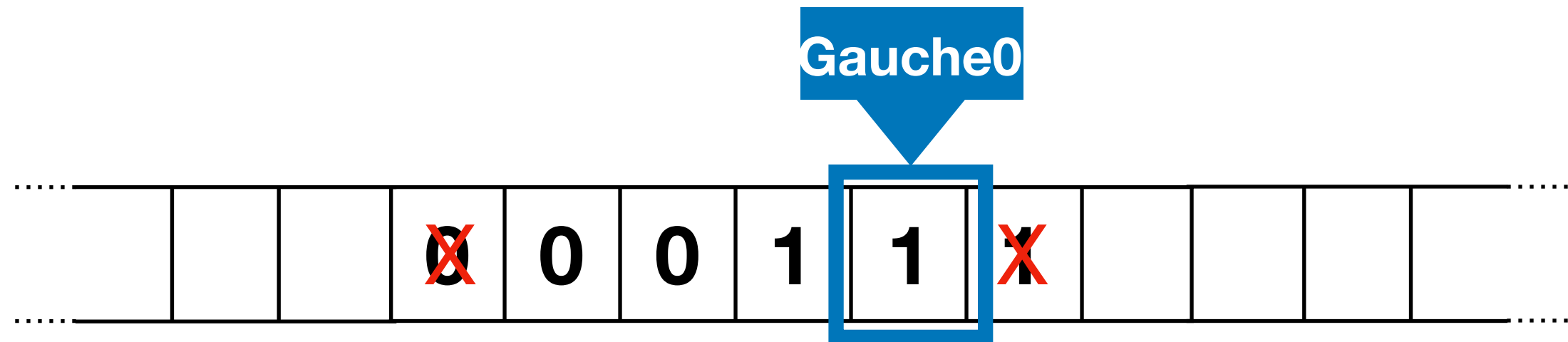
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0

... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0

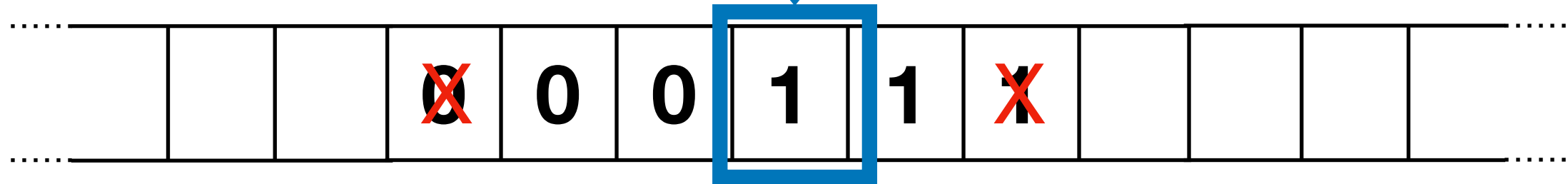
... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0

... et écrire sur la bande

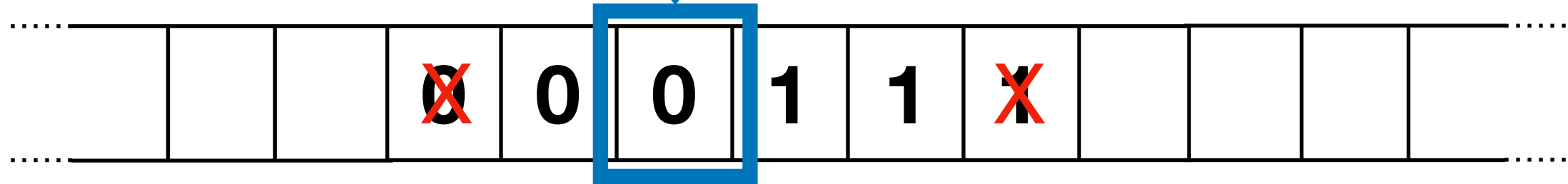
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0

... et écrire sur la bande

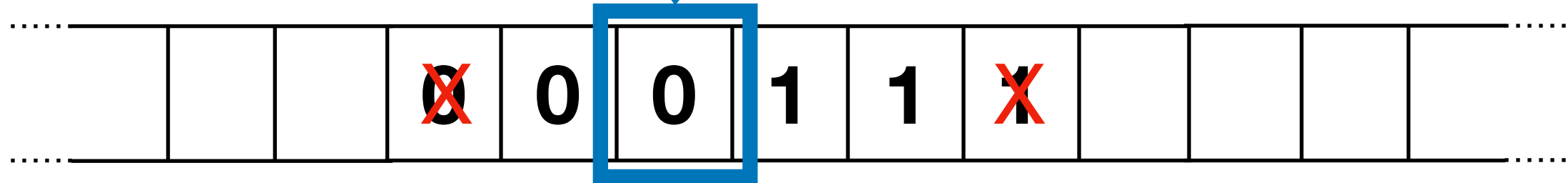
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0

... et écrire sur la bande

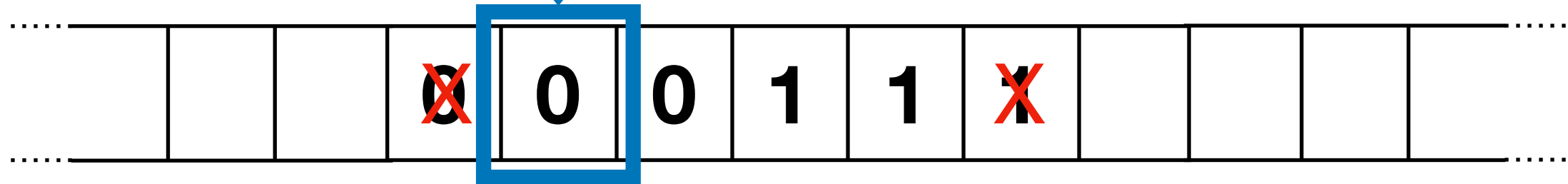
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0

... et écrire sur la bande

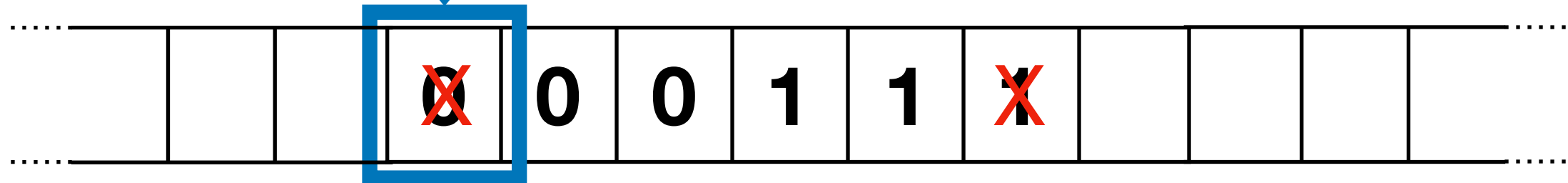
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0

... et écrire sur la bande

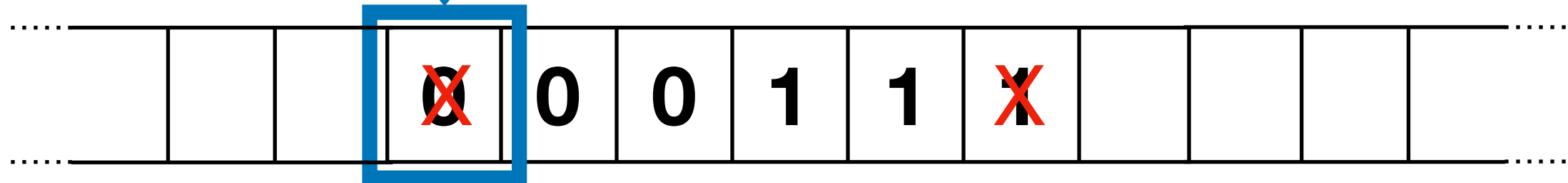
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0

... et écrire sur la bande

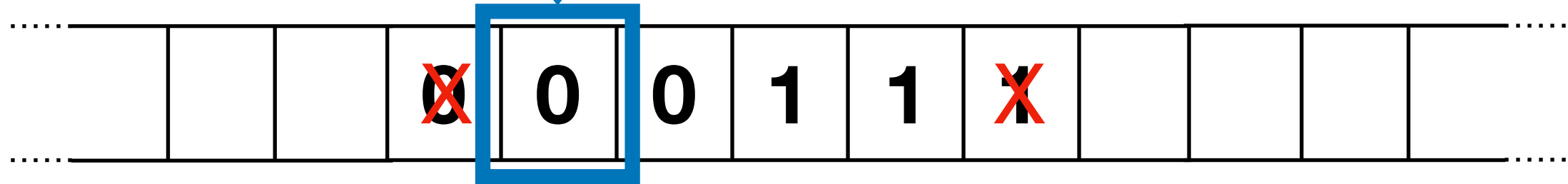
Gauche0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

... et écrire sur la bande

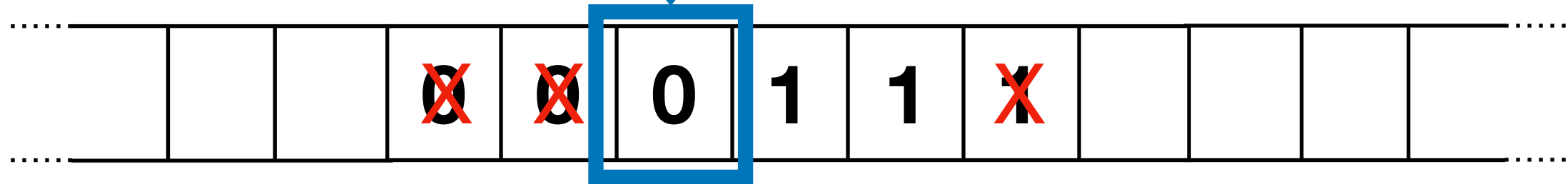
Droite0



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

... et écrire sur la bande

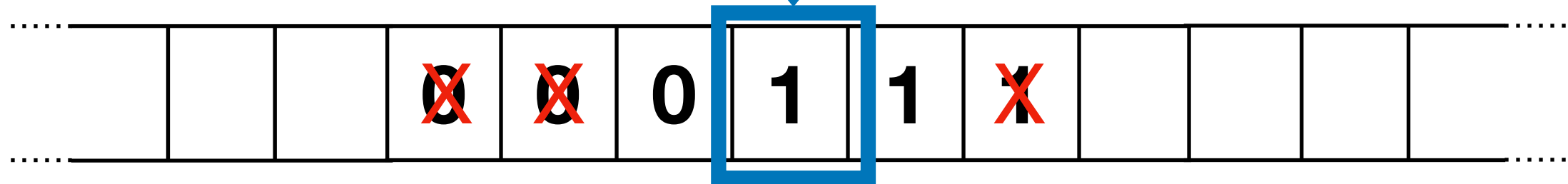
Droite1



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

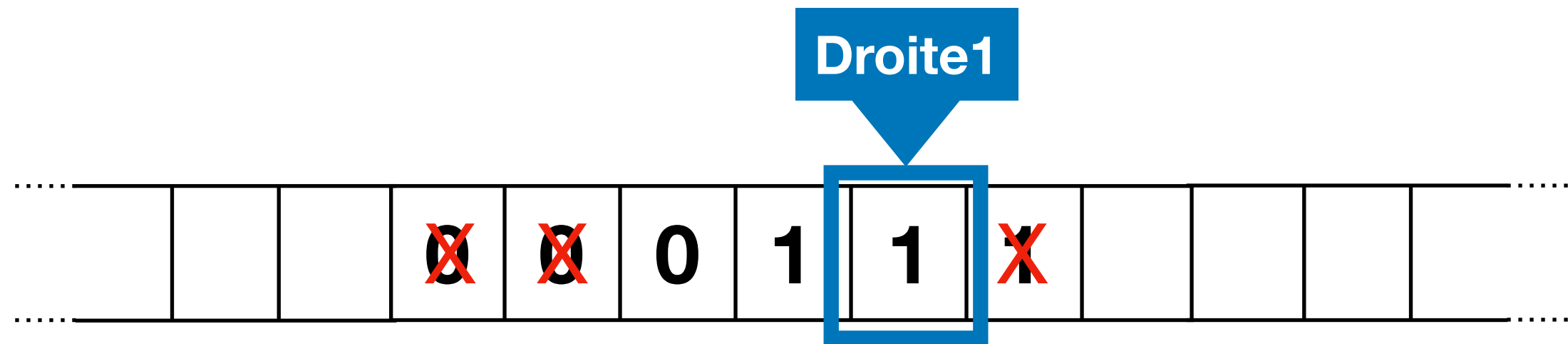
... et écrire sur la bande

Droite1



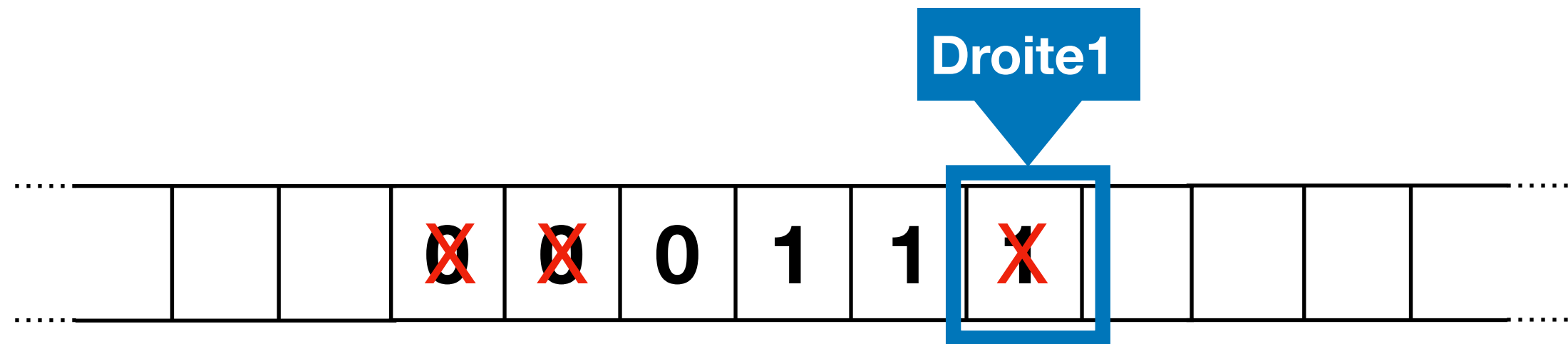
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

... et écrire sur la bande



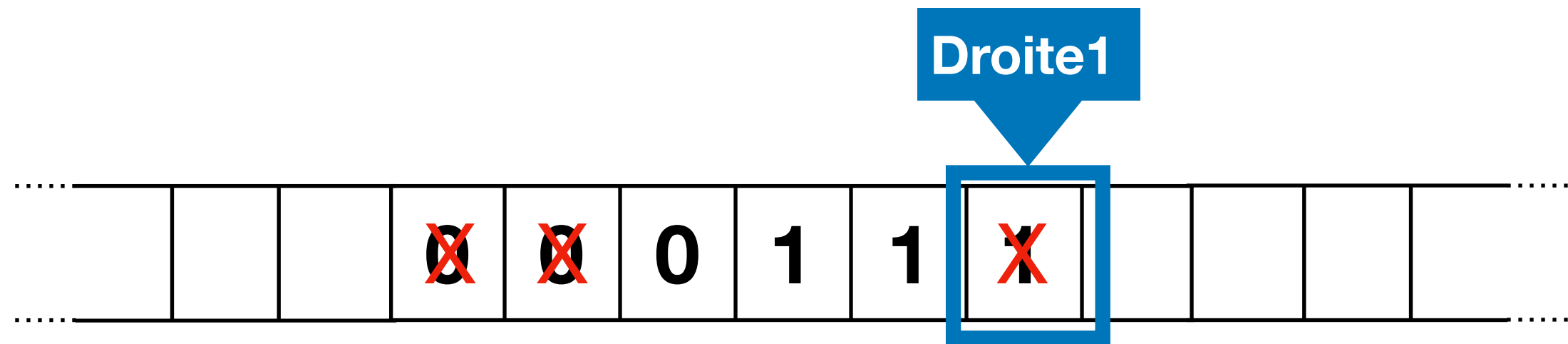
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

... et écrire sur la bande



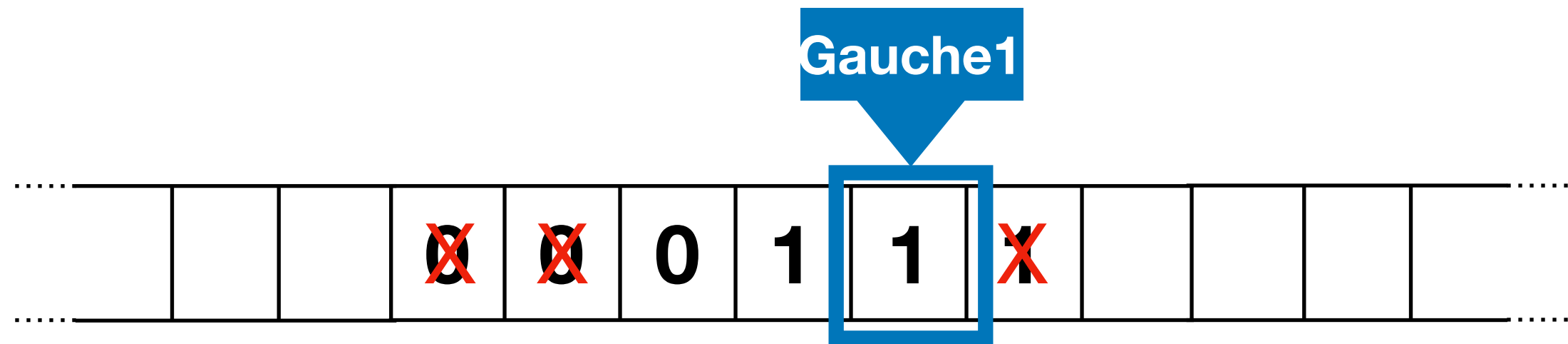
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0

... et écrire sur la bande



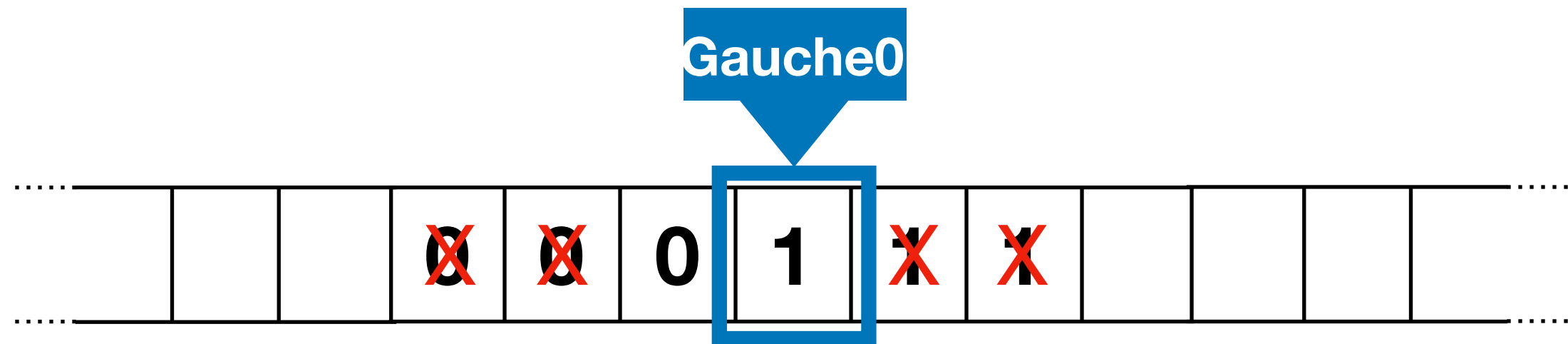
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



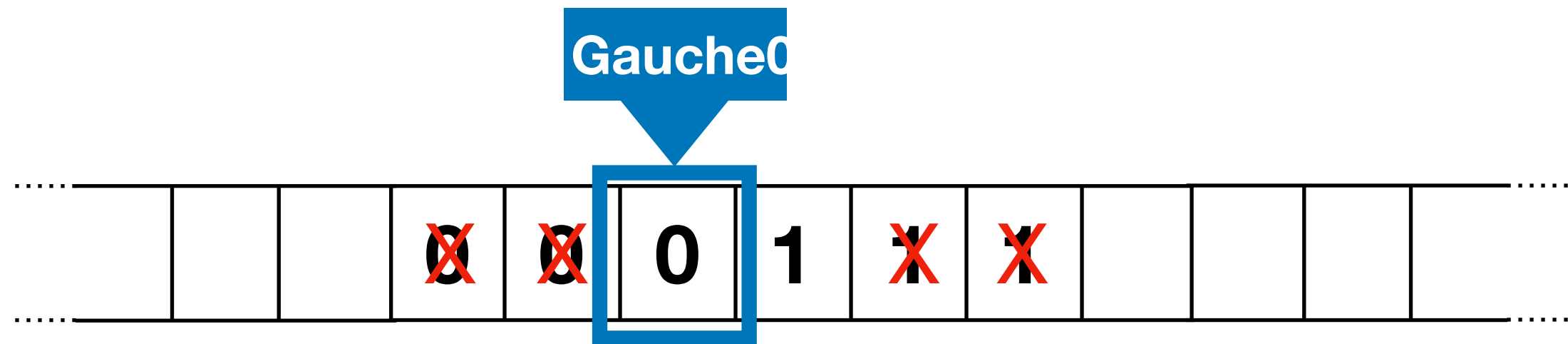
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



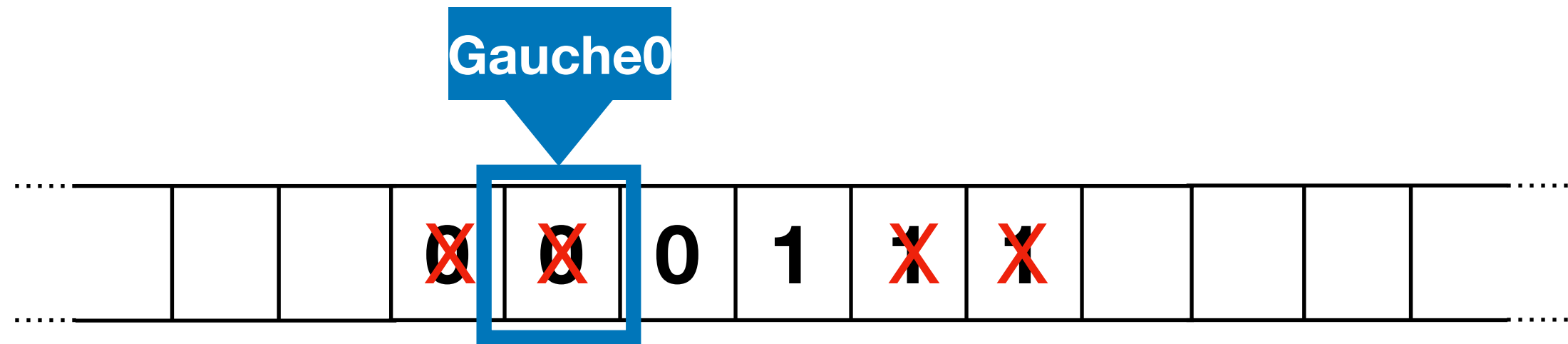
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



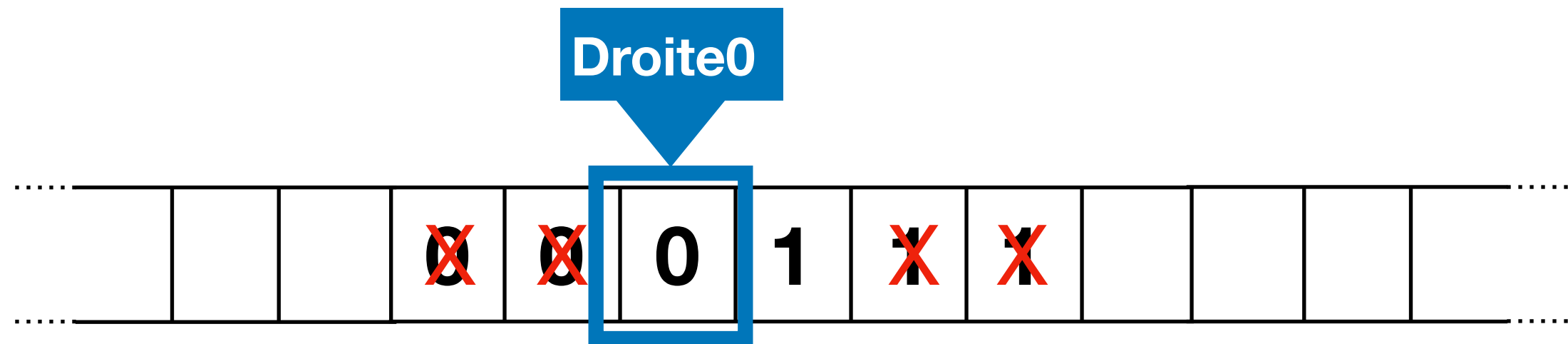
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

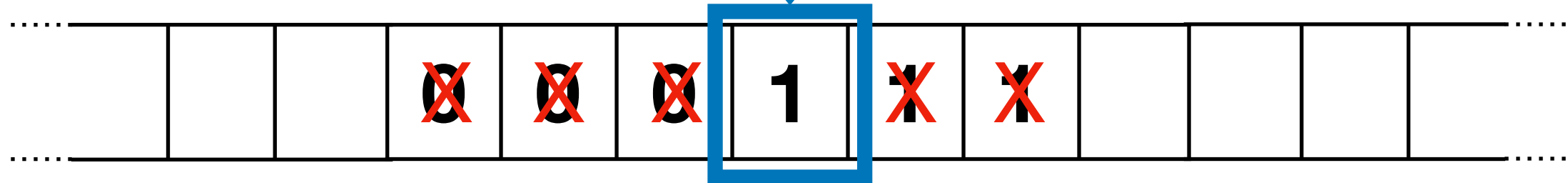
... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

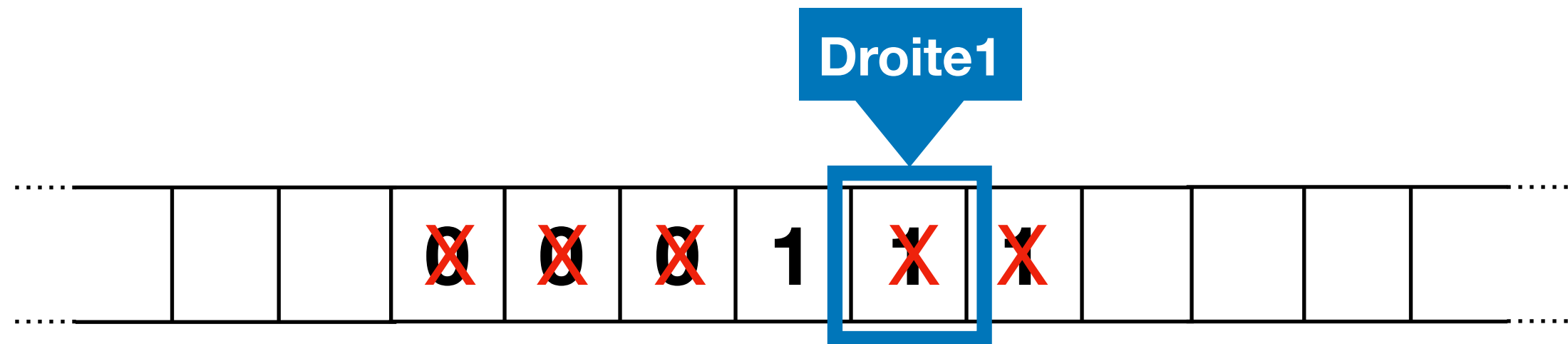
... et écrire sur la bande

Droite1



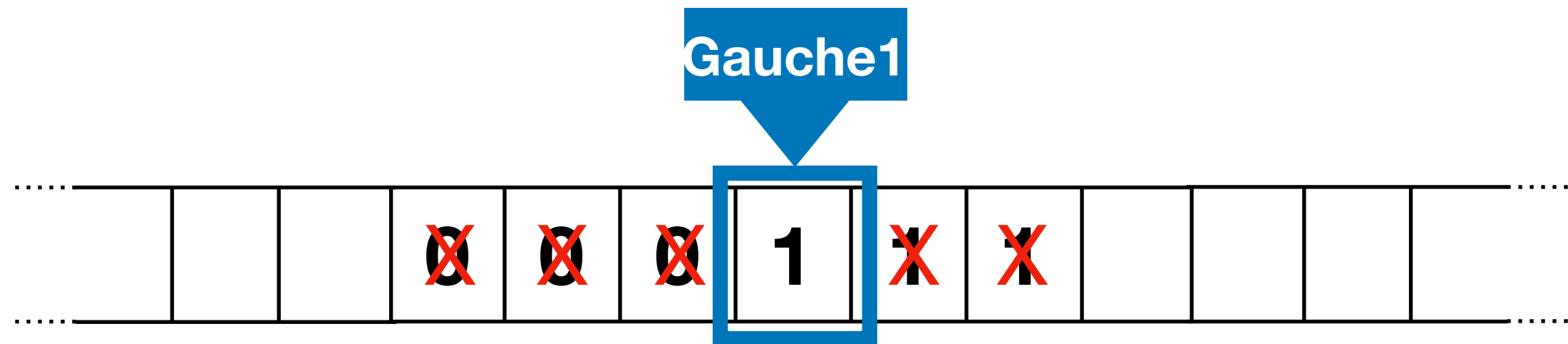
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



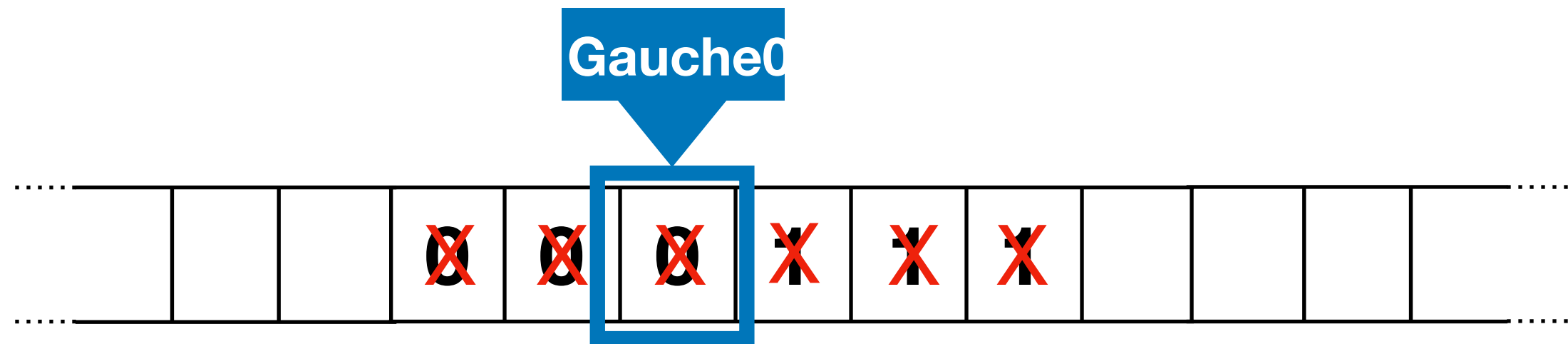
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



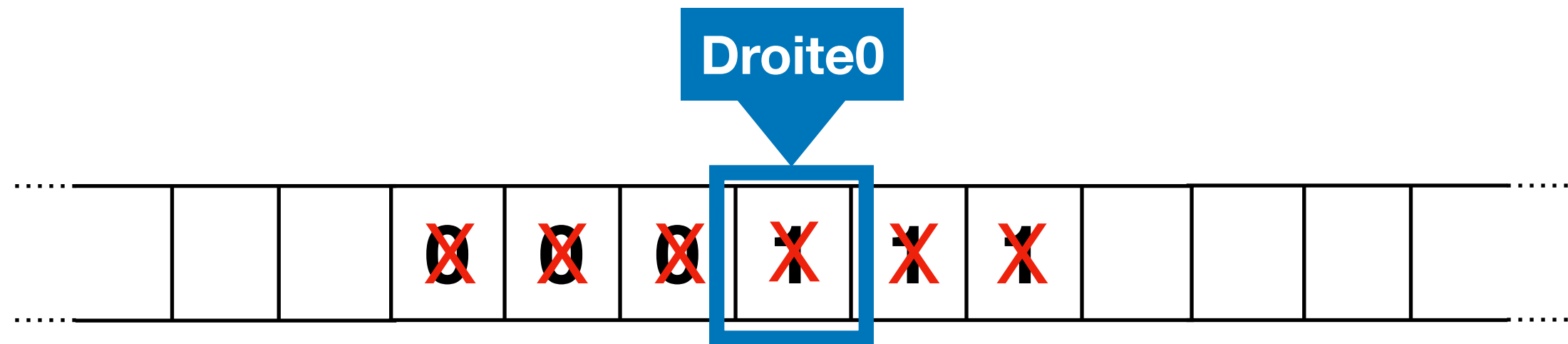
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



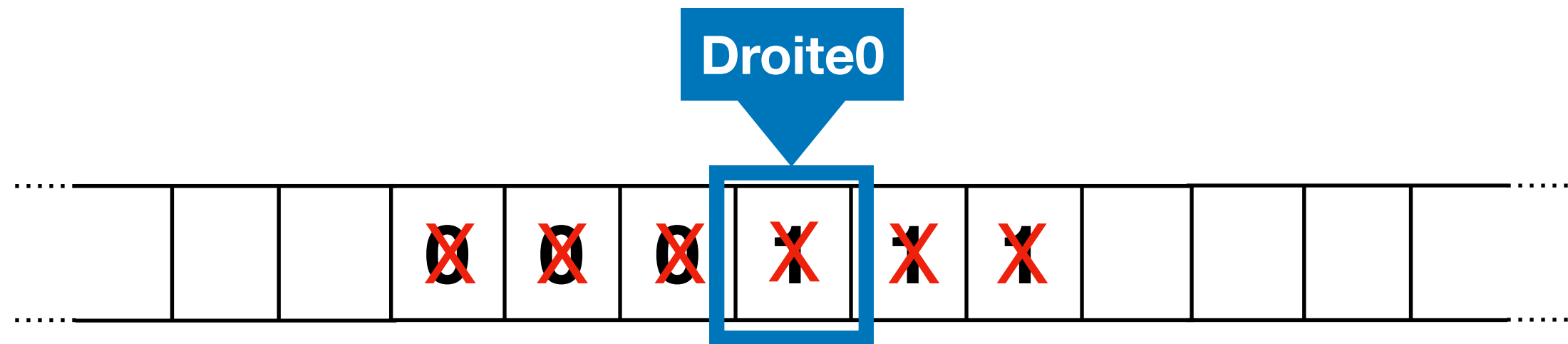
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



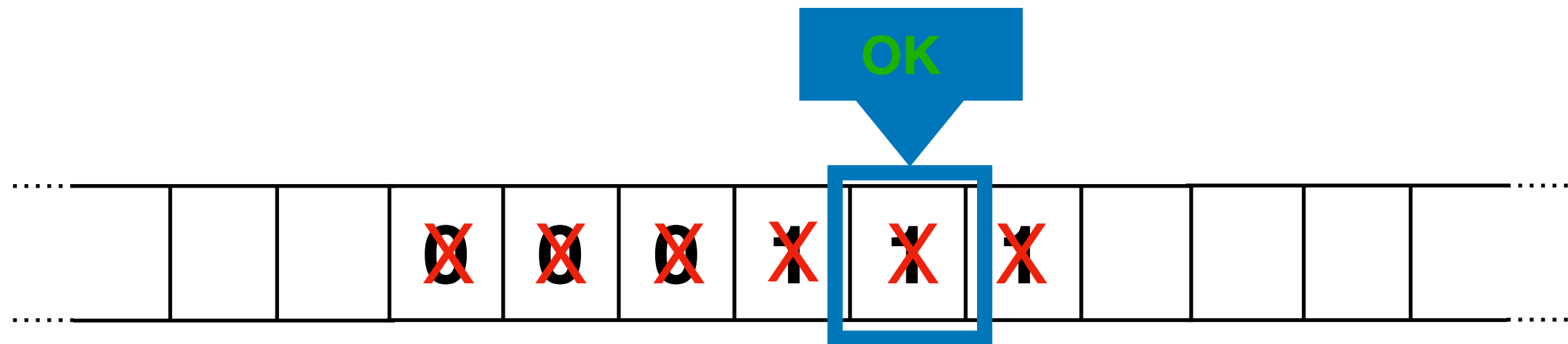
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1

... et écrire sur la bande



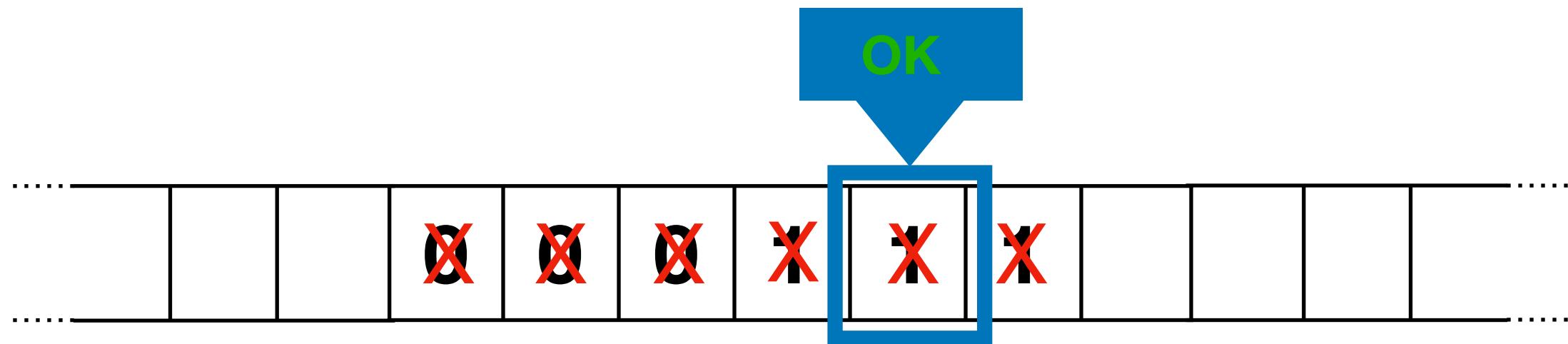
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1
Droite0	1	→	1	OK

... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1		←		Gauche1
Gauche1	1	←	1	Gauche0
Gauche0	1	←	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1
Droite0	1	→	1	OK

... et écrire sur la bande



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
<div>Si état = Droite0 et on lit 0 alors état := Droite1 écrire 1 sur la bande se déplacer à droite ...</div>				Gauche1
			1	Gauche0
			1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	1	←	1	Gauche1
Droite0	1	→	1	OK

**Machine de Turing =
calculateur humain avec
papier et crayon**



Calculateurs humains

NACA (Comité consultatif national pour
l'aéronautique), USA, 1950s

« Normalement on calcule en écrivant certains symboles sur le papier. [...] Je considère qu'on effectue le calcul sur un **papier unidimensionnel**, c'est-à-dire, sur un **ruban divisé en carrés**. »

– Alan M. Turing, *On computable numbers*

Papier 2D vs ruban 1D

A	B	C
D	E	F
G	H	I
J	K	L

Papier 2D vs ruban 1D

A	B	C
D	E	F
G	H	I
J	K	L

A	B	C	;	D	E	F	;	G	H	I	;	J	K	L
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Papier 2D vs ruban 1D

A	B	C
D	E	F
G	H	I
J	K	L

M	N	O
P	Q	R
S	T	U
V	W	X

E	F	:	G	H	I	:	J	K	L	:	M	N	O	:	P	Q	R	:	S	T
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

« Je suppose aussi que le **nombre de symboles** qu'on peut écrire soit **fini**. Si on permettait une infinité de symboles, il y aurait des symboles qui diffèrent dans une mesure arbitrairement faible [...] **On peut toujours utiliser une séquence de symboles** au lieu d'un symbole simple. »

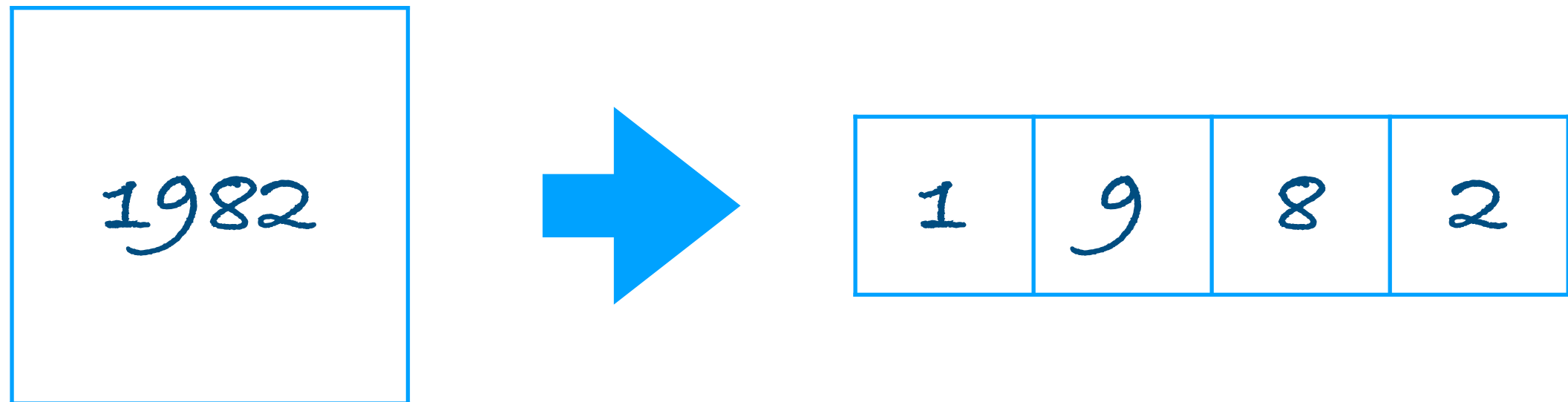
– Alan M. Turing, *On computable numbers*

Symboles atomiques vs composites



1982

Symboles atomiques vs composites



« La différence, de notre point de vue, entre les symboles simples et composites est qu'**on ne peut pas observer les symboles composites en un coup d'œil**, s'ils sont trop longs. Cela est conforme à l'expérience. On ne peut pas établir en un coup d'œil si 99999999999999999999 et 99999999999999999999 sont égales. »

– Alan M. Turing, *On computable numbers*

« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---

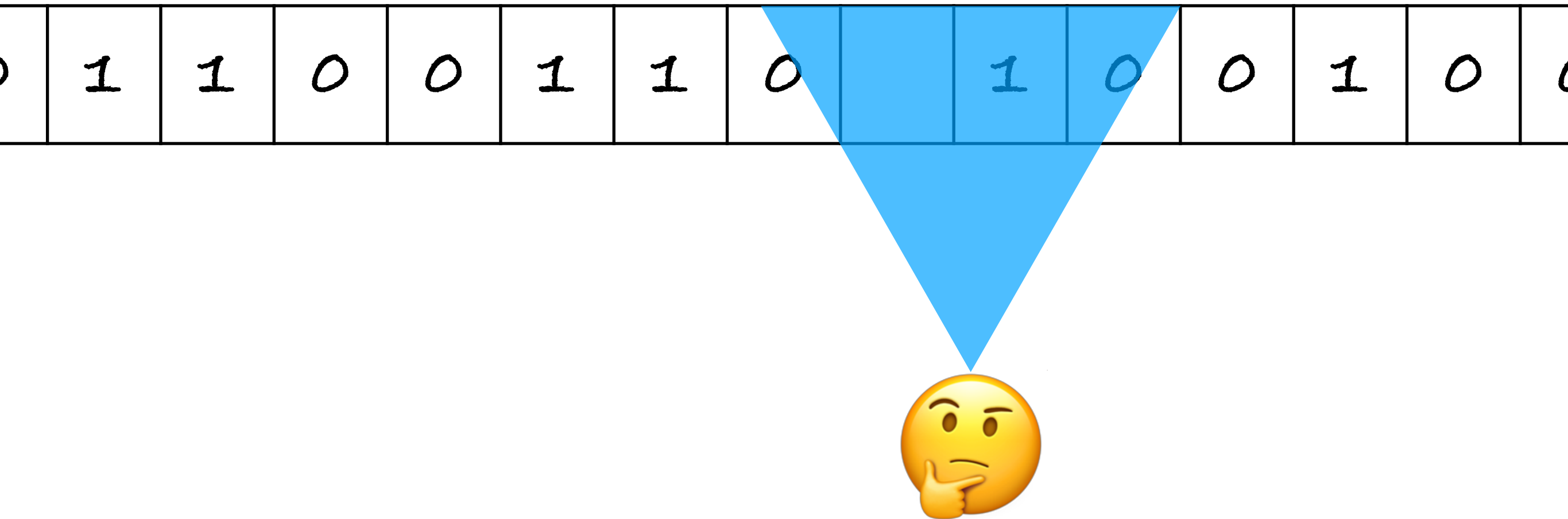


« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---



« Champ visuel »



« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---



« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---



« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---

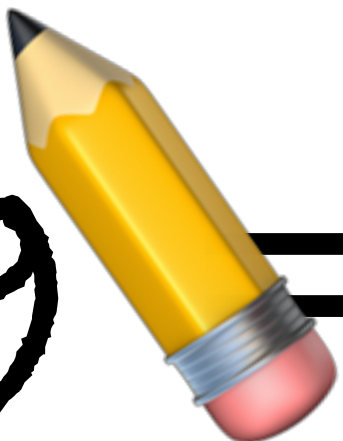


« Le **comportement** du calculateur à chaque moment est **déterminé par le symbole qu'il observe et son "état d'esprit"** à ce moment. »

– Alan M. Turing, *On computable numbers*

« États d'esprit »

12932 +
19 =



J'ai lu le
chiffre 2

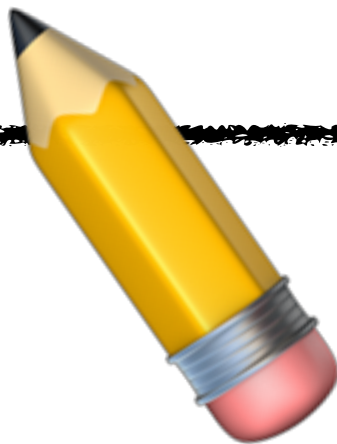


« États d'esprit »

12932 +

19 =

J'ai lu le
chiffre 2 et le
chiffre 9



« États d'esprit »

$$\begin{array}{r} 12932 \\ + \\ 19 \\ \hline \end{array}$$

Il faut que
j'écrive 1 et que je
garde 1 comme
retenue



« États d'esprit »

$$\begin{array}{r} 12932 \\ + \\ 19 \\ \hline \end{array}$$

Il faut que je
me déplace à
gauche ; la
retenue est 1

1



« États d'esprit »

$$\begin{array}{r} 12932 \\ + \\ 19 \\ \hline \end{array}$$

J'ai lu le
chiffre 3 ; avec la
retenue de 1 ça
fait 4

1



« États d'esprit »

12932 +

19 =

1

J'ai lu 4 et le
chiffre 1



« États d'esprit »

$$\begin{array}{r} 12932 \\ + \\ 19 \\ \hline \end{array}$$

Il faut que
j'écrive 5 ; pas
de retenue



1



« États d'esprit »

$$12932 + 19 =$$

Il faut que je
me déplace à
gauche

51



« On suppose également que le **nombre d'états d'esprit** qu'on doit prendre en compte soit **fini**. Les raisons pour cela sont de la même nature que celles qui restreignent le nombre de symboles. »

– Alan M. Turing, *On computable numbers*

États d'esprit trop proches

J'ai lu la
séquence
9999999999



États d'esprit trop proches

J'ai lu la
séquence
9999999999

J'ai lu la
séquence
9999999999



« On peut éviter l'utilisation d'états d'esprit plus compliqués **en écrivant plus de symboles** sur le ruban. »

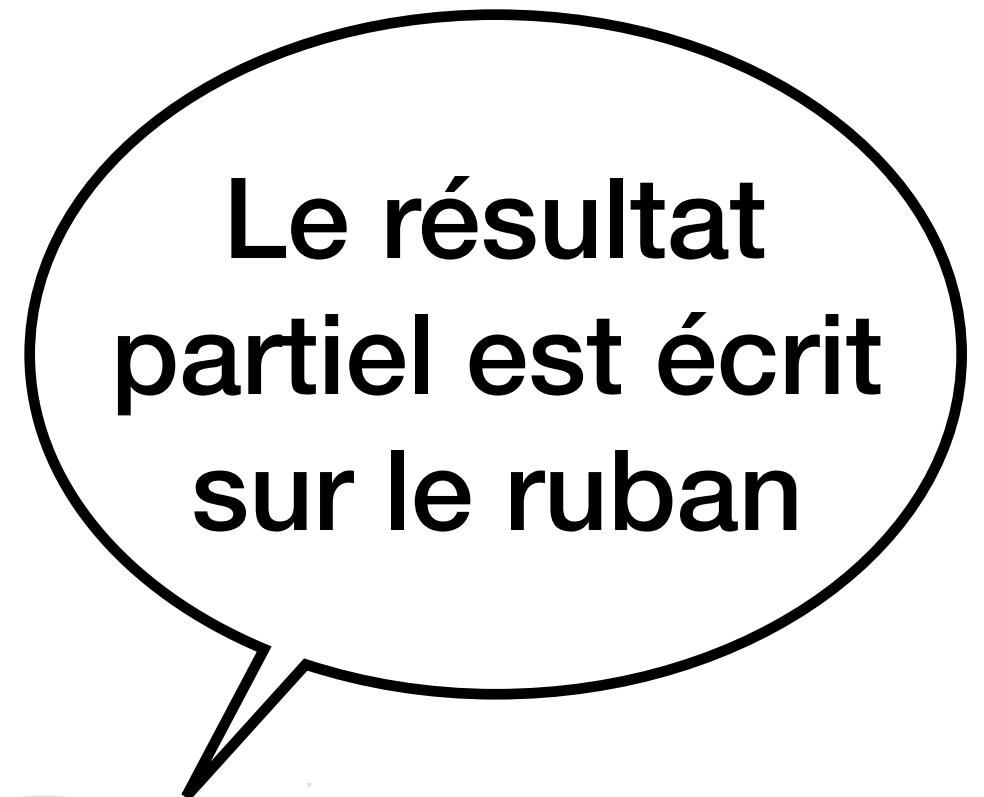
– Alan M. Turing, *On computable numbers*

Prendre note sur le ruban

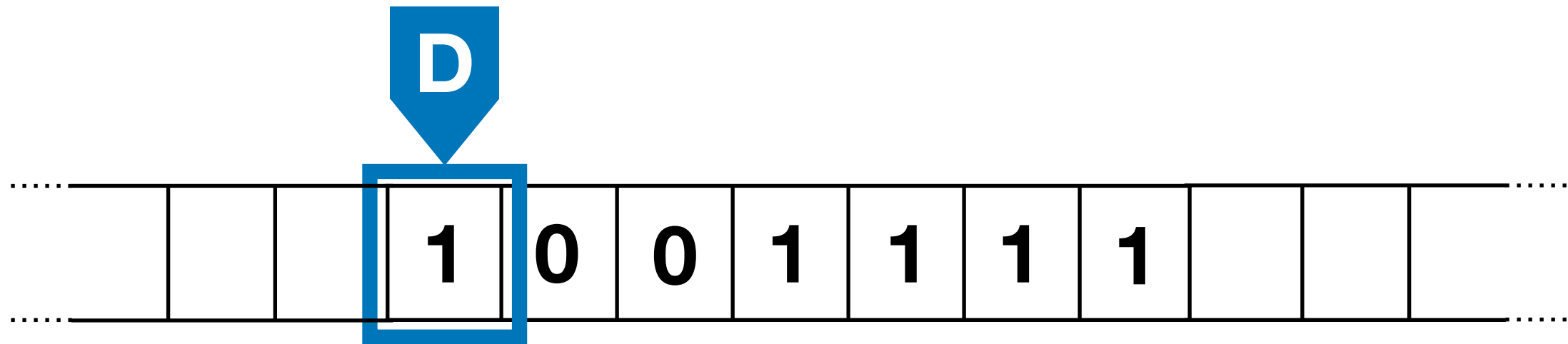
Le résultat
partiel est
9999999999



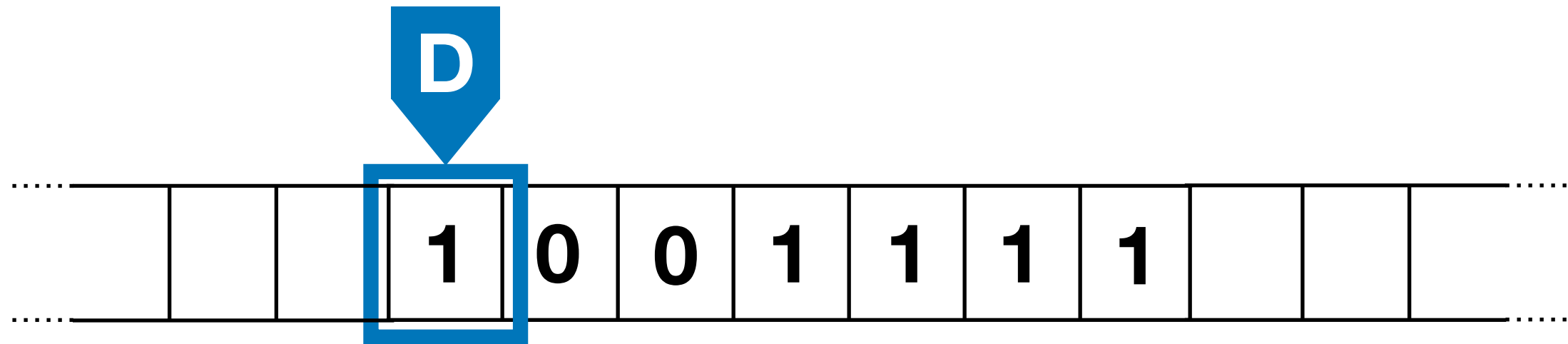
Prendre note sur le ruban



Incrémentation

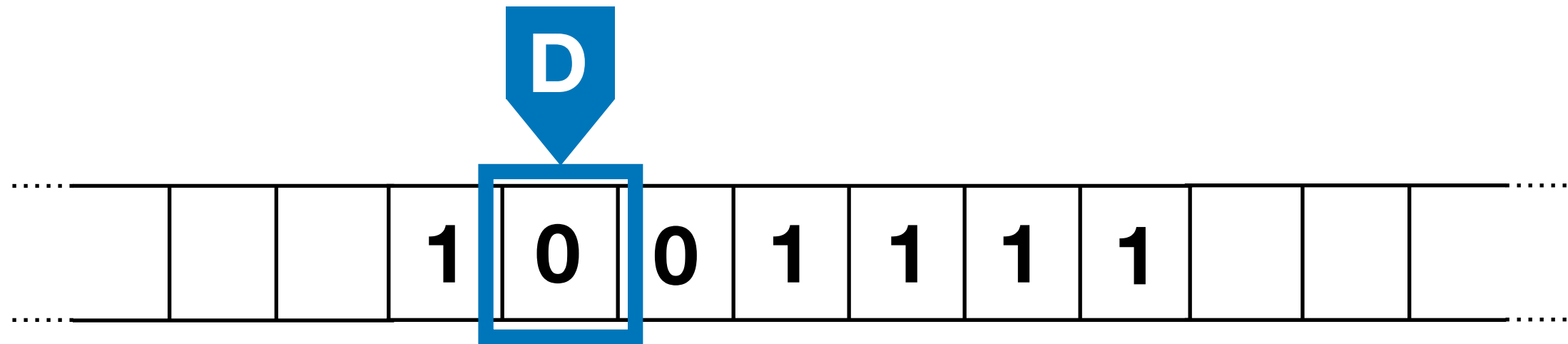


Incrémentation



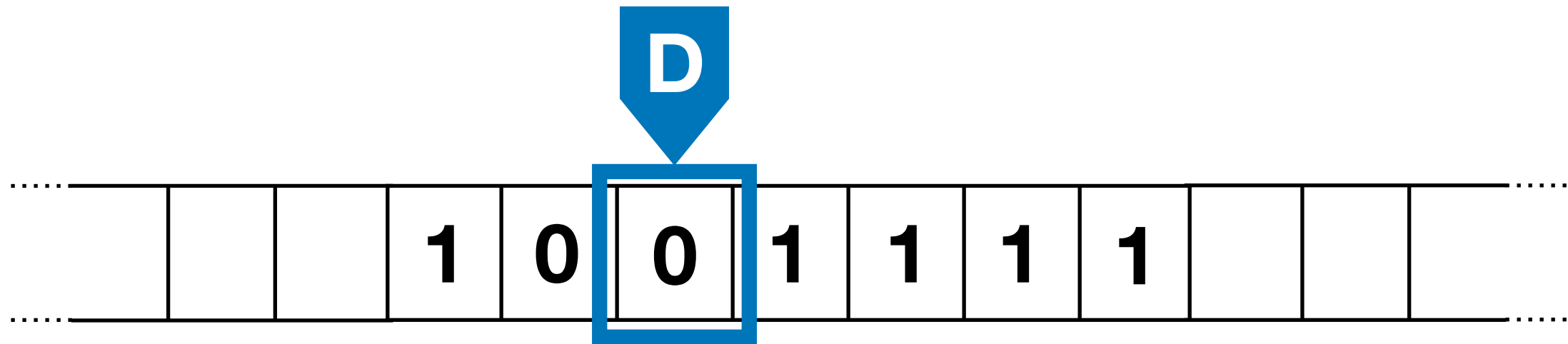
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



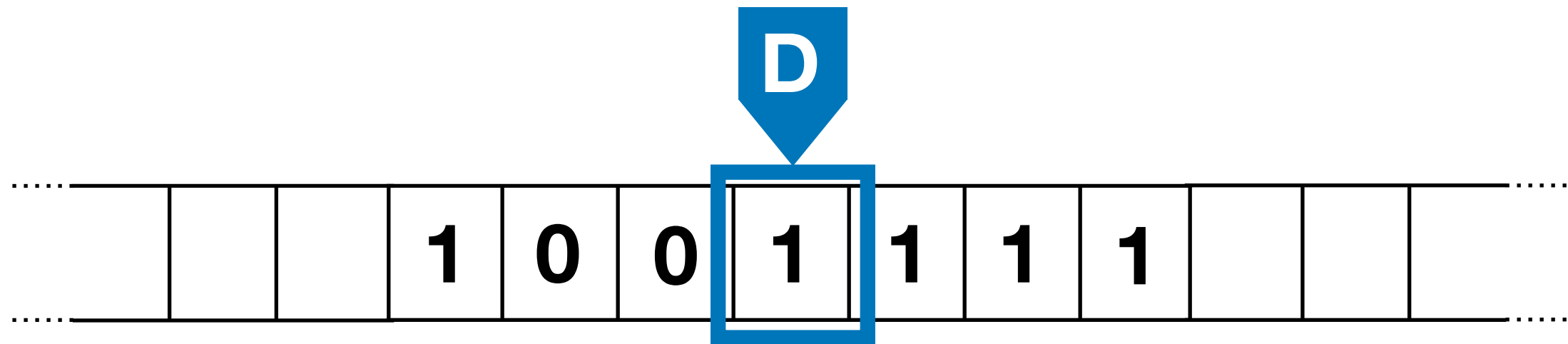
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



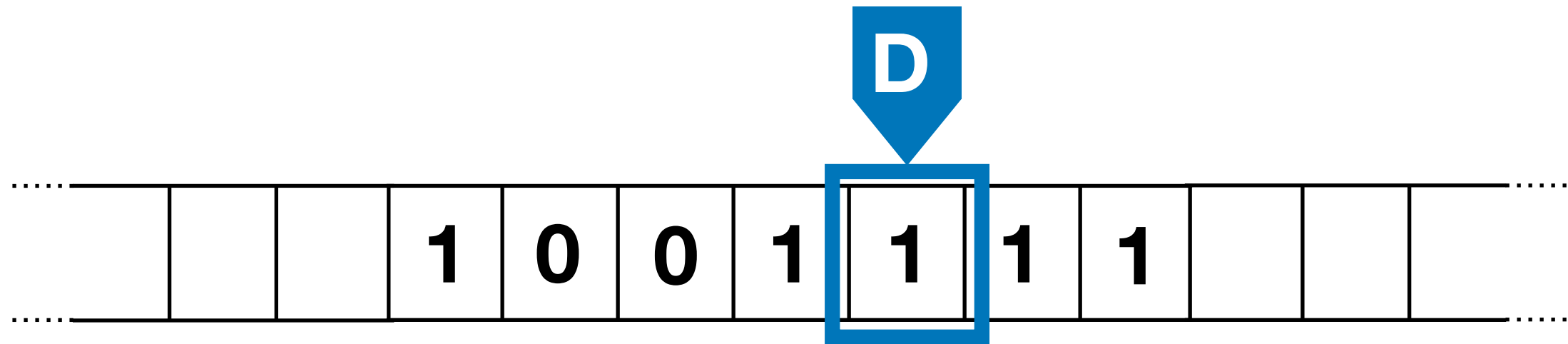
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



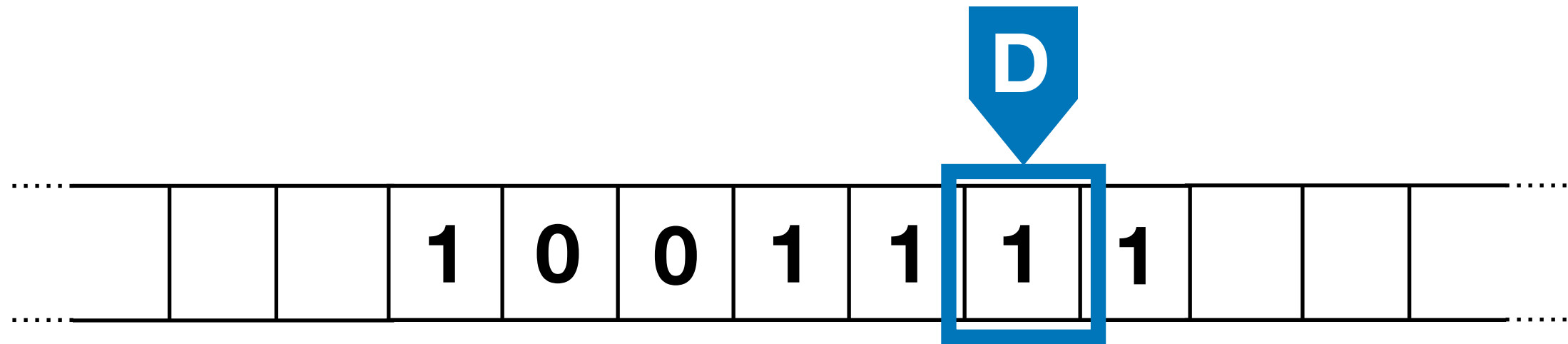
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



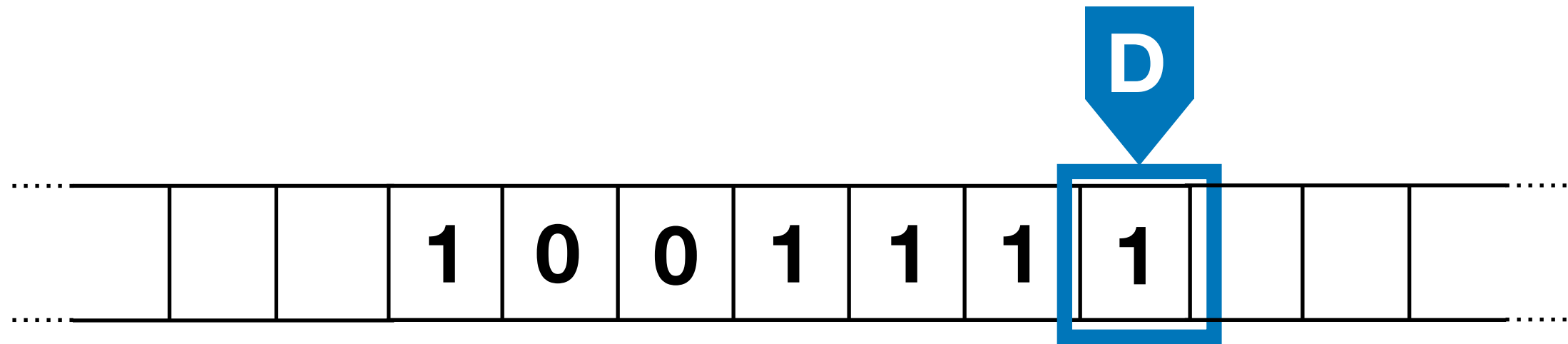
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



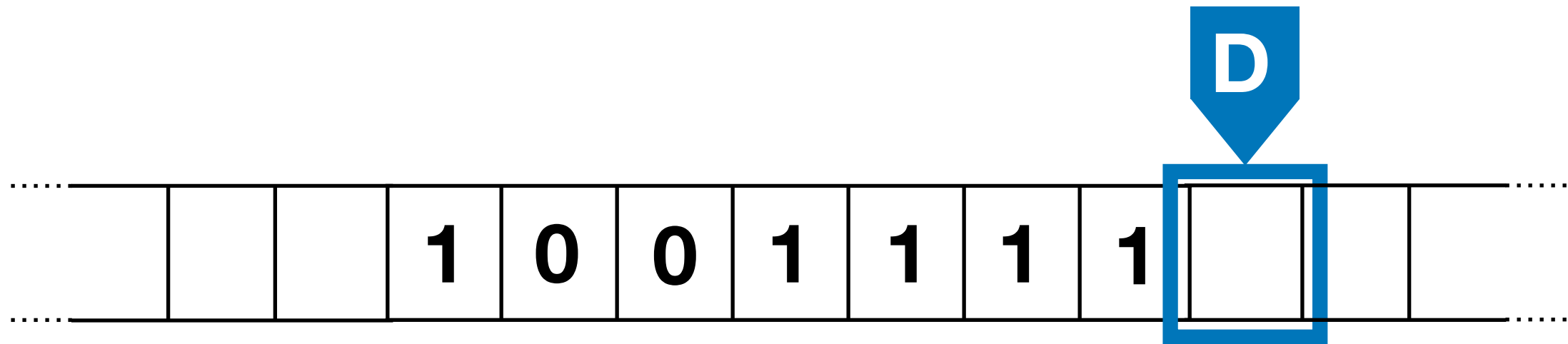
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



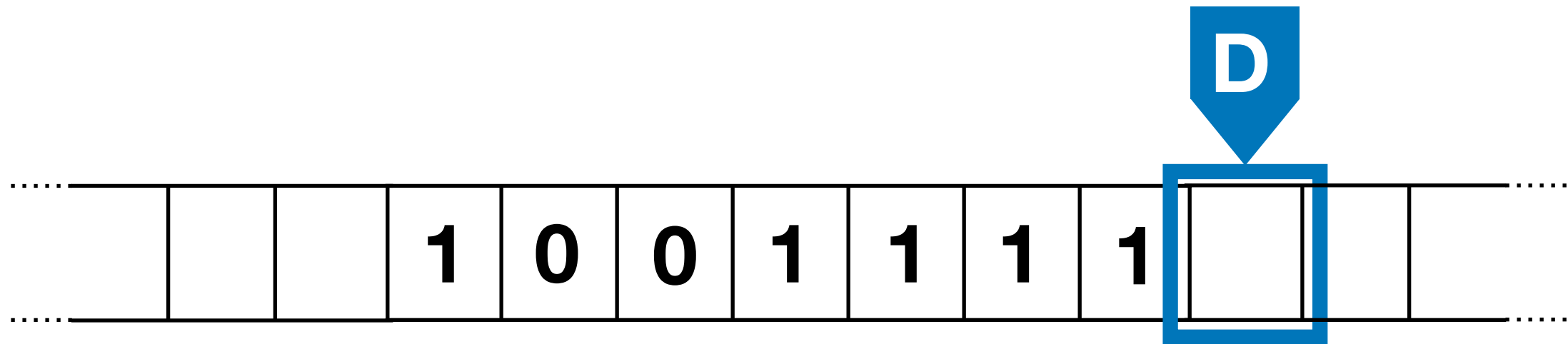
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



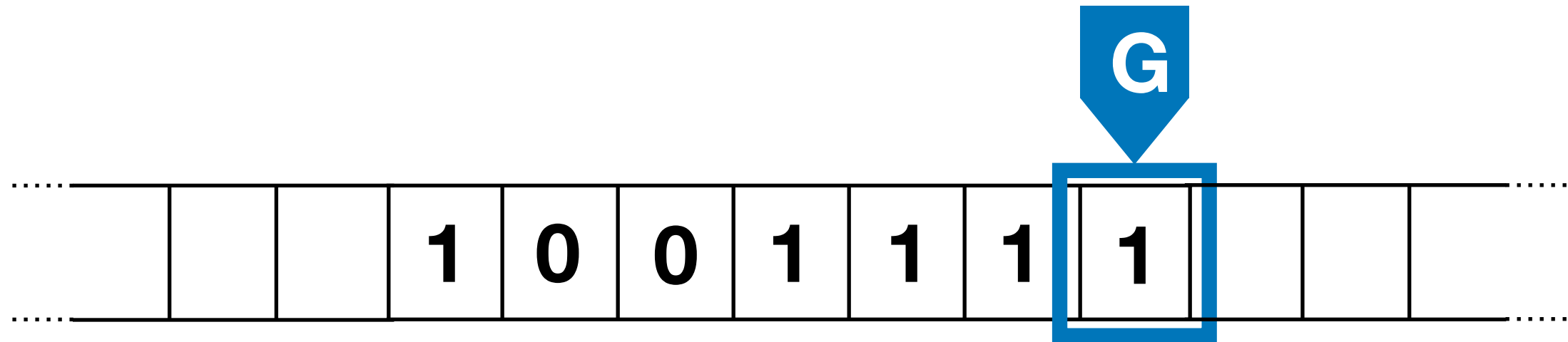
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D

Incrémentation



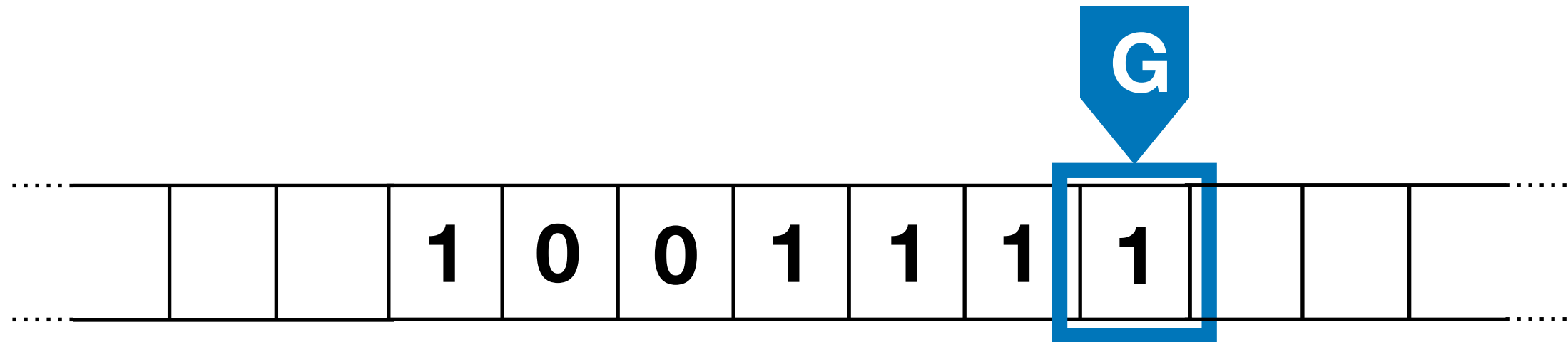
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G

Incrémentation



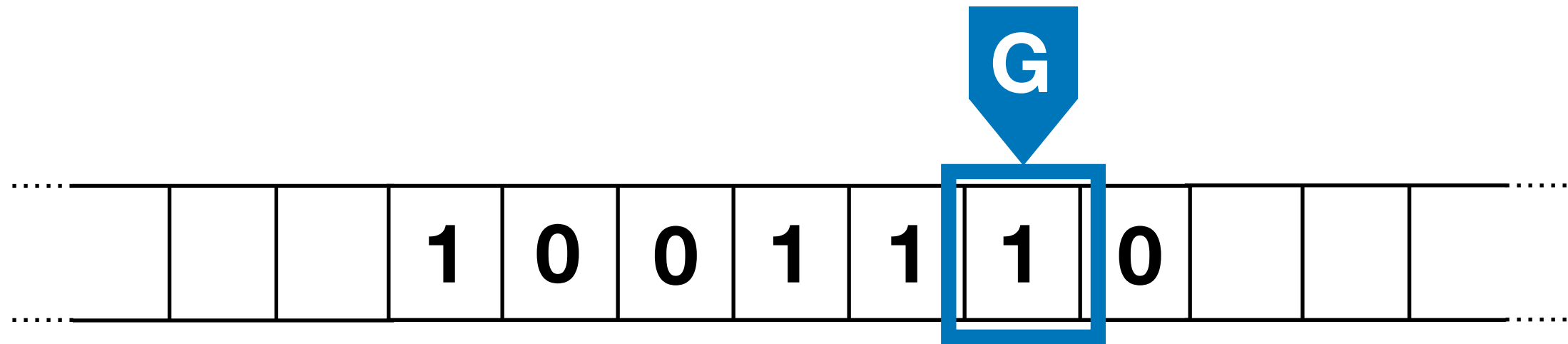
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G

Incrémentation



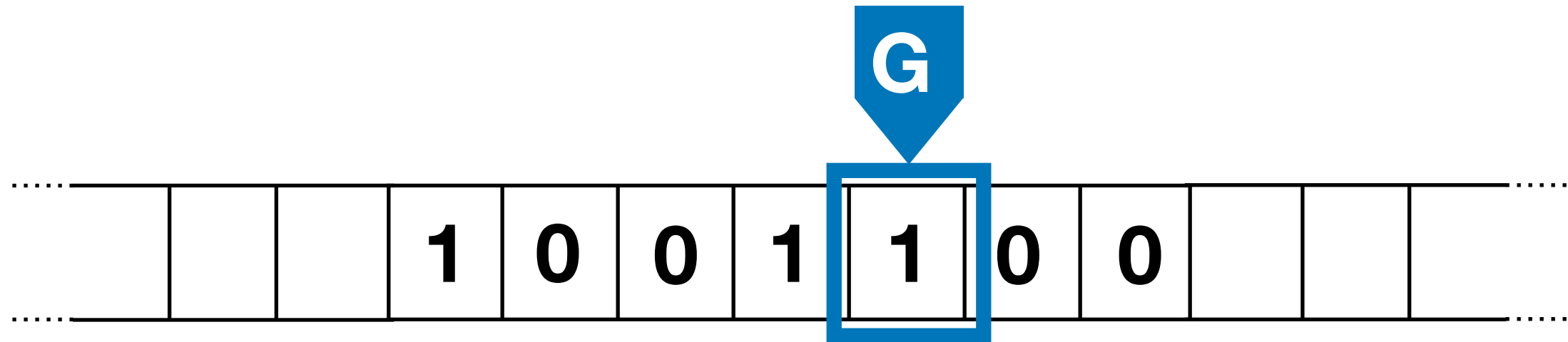
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		↑		G
G	1	↑	0	G

Incrémentation



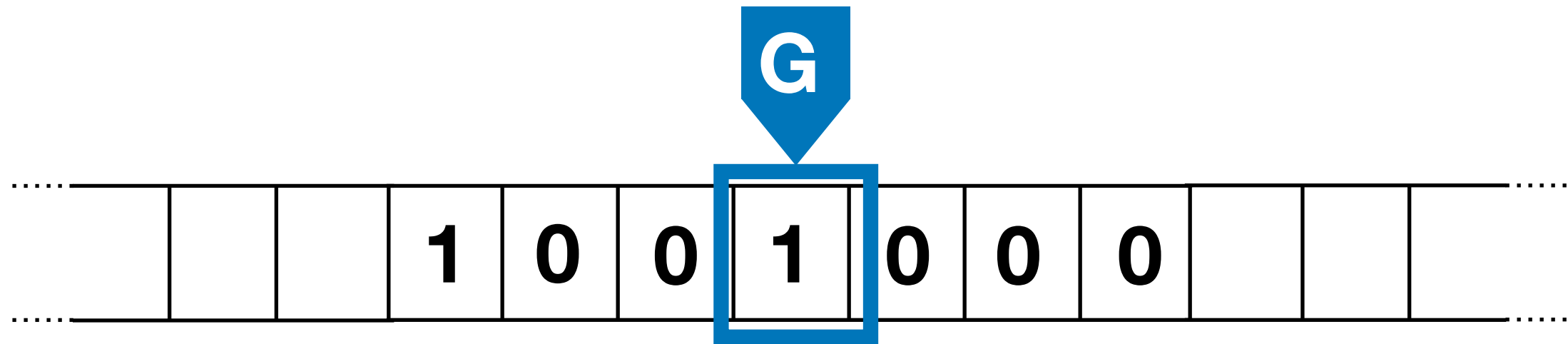
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		↑		G
G	1	↑	0	G

Incrémentation



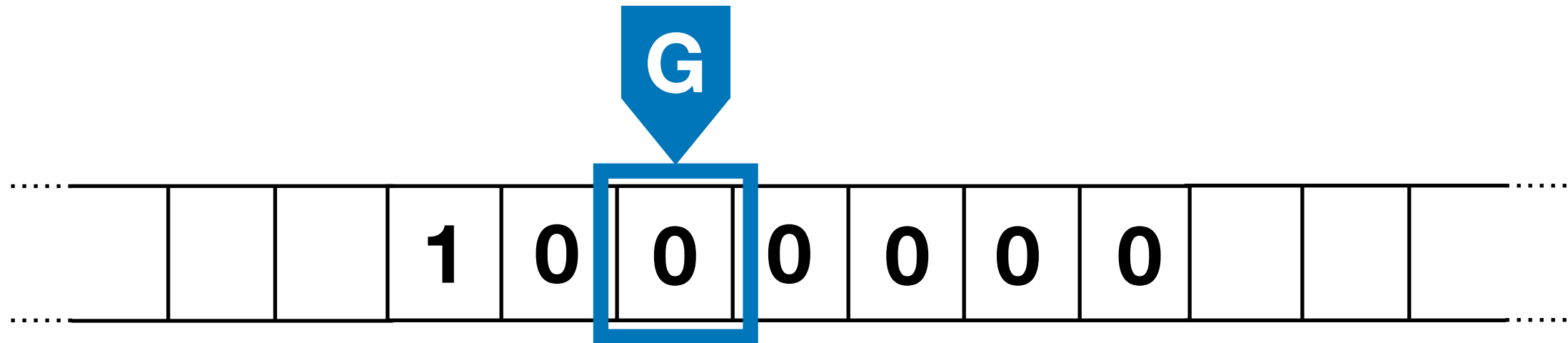
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		↑		G
G	1	↑	0	G

Incrémentation



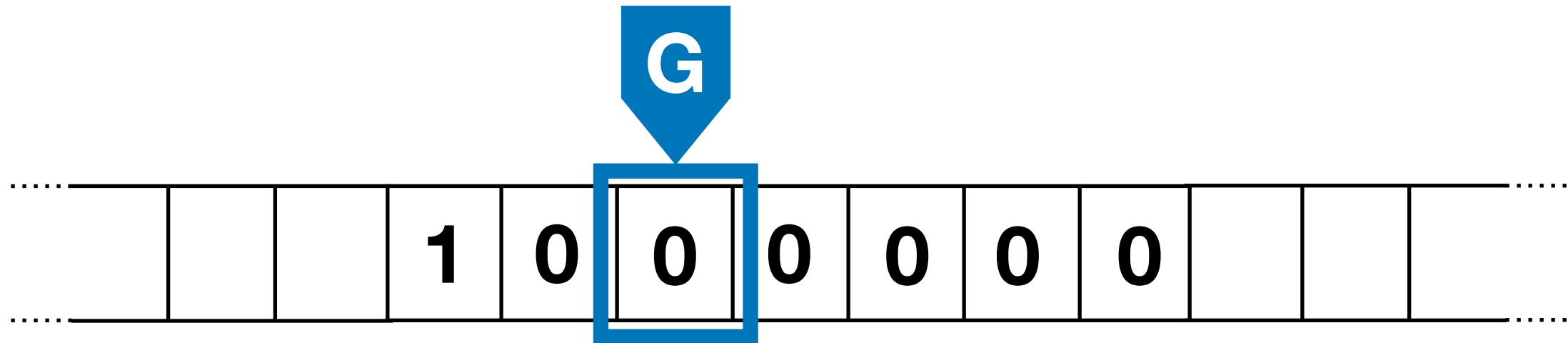
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G

Incrémentation



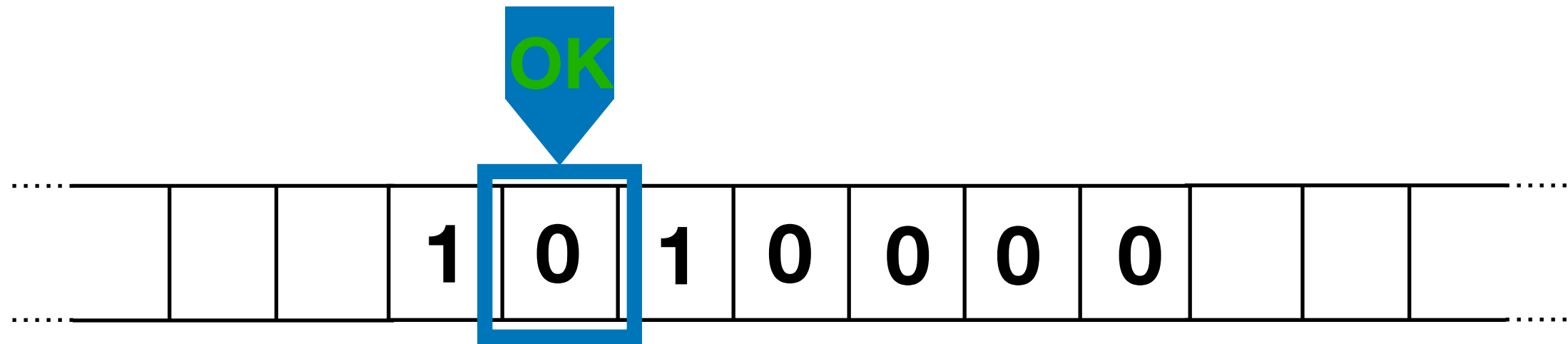
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G

Incrémentation



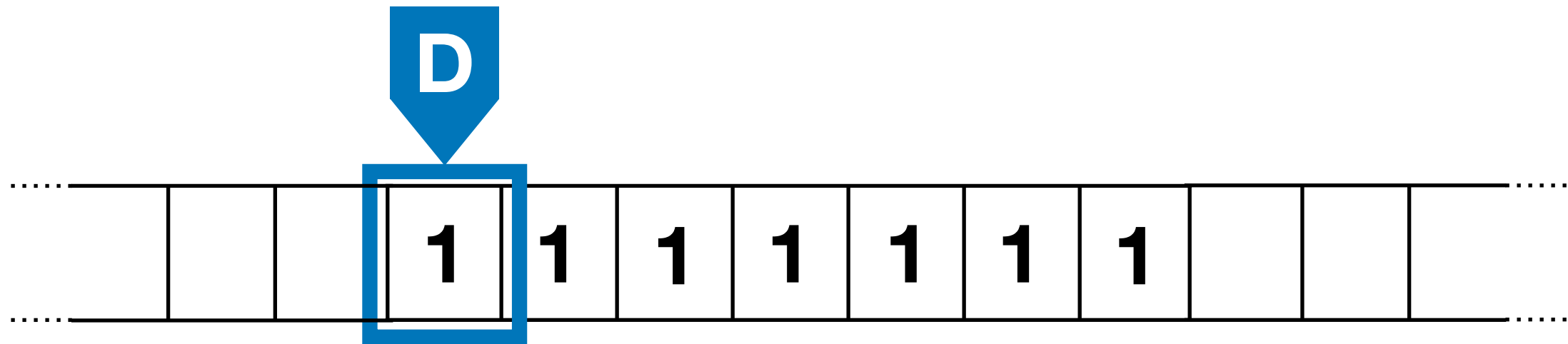
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



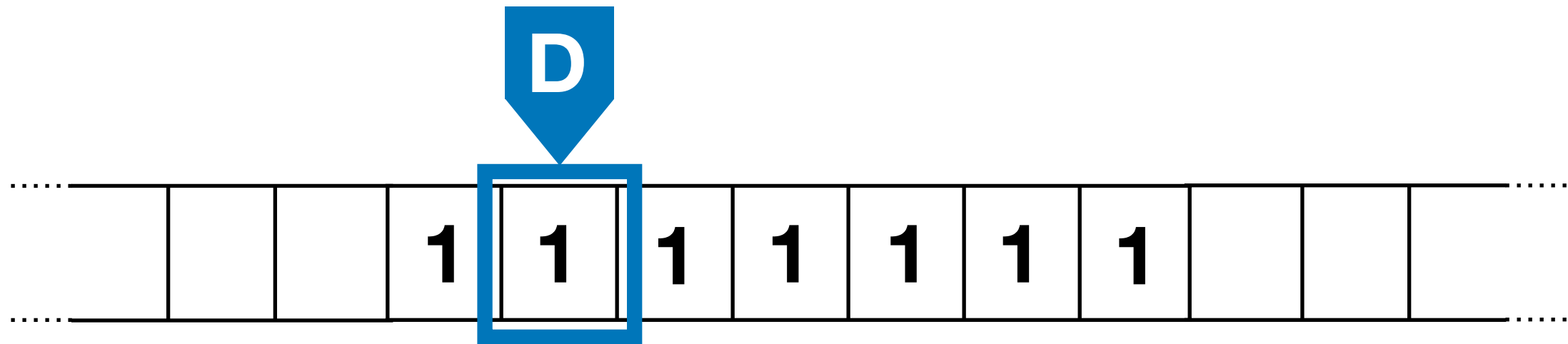
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



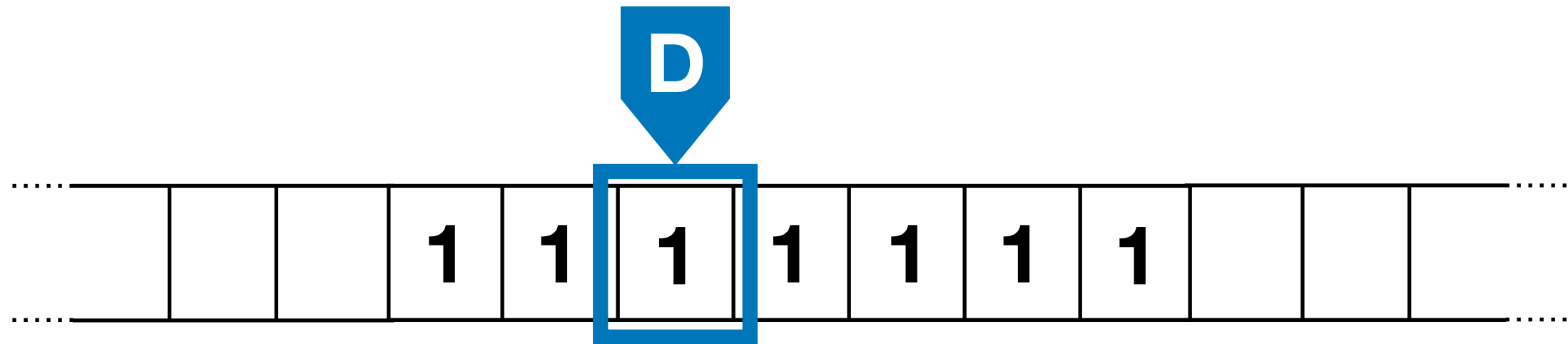
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



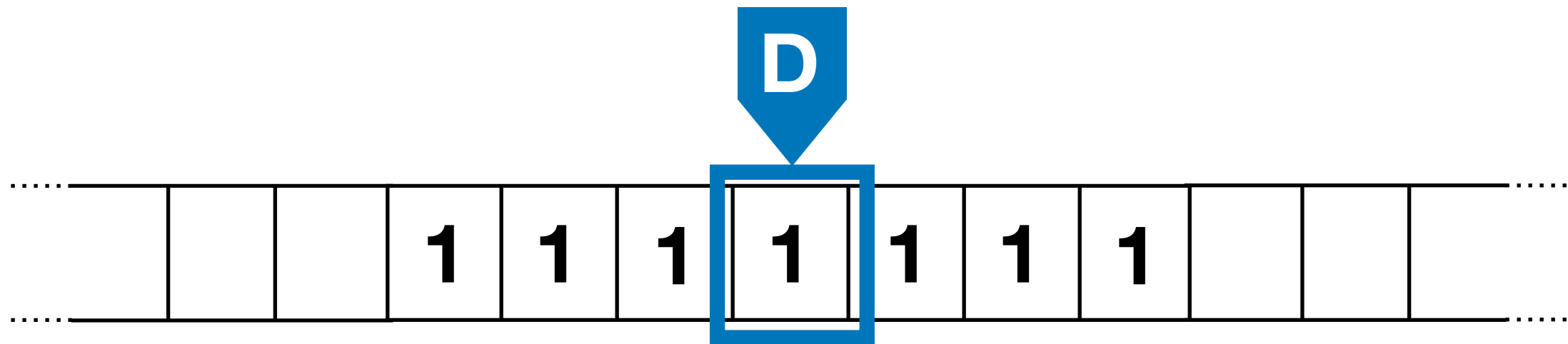
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



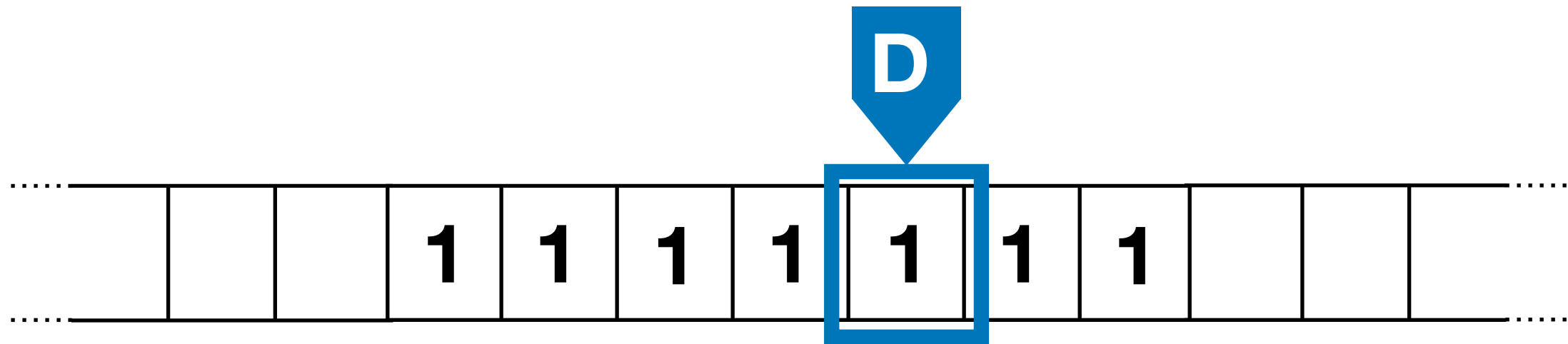
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



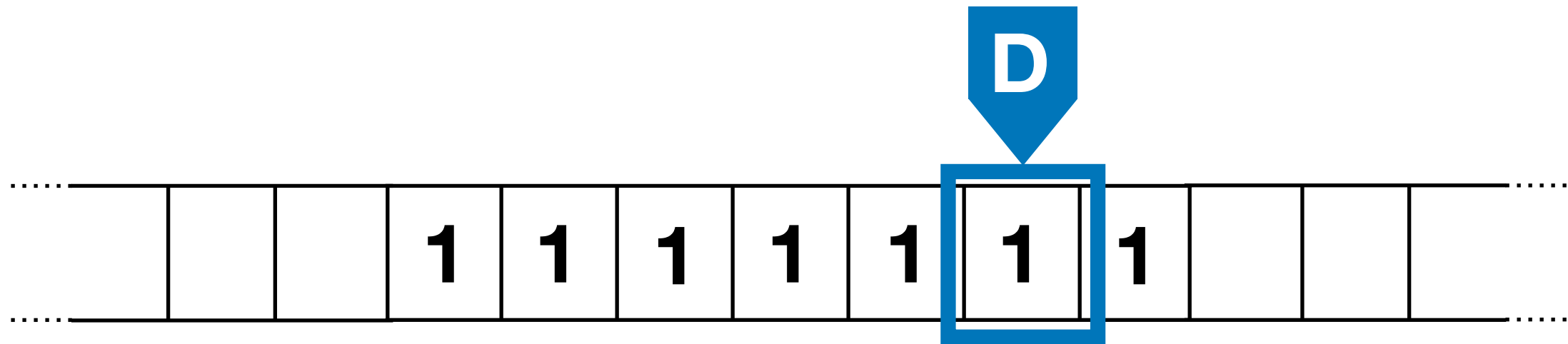
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



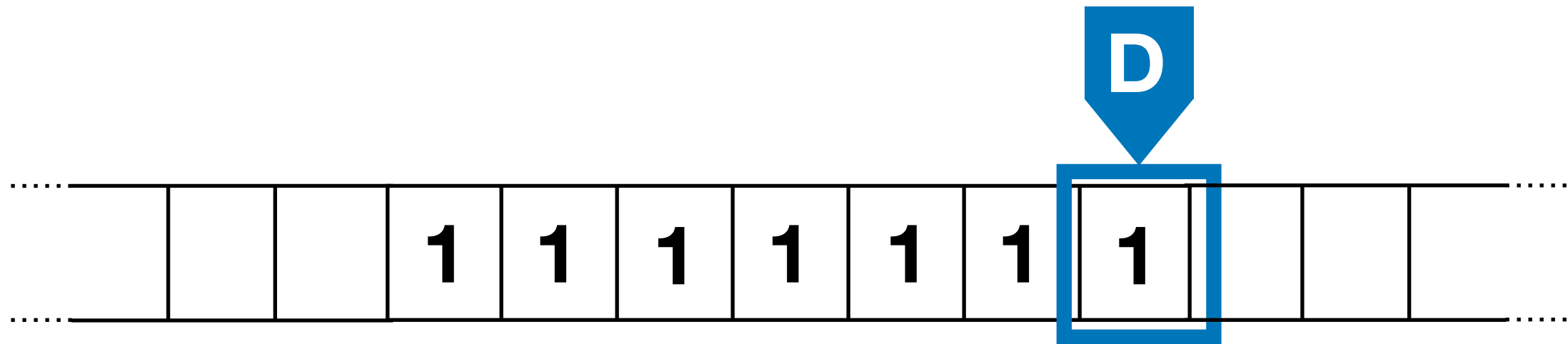
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



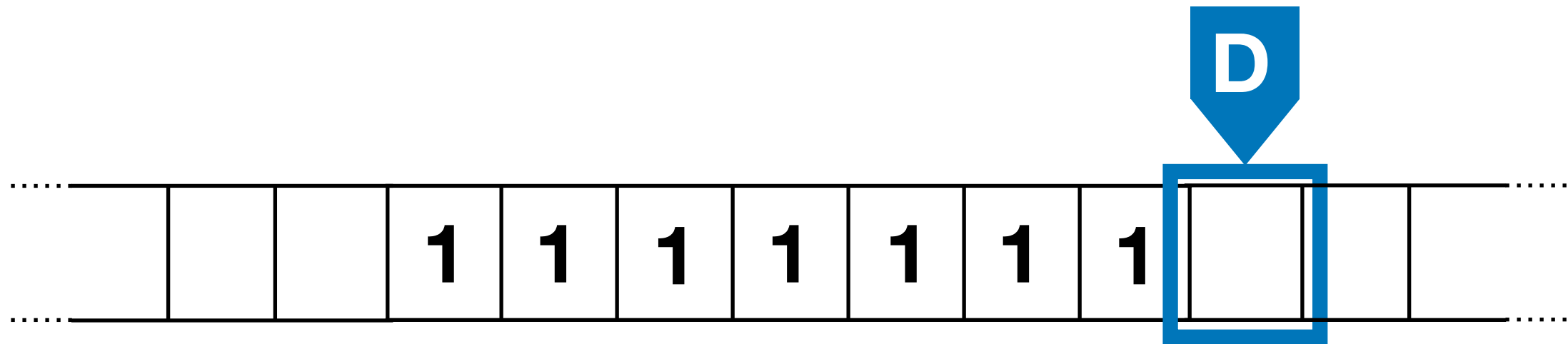
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



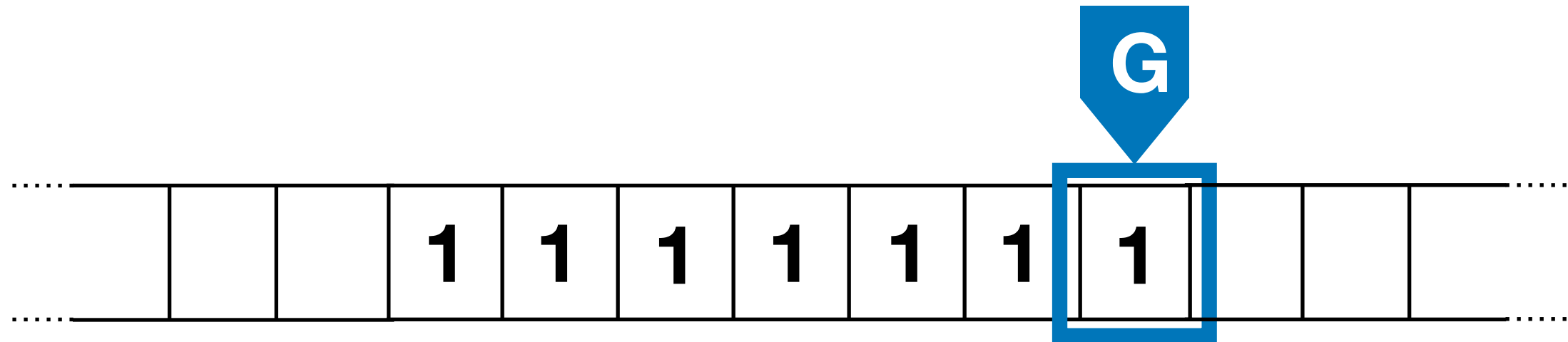
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



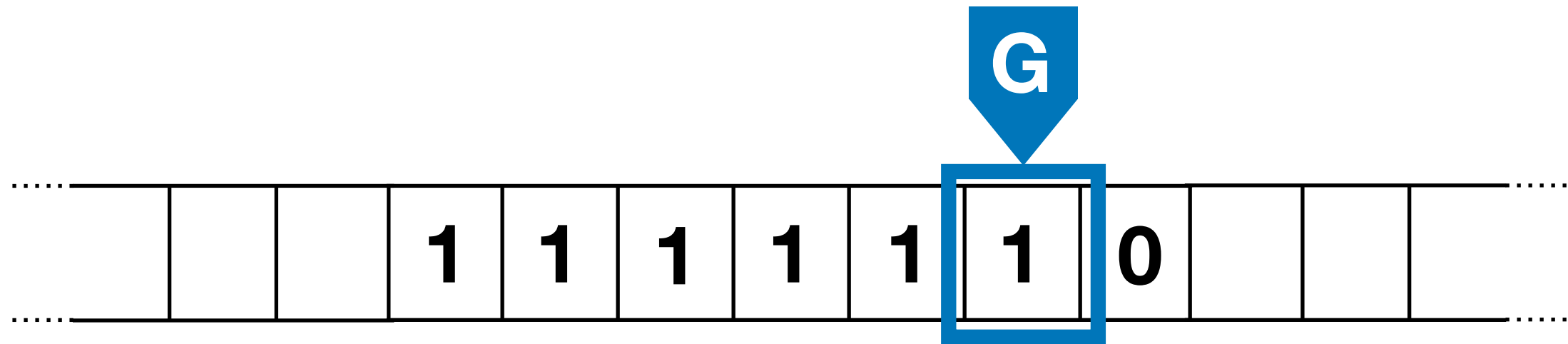
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



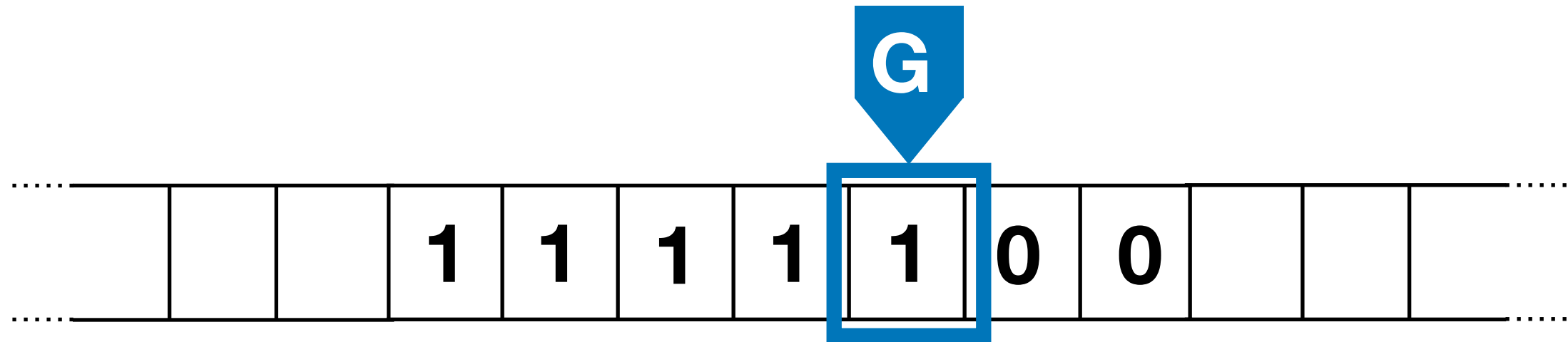
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



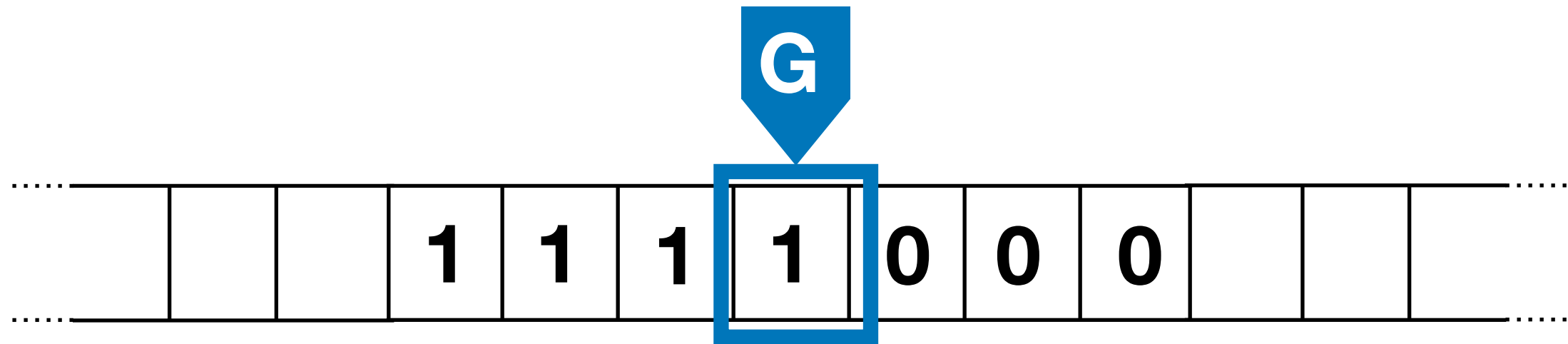
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



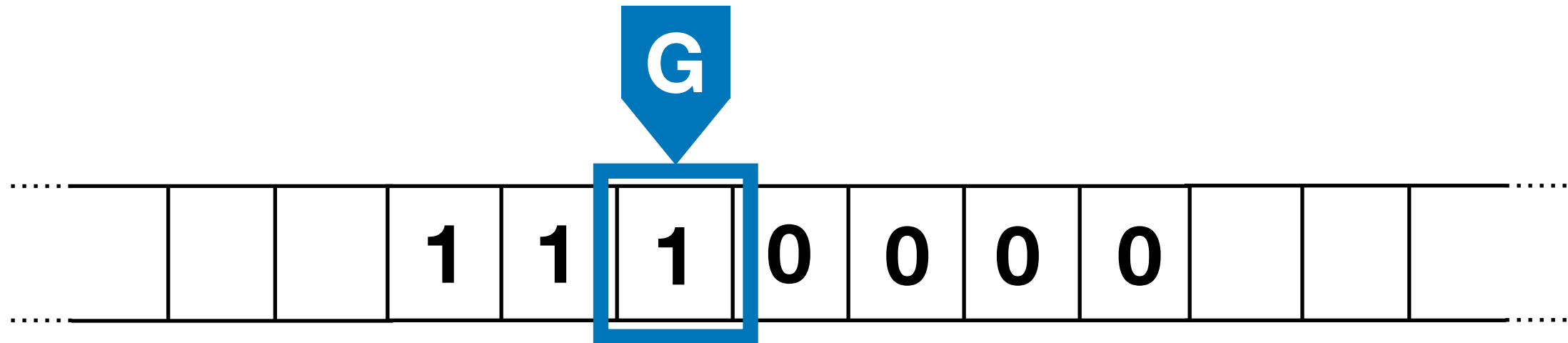
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



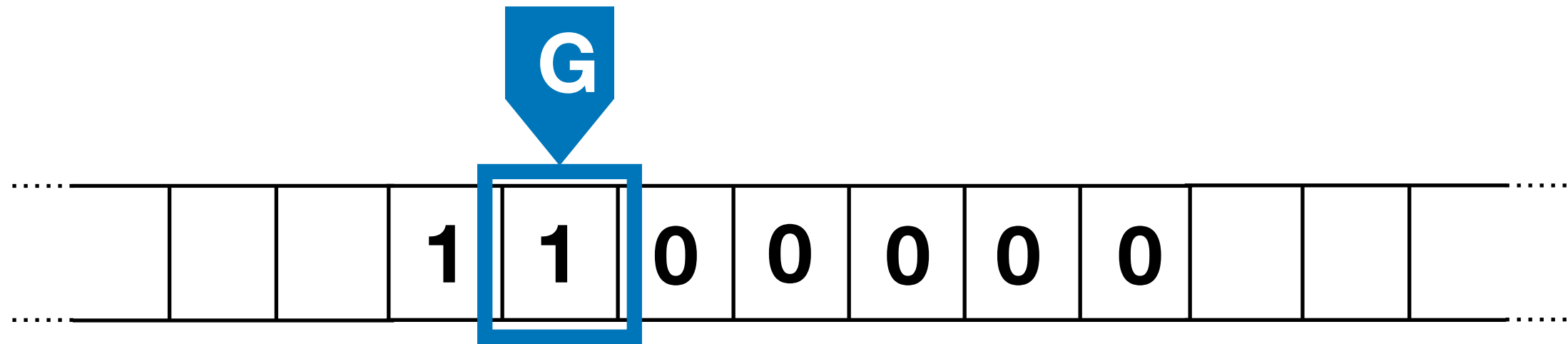
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



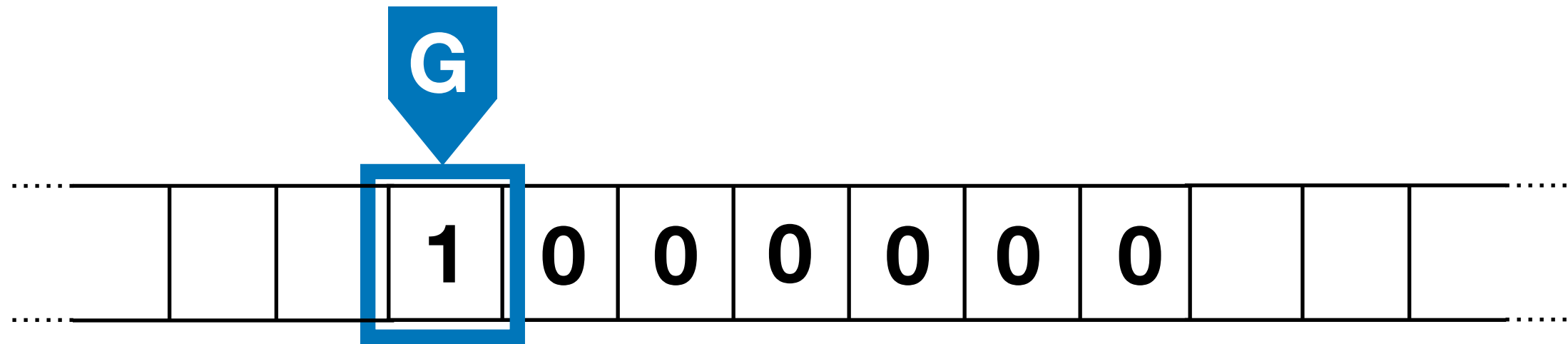
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



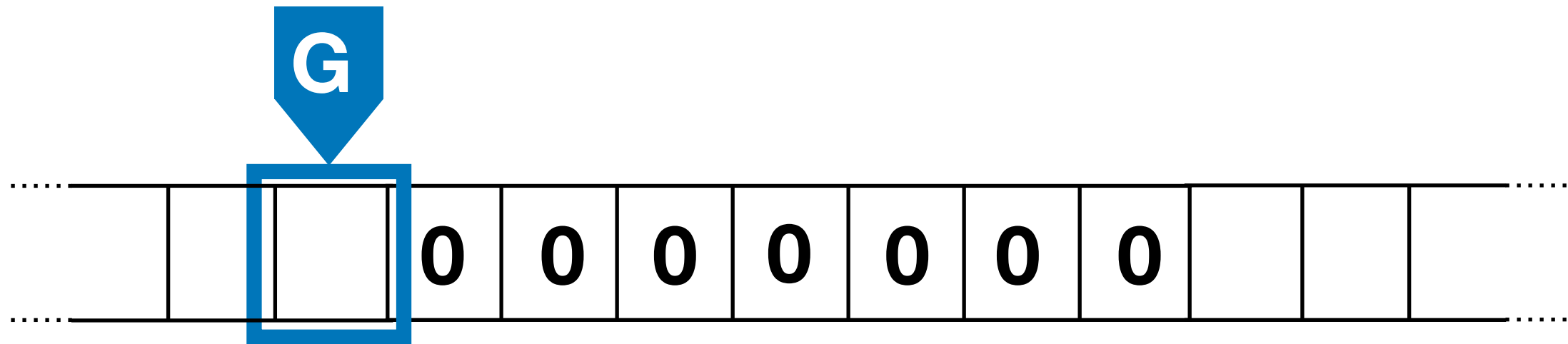
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



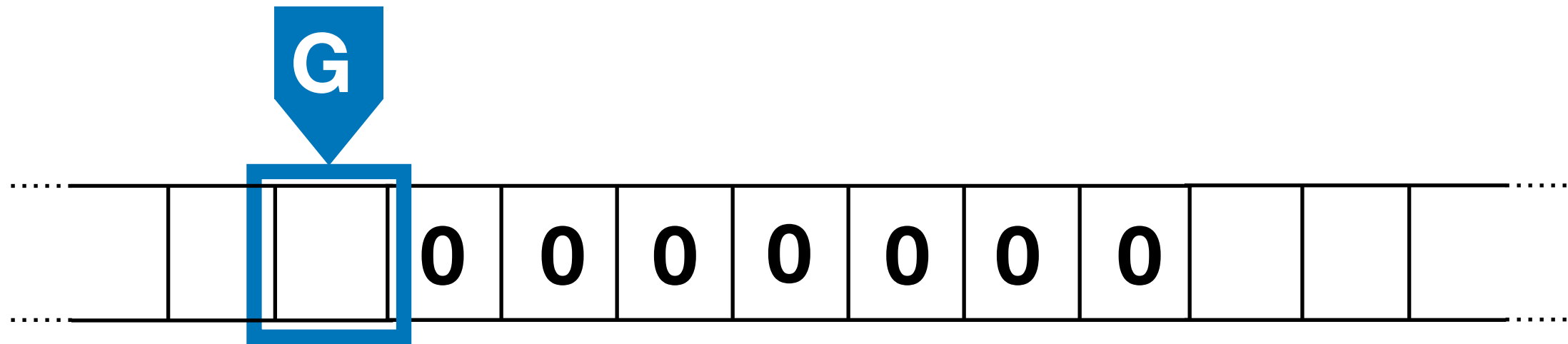
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



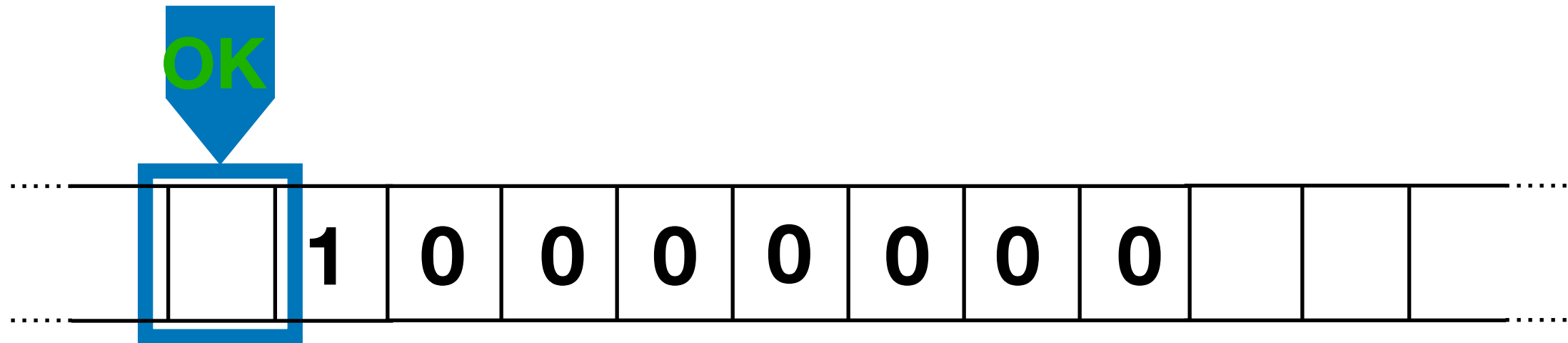
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK
G		←	1	OK

Incrémentation



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK
G		←	1	OK

**Machine de Turing =
pseudo-code =
ordinateur moderne**

Pseudo-code = machine de Turing

- Un peu d'arithmétique
- Séquence d'instructions
- Conditions (**si... alors... sinon**)
- Itération (boucles **tant que... faire**, **pour... faire**)

Pseudo-code = machine de Turing

- Un peu d'arithmétique → voir exemples précédents
- Séquence d'instructions → séquence d'états de la MT
- Conditions (**si... alors... sinon**) → table de transition
- Itération (boucles **tant que... faire**, **pour... faire**)
→ répétition d'états de la MT

Algorithmes en pseudo-code sur la machine de Turing

$i = 5$ $j = 7$ $T =$

4	2	3	1	0	1	6	0
---	---	---	---	---	---	---	---

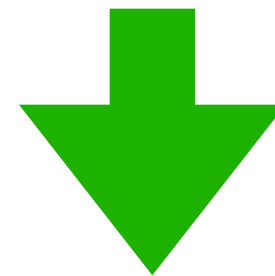
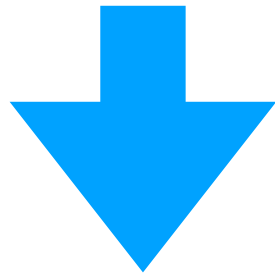
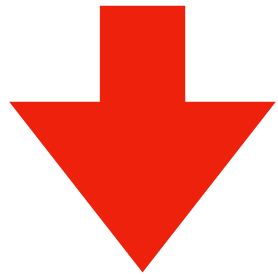
Algorithmes en pseudo-code sur la machine de Turing

$i = 5$

$j = 7$

$T =$

4	2	3	1	0	1	6	0
---	---	---	---	---	---	---	---



i	1	0	1	j	1	1	1	T	1	0	0	,	1	0	,	1	1	,
-----	---	---	---	-----	---	---	---	-----	---	---	---	---	---	---	---	---	---	---

Simuler la machine de Turing en pseudo-code

Etat	Symbole	Sens	Nouvel eau symbole	Nouvel état
A	0	→	1	B
A	1	←	1	A
A		←	0	B
B	0	←	1	A
B	1	→	0	B
B	0	→	1	B

tant que état ≠ OUI **et** état ≠ NON **faire**

si état = A **et** on lit 0 **alors**

état := B

écrire 1

se déplacer à droite

sinon si état = A **et** on lit 1 **alors**

état := A

écrire 1

se déplacer à gauche

sinon si...

Peut-on tout calculer ?

- Étant donné un problème bien formulé d'un point de vue mathématique...
- ...avec des entrées sous la forme de séquence de symboles et sortie **oui** ou **non**...
- ...existe-t-il toujours une machine di Turing (ou un algorithme en pseudo-code) qui puisse résoudre le problème ?

Peut-on tout calculer ?



Peut-on tout calculer ?



Est-ce que pour tout problème, il existe une machine de Turing le résolvant ?

Peut-on tout calculer ?



Est-ce que pour tout problème, il existe une machine de Turing le résolvant ?

avec des entrées sous la
forme d'une séquence et des
sorties « OK / pas OK »

Problème de l'arrêt



Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

+

Ruban de départ

Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

**+
Ruban de départ**

**Est-ce que la machine
/ le pseudo-code s'arrête
sur le ruban de départ ?**

Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

+

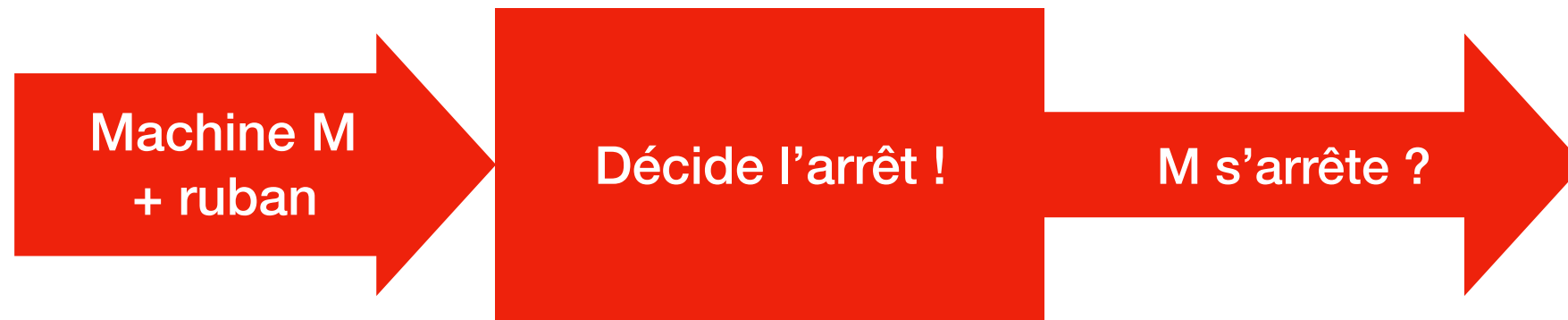
Ruban de départ

**Est-ce que la machine
/ le pseudo-code s'arrête
sur le ruban de départ ?**

Supposons qu'il existe une machine / du pseudo-code qui réponde à cette question !



Problème de l'arrêt



Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):
```

Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
    Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors
```

Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
    Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors  
        x := 0  
        Tant que x ≥ 0 faire  
            x := x+1  
        FinTantQue
```

Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
    Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors  
        x := 0  
        Tant que x ≥ 0 faire  
            x := x+1  
        FinTantQue  
    Sinon  
        retourner(OK)  
    FinSi
```

Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
  Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors  
    x := 0  
    Tant que x ≥ 0 faire  
      x := x+1  
    FinTantQue  
  Sinon  
    retourner(OK)  
  FinSi
```

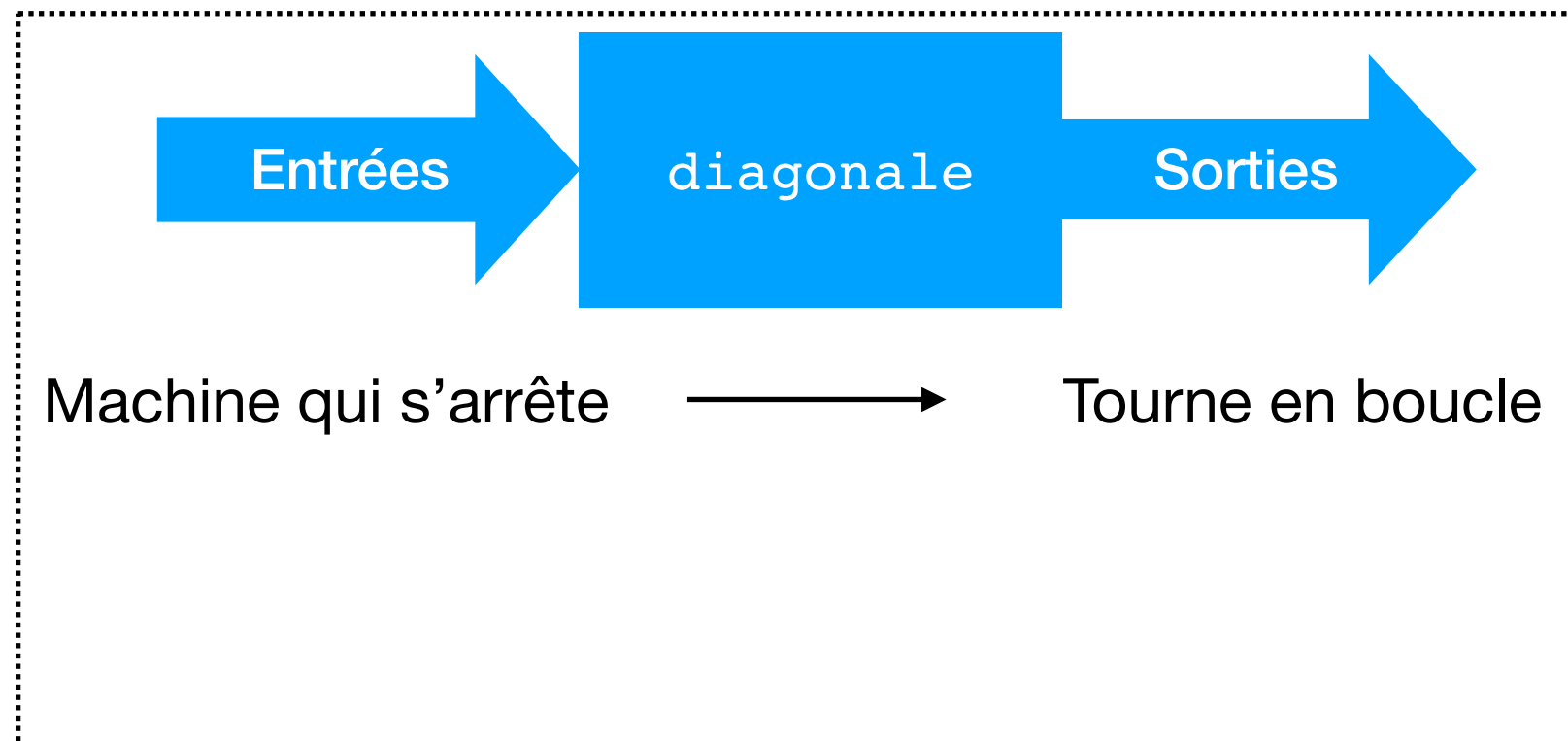


Problème de l'arrêt



Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
  Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors  
    x := 0  
    Tant que x ≥ 0 faire  
      x := x+1  
    FinTantQue  
  Sinon  
    retourner(OK)  
  FinSi
```

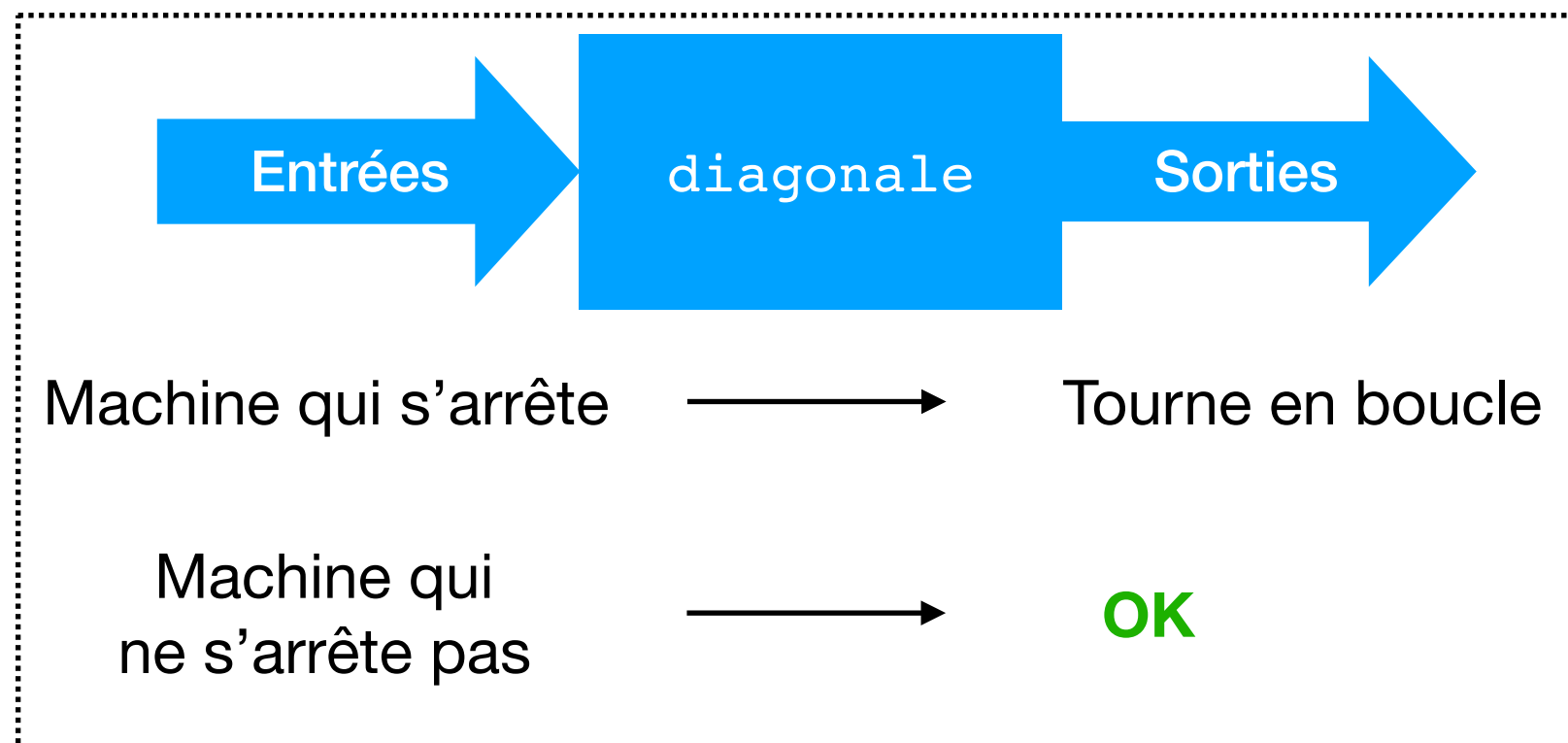


Problème de l'arrêt

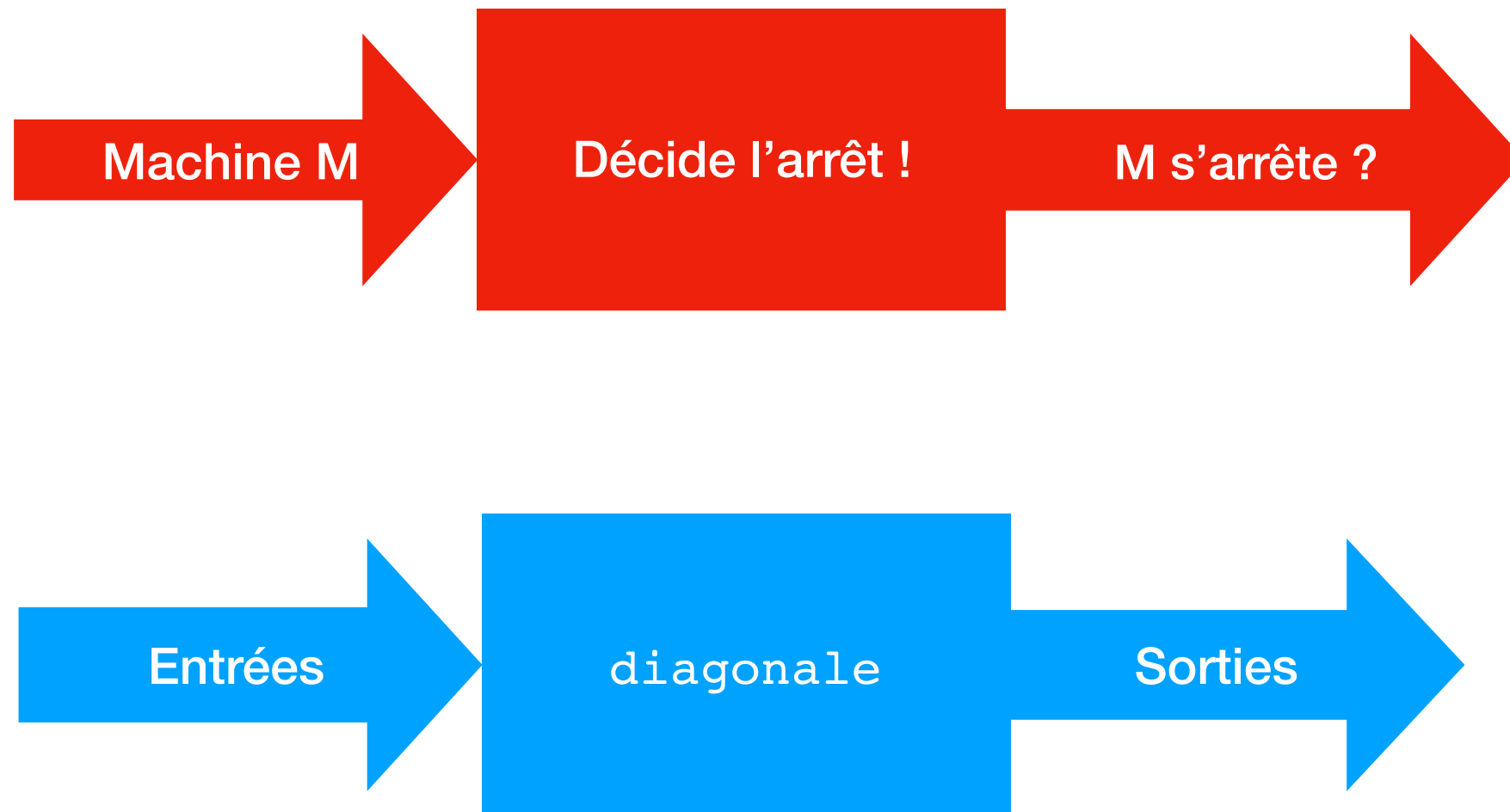


Construisons une nouvelle machine
qui suit le pseudo-code suivant :

```
fonction diagonale(M : machine):  
  Si M s'arrête sur un ruban de départ contenant le pseudo-code de M alors  
    x := 0  
    Tant que x ≥ 0 faire  
      x := x+1  
    FinTantQue  
  Sinon  
    retourner(OK)  
FinSi
```



Problème de l'arrêt



Problème de l'arrêt



diagonale



???

Problème de l'arrêt



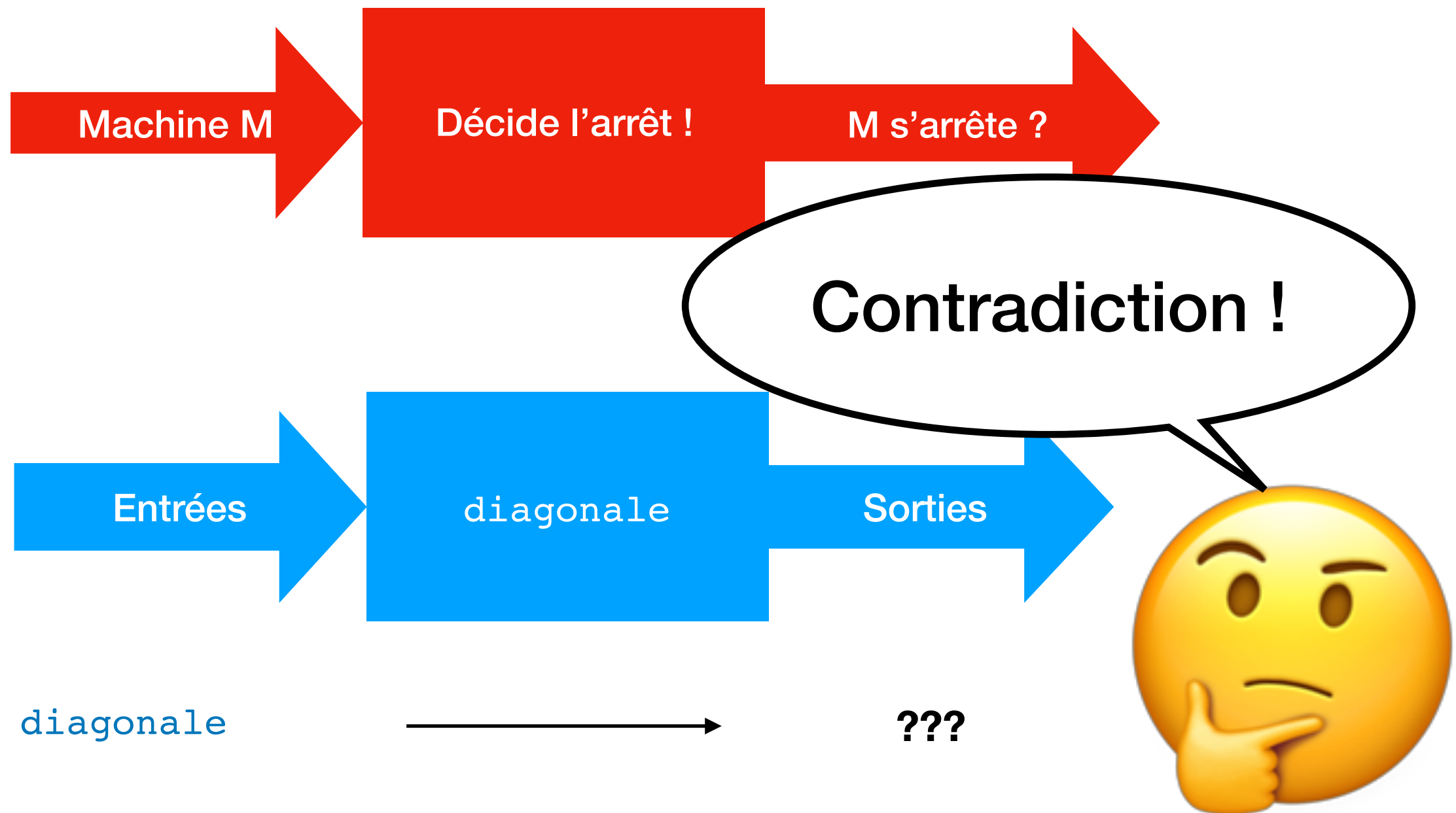
diagonale



???

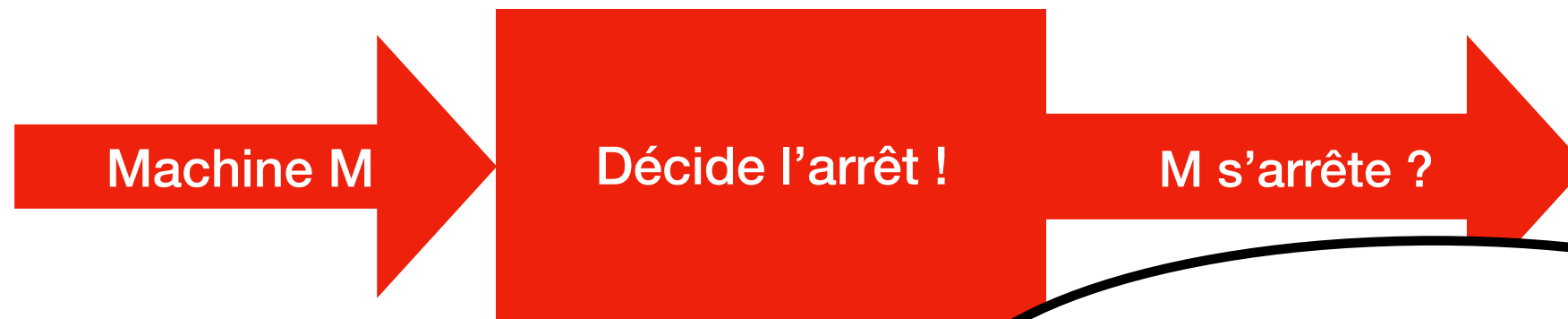
- Si `diagonale(diagonale)` s'arrête, alors on ne s'arrête pas
- Si `diagonale(diagonale)` ne s'arrête pas, alors on retourne OK (donc on s'arrête)

Problème de l'arrêt



- Si `diagonale(diagonale)` s'arrête, alors on ne s'arrête pas
- Si `diagonale(diagonale)` ne s'arrête pas, alors on retourne OK (donc on s'arrête)

Problème de l'arrêt



« ... Supposons qu'il existe une machine... »



diagonale



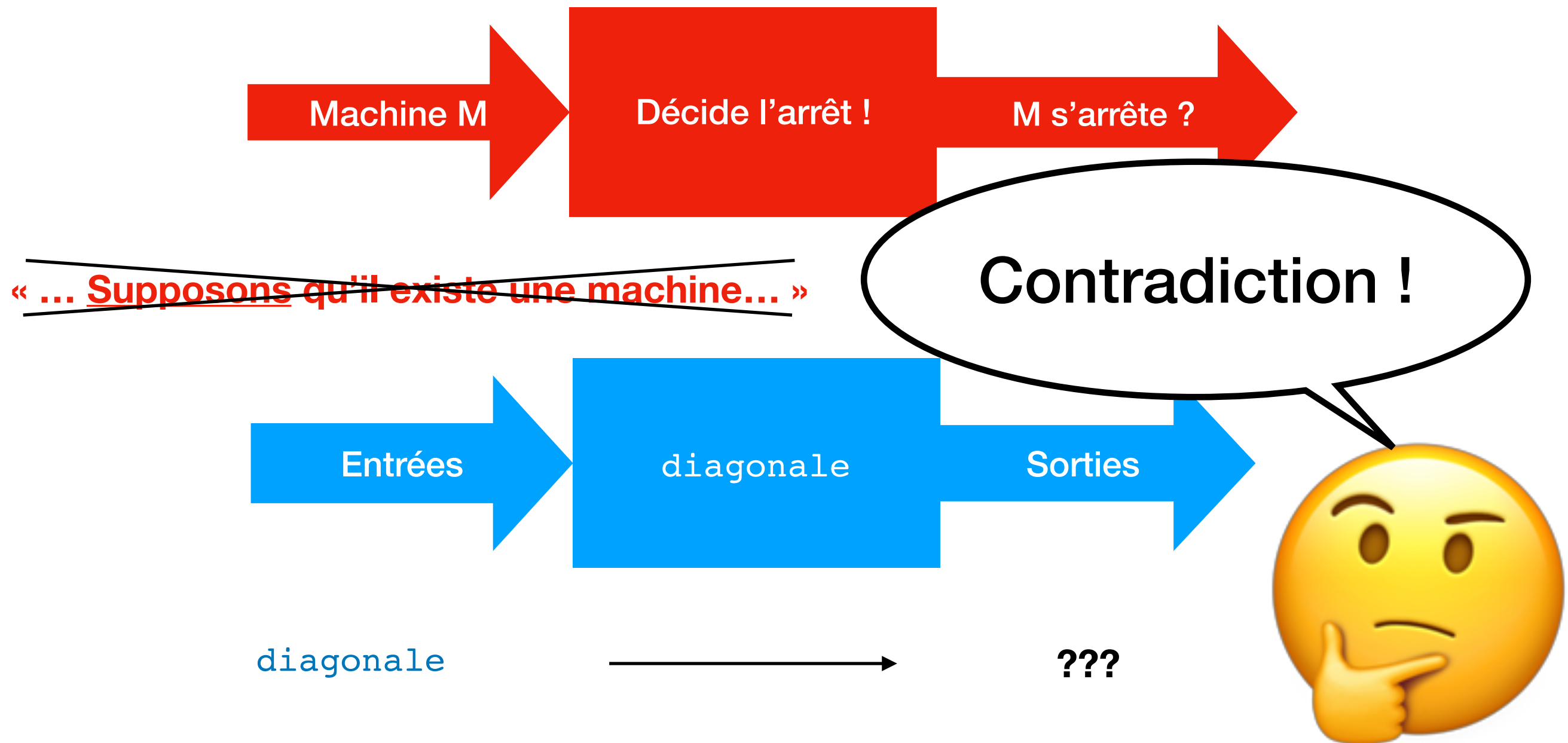
???

Contradiction !



- Si `diagonale(diagonale)` s'arrête, alors on ne s'arrête pas
- Si `diagonale(diagonale)` ne s'arrête pas, alors on retourne OK (donc on s'arrête)

Problème de l'arrêt



- Si `diagonale(diagonale)` s'arrête, alors on ne s'arrête pas
- Si `diagonale(diagonale)` ne s'arrête pas, alors on retourne OK (donc on s'arrête)

Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

**+
Ruban de départ**

**Est-ce que la machine
/ le pseudo-code s'arrête
sur le ruban de départ ?**

Problème de l'arrêt



**Machine de Turing
(= Pseudo-code)**

+

Ruban de départ

**Est-ce que la machine
/ le pseudo-code s'arrête
sur le ruban de départ ?**

**Il n'existe pas de machine de Turing résolvant le problème de l'arrêt !
On dit que le problème de l'arrêt est indécidable.**

Autres problèmes indécidables

- Est-ce qu'une machine de Turing donnée atteint un certain état (par exemple, d'acceptation) ?
- Est-ce qu'une proposition mathématique formalisée donnée est un théorème (c'est-à-dire, a-t-elle une preuve) ? ← « Entscheidungsproblem » (problème de la décision) de Hilbert
- Est-ce qu'un polynôme à plusieurs variables et à coefficients entiers donné admet une racine entière ?
← 10ème problème de Hilbert