

3.6 Sémaphores IPC

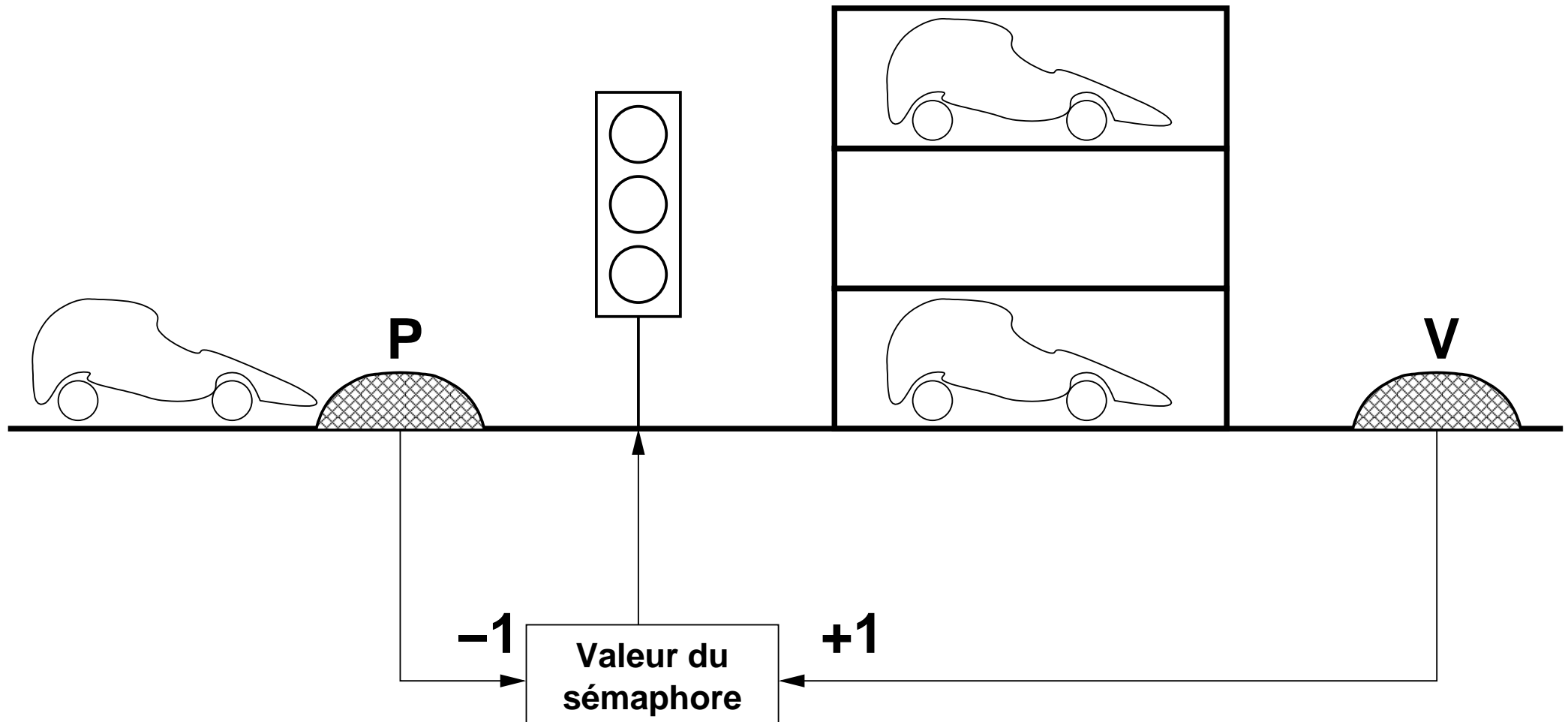
3.6.1 Introduction aux sémaphores	21
3.6.2 Introduction aux sémaphores : analogie	22
3.6.3 Introduction aux sémaphores : algorithmes P ET V	23
3.6.4 Sémaphores IPC	24
3.6.5 Utilisation des sémaphores IPC	25

3.6.1 Introduction aux sémaphores

- Sémaphore = objet composé :
 - ◆ D'une variable (sa valeur)
 - ◆ D'une file d'attente (les processus bloqués)
- Primitives associées :
 - ◆ Initialisation (avec une valeur positive ou nulle)
 - ◆ Manipulation :
 - ▶ Prise (P ou Wait) = demande d'autorisation
 - ▶ Validation (V ou Signal) = fin d'utilisation
- Principe : sémaphore associé à une ressource
 - ◆ Prise = demande d'autorisation (Puis-je ?)
si *valeur* > 0 accord, sinon blocage
 - ◆ Validation = restitution d'autorisation (Vas-y)
si *valeur* < 0 déblocage d'un processus

3.6.2 Introduction aux sémaphores : analogie

- Parking de N places contrôlé par un feu



3.6.3 Introduction aux sémaphores : algorithmes P ET V

■ Initialisation(sémaphore,n)

```
valeur[sémaphore] = n
```

■ P(sémaphore)

```
valeur[sémaphore] = valeur[sémaphore] - 1
```

```
si (valeur[sémaphore] < 0) alors
```

```
    étatProcessus = Bloqué
```

```
    mettre processus en file d'attente
```

```
finSi
```

```
invoquer l'ordonnanceur
```

■ V(sémaphore)

```
valeur[sémaphore] = valeur[sémaphore] + 1
```

```
si (valeur[sémaphore] == 0) alors
```

```
    extraire processus de file d'attente
```

```
    étatProcessus = Prêt
```

```
finSi
```

```
invoquer l'ordonnanceur
```

3.6.4 Sémaphores IPC

■ Création :

```
int semget ( key_t clé, int nsems, int semflg ) : retour = semid
```

■ Opérations :

```
int semop ( int semid, struct sembuf *spos, int nsops )
```

■ Opérations de contrôle :

```
int semctl ( int semid, int semno, int cmd, union semun arg )
```

3.6.5 Utilisation des sémaphores IPC

■ Structure :

```
struct sembuf {  
    u_short sem_num ;  
    short sem_op ;  
    short sem_flg ;  
}
```

■ Opérations de base :

- ◆ P (s) : { numSem, -1, 0 }
- ◆ V (s) : { numSem, 1, 0 }