

# Aufgabe 5: QNX - IO: LEDs

Embedded Computing, Teilgruppe 2, Florian Tobusch - Chris Brammer

## Device öffnen und schließen

Der Zugriff auf die LEDs wird über das Device `/dev/leds` gesteuert. Da wir nur die LEDs an- und ausschalten möchten, reicht der Zugriffsmodus `O_WRONLY` (Writing only).

```
int fid = open("/dev/leds", O_WRONLY);
```

`open()` liefert als Rückgabewert die ID eines neuen File-Descriptors oder `-1`, falls es zu einem Fehler kam. Mögliche Fehler können beispielsweise fehlende Zugriffsrechte oder falsche Pfadangaben sein.

Bevor das Programm beendet wird, gibt `close()` die Ressource wieder frei.

## Byte für Device berechnen und schreiben

Beim Start werden die Bits für die ersten beiden LEDs auf 0 gesetzt. Die Dritte LED ergibt sich aus dem XOR und die Vierte aus dem OR der ersten beiden LEDs. Der Benutzer kann nach der Initialisierung selbst den Zustand der ersten beiden LEDs über eine Eingabe definieren.

Das Byte für das LED-Device wird über shiften der einzelnen Bits und anschließendes verodern erstellt. Danach wird das Byte über `write()` in die Gerätedatei `/dev/leds` geschrieben.

## Benutzereingabe abfangen

```
Enter '0' or '1' to toggle LED 0/1. Enter 'q' to quit.
```

`getchar()` wartet solange, bis der Benutzer entweder `0`, `1` oder `q` eingibt. Mit `0` und `1` wird der Zustand der ersten bzw. der zweiten LED über einen Toggle-Mechanismus geändert. Durch die Eingabe von `q` wird das Programm beendet.