

Abgabe Aufgabenblatt Nr. 5

Markus Schmidt

Maximilian Gaul

Zuerst lesen wir von der Eingabe *STDIN_FILENO* (mit zusätzlicher Überprüfung, ob die Ausführung des Befehls durch ein Signal unterbrochen wurde):

```
uint8_t buf[2];
int read_bytes = 0;
while( (read_bytes = read(STDIN_FILENO, &buf, sizeof(buf))) < 0 && errno == EINTR )
    ;
```

Wir erwarten von der Eingabe zwei Zeichen, die den Zustand von *LED0* und *LED1* angeben (z.B. *11* um beide LEDs an bzw. *00* um beide auszuschalten).

Für die 4 LEDs ergeben sich folgende Zustände abhängig von der Benutzereingabe:

```
uint8_t led0 = (buf[0] - '0');
uint8_t led1 = (buf[1] - '0');
uint8_t led2 = (led0 ^ led1);
uint8_t led3 = (led0 | led1);
const char toWrite = (led3 << 3) | (led2 << 2) | (led1 << 1) | led0;
```

XOR-Verknüpfung:

Über das “^” werden die beiden Werte XOR-Verknüpft

OR-Verknüpfung:

Über das “|” werden die beiden Werte mit OR verknüpft.

Shift-Parameter

<< Mit diesem Parameter wird auf Bitebene verschoben.

Anschließend öffnen wir das entsprechende Device für die LEDs, schreiben den zuvor berechneten Wert und schließen es wieder:

```
fd = open("/dev/leds", O_RDWR);
write(fd, &toWrite, sizeof(toWrite));
close(fd);
```