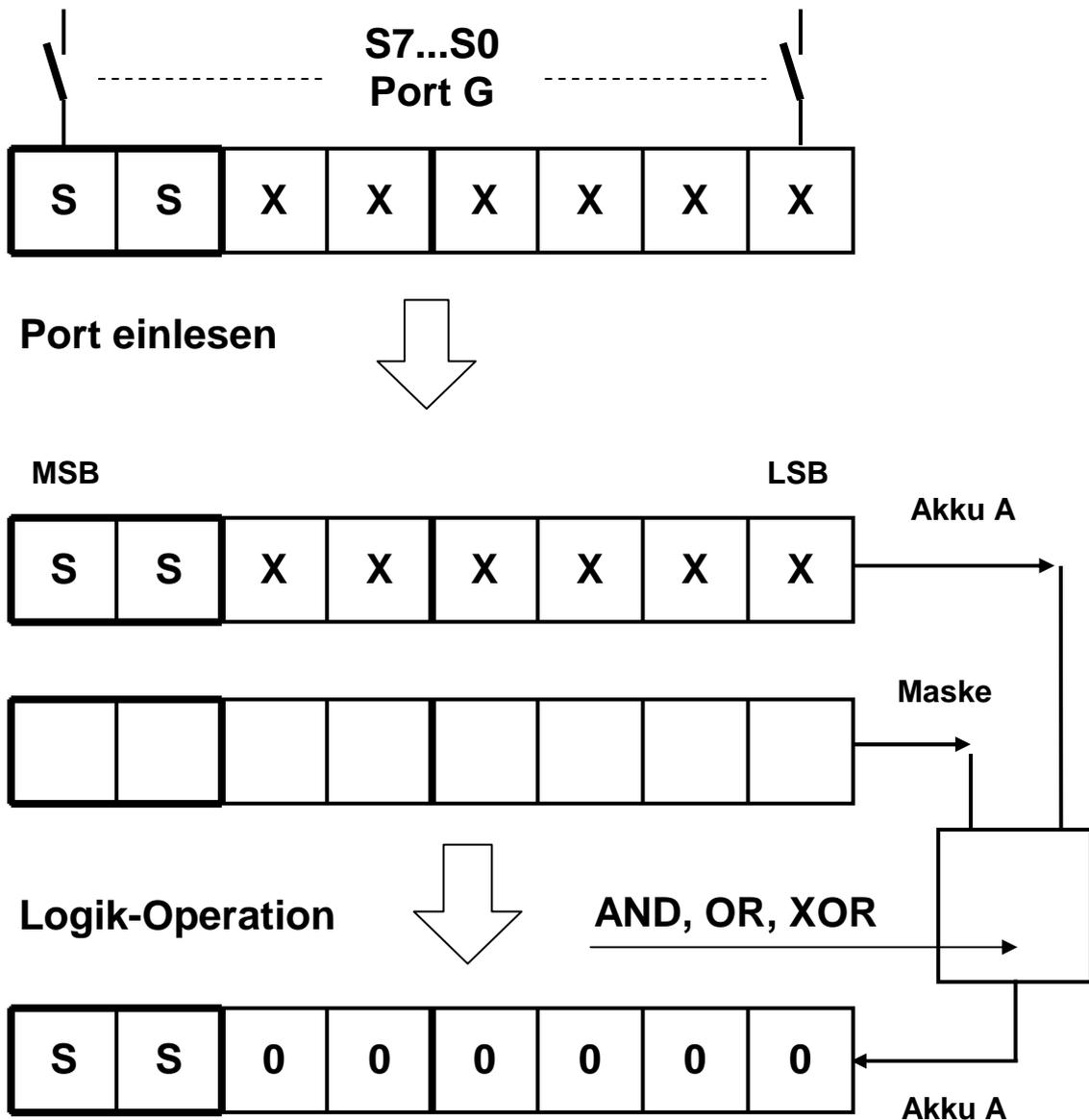


## Logikoperationen des Mikrocontrollers

### Anwendungs-Beispiel: Schalterbits maskieren

Für eine Anwendung werden die Schalter S0 bis S7 von Port G in den Akku A der CPU eingelesen. Benötigt und ausgewertet werden dabei vom Programm nur die Schalter S6 und S7. Gefordert ist, dass die übrigen Schalterbits "X" (nach dem Einlesen in Akku A) immer den Wert 0 (= Low) aufweisen, auch wenn die zugehörigen Schalter betätigt werden.

Die folgende Grafik zeigt, wie der Akku A mit einer **Bitmaske** (Maskierungs-Byte) logisch verknüpft werden kann, um gezielt einzelne Bits zu setzen (log. Wert = 1) oder rückzusetzen (log. Wert = 0). Dabei wird jedes einzelne Bit des Akku A mit dem entsprechenden Bit der Maske mittels der gewünschten Logikoperation verknüpft. Der Inhalt des Akku A wird mit dem Resultat dieser Operation überschrieben.



**Uebungen zu obigem Anwendungs-Beispiel****Aufgabe 1:**

- a) Wie sieht die Maske als binäre und hexadezimale Zahl aus?
- b) Mit welcher Logikoperation muss sie mit dem Akku A verknüpft werden?

a) .....

b) .....

**Aufgabe 2:**

- a) Welche Maske und
- b) welche Logikoperation muss angewendet werden, falls die nicht ausgewerteten Schalterbits "X" immer den Wert 1 (= High) im Akku A aufweisen müssen?

a) .....

b) .....

**Aufgabe 3:**

- a) Mit welcher Logikoperation und
- b) welchem Byte-Operand kann erreicht werden, dass im Akku A für den Schalter S7 immer der inverse Wert des Schalterstandes vorhanden ist (Bit invertieren!)?

a) .....

b) .....