

# Alarmanlage

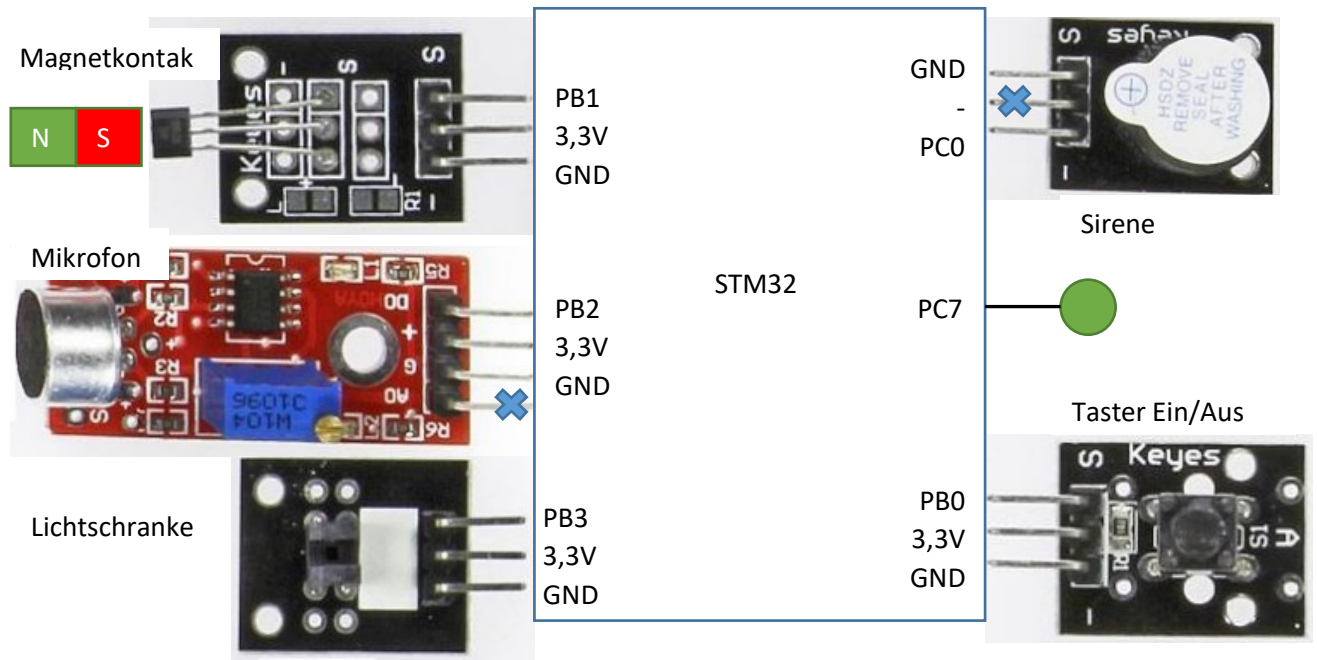


Abbildung 1 Technologieschema

## Anschlussbelegung:

- PB3: Lichtschanke (High aktiv)
- PB1: Magnetfeld (Low aktiv)
- PB2: Geräusch (Low aktiv)
- PC0: Sirene
- PB0: Taster Ein/Aus (Low aktiv)
- PC1..PC3: Optischer Alarm mit 3 Farben LED
- PC7: Anzeige Scharf

## Aufgaben:

### Stufe 0: Neues Projekt anlegen mit:

- Takteinstellung 32MHz
- Konfiguration der Ein- und Ausgänge

### Stufe 1: Umschaltung scharf

Bei jedem Tastendruck auf die Ein/Aus-Taste soll eine Variable `int scharf=0`; zwischen 0 und 1 hin- und herwechseln. Beachten Sie:

- Gedrückt bedeutet 0 (low aktiv), nicht gedrückt bedeutet 1
- Um Mehrfachwechsel zu vermeiden muss, nach jedem Tastendruck, gewartet werden bis die Taste wieder losgelassen wurde.
- Da die Taste prellt soll nach jeder Änderung des Tastensignals 20 ms gewartet werden

## Stufe 2: Melder auswerten

Wenn scharf=0 dann

- Anzeigelampe an PC7 aus
- Alarm PC0 aus

Wenn scharf=1 dann

- Anzeigelampe an PC7 ein
- Wenn Magnetfeldsensor = 0 dann Alarm
- Wenn Geräusch =0 dann Alarm
- Wenn Lichtschranke=1 dann Alarm

## Stufe 3: 3 Geräusche

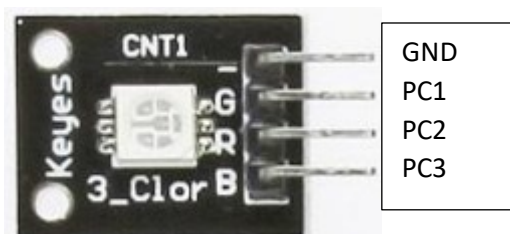
Es hat sich in der Praxis gezeigt, dass die Geräuschüberwachung zu empfindlich reagiert. Deshalb soll das Programm so abgeändert werden, dass erst nach dem 3. Geräusch der Alarm ausgelöst wird.

Nach jedem Geräusch soll 1 Sekunde gewartet werden (`HAL_Delay(1000);`)

## Stufe 4: Optische Meldung mit 3-Farben LED

Zusätzlich zum akustischen Alarm soll jetzt noch ein optischer Alarm aktiviert werden.

Dazu wird die 3-Farben-LED angeschlossen:



Im Alarmfall wechselt die LED im Sekunden-Rhythmus (`HAL_Delay(1000)`) zyklisch von Grün nach Rot nach Gelb.

Frage: Wie lange muss die Taste dann mindestens gedrückt werden um die Alarmanlage zu deaktivieren?

Versehen Sie Ihre Programme mit aussagekräftigen Kommentaren und stellen Sie Ihre Lösungen in der Klasse vor.