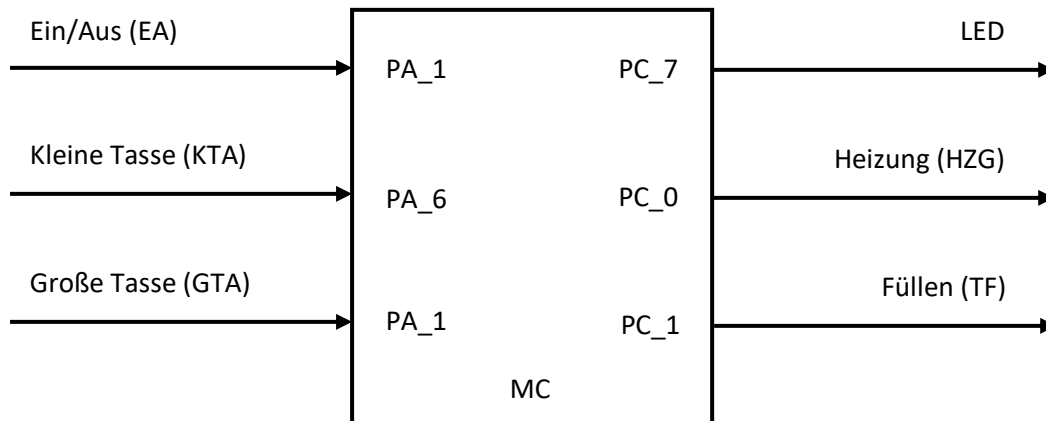


Steuerung eines Kaffeeautomaten



Alternativ soll die Funktion des Kaffeeautomaten mit einem Mikrocontroller realisiert werden.

- Mit dem Taster EA schaltet man die Maschine EIN und AUS.
- Der Heizvorgang wird begonnen, wenn die Maschine eingeschaltet wurde.
- Mit den Tastern KTA und GTA wird der Füllvorgang gestartet (TF=1); das ist nur möglich, wenn der Heizvorgang abgeschlossen ist (HZG=0).
- Die Kaffeemaschinensteuerung steuert die LED an und sorgt außerdem dafür, dass die Heizung (HZG) nach dem Einschalten und nach jedem Füllvorgang für 30 Sekunden auf High („1“) geht; solange kann kein weiterer Füllvorgang gestartet werden.

Aufgaben:

1. Deklarieren Sie die Ein- und Ausgänge gemäß obigem Blockdiagramm.
Der Taster Ein/Aus (EA) soll in der Lage sein einen externen Interrupt auszulösen.
2. Ein- und Ausschalten des Kaffeeautomaten mit externem Interrupt.
 1. Schreiben Sie die Interruptserviceroutine (isr_ea) für den Taster EA.
 - Die ISR negiert die globale Variable int ein=0;
 - Ruft, je nachdem ob ein- oder ausgeschaltet das Unterprogramm heizung() oder allesAus() auf.
 2. Schreiben Sie ein Unterprogramm void init() mit folgendem Inhalt:
 - Alle Taster mit PullDown
 - Taster EA: Bei steigender Flanke soll die isr_ea starten
3. Schreiben Sie das Hauptprogramm
 1. Initialisieren (Unterprogramm init())
 2. Wenn eingeschaltet ist
 - Wenn Taste KTA gedrückt ist und der Heizvorgang abgeschlossen ist und wenn der letzte Füllvorgang abgeschlossen ist (TF=0)
 1. Unterprogramm kleineTasse()
 2. Warten auf Füllvorgang abgeschlossen (TF=0)
 3. Heizen (Unterprogramm heizung())
 - Ebenso für große Tasse mit Unterprogramm grosseTasse()

•

4. Schreiben Sie die Timer-ISR: void tim6_isr()

1. Blinken der LED
2. Globale variable int dauer=0; herunterzählen
3. Bei 0:
 - Timer stoppen
 - Heizung ausschalten (HZG=0);
 - Füllen ausschalten (TF=0);
 - LED einschalten (LED=1);
4. Pending- und Überlaufbit zurücksetzen

5. Timer und Timerinterrupt initialisieren.

Ergänzen Sie das Interprogramm init() um folgende Anforderungen:

1. Prescaler für 1 ms
2. Autoreload für eine Blinkfrequenz der LED von 4 Hz
3. Timerinterrupt initialisieren

6. Unterprogramme

1. Heizung()
 - HZG=1
 - Autoreload für Blinkfrequenz der LED von 4Hz einstellen
 - Globale Variable int dauer=0; für eine Heizdauer von 30s initialisieren
 - Timer starten
2. kleineTasse()
 - TF=1
 - Autoreload für Blinkfrequenz der LED von 1Hz einstellen
 - Globale Variable int dauer=0; für eine Fülldauer von 10s initialisieren
 - Timer starten
3. grosseTasse()

ebenso wie kleineTasse, nur dass die Fülldauer jetzt 20s beträgt
4. allesAus()
 - Alle Ausgänge auf 0 schalten
 - Timer stoppen