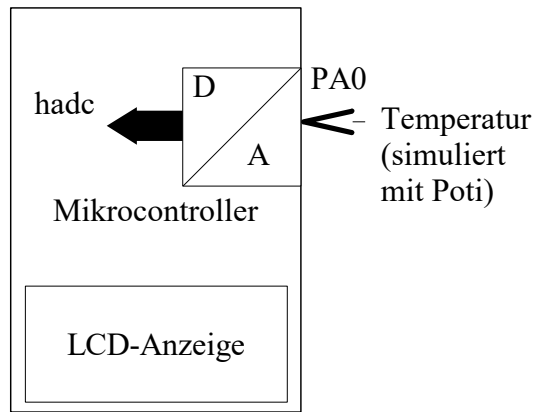


Technologieschema:



Beschreibung: Für eine Wetterstation sollen Temperaturmesswerte aufgezeichnet und ausgewertet werden.

- Messintervall: 1s
- bis 20 Messwerte, weitere Messwerte überschreiben den Speicher wieder von Anfang an.
- Aktueller Messwert in Zeile 1 anzeigen
- Da die Messwerte stark schwanken soll der Mittelwert von 20 Messungen in der 2. Zeile des LCD-Displays angezeigt werden.

Aufgabe:

- Was bedeutet A/D? In welchem Wertebereich liegen die "Temperaturen"
In welchem Speicher wird aufgezeichnet?
Welche Instruktionen sind erforderlich?
- Programm in ASM

Das ist ein dickes Brett! Deshalb nachfolgend ein paar Tips

Lösungshinweise:

RAM verwenden: Das RAM nimmt den Adressbereich von 0x20000000 bis 0x20013FFF ein.

Zugriffsmöglichkeiten:

ldr R6,=Messwerte //gewünschte RAM-Adresse laden

ldr R2,[R6,0] //ein Int-Wert aus dem RAM lesen Adresse 0x20000000 und in R2 eintragen

str R3,[R6,4] //ein Int-Wert aus R3 im RAM an der Adresse 0x20000004 abspeichern

mov R4,#8

str R3,[R6,R4] //speichert ein Byte aus R3 in der Adresse R1+R4 also 0x20000008

add R4,#4

ldr R5,[R6,R4] //lädt ein Byte aus R1+R4 also 0x2000000C in R5

Am Programmende: (vor end)

...

.data //Datenbereich im RAM

Messwerte:

.space 80,0 //reserviert 20 Bytes entspricht: char Messwerte[20];

.end

.end

AD-Wandler:

Initialisierung

ldr R0,=hadc

bl HAL_ADC_Start

Endlosschleife

ldr r0,=hadc

bl HAL_ADC_GetValue

//überträgt den Einstellwert des Potis nach R0

(bei SW4STM32: bl Ain0)

Zeitschleife:

mov R0,#1000 //1000ms = 1s

bl HAL_Delay

Display:

Initialisierung:

bl startLCD

weitere Befehle:

bl LCD_i2c_clear //löscht das display

mov R0,#0

bl LCD_i2c_cursorpos //Setzt den Cursor auf den Anfang der 1. Zeile

mov R0,#0x40

bl LCD_i2c_cursorpos //Setzt den Cursor auf den Anfang der 2. Zeile

bl LCD_i2c_dezaus // gibt R0 als Dezimalzahl aus.

Nützliche Assemblerbefehle:

add R8,R0 //R8=R8+R0

udiv R0,R8,R7 //R0=R8/R7