Work package 5, Part 2 Report, Group 3

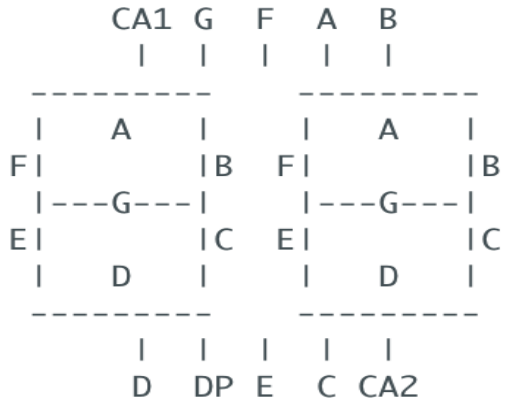
Contributors:
Samer Daknache
Martin Chukaleski
Yazan Alsahhar

Description of the system

The hardware of the system:

- Arduino UNO CPU board
- 1 **x** Dual digit 7 segment display
- 1 **x** Resistor
- 1 **x** LM35 temperature sensor module
- 1 **x** 2-bit dip switch
- 2 **x** Breadboard (Extra for digit display)
- 1 **x** Resistance net in a DIL

Pins & segment for the dual digit display:



Segments that make each number when lit up:

0 => -FEDCBA
1 => ----BC-
2 => G-ED-BA
3 => G--DCBA
4 => GF--CB-
5 => GF-DC-A
6 => GFEDC-A
7 => ----CBA
8 => GFEDCBA
9 => GF-DCBA

Description for ADC on Arduino UNO

The Arduino UNO has 6 Analog to Digital Conversion channels that can be used as analog voltage inputs. Since the conversion on the UNO has resolution of 10-bit, voltages between 0V-5V will map to integer values between 0-1023 (5V / 1024 units).

Description of the analogRead() function

The function analogRead() reads the voltage from the specified pin (for example, A0) and map to its respective integer value. The average time to read an analog input is 100 microseconds, or about 10 000/s. As stated previously, the function takes a pin as an input parameter and returns an integer value in the range 0-1023.

Description of the program structure

First we define all the necessary macros for the pins for better code readability and better code clarity. We then declare all the variables we use for data storing and manipulation.

In **setup()**, we set the pin modes for every pin (including the pullup) and we start the serial port at baud rate 9600.

In **loop()**, we read the temperature from analog into digital (integer) value and convert it to Celsius. We also keep track of the maximum measured temperature. We then check what's the state of the DIP switch and if it's on, we show the maximum measured temperature. In displaying it, we account for whether the temperature is as low as -9℃, above 0℃ or above 10℃ (i.e. has 2 digits). Thus, our range is -9℃ to 99℃. If, however, the dip switch is off, we display the current temperature in range from -9℃ to 99℃ using the same checks for the temperature digits. To update the digits on the display, we use delays of between 20 microseconds and 1 second. Finally, we also print all this temperature information to the serial port as well.

The function **selectFirstDigit()** selects the first digit or the cathodes D1 and D2 pins and sets their values. The function **selectSecondDigit()** selects the second digit (same cathodes) but with opposite values.

The function **drawMinus()** draws a minus on the selected digit by setting the segments of the digit accordingly. The function **drawNumber()** takes in a number between 0 and 9 that should be drawn up in the selected digit and uses a switch statement to set the segments of the selected digit accordingly.