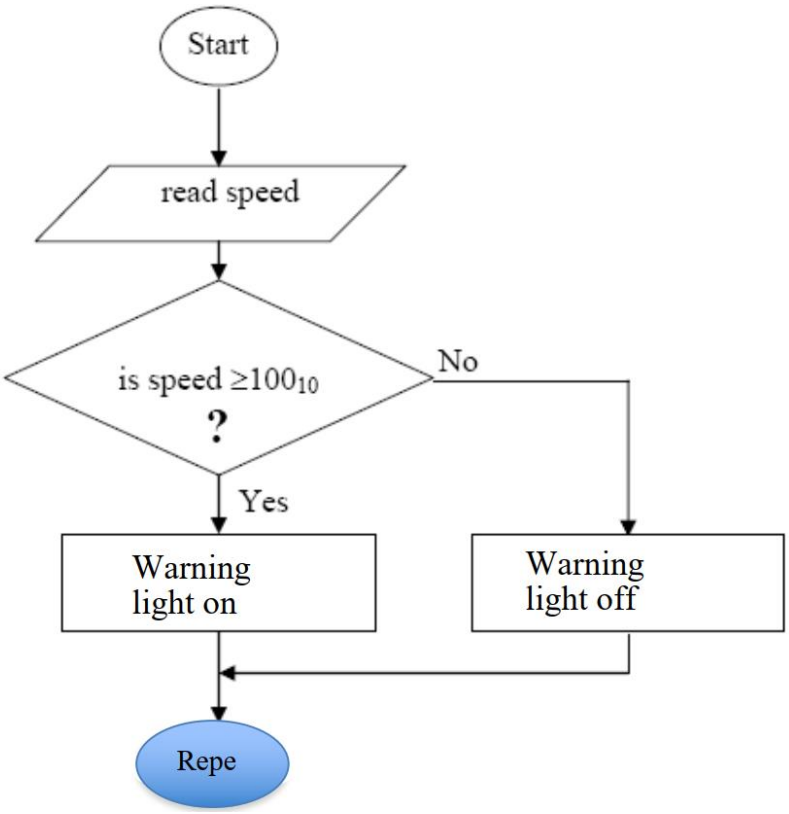
Homework 3 -- C Programming

1. Data Monitoring

Write a C subroutine "checkspeed" for the Nano controller to perform the following function:



Where ***speed*** is read as an 8-bit unsigned integer that needs to be read from port B. If ***speed*** is greater or equal to 100 (decimal) it switches on a warning light; otherwise, the light is turned off. The warning light is controlled by bit 4 on **port D**. A "1" on bit 4 turns on the light and a 0 turns it off. The other bits of port D are used for other functions. This program should loop forever.

2. Temperature Control

A Nano microcontroller is used to control the temperature in this room. The microcontroller has been hooked up such that the 8-bit signed number stored at address $0400 represents the temperature in this room in degrees Celsius. Also, a switch that controls the heater for this room is connected to Bit 0 of PORT D, and a switch that controls the air conditioner for this room is connected to Bit 4 of PORTD. Writing a 0 to Bit 0 turns the heater off; writing a 1 to Bit 0 turns the heater on. Writing a 0 to Bit 4 turns the air conditioner off; writing a 1 to Bit 4 turns the air conditioner on.

Write a C program on the microcontroller, which will do the following:

- Set up Bits 0 and 4 of PORTD as output bits. The other bits of PORTD should be set up as inputs.
- If the temperature in the room is below 20 degrees, make sure the air conditioner is off and the heater is on.
- If the temperature in the room is between 20 and 24 degrees, make sure both the heater and air conditioner are off.
- If the temperature in the room is above 24 degrees make sure the air conditioner is on

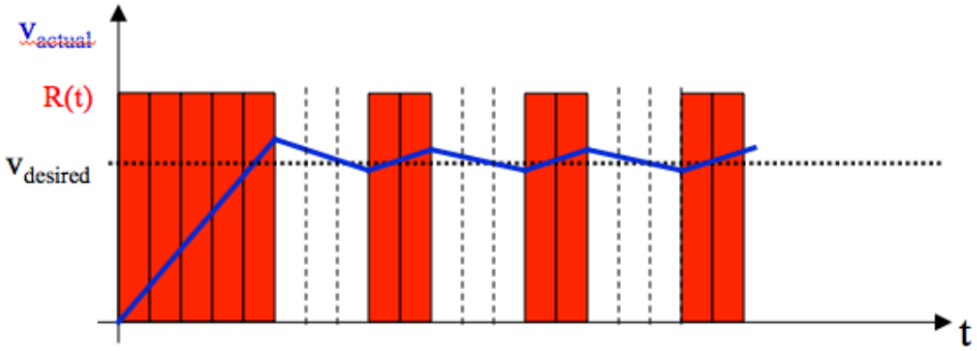
and the heater is off.

- Repeat this set of instructions forever.

3. On-Off Controller

A Nano microcontroller is used for on-off motor control (bangbang control), where an encoder is used to measure the speed of the motor. And some constraints are listed as following:

- The motor output pin is D0.
- The encoder data pin is connected to the ADC port to acquire the speed of motor .
- The motor is spinning only in one direction, so no motor direction pin is required and a single encoder input is sufficient.
- Assume some peripherals such as ADC module and Timer 1 module have been initialized and their corresponding functions can be called directly by adding the following statement '*#include <peripherals.h>*' in your programs, where the prototypes of initialization functions of ADC module and Timer 1 module are '*void initADC*()' and '*void initTimer1*()', respectively, and the prototype of function for reading data from ADC module is '*int readADC*() '.
- Assume the interrupter service routine (ISR) of timer 1 is called at 100Hz.



Implement a complete C program for on-off motor control (bangbang control), which will do the following:

- The **main** program has to initialize port D0, ADC module and Timer 1 module, and enable global interrupt.
- The **ISR** of timer 1 reads the encoder, calculates the current speed, then decides whether the motor should be switched on or off.