RC Car

Holly Krebs and Chris Willecke
Objective: To build a base platform for a mobile robot.
Design:

- PWM system and H-Bridges for motor control
- ADC for analog input
- X-bee wireless modules for communication
Construction
Controller Flowchart

Position of joystick sends two analog signals to HCS12

A/D converter converts two analog signals (forwardBack and leftRight)

XBee wirelessly transmits serial data
Controller Code:

```c
// leftRight = 45;
putcharSCI('0'); // Signifies start of transmission
for(i = 0; i < 10; i++){
    DelayuSec(1000);
}

forwardBack = ATDDR0L;
if (forwardBack == 0x00) {
    forwardBack = forwardBack + 1;
}

putcharSCI(forwardBack); // Sends forwardback value
for(i = 0; i < 10; i++){
    DelayuSec(1000);
}

leftRight = ATDDR1L;
if (leftRight == 0x00) {
    leftRight = leftRight + 1;
}

putcharSCI(leftRight); // Sends leftright value
for(i = 0; i < 10; i++){
    DelayuSec(1000);
}
```
Car Flowchart

1. XBee wirelessly receives serial data
2. leftmotor = forwardBack, rightmotor = forwardBack
3. Is the value of forwardBack > 130?
   - Yes: LR = leftright - 127
   - No: LR = 127 - leftright
4. LR = LR / 2
5. leftmotor = leftmotor + LR, rightmotor = rightmotor - LR
6. Set Duty Cycle for right and left motors
```c
leftMotor = rightMotor = forwardBack;
    if(forwardBack > 130 ){
        LR = leftRight - 127;
    } else if(forwardBack < 124 ){
        LR = 127 - leftRight;
    } else{
        LR = 0;
    }
LR = LR/2 ;
if(leftMotor < 124){
    // set left motor forwards at leftmotor - 127
    PWMPER2 = 127;
    PWMDTY2 = 127 - leftMotor;
    PWMDTY3 = 0;
    PWMPOL = 0xFF;
} else if(leftMotor > 130){
    // set left motor backwards at 127 - leftmotor
    PWMPER3 = 127;
    PWMDTY3 = (leftMotor - 127);
    PWMDTY2 = 0;
    PWMPOL = 0xFF;
} else{
    // turn left motor off
    PWMDTY2 = PWMDTY3 = 0;
}
```