MSP430

RST

TEST

GND

VCC

Using SET/RESET : TACCTL1 |= OUTMOD\_3 (or RESET/SET = OUTMOD\_7)

TACCR1 = 60; (60% duty cycle)

TACCR0 = 8000; (set period for # of clock cycles)

P1DIR |= BIT2; // TA0CCR1 on P1.2

P1SEL |= BIT2; // TA0CCR1 on P1.2

TACTL &= ~MC1|MC0; // stop timer A0

TACTL |= TACLR; // clear timer A0

TACTL |= TASSEL1; // configure for SMCLK

TACCRO will be set to the period.

Then, it is just a matter of switching between forward/backward and left/right motion. For now, I will use \_\_delay\_cycles(number) for the length each movement is carried out. Eventually, I will work on converting this to PWM.

Initialize

moveForward()

moveBackward()

leftTurn() - short

rightTurn() - short

leftTurn() - long

rightTurn() - long

moveForward() makes the robot move forward by having both grounds on the side connected to P0. Vcc on the side connected to P1.

moveBackward() makes the robot move in the opposite direction by reversing the ground on each side. In other words, what was a ground moving forward is now Vcc, and what was Vcc moving forward is now ground.

leftTurn() will turn left for a specified period of time (long or short). A short turn will create an angle of 45 degrees or less while a long turn will create a 90 degree turn. The left wheel needs to move forward while the right wheel turns backward.

rightTurn(time) will turn right for the specified period of time. This function will be very similar to leftTurn(), it will just set the wheels turning the opposite direction.

Overall, it makes the most sense to have the above functions just set the wheel direction. From there, I should create a separate function that specifies the amount of time each wheel will be turning. This time specification will create the difference between a small and a big left/right turn.

I will use PWM signals directly. I believe it makes the most sense to start with PWM because that is what a functionality is asking for.