

Circle :

```

#include <Romi32U4.h>
#include <PololuRPiSlave.h>

Romi32U4Motors motors;
Romi32U4Encoders encoders;
Romi32U4ButtonA buttonA;

void setup() {
    Serial.begin(57600);
    buttonA.waitForButton();
    delay(2000);
    // put your setup code here, to run once:
    ledYellow(false);
    ledGreen(true);
    ledRed(false);
}

float _debug_linear_ms = 0.25;
float _debug_angle_rs = 0.0;
void _DEBUG_PID_CONTROL() {
    static float _linear_ms_change = 0.1;
    set_twist_target(_debug_linear_ms,
        _debug_angle_rs);
}

void loop() {

    _DEBUG_PID_CONTROL();
    for (int speed = 0; speed <= 100; speed++)
    {
        motors.setLeftSpeed(speed);
    }

    // Run right motor forward.
    ledRed(1);
    for (int speed = 0; speed <= 200; speed++)
    {
        motors.setRightSpeed(speed);
    }
    if (everyNmillisec(10)) {
        // ODOMETRY
        calculateOdom();
        doPID();
    }
}

```

## Square

```

#include <Romi32U4.h>
#include <PololuRPiSlave.h>

Romi32U4Motors motors;
Romi32U4Encoders encoders;
Romi32U4ButtonA buttonA;

void setup() {
    Serial.begin(57600);
    buttonA.waitForButton();
    delay(2000);
    // put your setup code here, to run once:
    ledYellow(false);
    ledGreen(true);
    ledRed(false);
}

float _debug_linear_ms = 0.25;
float _debug_angle_rs = 0.0;
void _DEBUG_PID_CONTROL() {
    static float _linear_ms_change = 0.1;
    set_twist_target(_debug_linear_ms,
        _debug_angle_rs);
}

void loop() {

    _DEBUG_PID_CONTROL();
    for (int speed = 0; speed <= 100; speed++)
    {
        motors.setLeftSpeed(speed);

```

```

    }
    for (int speed = 0; speed <= 100; speed++)
    {
        motors.setRightSpeed(speed);
    }
    delay(1000);
    for (int speed = 100; speed >= 0; speed--)
    {
        motors.setRightSpeed(speed);
        motors.setLeftSpeed(speed);
    }
    delay(1000);

    for (int speed = 0; speed <= 100; speed++)
    {
        motors.setRightSpeed(speed);
    }
    for (int speed = 0; speed <= -50; speed--)
    {
        motors.setLeftSpeed(speed);
    }
    delay(1000);
    if (everyNmillisec(10)) {
        // ODOMETRY
        calculateOdom();
        doPID();
    }
}
}”

```