To create an intelligent wheelchair controlled by an Arduino Uno and a joystick, you need to set up the hardware components and code the Arduino to control the wheelchair's motors based on input from the joystick. Below is a guide on how to create this project:

**### \*Hardware Components:\***

1. \*Arduino Uno\*: The main microcontroller for controlling the wheelchair.

2. \*Joystick Module\*: To receive directional input from the user.

3. \*DC Motors and Motor Driver\*: The motors will drive the wheelchair, and the motor driver will control them based on input from the Arduino.

4. \*Battery\*: A power source for the Arduino and motors.

5. \*Wiring and connectors\*: To connect the components.

6. \*Wheelchair\*: The platform for the intelligent wheelchair.

**### \*Setup:\***

1. \*Assemble the Wheelchair\*: Install the motors on the wheelchair and connect them to the motor driver. Connect the joystick to the Arduino.

2. \*Power\*: Connect the battery to the Arduino and the motor driver.

**### \*Code:\***

You can write a program in the Arduino IDE that reads the input from the joystick and controls the motors accordingly. Here's an example code:

cpp

// Include necessary libraries

#include <Wire.h>

// Define pins

#define JOYSTICK\_X\_PIN A0 // Joystick X-axis pin

#define JOYSTICK\_Y\_PIN A1 // Joystick Y-axis pin

#define MOTOR\_LEFT\_PWM 9 // Left motor PWM pin

#define MOTOR\_RIGHT\_PWM 10 // Right motor PWM pin

#define MOTOR\_LEFT\_DIR 2 // Left motor direction pin

#define MOTOR\_RIGHT\_DIR 3 // Right motor direction pin

// Threshold values for joystick

int threshold = 100;

// Function to control motors

void controlMotors(int speed, int direction) {

if (direction > 0) {

// Move forward

digitalWrite(MOTOR\_LEFT\_DIR, HIGH);

digitalWrite(MOTOR\_RIGHT\_DIR, HIGH);

analogWrite(MOTOR\_LEFT\_PWM, speed);

analogWrite(MOTOR\_RIGHT\_PWM, speed);

} else if (direction < 0) {

// Move backward

digitalWrite(MOTOR\_LEFT\_DIR, LOW);

digitalWrite(MOTOR\_RIGHT\_DIR, LOW);

analogWrite(MOTOR\_LEFT\_PWM, speed);

analogWrite(MOTOR\_RIGHT\_PWM, speed);

} else {

// Stop

analogWrite(MOTOR\_LEFT\_PWM, 0);

analogWrite(MOTOR\_RIGHT\_PWM, 0);

}

}

void setup() {

// Initialize pins

pinMode(MOTOR\_LEFT\_PWM, OUTPUT);

pinMode(MOTOR\_RIGHT\_PWM, OUTPUT);

pinMode(MOTOR\_LEFT\_DIR, OUTPUT);

pinMode(MOTOR\_RIGHT\_DIR, OUTPUT);

}

void loop() {

// Read joystick input

int x = analogRead(JOYSTICK\_X\_PIN);

int y = analogRead(JOYSTICK\_Y\_PIN);

// Calculate speed and direction

int speed = map(y, 0, 1023, -255, 255);

int direction = map(x, 0, 1023, -255, 255);

// Control motors based on joystick input

if (abs(speed) < threshold) {

// Stop if within the dead zone

speed = 0;

controlMotors(0, 0);

} else {

// Move according to speed and direction

controlMotors(abs(speed), speed);

}

}

### \*How the code works:\*

- The program starts by defining the pins and threshold value.

- The controlMotors function controls the motors based on the speed and direction provided.

- In the setup function, the motor pins are initialized as output.

- In the loop function, the joystick values are read, and the speed and direction are calculated.

- If the speed is within the threshold, the motors stop. Otherwise, they move according to the calculated speed and direction.

### \*Safety Considerations:\*

- Always ensure that the wheelchair's speed and direction are appropriately limited for safe operation.

- Test the wheelchair in a controlled environment before actual use.

### \*Conclusion:\*

This basic Arduino Uno and joystick-based intelligent wheelchair code and setup guide provides a foundation for creating a controllable wheelchair. Feel free to customize and extend the project according to your specific needs.