

```

// Arduino_receiver.c

// Code for an Arduino version of the receiver PIC in an example illustrating
// A/D conversion, hand shaking, serial communication, and LCD output

// NOTE:
// Be sure to disconnect the pin 0 (RX) line before re-programming the Arduino
// Select "Serial Monitor" under the "Tools" menu to see the printed messages

// Define global variables and constants
const byte led=13;
const byte sample_button=12;
const byte hand_shake=11;
byte pot_value=0;

// The setup function runs once when you press reset or power the board
void setup() {
  // Initialize pin I/O status
  pinMode(led, OUTPUT);
  pinMode(sample_button, INPUT);
  pinMode(hand_shake, OUTPUT);

  // Open serial communication for write read on pin 0 (RX)
  Serial.begin(9600);
  Serial.println("Serial communication initiated.");

  // Blink the LED three times to indicate the Arduino is running
  blink(); blink(); blink();

  // Make sure the handshake line is off initially
  digitalWrite(hand_shake, LOW);
}

// Main program loop
void loop() {
  // Wait for the button to be pressed
  while (digitalRead(sample_button) == LOW);
  Serial.println(" button pressed");
}

```

```
// Handshake with the sender PIC and receive the POT value serially
digitalWrite(hand_shake, HIGH); // signals the sender PIC to send
// Wait for serial data to arrive
while (Serial.available() == 0);
// Read serial data
pot_value = Serial.read();
digitalWrite(hand_shake, LOW);

// Display the received POT value on the serial monitor and blink the LED
Serial.print("POT value = ");
Serial.println(pot_value, DEC);
blink();
}
```

```
// Subroutine to blink the LED on and off once
void blink() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(250);              // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(250);              // wait for a second
}
```