main.c

```c
#include "pico/stdlib.h"
#include "hardware/adc.h"
#include <stdio.h>
#include <string.h>

#define LED_PIN 15
#define BUFFER_SIZE 64

int main() {
    stdio_init_all(); //initialize USB serial

    gpio_init(LED_PIN);
    gpio_set_dir(LED_PIN, GPIO_OUT);

    adc_init();
    adc_gpio_init(26);
    adc_select_input(0);

    char incoming[BUFFER_SIZE];
    int index = 0;
    uint16_t last_adc = 0;
    uint64_t last_send = 0;

    sleep_ms(1500);

    while (true) {
        if(time_us_64() - last_send >= 2000000){
            uint16_t raw = adc_read();
            if (raw != last_adc) {
                last_adc = raw;
                printf("%u\n", raw);  // C# reads this
            }
            last_send = time_us_64();
        }
        int c = getchar_timeout_us(0);  // non-blocking
        if (c == PICO_ERROR_TIMEOUT) continue;

        if (c == '\n' || c == '\r') {
            incoming[index] = 0;  // null-terminate
            if (strcmp(incoming, "LedOn") == 0) {
                gpio_put(LED_PIN, 1);
                printf("LED is ON\n");
            } else if (strcmp(incoming, "LedOff") == 0) {
                gpio_put(LED_PIN, 0);
                printf("LED is OFF\n");
            }
            index = 0;  // reset buffer
        } else {
            if (index < BUFFER_SIZE - 1) incoming[index++] = (char)c;

        }
    }
}

```