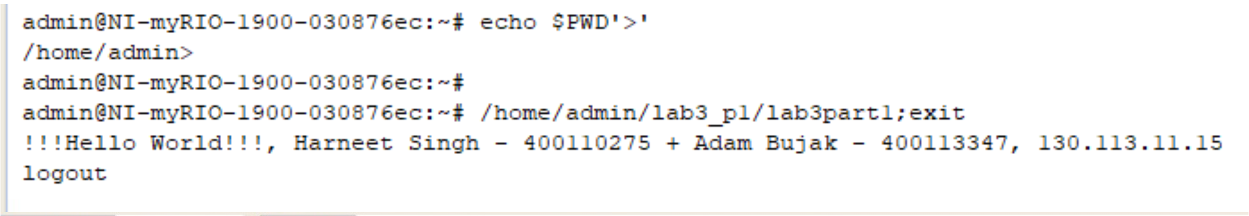
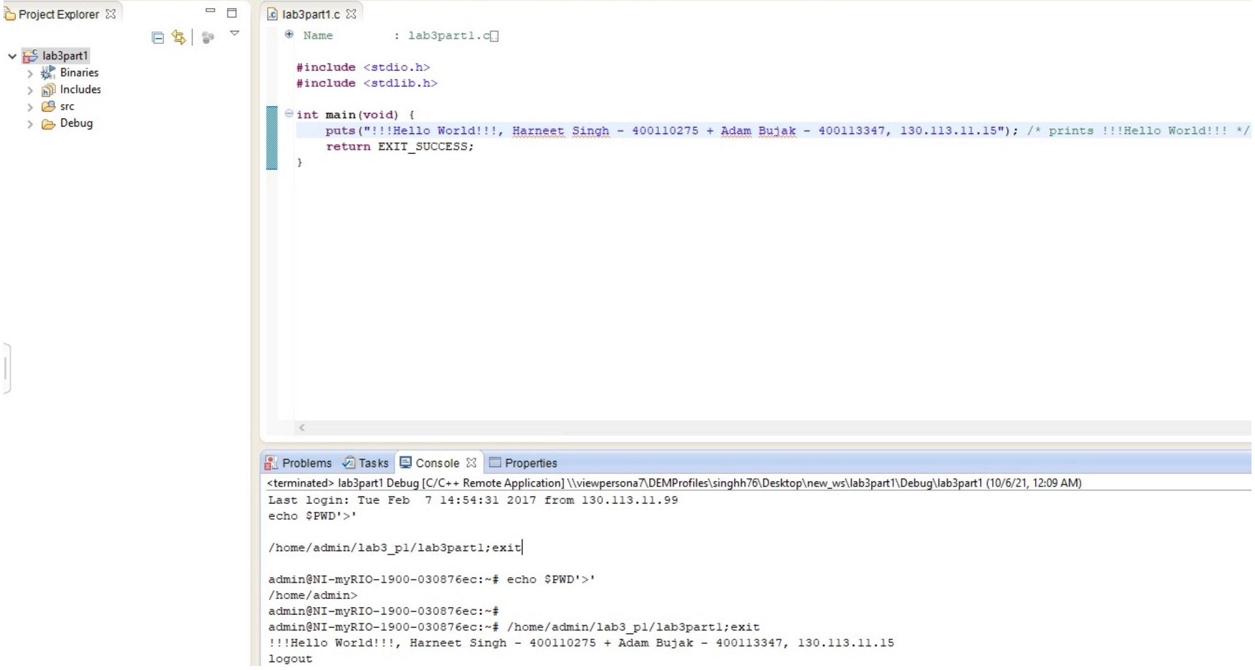
MECHTRON 4AA4
Lab 3 Report

Adam Bujak – 400113347
Harneet Singh – 400110275

Part 1.



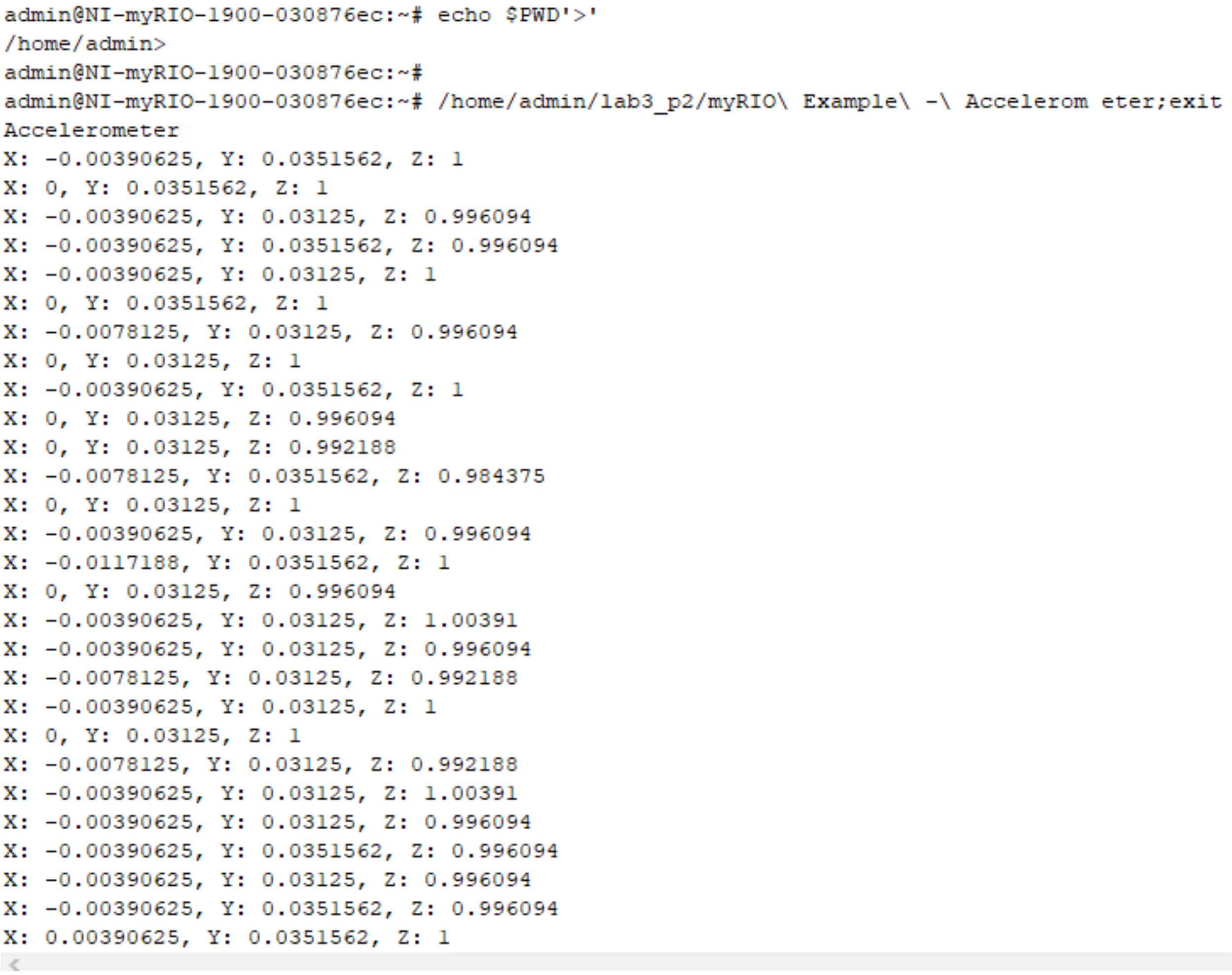
```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p1/lab3part1;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
logout
```



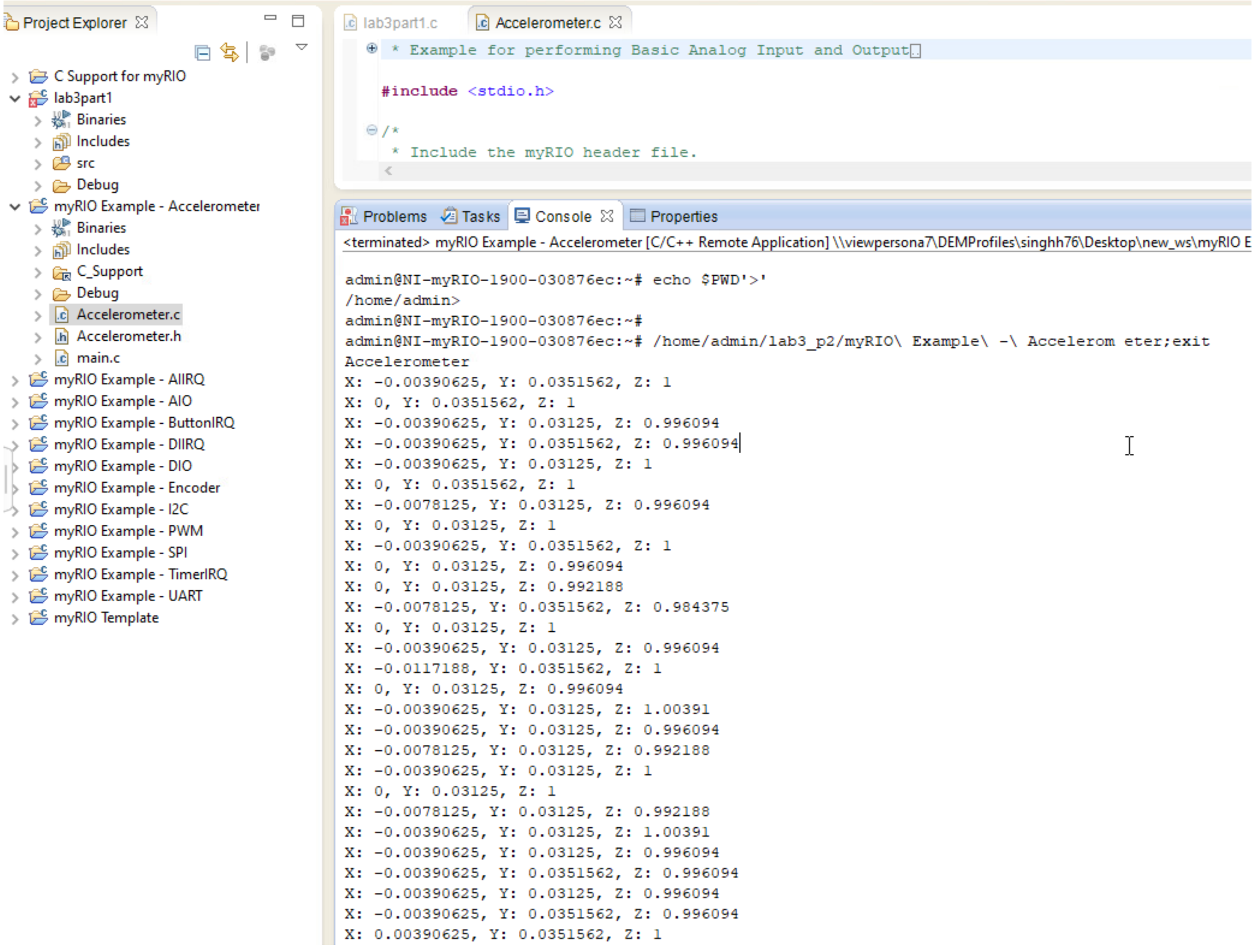
```
Name        : lab3part1.c

#include <stdio.h>
#include <stdlib.h>

int main(void) {
    puts("!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15"); /* prints !!!Hello World!!! */
    return EXIT_SUCCESS;
}
```

```
<terminated> lab3part1 Debug [C/C++ Remote Application] \\viewpersona7\DEMProfiles\singhh76\Desktop\new_ws\lab3part1\Debug\lab3part1 (10/6/21, 12:09 AM)
Last login: Tue Feb  7 14:54:31 2017 from 130.113.11.99
echo $PWD'>'

/home/admin/lab3_p1/lab3part1;exit

admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p1/lab3part1;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
logout
```

Part 2.



```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p2/myRIO\ Example\ -\ Accelerom eter;exit
Accelerometer
X: -0.00390625, Y: 0.0351562, Z: 1
X: 0, Y: 0.0351562, Z: 1
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 1
X: 0, Y: 0.0351562, Z: 1
X: -0.0078125, Y: 0.03125, Z: 0.996094
X: 0, Y: 0.03125, Z: 1
X: -0.00390625, Y: 0.0351562, Z: 1
X: 0, Y: 0.03125, Z: 0.996094
X: 0, Y: 0.03125, Z: 0.992188
X: -0.0078125, Y: 0.0351562, Z: 0.984375
X: 0, Y: 0.03125, Z: 1
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.0117188, Y: 0.0351562, Z: 1
X: 0, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 1.00391
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.0078125, Y: 0.03125, Z: 0.992188
X: -0.00390625, Y: 0.03125, Z: 1
X: 0, Y: 0.03125, Z: 1
X: -0.0078125, Y: 0.03125, Z: 0.992188
X: -0.00390625, Y: 0.03125, Z: 1.00391
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: 0.00390625, Y: 0.0351562, Z: 1
```

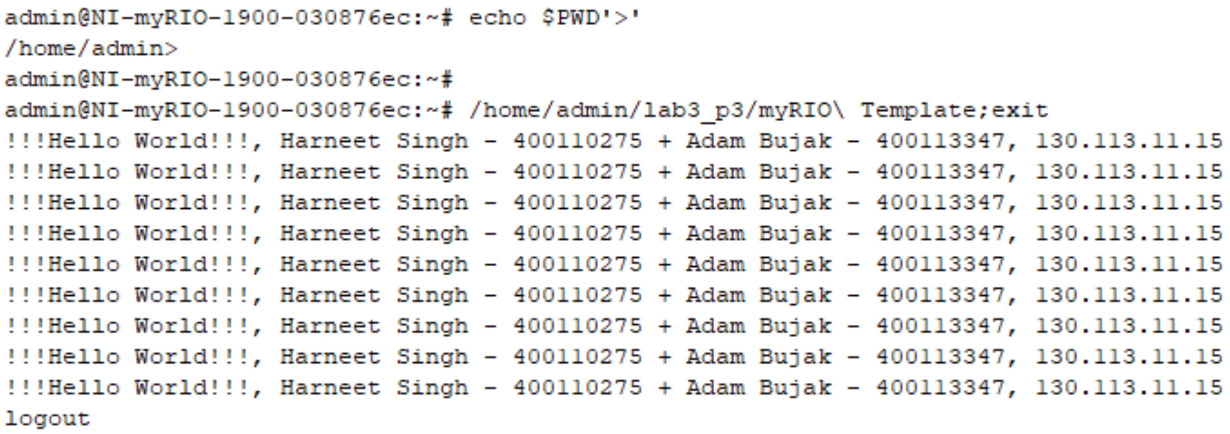


```
 * Example for performing Basic Analog Input and Output

#include <stdio.h>

/*
 * Include the myRIO header file.
```

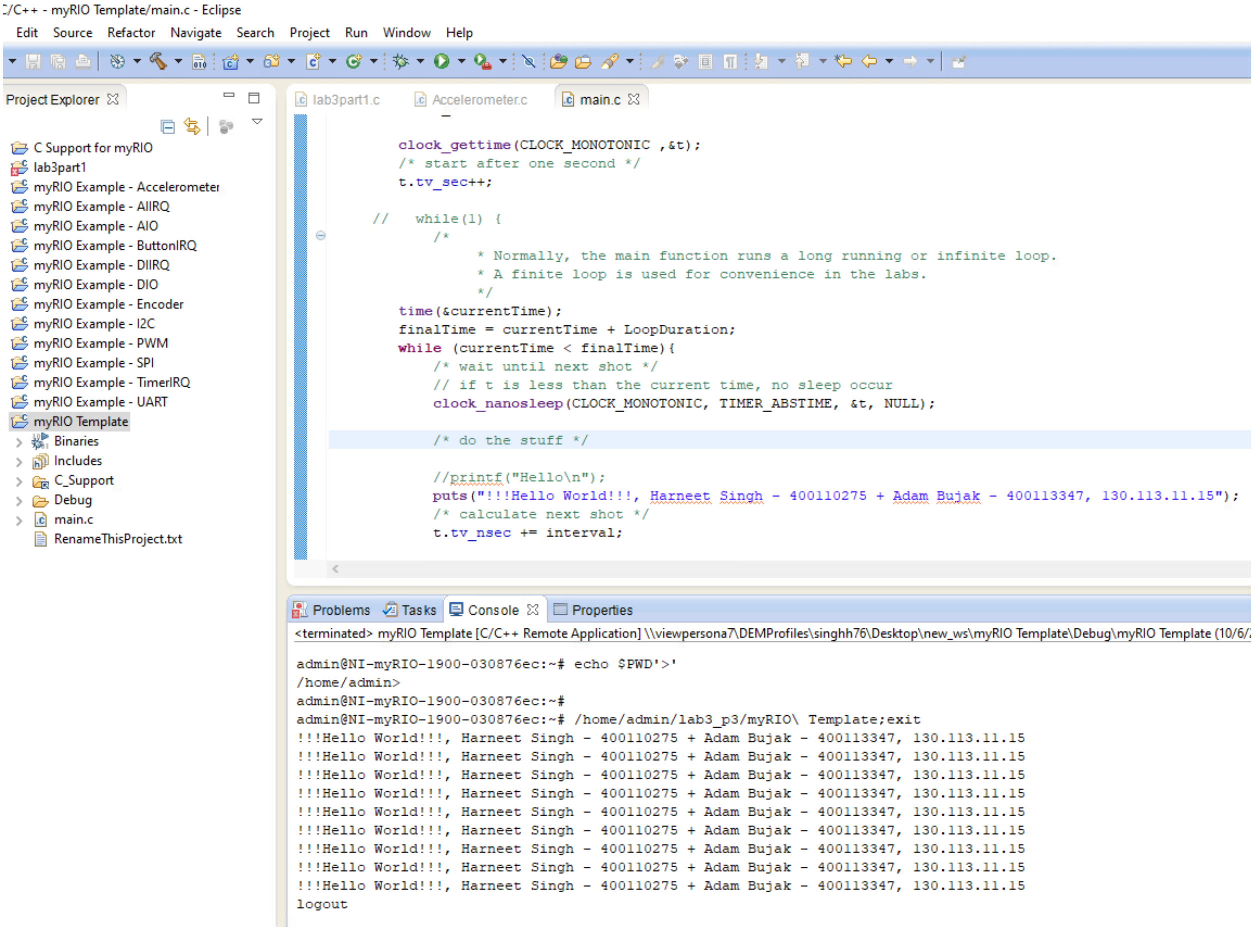
<terminated> myRIO Example - Accelerometer [C/C++ Remote Application] \\viewpersona7\DEMProfiles\singhh76\Desktop\new_ws\myRIO E

```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p2/myRIO\ Example\ -\ Accelerom eter;exit
Accelerometer
X: -0.00390625, Y: 0.0351562, Z: 1
X: 0, Y: 0.0351562, Z: 1
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 1
X: 0, Y: 0.0351562, Z: 1
X: -0.0078125, Y: 0.03125, Z: 0.996094
X: 0, Y: 0.03125, Z: 1
X: -0.00390625, Y: 0.0351562, Z: 1
X: 0, Y: 0.03125, Z: 0.996094
X: 0, Y: 0.03125, Z: 0.992188
X: -0.0078125, Y: 0.0351562, Z: 0.984375
X: 0, Y: 0.03125, Z: 1
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.0117188, Y: 0.0351562, Z: 1
X: 0, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 1.00391
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.0078125, Y: 0.03125, Z: 0.992188
X: -0.00390625, Y: 0.03125, Z: 1
X: 0, Y: 0.03125, Z: 1
X: -0.0078125, Y: 0.03125, Z: 0.992188
X: -0.00390625, Y: 0.03125, Z: 1.00391
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: -0.00390625, Y: 0.03125, Z: 0.996094
X: -0.00390625, Y: 0.0351562, Z: 0.996094
X: 0.00390625, Y: 0.0351562, Z: 1
```

Part 3.



```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p3/myRIO\ Template;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
logout
```

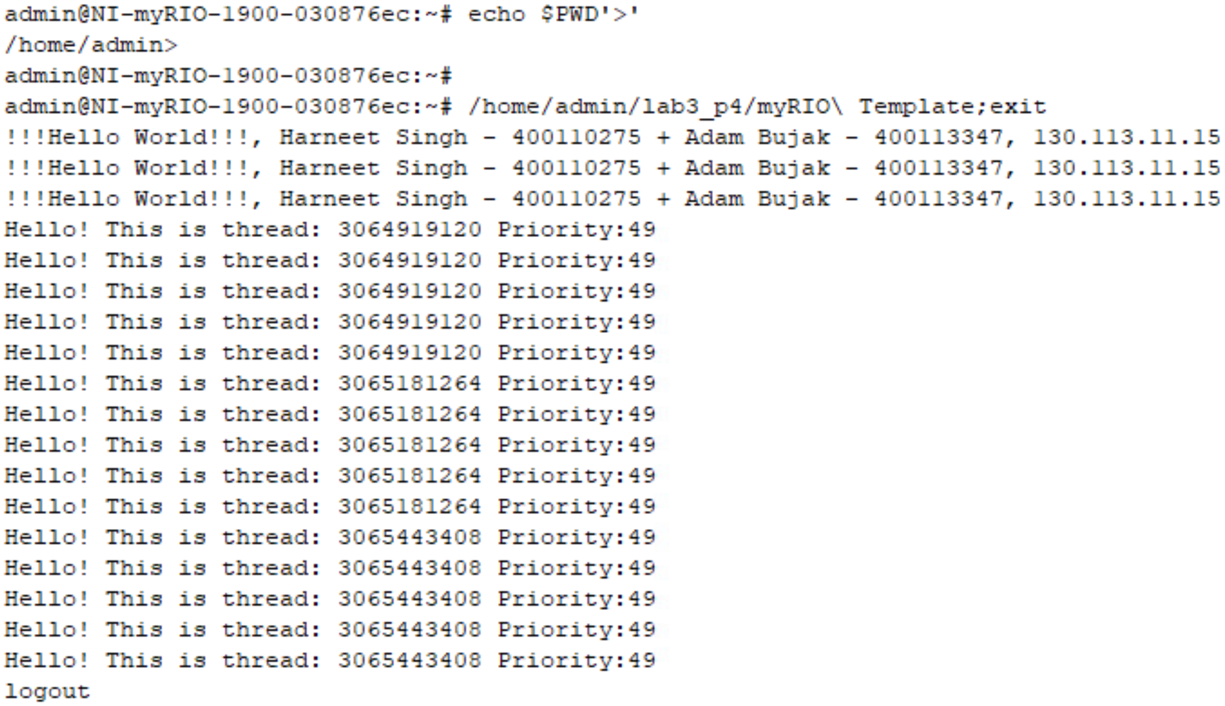


```
        clock_gettime(CLOCK_MONOTONIC ,&t);
        /* start after one second */
        t.tv_sec++;

//      while(1) {
            /*
                 * Normally, the main function runs a long running or infinite loop.
                 * A finite loop is used for convenience in the labs.
                 */
        time(&currentTime);
        finalTime = currentTime + LoopDuration;
        while (currentTime < finalTime){
            /* wait until next shot */
            // if t is less than the current time, no sleep occur
            clock_nanosleep(CLOCK_MONOTONIC, TIMER_ABSTIME, &t, NULL);

            /* do the stuff */

            //printf("Hello\n");
            puts("!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15");
            /* calculate next shot */
            t.tv_nsec += interval;
```

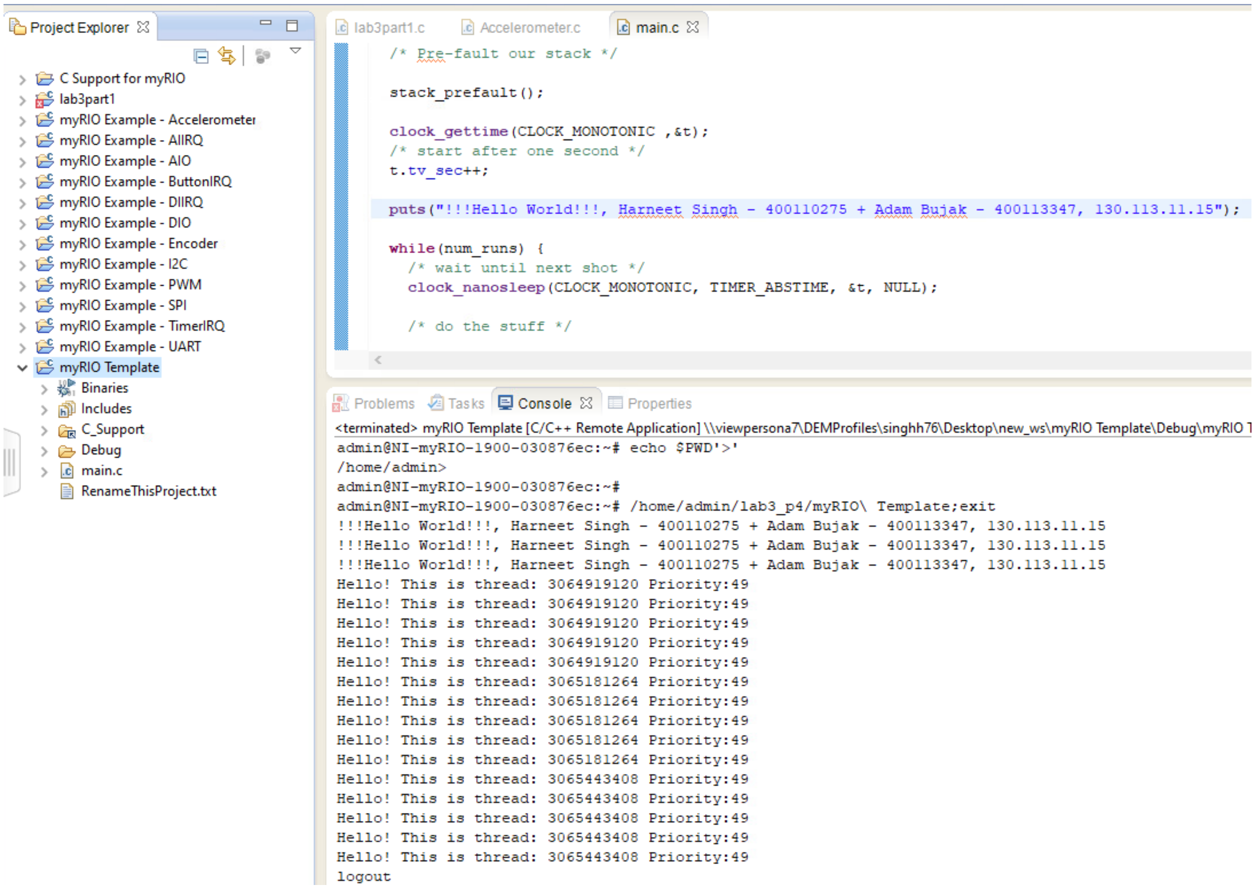
<terminated> myRIO Template [C/C++ Remote Application] \\viewpersona7\DEMProfiles\singhh76\Desktop\new_ws\myRIO Template\Debug\myRIO Template (10/6/

```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p3/myRIO\ Template;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
logout
```

Part 4.



```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p4/myRIO\ Template;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
logout
```



```
/* Pre-fault our stack */

stack_prefault();

clock_gettime(CLOCK_MONOTONIC ,&t);
/* start after one second */
t.tv_sec++;

puts("!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15");

while(num_runs) {
  /* wait until next shot */
  clock_nanosleep(CLOCK_MONOTONIC, TIMER_ABSTIME, &t, NULL);

  /* do the stuff */
```

<terminated> myRIO Template [C/C++ Remote Application] \\viewpersona7\DEMProfiles\singhh76\Desktop\new_ws\myRIO Template\Debug\myRIO T

```
admin@NI-myRIO-1900-030876ec:~# echo $PWD'>'
/home/admin>
admin@NI-myRIO-1900-030876ec:~#
admin@NI-myRIO-1900-030876ec:~# /home/admin/lab3_p4/myRIO\ Template;exit
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
!!!Hello World!!!, Harneet Singh - 400110275 + Adam Bujak - 400113347, 130.113.11.15
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3064919120 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065181264 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
Hello! This is thread: 3065443408 Priority:49
logout
```

Report

1. **Explain the purpose of functions: `sched_setscheduler()`, `mlockall()` and `stack_prefault()`**

The purpose of `sched_setscheduler` is to set the specified process' scheduling policy. The supported options are SCHED_OTHER, SCHED_BATCH, SCHED_IDLE, SCHED_FIFO, and SCHED_RR.

The purpose `mlockall()` is to prevent page faults by locking the process' memory in RAM and not allowing it to be moved to the swap (secondary memory emulating RAM), as this is a slow operation since secondary memory is slow.

The purpose of `stack_prefault()` is to prefault the entire memory of the process. This means to move any of the process' data stored in the swap area to the primary memory. By accessing every memory address, we force anything stored in secondary memory to be brought back into primary memory.

2. **We used a function clock_gettime(CLOCK_MONOTONIC ,&t), where t is struct of type timespec. Give details of the struct timespec and explain the meaning of CLOCK_MONOTONIC.**
   The struct timespec has the following structure:

```
struct timespec {
 time_t   tv_sec;        /* seconds */
 long     tv_nsec;       /* nanoseconds */
};
```

So, by calling clock_gettime(CLOCK_MONOTONIC, &t) the function sets our local t variable's seconds and nanoseconds fields. The CLOCK_MONOTONIC field indicates to the clock_gettime() function that we want the time elapsed from some fixed point in time which can never be changed. This is useful because we know that if we are comparing CLOCK_MONOTONIC times they will always be compared to a fixed point in time and won't be changed by some other process.

3. **What is the time period of this task? How can you assign a time period of 20ms to this task?**
   Since the interval is 50,000,000 this corresponds to 50,000,000ns which makes the time period 50ms. To make the time period 20ms we should set `int interval = 20000000;`
4. **Explain the purpose of the function sched_setaffinity() and its parameters.**

   A CPU affinity mask is a number that indicates on which CPUs a process should run. The sched_setaffinity() function sets the specified process' CPU affinity mask.

```
int sched_setaffinity(pid_t pid, size_t cpusetsize,
                             const cpu_set_t *mask);
```

Parameter	Description
pid	The process ID of the process whose cpu affinity the user wishes to modify
cpusetsize	The size of the mask the user is specifying in bytes

mask	Pointer to the mask which the user wishes to set. ie if the user whishes to use CPUs 0 and 2 the user would do the following:
	cpu_set_t mask = 0x5;
	sched_setaffinity(PID, sizeof(cpu_set_t), &mask);

5. Explain the purpose of the function pthread_create() and its parameters.

pthread_create() is used to create a new thread in the current process.

```
int pthread_create(pthread_t *restrict thread,
                          const pthread_attr_t *restrict attr,
                          void *(*start_routine)(void *),
                          void *restrict arg);
```

Parameter	Description
thread	Once the thread is created the thread ID is stored in this pointer
attr	Pointer to a structure that specifies some configuration options for the thread
start_routine	The function which is executed once the thread starts
arg	Some value the user wishes to be passed to the start_routine