Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Total: **50 marks**

Problem 1. (10 points) Provide the state-space representation of the dynamics of a DC Motor. Assume that there is no additional load on the motor. Next, Design a Simulink model to capture the dynamics and simulate the model for an input PWM voltage signal with magnitude 1V, frequency 1 kHz and duty cycle 0.1. Assume that the kinetic friction of the motor is negligible. Take the values of the other parameters from Example 7.13 in [LS15]. Provide the plot showing how the angular velocity of the motor varies with time.
[LS15] Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Second Edition, http://LeeSeshia.org, ISBN 978-1-312-42740-2, 2015.

Parameters of motor
The motor parameters are defined as follows:

$$I = 3.88 \times 10^{-7} \text{ kg} \cdot \text{m}^2$$
$$k_b = 2.75 \times 10^{-4} \text{ volts/RPM}$$
$$k_T = 5.9 \times 10^{-3} \text{ newton} \cdot \text{meters/amp}$$
$$R = 1.71 \text{ ohms}$$
$$L = 1.1 \times 10^{-4} \text{ henrys}$$

Mathematical Modelling:

Initially, at $t = 0^+$,

$$V(t) = Ri(t) + L\frac{di(t)}{dt} \quad \iff \quad V(s) = (R + sL)i(s)$$

Due to this current $i(t)$, torque is generated in the motor shaft,

$$\tau(t) = k_T i(t) \quad \iff \quad \tau(s) = k_T i(s)$$

This leads to the torque equation (with constant load torque τ_L, which equals 0 in this case):

$$I\frac{d\omega(t)}{dt} = \tau(t) - \tau_L \quad \iff \quad sI\omega(s) = \tau(s) - \tau_L \tag{1}$$

Now, due to the rotation of the motor, there is an armature reaction leading to back e.m.f generated,

$$E_b(t) = k_b\omega(t) \quad \iff \quad E_b(s) = k_b\omega(s)$$

This leads to a change in the original equation:

$$V(t) - E_b(t) = Ri(t) + L\frac{di(t)}{dt} \quad \iff \quad V(s) - k_b\omega(s) = Ri(s) + sLi(s) \tag{2}$$

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

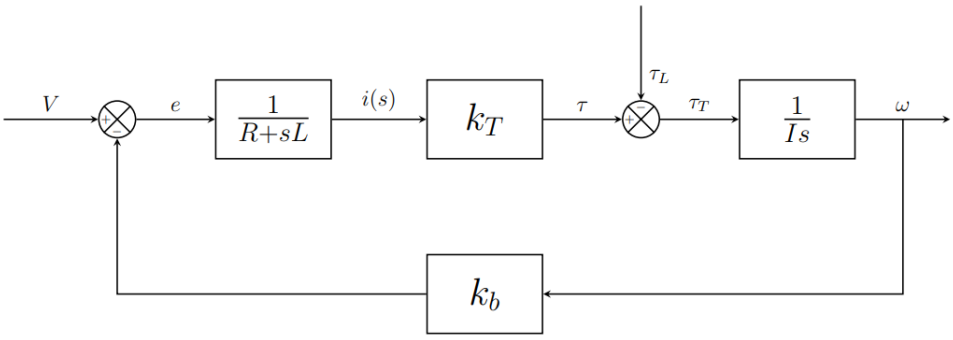


Figure 1: Block Diagram for Voltage Controlled DC Motor

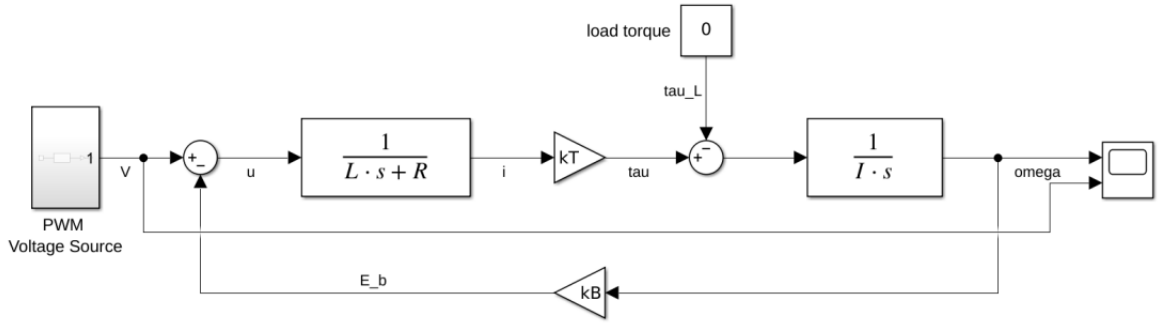


Figure 2: Transfer Function

For obtaining the state-space equations, consider equations (1) and (2), we get:

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} 0 & \frac{k_T}{I} \\ -\frac{k_B}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 & -\frac{1}{I} \\ \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} V \\ \tau_L \end{bmatrix} \tag{3}$$

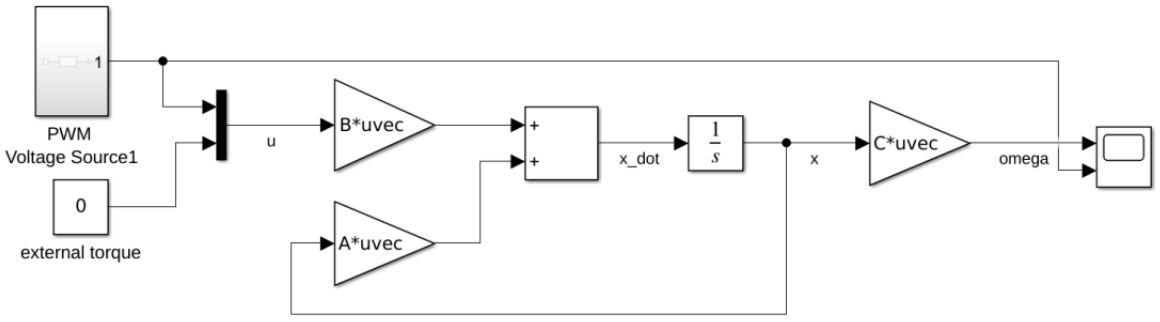


Figure 3: Steady Space

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Resultant response of both representation for 1V DC Input with 0.1 duty cycle at 1kHz:

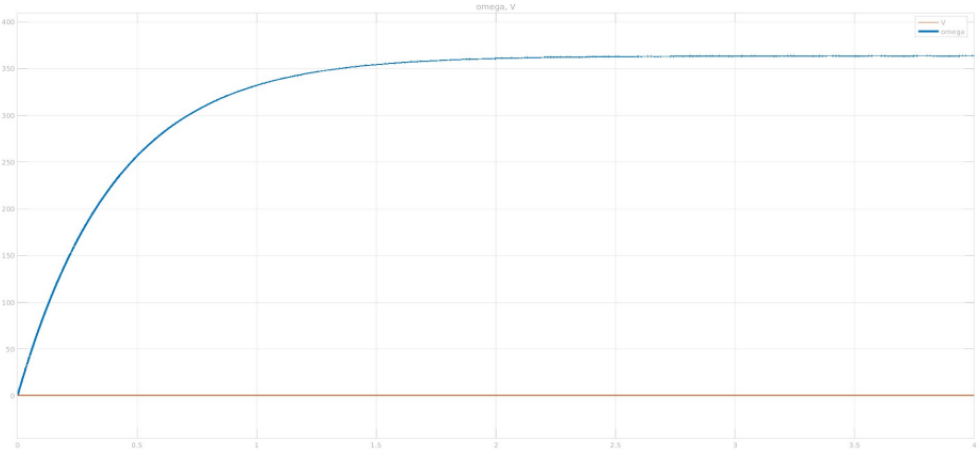


Figure 4: Response to 0.1 V

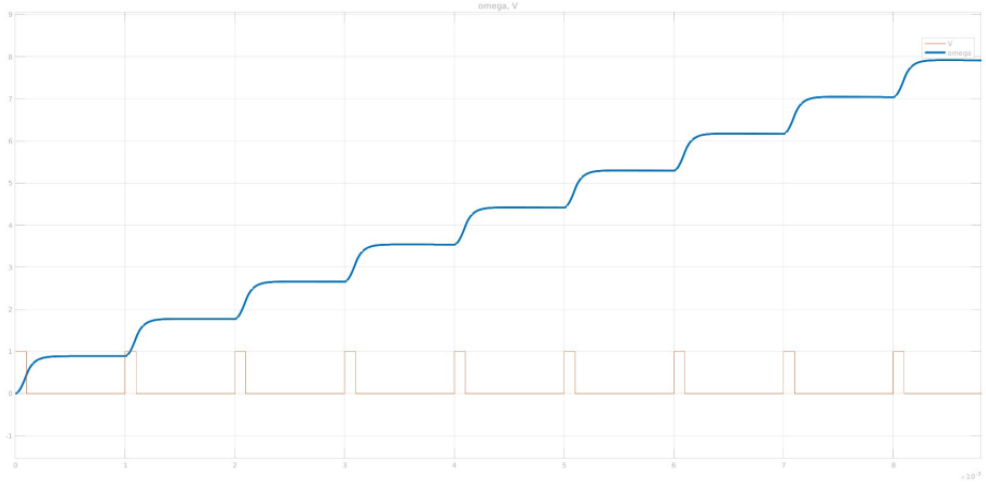


Figure 5: Initial response (blue), Input (red)

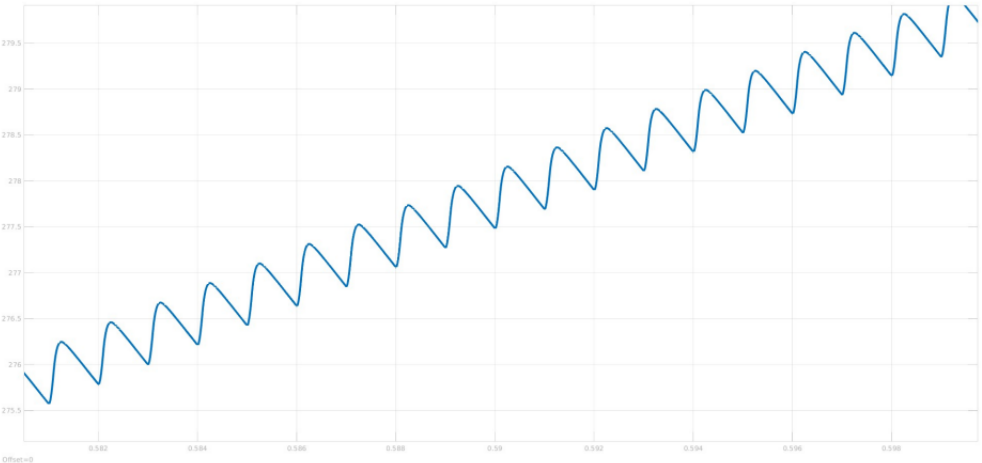


Figure 6: PWM rise and fall visible in response curve

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

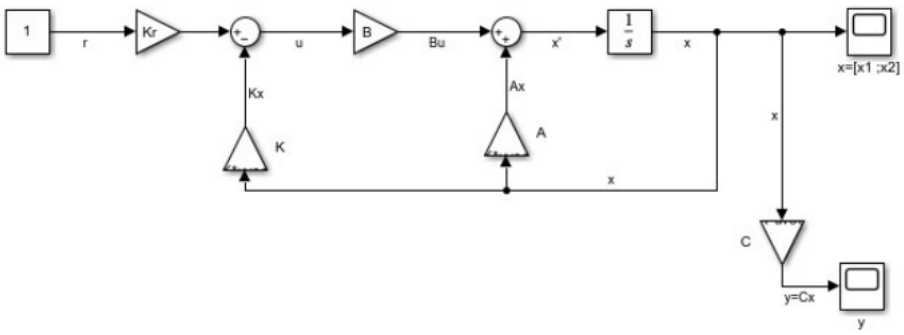
Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Problem 2. (10 points) Consider the vehicle steering control problem in Example 6.4 in [AM09]. Assume that $k_1 = 1$, $k_2 = 1.6$, and $k_r = 1$. Model the control system in Simulink using double precision floating point arithmetic. Now replace the model of the controller with the ones that use 16 bit and 8-bit fixed-point arithmetic. In each case, determine the fixed-point data types precisely. Plot the difference between the first state for the floating-point controller and that for the fixed-point controllers. Generate code for both the floating point controller and the fixed-point controllers using different optimization options. Describe the steps you have followed for the code generation.

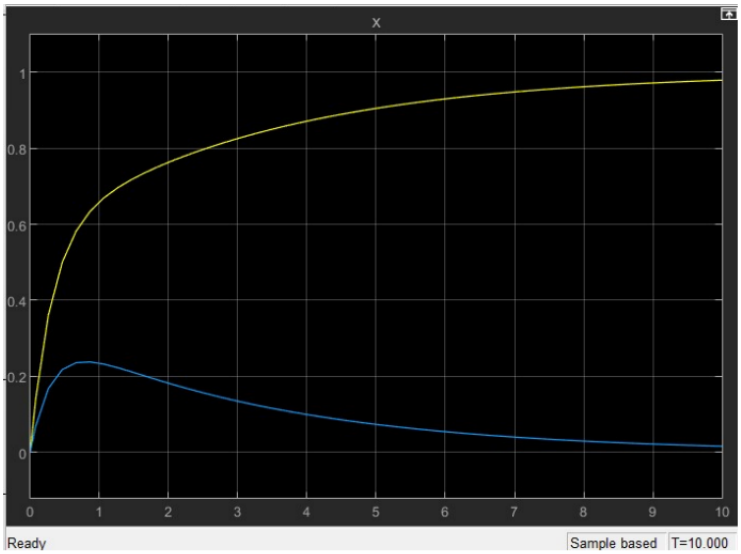
[AM09] K. J. Astrom and R. M. Murray. Feedback Systems: An Introduction for Scientists and Engineers. Princeton University Press, 2009.
http://www.cds.caltech.edu/∼murray/books/AM05/pdf/am08-complete_22Feb09.pdf.

Solution:

Simulink Model:



Double Precision Point Arithmetic:



Name: Himesh Baraik

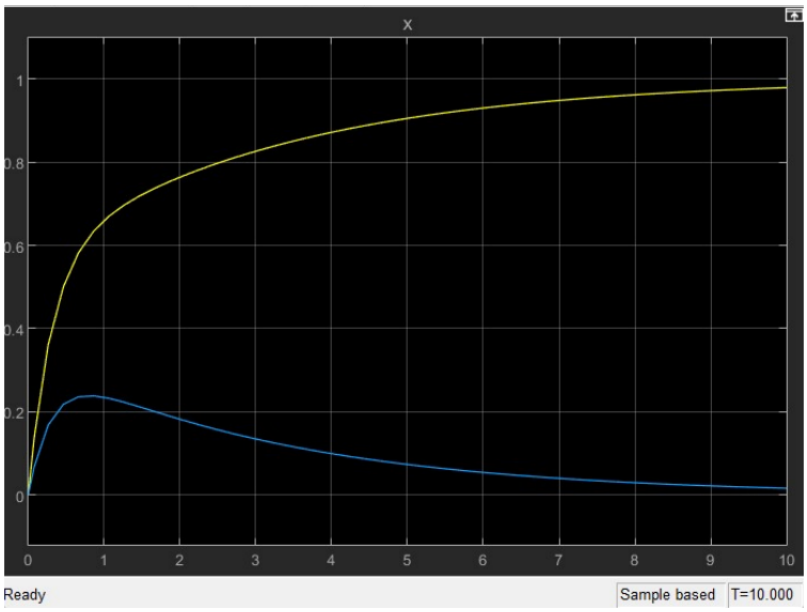
Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

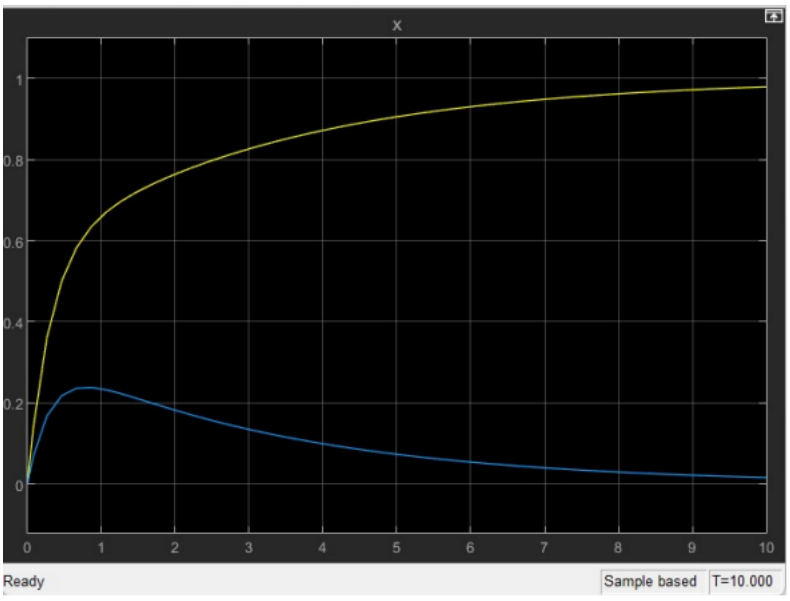
Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

---

8 Bit Fixed Point Arithmetic:



16 Bit Fixed Point Arithmetic:



Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

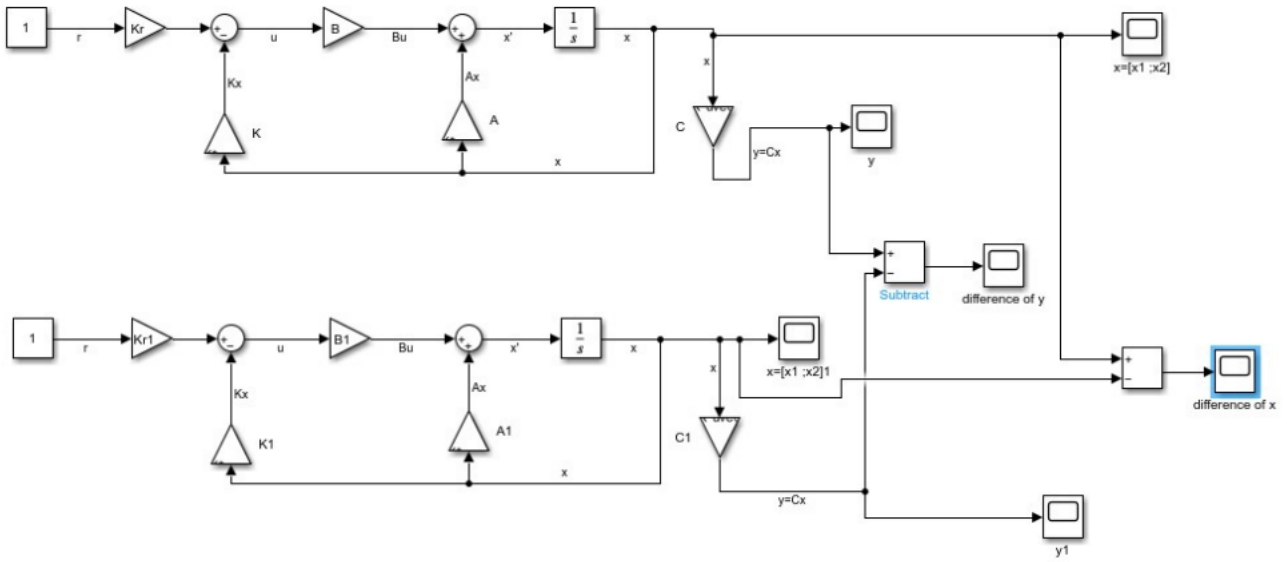
```
>> A=sfi(A,8);
>> gamma=sfi(gamma,8);
>> B=sfi(B,8);
>> K=sfi(K,8)

K =

    1.0000    1.5938

          DataTypeMode: Fixed-point: binary point scaling
            Signedness: Signed
            WordLength: 8
        FractionLength: 6
>> Kr=sfi(Kr,8);
>> C=sfi(Kr,8);
>> C=sfi(C,8);
>> C=[1 0];
>> C=sfi(C,8);
>> A=sfi(A,16);
>> gamma=sfi(gamma,8);
>> gamma=sfi(gamma,16);
>> B=sfi(B,16);
>> K=sfi(K,16);
>> C=sfi(C,16);
>> Kr=sfi(Kr,16);
```

Name: Himesh Baraik

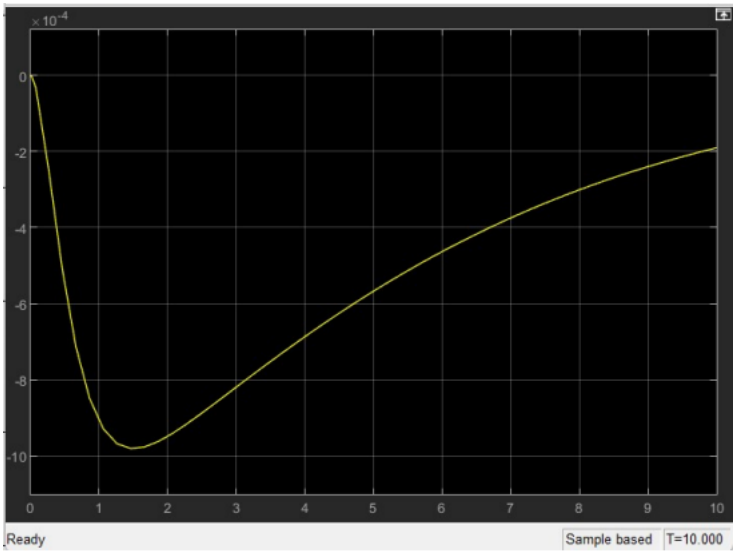
Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024



Differnce of Y in floating point controller and fixed point Controller



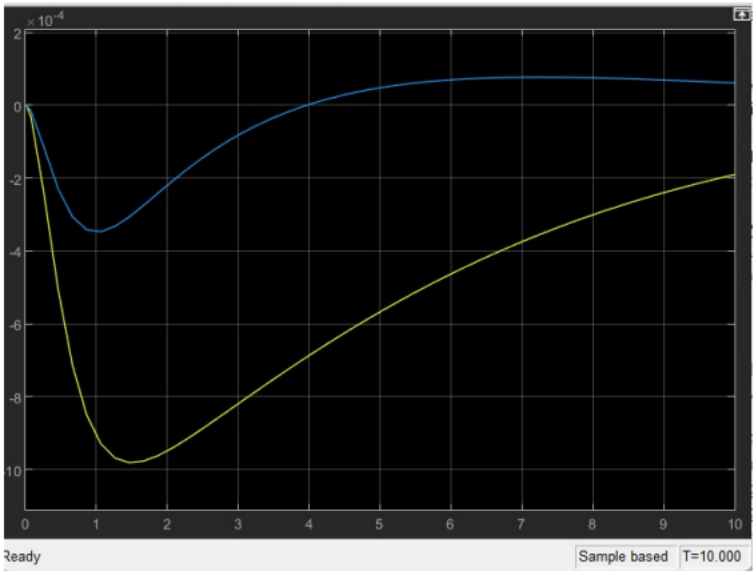
Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Differnce of X in floating point controller and fixed point Controller



Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Problem 3. (10 points) Work out Problem 1 in the Exercises of Chapter 9 in [LS15].
[LS15] Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Second Edition, http://LeeSeshia.org, ISBN 978-1-312-42740-2, 2015.

Solution:

```
int data[N];

int compute_variance() {
    int sum1 = 0, sum2 = 0, result;
    int i;

    for(i=0; i < N; i++) {
        sum1 += data[i];
    }
    sum1 /= N;

    for(i=0; i < N; i++) {
        sum2 += data[i] * data[i];
    }
    sum2 /= N;

    result = (sum2 - sum1*sum1);

    return result;
}
```

Cache Memory Analysis

Main Memory:
Main memory is divided into blocks of 4 words (16 bytes), with a total of 4094 blocks, making up 64 kB.

Cache:
The cache is divided into 128 blocks (each block has 4 words = 16 bytes), giving a total cache size of 2 kB. The cache is direct-mapped, meaning each memory block i is mapped to cache block $i \mod 128$.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Part 1

Let $N = 16$.

Loop 1:

1. **Step 1 (data[0]):** Cache miss occurs as data[0] is not in cache. Memory block $0x00$ to $0x1F$ is loaded into cache set 0.
2. **Step 2 (data[1]):** Cache hit; data[1] is already in cache set 0.
3. **Step 3 (data[2]):** Cache miss occurs. Memory block $0x20$ to $0x3F$ is loaded into cache set 1.
4. **Step 4 (data[3]):** Cache hit; data[3] is already in cache set 1.

This pattern continues, resulting in:

- Every even-indexed access results in a cache miss, while every odd-indexed access results in a cache hit.

Total Cache Misses:
8 out of 16 accesses (50% miss rate).

Loop 2:
Since the data array is already in cache, all accesses will be cache hits.

Total Cache Misses:
0 out of 16 accesses (0% miss rate).

Summary for $N = 16$:

Loop 1: 8 misses
Loop 2: 0 misses
Total: 8 out of 32 accesses (25% miss rate).

Part 2:

Let $N = 32$.

Loop 1:
The first 16 elements (data[0] to data[15]) are loaded into cache similarly as in Part 1, with 8 cache misses. When accessing data[16], a cache miss occurs as it maps to cache set 0, which now contains data[0] and data[1]. A cache miss occurs, loading block $0x100$ to $0x11F$, evicting data[0] and data[1].
This pattern continues, resulting in: A cache miss for every even-indexed access from data[16] onwards.

Total Cache Misses:
16 out of 32 accesses (50% miss rate).

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Loop 2:
After Loop 1, accessing data[0] results in a cache miss (loading back block $0x00$ to $0x1F$). Every two iterations result in a cache miss as blocks are reloaded into cache.

Total Cache Misses:
16 out of 64 accesses (25% miss rate).

Summary for $N = 32$:

Loop 1: 16 misses
Loop 2: 16 misses
Total: 32 out of 96 accesses (33% miss rate).

Part 3:

Let $N = 16$ with Cache Configuration $(m, S, E, B) = (32, 8, 2, 4)$.

Cache Configuration:

- 8 sets ($S = 8$)
- 2 lines per set ($E = 2$)
- Each block in the cache stores only one word ($B = 4$).

Loop 1:
Iteration 0 (data[0]): Cache miss occurs, loading block $0x00$ to $0x0F$ into cache set 0, line 0.
Iteration 1 (data[1]): Cache miss occurs, loading block $0x10$ to $0x1F$ into cache set 1, line 0.
This continues until data[8], which maps back to set 0. Since there are 2 lines per set, data[8] is stored in line 1 of cache set 0 without evicting data[0].This prevents immediate evictions.

Total Cache Misses:
16 cache misses out of 16 accesses in Loop 1.

Loop 2:
All data is already loaded in the cache from Loop 1, resulting in all accesses being cache hits.

Total Cache Misses:
0 cache misses out of 16 accesses.

Summary for $N = 16$ (2-way Set-Associative Cache):

Loop 1: 16 misses
Loop 2: 0 misses
Total: 16 out of 48 accesses (33% miss rate).

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Conclusion

Part 1 ($N = 16$):
8 misses in Loop 1, 0 in Loop 2. Total: 8 out of 32 accesses (25% miss rate).

Part 2 ($N = 32$):
32 misses out of 96 accesses (33% miss rate).

Part 3 ($N = 16$, 2-way associative cache):
16 misses in Loop 1, 0 in Loop 2. Total: 16 out of 48 accesses (33% miss rate).

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Problem 4. (10 points) Work out Problem 3 in the Exercises of Chapter 12 in [LS15].

[LS15] Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Second Edition, http://LeeSeshia.org, ISBN 978-1-312-42740-2, 2015.

Solution:

This problem compares RM and EDF schedules.
Consider two tasks with periods p1 = 2 and p2 = 3 and execution times e1 = e2 = 1. Assume that the deadline for each execution is the end of the period.

1. RM Schedule and Processor Utilization

Task Characteristics

$$\begin{aligned} &\text{Task 1 (T1):} \quad p_1 = 2, \quad e_1 = 1 \\ &\text{Task 2 (T2):} \quad p_2 = 3, \quad e_2 = 1 \end{aligned}$$

RM Schedule
Using the Rate Monotonic (RM) scheduling, we assign priorities based on the periods of the tasks: the task with the shorter period gets a higher priority. Thus:

- T1 has a higher priority than T2.

We can construct the schedule as follows:

- Time 0: T1 executes from 0 to 1 (remaining time 1)
- Time 1: T1 completes, T2 executes from 1 to 2 (remaining time 0)
- Time 2: T1 starts executing again from 2 to 3 (remaining time 1)
- Time 3: T1 completes; T2 executes from 3 to 4 (remaining time 0)
- Time 4: T1 starts executing again from 4 to 5 (remaining time 1)
- Time 5: T1 completes; T2 executes from 5 to 6 (remaining time 0)
- Time 6: T1 starts executing again from 6 to 7 (remaining time 1)
- Time 7: T1 completes; T2 executes from 7 to 8 (remaining time 0)

This pattern continues, and the schedule can be repeated every 6 time units since

$$\text{LCM}(2, 3) = 6.$$

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Processor Utilization
The utilization U for RM scheduling can be calculated as follows:

$$\begin{aligned} U &= \frac{e_1}{p_1} + \frac{e_2}{p_2} \\ &= \frac{1}{2} + \frac{1}{3} \\ &= \frac{3}{6} + \frac{2}{6} \\ &= \frac{5}{6} \approx 0.8333. \end{aligned}$$

Liu and Layland Utilization Bound
The utilization bound for n tasks is given by:

$$U(n) = n \cdot \left(2^{1/n} - 1\right).$$

For $n = 2$:

$$\begin{aligned} U(2) &= 2 \cdot \left(2^{1/2} - 1\right) \\ &= 2 \cdot (1.414 - 1) \\ &= 2 \cdot 0.414 \\ &= 0.828. \end{aligned}$$

Thus, we have:

$$U = 0.8333, \quad U(2) \approx 0.828.$$

Since $U > U(2)$, this implies the tasks are schedulable under RM but close to the utilization bound.

2. Feasibility with Increased Execution Times

Increase in e_1 or e_2
If we increase either e_1 or e_2:

- If e_1 increases:

$$U = \frac{e_1}{p_1} + \frac{1}{2} + \frac{1}{3} > 1$$

  implies

$$U = \frac{2}{e_1} + \frac{1}{3} > 1 \quad \text{(infeasible)}$$

- If e_2 increases:

$$U = \frac{1}{2} + \frac{e_2}{3} + \frac{1}{3} > 1$$

  implies

$$U = \frac{1}{2} + \frac{3}{e_2} > 1 \quad \text{(infeasible)}$$

Thus, any increase in execution time makes the RM schedule infeasible.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Reducing p_1
If we hold $e_1 = e_2 = 1$ and $p_2 = 3$ constant, we can try reducing p_1:

- If $p_1 = 1$:
$$U = \frac{1}{1} + \frac{1}{3} = 1 + 0.333 > 1 \quad \text{(infeasible)}$$
- If $p_1 = 1.5$:
$$U = \frac{1}{1.5} + \frac{1}{3} = 0.666 + 0.333 = 1 \quad \text{(feasible)}$$

Thus, we can reduce p_1 to a maximum of 1.5.

Reducing p_2
If we hold $e_1 = e_2 = 1$ and $p_1 = 2$ constant, we can try reducing p_2:

- If $p_2 = 2.5$:
$$U = \frac{1}{2} + \frac{1}{2.5} = 0.5 + 0.4 = 0.9 \quad \text{(feasible)}$$
- If $p_2 = 2$:
$$U = \frac{1}{2} + \frac{1}{2} = 1 \quad \text{(infeasible)}$$

Thus, we can reduce p_2 to a maximum of 2.5.

3. EDF Schedule with Increased Execution Time of Task 2

New Task Characteristics

- Task 1 (T_1):
$$p_1 = 2, \quad e_1 = 1$$
- Task 2 (T_2):
$$p_2 = 3, \quad e_2 = 1.5$$

EDF Schedule

Using the Earliest Deadline First (EDF) scheduling, we schedule the tasks based on their deadlines:

- Time 0: T_1 executes from 0 to 1 (remaining time 1)
- Time 1: T_2 executes from 1 to 2.5 (remaining time 0.5)
- Time 2.5: T_1 executes from 2.5 to 3 (remaining time 0)
- Time 3: T_2 executes from 3 to 4 (remaining time 0)
- Time 4: T_2 executes from 4 to 5 (remaining time 0)

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

We can see that:

- At Time 2: T_1 is scheduled first due to its earlier period.
- At Time 3: T_2 has remaining time to complete but its deadline is missed, as it was only supposed to run until time 3, which is infeasible.

Feasibility
Since T_2 cannot complete within its deadline, the EDF schedule is not feasible.

Processor Utilization
The utilization U for EDF can be calculated as follows:

$$U = \frac{e_1}{p_1} + \frac{e_2}{p_2} = \frac{1}{2} + \frac{1.5}{3} = 0.5 + 0.5 = 1$$

Thus, the processor utilization is $U = 1$, which means it is fully utilized but cannot meet the deadlines due to the higher execution time of T_2.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Problem 5. (10 points) Work out Problem 6 in the Exercises of Chapter 12 in [LS15].

[LS15] Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Second Edition, http://LeeSeshia.org, ISBN 978-1-312-42740-2, 2015.

Solution:

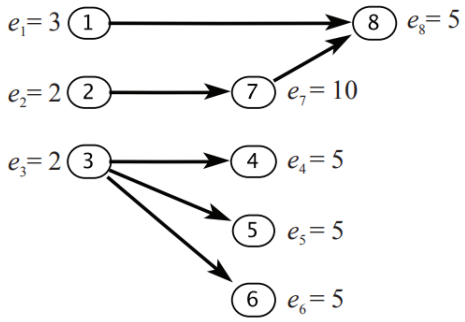


Figure 12.16: Precedence Graph for Exercise 6.

Precedence Constraints

- T_1 must finish before T_8
- T_2 must finish before T_7
- T_7 must finish before T_8
- T_3 must finish before T_4, T_5, and T_6

Task Execution Times

- Task 1 (T_1): $e_1 = 3$
- Task 2 (T_2): $e_2 = 2$
- Task 3 (T_3): $e_3 = 2$
- Task 4 (T_4): $e_4 = 5$
- Task 5 (T_5): $e_5 = 5$
- Task 6 (T_6): $e_6 = 5$
- Task 7 (T_7): $e_7 = 10$
- Task 8 (T_8): $e_8 = 5$

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

(a) Scheduling on Two Processors

Schedule:

- **Processor 1 (P1):**

$$\boxed{\text{T1 (3)}} \rightarrow \boxed{\text{T3 (2)}} \rightarrow \boxed{\text{T4 (5)}}$$

- **Processor 2 (P2):**

$$\boxed{\text{T2 (2)}} \rightarrow \boxed{\text{T7 (10)}} \rightarrow \boxed{\text{T8 (5)}}$$

Execution Order:

- $T_1\,(3)$ on P_1: 0 to 3
- $T_2\,(2)$ on P_2: 0 to 2
- $T_3\,(2)$ on P_1: 3 to 5
- $T_4\,(5)$ on P_1: 5 to 10
- $T_7\,(10)$ on P_2: 2 to 12
- $T_8\,(5)$ on P_2: 12 to 17

Makespan:

The final completion time is 17 time units.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

(b) Scheduling on Three Processors

Schedule:

- **Processor 1 (P1):**

  T1 (3) → T3 (2)

- **Processor 2 (P2):**

  T2 (2) → T4 (5)

- **Processor 3 (P3):**

  T5 (5) → T6 (5)

Execution Order:

- $T_1\,(3)$ on P_1: 0 to 3
- $T_2\,(2)$ on P_2: 0 to 2
- $T_3\,(2)$ on P_1: 3 to 5
- $T_4\,(5)$ on P_2: 2 to 7
- $T_5\,(5)$ on P_3: 5 to 10
- $T_6\,(5)$ on P_3: 10 to 15
- $T_7\,(10)$ on P_2: 7 to 17
- $T_8\,(5)$ on P_2: 17 to 22

Makespan:

The final completion time is 22 time units.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001) Dept.: EE (e.g. CSE)

(c) Scheduling on Two Processors with Reduced Execution Times

Execution Times:

- Task 1 (T_1): $e_1 = 2$
- Task 2 (T_2): $e_2 = 1$
- Task 3 (T_3): $e_3 = 1$
- Task 4 (T_4): $e_4 = 4$
- Task 5 (T_5): $e_5 = 4$
- Task 6 (T_6): $e_6 = 4$
- Task 7 (T_7): $e_7 = 9$
- Task 8 (T_8): $e_8 = 4$

Schedule:

- **Processor 1 (P1):**

$$\boxed{\text{T1 (2)}} \rightarrow \boxed{\text{T3 (1)}} \rightarrow \boxed{\text{T4 (4)}}$$

- **Processor 2 (P2):**

$$\boxed{\text{T2 (1)}} \rightarrow \boxed{\text{T7 (9)}} \rightarrow \boxed{\text{T8 (4)}}$$

Execution Order:

- $T_1\,(2)$ on P_1: 0 to 2
- $T_2\,(1)$ on P_2: 0 to 1
- $T_3\,(1)$ on P_1: 2 to 3
- $T_4\,(4)$ on P_1: 3 to 7
- $T_7\,(9)$ on P_2: 1 to 10
- $T_8\,(4)$ on P_2: 10 to 14

Makespan:

The final completion time is 14 time units.

Name: Himesh Baraik

Roll No: 210447 (e.g. 170001)

Dept.: EE (e.g. CSE)

Indian Institute of Technology Kanpur
CS637 Embedded and Cyber-Physical Systems
Homework Assignment 2
Deadline: October 11, 2024

Summary of Makespans

- **Part (a)**: 17 time units (two processors)
- **Part (b)**: 22 time units (three processors)
- **Part (c)**: 14 time units (two processors with reduced execution times)

Conclusions

- The makespan is greater when using three processors (22) compared to using two processors (17).
- Reducing the execution times of tasks results in a smaller makespan (14) compared to the original two-processor scenario (17).