DVA421: Software Development for Real-Time Systems

The anti-slip regulation system (ASR)
The brake-by-wire system (BBW)

Students: Simone Aglianò
Léa Brunschwig
Date: 2017-04-16

Assignment 1: Requirements and High Level Design

1. Introduction

The purpose of this exercise is to specify two sub-system within a car. The two sub-systems which are going to be tackled in this exercise are the anti-slip regulation system (ASR) and the brake-by-wire system (BBW).

The goal of the ASR is to avoid that one or more wheels are spinning too fast comparing to the others wheel due to a hard acceleration or a slippery surface. Thus, the system has to reduce the speed engine or to apply the brake on the concerned wheel(s). It also warns the driver while this system is in action by switching on an alarm light.

The brake-by-wire system has for main purpose to replace the physical connection from cars between the brake pedal and the brakes by a computer system. The challenge of this sub-system is to make sure that the information given by the brake pedal will reach in time the brakes and that in feedback the brakes will give the current pressure applied on them in order to give the feeling of resistance on the pedal while the driver is pushing it.

Firstly, a global overview of the sub-systems architecture is going to be showed, then the different functions with the time requirements of each of the sub-systems are going to be described and finally we will give a high-level software design of both of the sub-systems.

2. System Architecture

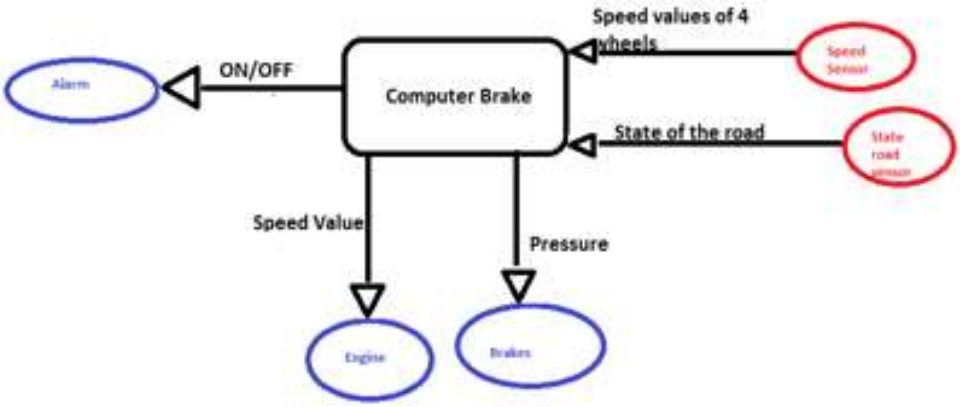

Figure 1.1: The system architecture of the Anti-slip regulation (ASR)

a. Sensors

The system will make use of the following sensors:

Sensor 1: Speed-Sensor

There are 4 sensors, one for each wheel. Each sensor catches the speed of the wheel.

Sensor 2: State-Road-Sensor

There are 4 sensors, one for each wheel. Each sensor analyses the state of the road in order to know if it is a slippery surface.

b. Actuators

The system will make use of the following actuators.

Actuator 1: Engine-Actuator

There are 4 actuators for each wheel. Each actuator reduces or increases the power of the engine. They are activated when the system is asking for it.

Actuator 2: Alarm-Actuator

There is 1 actuator of this type which is placed on the display. It is activated when the ASR is working.

Actuator 3: Brake-Actuator

There are 4 actuators, one for each brake. Each actuator reduces or increases the pression of the brakes on the wheel when there is a slippery situation. They are activated when the system is asking for it.

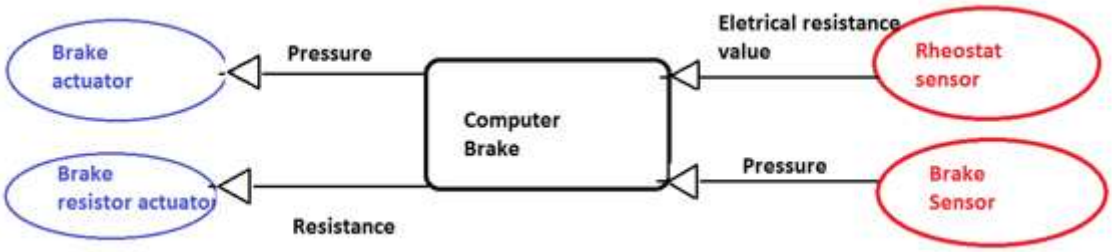

Figure 1.2: The system architecture of the Brake-by-wire (BBW)

a. Sensors

The system will make use of the following sensors:

Sensor 1: Rheostat-Sensor

This sensor measures electrical resistance on the brake pedal. There is only one of it on the brake pedal (and it might have one on the acceleration pedal but it not part of our subject). It might change when the driver push or release the brake pedal.

Sensor 2: Brake-Sensor

This sensor measures the pressure on the brake. There is one sensor per brake, thus four in our system. It might change when the brake compute signals servo pumps how much pressure to put into secondary circuit.

b. Actuators

The system will make use of the following actuators.

Actuator 1: Brake-Actuator

This actuator will apply a pressure on the brake when the brake pedal has been pushed (or it will release the pressure if the brake pedal is released). There is one actuator on every brake.

Actuator 2: Brake-Resistor-Actuator

This actuator will apply a resistance on the brake pedal to make the driver feels the pressure applied on the brakes. There is one on the brake pedal.

3. Functions and Timing Requirements

a. Anti-Slip Regulation system

Function 1: Hard acceleration

The system has to reduce the speed of one or more wheels if there is a hard acceleration.

Stimulus: Speed values of wheels (every wheel in order to compare their speeds)
Response: Reducing of engine speed
Additional information needed: State road sensor value

TR 1.1: The system has to react within 300 ms.
Motivation: Because this is a critical situation, thus it has to react fast.

TR 1.2: Action has to be done in at least 1 second from the start.
Motivation: Because if the action is done too fast it might be dangerous.

Function 2: Slippery surface

Firstly, the system has to reduce the power given to the engine if there is hard acceleration is detected. Secondly, if there is also a slippery surface the system will apply the brakes.

Stimulus: Speed values of wheels (every wheel in order to compare their speeds)
Response: Apply the brake on the concerned wheel.
Additional information needed: State road sensor value

TR 2.1: The system has to react within 300 ms.
Motivation: Because this is a critical situation, thus it has to react fast.

TR 2.2: Action has to be done in at least 1 second from the start.
Motivation: Because if the action is done too fast it might be dangerous.

Function 3: Slippery surface and hard acceleration

The system has to apply the brakes on one or more wheels if there is a slippery surface under one or more wheels.

Stimulus: Speed values of wheels (every wheel in order to compare their speeds)
Response: Reduce the engine power and then apply the brake on the concerned wheel.
Additional information needed: State road sensor value

TR 3.1: The system has to react within 150 ms.

Motivation: Because this is a critical situation, thus it has to react fast, moreover two different actions have to be done.

TR 3.2: Reduce engine power has to be done in 500 ms from the start.
Motivation: Because if the action is done too fast it might be dangerous.

TR 3.3: Apply the brake has to be done in at least 1 second and it has to start within 650ms even if the first action is not over.
Motivation: Because the second action is more important than the first one and need to be completed in time.

Function 4: Warning alarm

The system has to switch on the light for the alarm when the ASR is activated.

Stimulus: ASR is activated
Response: The light is switched on.

TR 4.1: The system has to react within 500 ms.
Motivation: Because it is not critical situation and the driver does not need to see this information quickly.

b. Brake-by-wire system

Function 1: Driver brakes

The driver pushes the brake pedal in order to make the vehicle braking.

Stimulus: Brake pedal is pushed.
Response: The car brakes.
Additional information needed: Current pressure on the pedal

TR 1.1: Within 100 ms.
Motivation: The brakes are critical devices and it has to be extremely fast.

Function 2: Car brakes

The car is braking and send to the computer the force feedback in order to create the feeling of resistance under the brake pedal.

Stimulus: Car is braking
Response: The drive can feel a resistance regarding the brake pedal.

TR 2.1: Within 100ms
Motivation: The brakes are critical devices and it has to be extremely fast, moreover it is really important for the driver to feel that resistance in order to know if he needs to brake more or not.

4. High-level Software Design

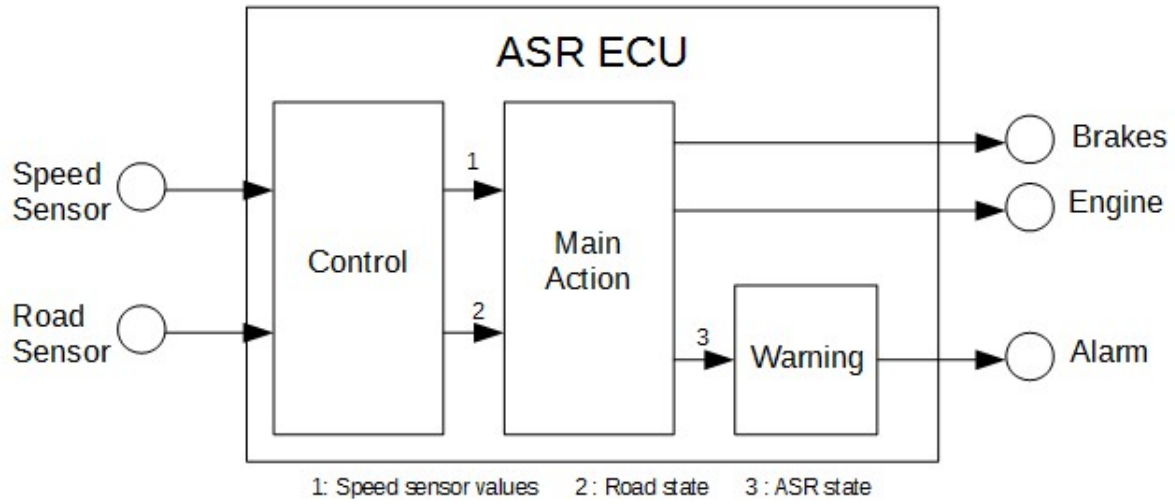

Figure 1.3: The software design of the anti-spin regulation system

Task 1: Control
Inputs: Speed of the wheels and state of the road
Outputs: Speed of the wheels and state of the road
Description: The system would sample the sensors for the next task.

Task 2: Main Action
Inputs: Speed of the wheels and state of the road
Outputs: Speed of the wheels and state of the road
Description: The system would first reduce the power of the engine and then would brake.

Task 3: Warning
Inputs: ASR State
Outputs: Alarm State
Description: If the ASR function is activated then the alarm is switched on.

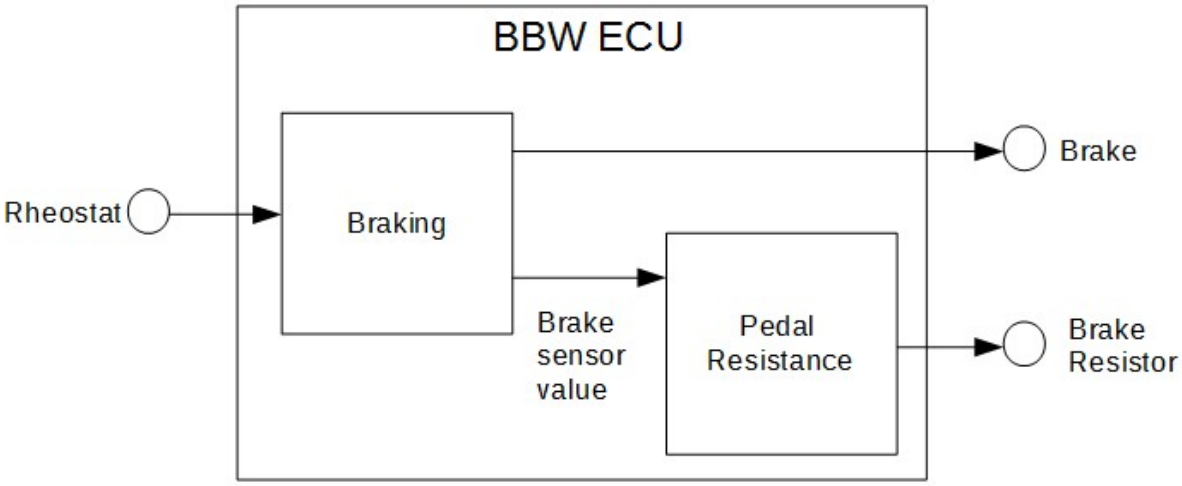

Figure 1.4: The software design of the brake-by-wire system

Task 1: Braking
Inputs: Rheostat
Outputs: Brake and Brake resistance
Description: The system would brake depending of the pressure on the pedal.

Task 2: Pedal resistance
Inputs: Brake resistance
Outputs: Pressure value on the brake resistor
Description: The system would apply a pressure on the pedal depending on the brake resistance collected.

Assignment 2: Timing Parameter Assignment and Schedulability Analysis

1. Real-time Task Parameters

a. Anti-Slip Regulation system

Task	Task name	WCET	Period	Deadline	Precedes
T1	Control	2	10	10	T2
T2	Main Action	5	75	75	T3
T3	Warning	4	250	250	-
T4	Braking	4	50	50	T5
T5	Pedal Resistance	4	50	50	-

Figure 2.1: Summary of the tasks and their timing parameters

Motivation for the task parameters:
For the 3 first tasks: The sensors have to be sampled at least every 20 ms, thus, by applying the Nyquist-Shannon sampling theorem which is saying that the period has to be the half of the timing requirements, we set the period of the first task to 10. Regarding the second task, the shortest timing requirements between the functions 1, 2 and 3 is 150 ms, that is why we have chosen, by applying the theorem, to set the period to 75. Finally, the period for the Warning task has been chosen with the same logic.
For the 2 last tasks: Regarding Nyquist-Shannon sampling theorem, the period has been chosen to be the half of the timing requirements of the tasks. Thus the deadline cannot be longer than the period that is why it is equal. Finally, the first task has to precede the second task because the feeling on the pedal is created by the braking itself.

2. Offline Scheduling

The method consists in three steps:

1. Construct a joint graph with the period equals to the least common multiple of all precedence graphs periods
2. Generate search tree
3. Traverse the search tree to find a solution

The system will become more predictable with the offline scheduling but it will also become less dynamic.
The scheduling algorithm can be used with preemption or without preemption. With the preemption, we have more possibilities to find a solution but the size of the tree will increase and the complexity of the search tree will increase exponentially.

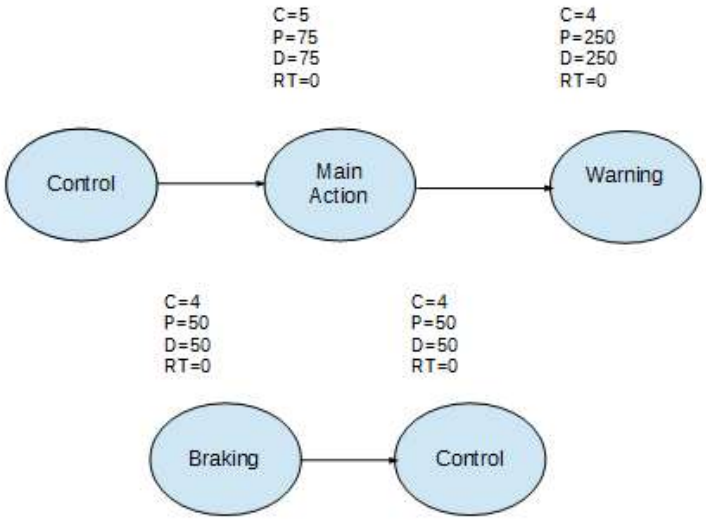

Figure 2.3: Precedence Graph

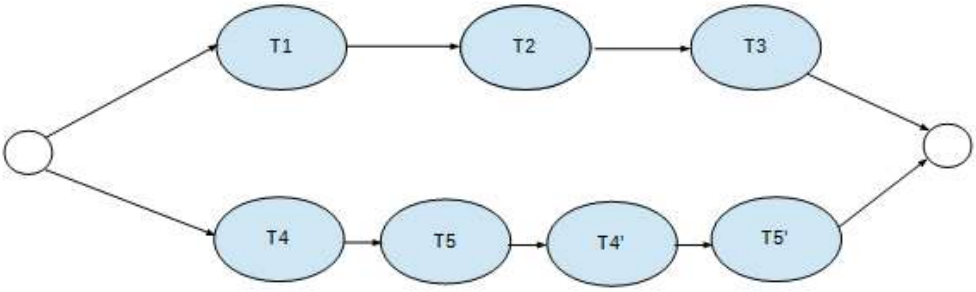

Figure 2.4: Joint Graph

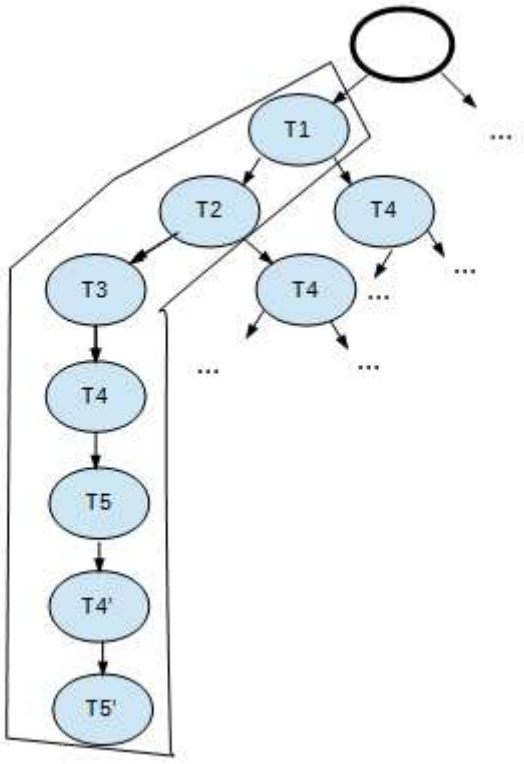

Figure 2.5: Tree Graph

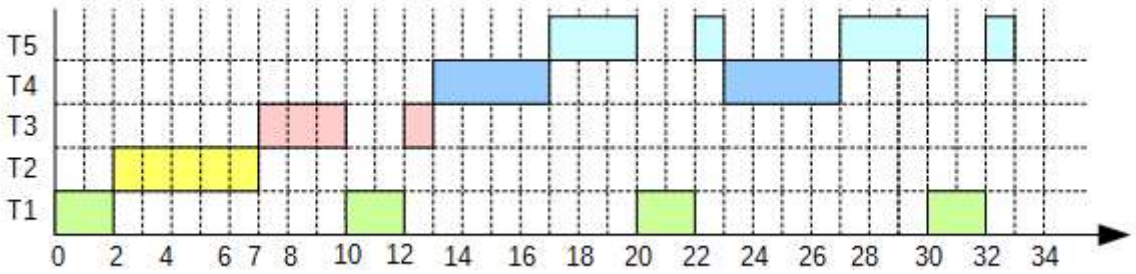

Figure 2.6: Offline schedule

Verdict: The task set is offline schedulable.

3. FPS scheduling

The deadline and the period are equal in every task, so there is no difference to choose Deadline Monotonic or Rate Monotonic. In order to solve this assignment, we have chosen Rate Monotonic.

Task	Task name	WCET	Period	Deadline	Priority
T1	Control	2	10	10	5
T2	Main Action	5	75	75	2
T3	Warning	4	250	250	1
T4	Braking	4	50	50	4
T5	Pedal Resistance	4	50	50	3

Figure 2.7: Task priorities

Schedulability test: All the tasks are independent of each other; we can use this scheduling algorithm.

We can also prove it by using Response Time Analysis (RTA):

$$R_i^{n+1} = C_i + \sum_{\forall j \in hp(i)} \left\lceil \frac{R_i^n}{T_j} \right\rceil C_j$$

$R_{T1} = C_{T1} = 2$

$R^0_{T4} = C_{T4} = 4$
$R^1_{T4} = C_{T4} + (R^0_{T4}/T_{T1})*C_{T1} = 4 + (4/10)*2 = 6$
$R^2_{T4} = C_{T4} + (R^1_{T4}/T_{T1})*C_{T1} = 4 + (6/10)*2 = 6$

$R^0_{T5} = C_{T5} = 4$
$R^1_{T5} = C_{T5} + (R^0_{T5}/T_{T1})*C_{T1} + (R^0_{T5}/T_{T4})*C_{T4} = 4 + (4/10)*2 + (4/50)*4 = 10$
$R^2_{T5} = C_{T5} + (R^1_{T5}/T_{T1})*C_{T1} + (R^1_{T5}/T_{T4})*C_{T4} = 4 + (10/10)*2 + (10/50)*4 = 10$

$R^0_{T2} = C_{T2} = 5$
$R^1_{T2} = C_{T2} + (R^0_{T2}/T_{T1})*C_{T1} + (R^0_{T2}/T_{T4})*C_{T4} + (R^0_{T2}/T_{T5})*C_{T5} = 15$

$R^2_{T2} = C_{T2} + (R^1_{T2}/T_{T1})*C_{T1} + (R^1_{T2}/T_{T4})*C_{T4} + (R^1_{T2}/T_{T5})*C_{T5} = 17$
$R^3_{T2} = C_{T2} + (R^2_{T2}/T_{T1})*C_{T1} + (R^2_{T2}/T_{T4})*C_{T4} + (R^2_{T2}/T_{T5})*C_{T5} = 17$

$R^0_{T3} = C_{T3} = 4$
$R^1_{T3} = C_{T3} + (R^0_{T3}/T_{T1})*C_{T1} + (R^0_{T3}/T_{T4})*C_{T4} + (R^0_{T3}/T_{T5})*C_{T5} + (R^0_{T3}/T_{T2})*C_{T2} = 18$
$R^2_{T3} = C_{T3} + (R^1_{T3}/T_{T1})*C_{T1} + (R^1_{T3}/T_{T4})*C_{T4} + (R^1_{T3}/T_{T5})*C_{T5} + (R^1_{T3}/T_{T2})*C_{T2} = 20$
$R^3_{T3} = C_{T3} + (R^2_{T3}/T_{T1})*C_{T1} + (R^2_{T3}/T_{T4})*C_{T4} + (R^2_{T3}/T_{T5})*C_{T5} + (R^2_{T3}/T_{T2})*C_{T2} = 20$

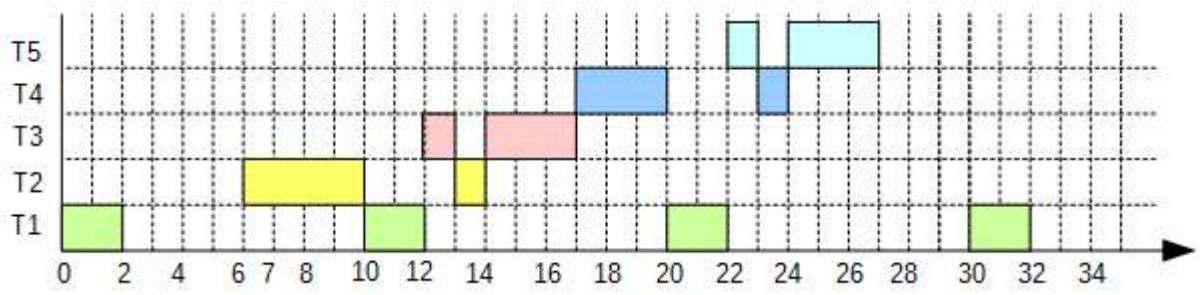

Figure 2.8: RTA Schedule

Verdict: The task set is FPS schedulable under the above priority assignment.

4. EDF scheduling

To test the schedulability of the system, we need to measure the CPU utilization and prove that U ≤ 1 but because we need to assure that T4 is executed before T5 and they both have the same deadline we cannot apply the EDF scheduling.

Verdict: The task set is EDF schedulable.

In order to make the task set EDF schedulable, we need to change the deadline of T4 to make sure that it would be executed before T5.
In this case, U ≤ 1 is no longer valid thus we need to us processor utilization demand analysis. We fix the new deadline of T4 to 40.

D = { 10, 20, 40, 50, 75, 90, 100, 250, 1000 }

$$C^{tot}(0,L) = \sum_{i=1}^{n} N_i C_i = \sum_{i=1}^{n} \left(\left\lfloor \frac{L - D_i}{T_i} \right\rfloor + 1 \right) C_i \leq L$$

$C^{tot}(0,20)$ = (((20 – 10)/10)+1)*2 + (((20-75)/75)+1)*5 + (((20-250)/250)+1)*4 + (((20-40)/50)+1)*4 + (((20-50)/50)+1)*4 = 4 ≤ 20

$C^{tot}(0,40)$ = (((40 – 10)/10)+1)*2 + (((40-75)/75)+1)*5 + (((40-250)/250)+1)*4 + (((40-40)/50)+1)*4 + (((40-50)/50)+1)*4 = 9 ≤ 40

$C^{tot}(0,50) = 10 \leq 50$

$C^{tot}(0,75) = 14 \leq 75$

$C^{tot}(0,90) = 26 \leq 90$

$C^{tot}(0,100) = 36 \leq 100$

$C^{tot}(0,250) = 105 \leq 250$

$C^{tot}(0,1000) = 441 \leq 1000$

Assignment 3: Resource Sharing and Communication

1. Shared resources

Brake actuator value is the common actuator which is used by both of the systems. It would initiates the braking during the execution of T2 and T4.
There are also other shared resources but for the analysis we're taking care just for the most important ones that are shared between the two nodes.

Task	WCET	Deadline	Period	Shared Resource	Critical section
T1	2	10	10	None	0
T2	5	75	75	Brake Actuator value	2
T3	4	250	250	None	0
T4	4	50	50	Brake Actuator value	2
T5	4	50	50	None	0

Figure 3.1: Shared resources and length of critical sections

Motivation: We have chosen the critical section of 2 because the braking the complete execution of the task and it has to be less than the worst case execution time of the task.

Firstly, we need to find the blocking time which is defined in PCP as:
$B_i = max(cs_i,\ldots, cs_k)$ where $cs_i \ldots cs_k$ are the critical sections of all lower priority tasks that can block task τ_i

We are calculating the blocking time for T2 and T4 and S1 is the semaphore:

ceil(S1) = max[prio(T2), prio(T4)] = prio(T4) = 4

$B_{T2} = 0$ because there is no lower priority task

$B_{T4} = 2$ because T2 uses S1 and prio(B_{T4}) > 4 is not true so T2 can block T4

$R_{T1} = C_{T1} + B_{T1} = 2$

$R^0_{T4} = C_{T4} + B_{T4} = 4 + 2 = 6$
$R^1_{T4} = C_{T4} + B_{T4} + (R^0_{T4}/T_{T1})*C_{T1} = 4 + 2 + (4/10)*2 = 8$

$R^2_{T4} = C_{T4} + B_{T4} + (R^1_{T4}/T_{T1})*C_{T1} = 4 + 2 + (6/10)*2 = 8$

$R^0_{T5} = C_{T5} + B_{T5} = 4$
$R^1_{T5} = C_{T5} + B_{T5} + (R^0_{T5}/T_{T1})*C_{T1} + (R^0_{T5}/T_{T4})*C_{T4} = 4 + 0 + (4/10)*2 + (4/50)*4 = 10$
$R^2_{T5} = C_{T5} + B_{T5} + (R^1_{T5}/T_{T1})*C_{T1} + (R^1_{T5}/T_{T4})*C_{T4} = 4 + 0 + (10/10)*2 + (10/50)*4 = 10$

$R^0_{T2} = C_{T2} + B_{T2} = 5$
$R^1_{T2} = C_{T2} + B_{T2} + (R^0_{T2}/T_{T1})*C_{T1} + (R^0_{T2}/T_{T4})*C_{T4} + (R^0_{T2}/T_{T5})*C_{T5} = 15$
$R^2_{T2} = C_{T2} + B_{T2} + (R^1_{T2}/T_{T1})*C_{T1} + (R^1_{T2}/T_{T4})*C_{T4} + (R^1_{T2}/T_{T5})*C_{T5} = 17$
$R^3_{T2} = C_{T2} + B_{T2} + (R^2_{T2}/T_{T1})*C_{T1} + (R^2_{T2}/T_{T4})*C_{T4} + (R^2_{T2}/T_{T5})*C_{T5} = 17$

$R^0_{T3} = C_{T3} + B_{T3} = 4$
$R^1_{T3} = C_{T3} + B_{T3} + (R^0_{T3}/T_{T1})*C_{T1} + (R^0_{T3}/T_{T4})*C_{T4} + (R^0_{T3}/T_{T5})*C_{T5} + (R^0_{T3}/T_{T2})*C_{T2} = 18$
$R^2_{T3} = C_{T3} + B_{T3} + (R^1_{T3}/T_{T1})*C_{T1} + (R^1_{T3}/T_{T4})*C_{T4} + (R^1_{T3}/T_{T5})*C_{T5} + (R^1_{T3}/T_{T2})*C_{T2} = 20$
$R^3_{T3} = C_{T3} + B_{T3} + (R^2_{T3}/T_{T1})*C_{T1} + (R^2_{T3}/T_{T4})*C_{T4} + (R^2_{T3}/T_{T5})*C_{T5} + (R^2_{T3}/T_{T2})*C_{T2} = 20$

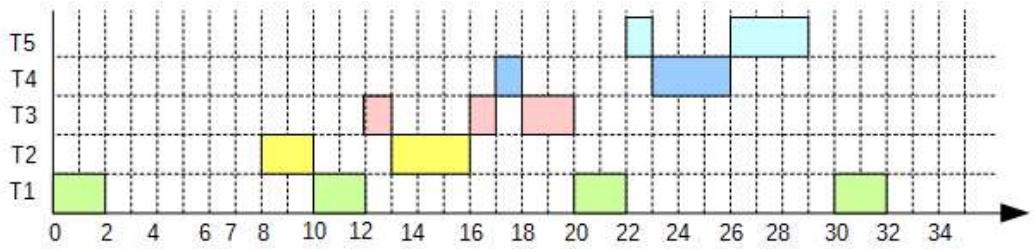

Figure 3.2: RTA Schedule with blocking ti1me

2. Communication

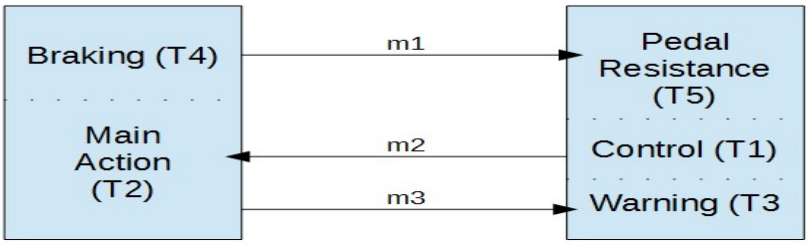

Figure 3.3: Communication between different nodes

Task	WCET	Deadline	Period	Sends Message To
T1	2	10	10	T2
T2	5	75	75	T3
T3	4	250	250	None
T4	4	50	50	T5

T5	4	50	50	None

Figure 3.4: Communication between different nodes

Task	T	D	C	Message	Priority
T4	50	50	4	m1	000
T2	75	75	5	m3	010

Figure 3.5: Node 1

Task	T	D	C	Message	Priority
T1	10	10	2	m2	001
T5	50	50	4	-	
T3	250	250	4	-	

Figure 3.6: Node 2

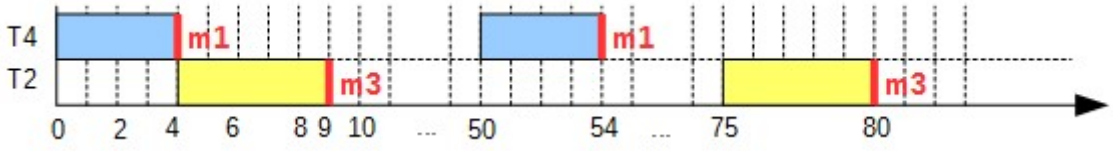

Figure 3.7: Jitter representation for Node 1

$J_{m1} = R_{max} - R_{min} = 4 - 4 = 0$ No Jitter
$J_{m3} = R_{max} - R_{min} = 9 - 5 = 4$ ms Jitter

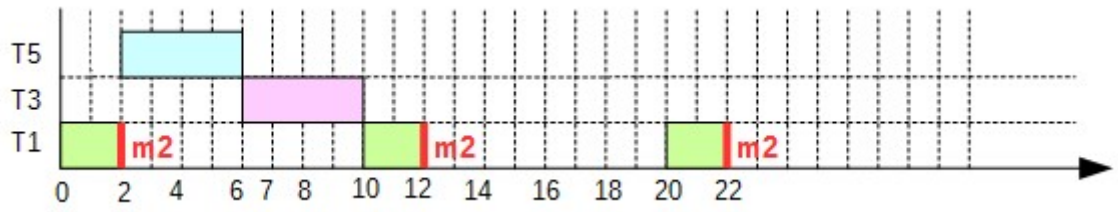

Figure 3.8: Jitter representation for Node 2

$J_{m2} = R_{max} - R_{min} = 2 - 2 = 0$ No Jitter

3. Holistic Schedulability Analysis

FORMULE

Calculation of the transaction T4 → m1 → T5:

We start by calculating response time of Braking (T4):
$w^0_{T4} = C_{T4} = 4$ ms
$w^1_{T4} = C_{T4} + B_{T4} = 6$ ms
$R_{T4} = J_{m1} + w^1_{T4} = 6$ ms that is less than the deadline 50 ms

We continue by calculating response time for m1:
$w_{m1} = B_{m1} = 0.135$ ms

Jitter is:
$J_{m1} = R^{max}_{T4} - R^{min}_{T4} = 6 - 4 = 2$ ms

Response time is:
$R_{m1} = J_{m1} + w_{m1} + C_{m1} = 2 + 0.135 + 0.135 = 2.27$ ms

Pedal resistance (T5) is the medium priority task, so it is delayed by Control task (T1) and blocked by Warning (T3).
$w^0_{T5} = C_{T5} = 4$
$w^1_{T5} = C_{T5} + B_{T3} + ((w^0_{T5} + J_{m2} + \tau_{bit}) / T_{T1})*C_{T1} = 4 + 0 + 2 = 6$
$w^2_{T5} = C_{T5} + B_{T3} + ((w^1_{T5} + J_{m2} + \tau_{bit}) / T_{T1})*C_{T1} = 4 + 0 + 2 = 6$

The best response time is when we received an empty message: $47* \tau_{bit} = 0.047$ ms
$J_{T5} = R^{max}_{m1} - R^{min}_{m1} = 2.27 – 0.047 = 2.223$ ms

Response time is:
$R_{T5} = J_{T5} + w_{T5} = 2.223 + 6 = 8.223$ ms that is less than the deadline 50 ms,

Time to execute the transaction:
$R_{T4} + R_{m1} + R_{T5} = 6 + 2.27 + 8.223 =$ 16.493 ms

Calculation of the transaction T1 → m2 → T2 → m3 → T3:

Verdict: The task set <is/is not> schedulable.
<If the task set is not schedulable, describe what can you do to make it schedulable>

4. Dependability

<Describe the overall dependability concerns in your systems>
Threats: <Describe the identified threats to your system.>
Task criticality: <Motivate the tasks criticalities you have identified (you can decide your own scale from non-critical to highly critical).>

Task	Task name	WCET	Period	Deadline	Criticality
T1	ComputePosition				<High>

Figure 9: Task criticalities

Time and space redundancy: <Discuss the possibilities to use time or space redundancy, and the resulting consequences.>