**Modeling discrete Event using Cadmium (2019 Fall)**

**Assignment 1:**

**Water Level Indicator**

Gurkaran Singh Kahlon, M.Eng

Carleton University, 101133588

**Description of the Model – Water Level Indicator**

The **Water Level Indicator** measures the level of water in tanks, wells. It consists of Four atomic model – ADC(Analog to Digital Convertor), Color Decider, Display Panel and Buzzer.

ADC and Color Decider form a coupled model called ‘ADC-Decide’. Following is graph representing relationship among each sub model.

WATER LEVEL INDICATOR

ADC-Decide

CDIn1

DigOut

SENSOR

greenOut

Display Panel

DPIn1

CDOut

Color Decider

CDIn2

LevelOut

ADC

SIn

yellowOut

DPIn2

ToBuzz

L\_Out

. redOut

DPLevel

buzzOut

buzzIn

Buzzer

OUT

**Model Description**

Sensor: There is a sensor S which reads level of water in the tank, values can be in between 0 to 11. When water level hits 3 or below, it outputs a red LED and a buzzer sound, which means water need to be filled into tank. Similarly, when water level reaches 10 or above, a buzzer sound and red LED shows water overflow.

ADC-Decide: This coupled model consists of ADC(Analog to Digital Convertor) and color decider,

ADC takes sensor reading as input and produces 3 outputs, out of these 2 outputs, DigOut and LevelOut acts as input to Color Decider.

Based on the value of Sin input to ADC we compute DigOut as

a) If SIn is between 0 and 3, then DigOut =0;

b) If SIn is between 3 and 7, then DigOut =1;

c) If SIn is between 7 and 10, then DigOut =2;

d) If SIn is between 10 and 11, then DigOut =3;

In Color Decider – The Decision of color of LED to be Output is decided in this model. It takes CDIn1 as input to the color decides and gives CDOut which is a value from 1 to 3 range.

Following logic is implemented

If(DigOut = 0 or DigOut = 3), CDOut = 1

If(DigOut = 1 ), CDOut = 2

If(DigOut = 2 ), CDOut = 3

The other output L\_Out is send to Display Panel

Display Panel : It shows Digital Water level value and the corresponding LED. Input DPIn1 produces 3 LED output( greenOut, YellowOut and RedOut). LED is red at two states when water is below 3 or above 10(overflow). Between 3 – 7 , LED color is Yellow and for 7-10 color is green.

Second Output is DPLevel ,this is Digital Output value that a user can see. It’s from range 0 to 11.

Buzzer: Alerts the person if water level is below 3 or above 11 to avoid overflow. As we can see buzzer is connected to ADC, Buzzer is connected to ADC. It works as follow:

When BuzzIn =0 or BuzzIn = 3, Output BuzzOut =1

**DEVS Formal Specification:**

The formal specifications <S, X, Y, δint, δext, λ, ta> and the state diagrams for both the atomic and coupled models are given as follows:

1. **ADC - (Atomic Model)**

DigOut

ADC

LevelOut

SIn

ToBuzz

S = {passive, active}

X = {SIn}

Y = {DigOut, levelOut, ToBuzz}

δint (active) = passive

δext (SIn, passive) = active

Level = SIn

λ(active)

{If 0 <= Level <= 3, send 0 to DigOut, ToBuzz

If 4 <= Level <=7, send 1 to DigOut

If 8 <= Level <= 10, send 2 to DigOut

If 10 <= Level <= 11, send 3 to DigOut, ToBuzz

}

ta(passive) = INFINITY

ta(active) = preparationTime

1. **COLOR DECIDER - (Atomic Model)**

COLOR DECIDER

CDIn1

CDOut

L\_Out

CDIn2

S = {passive, active}

X = {CDIn1, CDIn2}

Y = {CDOut, L\_Out}

δint (active) = passive

δext (CDIn, passive) = active

Disp = CDIn1

Clevel = CDIn2

λ(active)

{If (Disp == 0) || (Disp == 3), send 1 to CDOut,

ATM

If (Disp == 1), send 2 to CDOut,

If (Disp ==2), send 3 to CDOut

}

ta(passive) = INFINITY

ta(active) = decisionTime

1. **DISPLAY PANEL – (Atomic Model)**

DISPLAY PANEL

greenOut

yellowOut

DPIn1

redOut

DPIn2

DPLevel

S = {passive, active}

X = {DPIn1, DPIn2}

Y = {greenOut, yellowOut, redOut, DPLevel}

δint (active) = passive

δext (DPIn, passive) = active

Color = DPIn1

D\_level = 0

λ(active)

{If Color == 1, send 1 to redOut,

If Color == 2, send 1 to greenOut,

If Color == 3, send 1 to yellowOut,

}

ta(passive) = INFINITY

ta(active) = displayTime

1. **BUZZER – (Atomic Model)**

BUZZER

buzzOut

buzzIn

S = {passive, active}

X = {buzzIn}

Y = {buzzOut}

δint (active) = passive

δext (buzzIn, passive) = {

active

B = buzzIn

}

λ(active)

{If B == 0, send 1 to buzzOut,

If B == 3, send 1 to buzzOut

}

ta(passive) = INFINITY

ta(active) = buzzTime

**Coupled TOP Model:**

WATER LEVEL INDICATOR: The coupled model represents the input and output generated from the entire model.

WATER LEVEL INDICATOR

greenOut

yellowOut

SIn

redOut

DPLevel

buzzOut

Input Ports: SIn

Output Ports: greenOut, yellowOut, redOut, DPLevel, buzzOut

WATER LEVEL INDICATOR Formal specifications:

WATER LEVEL INDICATOR = <X, Y, D, EIC, EOC, IC, SELECT>

X = {SIn}

Y = {greenOut, yellowOut, redOut, DPLevel, buzzOut}

D={ADC-Decide, Buzzer, DisplayPanel}

EIC = {WLI.in, ADC.SIn}

EOC = {(DisplayPanel.greenOut, DisplayPanel.yellowOut, DisplayPanel.redOut, DisplayPanel.DPLevel,Buzzer.buzzOut)}

IC = {(ADC-Decide.CDOut, DisplayPanel.DPIn1),

(ADC-Decide.L-Out, DisplayPanel.DPIn2),

(ADC-Decide.ToBuzz,Buzzer.buzzIn) }

SELECT: {ADC-Decide, WLI} = WLI

{Buzzer, WLI} = WLI

{DisplayPanel,WLI}=WLI

**Testing Strategy**

The following describes the inputs and outputs for the test on each model, to evaluate the performance of the atomic models and the Water Level Indicator, in general.

1. **ADC (Analog to Digital Converter)**

**Input**

It consists of Time and input respectively in adc\_input\_test.txt file.

00:00:30 2

00:00:40 0

00:00:45 3

00:00:52 4

…..

**Output is stored in file adc\_test\_output\_messages.txt**

00:00:00:000

State for model input\_reader is next time: 00:00:00:000

State for model adc1 is level 0 digout 0

00:00:00:000

State for model input\_reader is next time: 00:00:30:000

State for model adc1 is level 0 digout 0

00:00:30:000

State for model input\_reader is next time: 00:00:10:000

State for model adc1 is level 2 digout 0

00:00:40:000

State for model input\_reader is next time: 00:00:05:000

State for model adc1 is level 0 digout 0

00:00:45:000

State for model input\_reader is next time: 00:00:07:000

State for model adc1 is level 3 digout 0

00:00:52:000

State for model input\_reader is next time: 00:00:33:000

State for model adc1 is level 4 digout 1

……

The 2 outputs (levelout, digout) are given for each input value in the model, the “tobuzz” output produces just two values (0 &3), for low and high-water level alerts respectively.

1. **COLOR DECIDER**

**Input**

For testing this Color Decider model, we take two input reader because there are two input ports of color reader CDIn1, CDIn2. These are defined in files - colordecider\_input\_1\_test.txt and colordecider\_input\_2\_test.txt.

**Input for CDIn1** from - colordecider\_input\_1\_test.txt

00:00:10 01

00:00:30 02

00:00:45 03

00:00:52 04

…..

**Input for CDIn2** from - colordecider\_input\_2\_test.txt

00:00:10 01

00:00:30 02

00:00:45 03

00:00:52 04

…..

**Output is stored in file displaypanel\_test\_output\_messages.txt**

00:00:00:000

State for model input\_reader\_1 is next time: 00:00:00:000

State for model input\_reader\_2 is next time: 00:00:00:000

State for model colordecider1 is CDOut: 0 L\_Out: 0

00:00:00:000

State for model input\_reader\_1 is next time: 00:00:10:000

State for model input\_reader\_2 is next time: 00:00:10:000

State for model colordecider1 is CDOut: 0 L\_Out: 0

00:00:10:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model colordecider1 is CDOut: 2 L\_Out: 1

00:00:15:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model colordecider1 is CDOut: 2 L\_Out: 1

00:00:20:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model colordecider1 is CDOut: 2 L\_Out: 1

00:00:25:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model colordecider1 is CDOut: 2 L\_Out: 1

00:00:30:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model colordecider1 is CDOut: 3 L\_Out: 2

00:00:35:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model colordecider1 is CDOut: 3 L\_Out: 2

00:00:40:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model colordecider1 is CDOut: 3 L\_Out: 2

00:00:45:000

State for model input\_reader\_1 is next time: 00:00:07:000

State for model input\_reader\_2 is next time: 00:00:07:000

State for model colordecider1 is CDOut: 1 L\_Out: 3

00:00:50:000

State for model input\_reader\_1 is next time: 00:00:07:000

State for model input\_reader\_2 is next time: 00:00:07:000

State for model colordecider1 is CDOut: 1 L\_Out: 3

00:00:52:000

State for model input\_reader\_1 is next time: 00:00:33:000

State for model input\_reader\_2 is next time: 00:00:33:000

State for model colordecider1 is CDOut: 0 L\_Out: 4

………

CDIn1 and CDIn2 both produce their respective values of CDOut and L\_Out for each values of their inputs. CDOut only ranges from 1 to 3, indicating the 3 different LED colors, and the L\_Out produces the display level values, from 0 to 11.

1. **DISPLAY PANEL**

For testing this Display Panel model, we take two input reader because there are two input ports of color reader DPIn1, DPIn2 from files displaypanel\_input\_1\_test.txt, displaypanel\_input\_2\_test.txt.

Input for DPIn1 from displaypanel\_input\_1\_test.txt.

00:00:10 1

00:00:30 2

00:00:45 3

00:00:52 4

….

Input for DPIn2 from displaypanel\_input\_2\_test.txt.

00:00:10 1

00:00:30 2

00:00:45 3

00:00:52 4

….

**Output is stored in displaypanel\_test\_output\_messages.txt**

00:00:00:000

State for model input\_reader\_1 is next time: 00:00:00:000

State for model input\_reader\_2 is next time: 00:00:00:000

State for model displaypanel1 is Color value: 0 Red Value: 0 Green value: 0 Yellow value: 0 Level of water: 0

00:00:00:000

State for model input\_reader\_1 is next time: 00:00:10:000

State for model input\_reader\_2 is next time: 00:00:10:000

State for model displaypanel1 is Color value: 0 Red Value: 0 Green value: 0 Yellow value: 0 Level of water: 0

00:00:10:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model displaypanel1 is Color value: 1 Red Value: 1 Green value: 0 Yellow value: 0 Level of water: 1

00:00:15:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model displaypanel1 is Color value: 1 Red Value: 1 Green value: 0 Yellow value: 0 Level of water: 1

00:00:20:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model displaypanel1 is Color value: 1 Red Value: 1 Green value: 0 Yellow value: 0 Level of water: 1

00:00:25:000

State for model input\_reader\_1 is next time: 00:00:20:000

State for model input\_reader\_2 is next time: 00:00:20:000

State for model displaypanel1 is Color value: 1 Red Value: 1 Green value: 0 Yellow value: 0 Level of water: 1

00:00:30:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model displaypanel1 is Color value: 2 Red Value: 0 Green value: 1 Yellow value: 0 Level of water: 2

00:00:35:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model displaypanel1 is Color value: 2 Red Value: 0 Green value: 1 Yellow value: 0 Level of water: 2

00:00:40:000

State for model input\_reader\_1 is next time: 00:00:15:000

State for model input\_reader\_2 is next time: 00:00:15:000

State for model displaypanel1 is Color value: 2 Red Value: 0 Green value: 1 Yellow value: 0 Level of water: 2

00:00:45:000

State for model input\_reader\_1 is next time: 00:00:07:000

State for model input\_reader\_2 is next time: 00:00:07:000

State for model displaypanel1 is Color value: 3 Red Value: 0 Green value: 0 Yellow value: 1 Level of water: 3

00:00:50:000

State for model input\_reader\_1 is next time: 00:00:07:000

State for model input\_reader\_2 is next time: 00:00:07:000

State for model displaypanel1 is Color value: 3 Red Value: 0 Green value: 0 Yellow value: 1 Level of water: 3

00:00:52:000

State for model input\_reader\_1 is next time: 00:00:33:000

State for model input\_reader\_2 is next time: 00:00:33:000

State for model displaypanel1 is Color value: 4 Red Value: 0 Green value: 0 Yellow value: 1 Level of water: 4

The two inputs taken into the display panel now produces final outputs to the user; the respective LED outputs to each display level value and the actual level of water in the tank.

1. **BUZZER**

**Input** is taken from buzzer\_input\_test.txt

00:00:30 3

00:00:40 3

00:00:45 3

00:00:50 4

**…..**

**Output** is stored in buzzer\_test\_output\_messages.txt

00:00:00:000

State for model input\_reader is next time: 00:00:00:000

State for model buzzer1 is buzzOut: 0

00:00:00:000

State for model input\_reader is next time: 00:00:30:000

State for model buzzer1 is buzzOut: 0

00:00:30:000

State for model input\_reader is next time: 00:00:10:000

State for model buzzer1 is buzzOut: 1

00:00:40:000

State for model input\_reader is next time: 00:00:05:000

State for model buzzer1 is buzzOut: 1

00:00:45:000

State for model input\_reader is next time: 00:00:07:000

State for model buzzer1 is buzzOut: 1

00:00:52:000

State for model input\_reader is next time: 00:00:33:000

State for model buzzer1 is buzzOut: 0

The buzzer Output (buzzout=1) is given at two states; when the water level (buzzIn) is between 0 &3; indicating low water level and when the water level is between 10 & 11; indicating high water level. Thereby alerting the user at both states, the buzzer sounds for 10 seconds.

**WATER LEVEL INDICATOR COUPLED TOP MODEL**

**Input** is obtained from wli\_input\_test.txt

00:00:30 2

00:00:40 0

00:00:45 3

00:00:52 4

…...

**Output is stored in wli\_test\_output\_messages.txt**

00:00:00:000

00:00:00:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<int>::out: {}] generated by model input\_reader

00:00:30:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<int>::out: {2}] generated by model input\_reader

00:00:40:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<int>::out: {0}] generated by model input\_reader

[adc\_defs::DigOut: {0}, adc\_defs::LevelOut: {2}, adc\_defs::ToBuzz: {0}] generated by model adc1

00:00:45:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<int>::out: {3}] generated by model input\_reader

[colordecider\_defs::CDOut: {1}, colordecider\_defs::L\_Out: {2}] generated by model colordecider1

00:00:50:000

[colordecider\_defs::CDOut: {1}, colordecider\_defs::L\_Out: {2}] generated by model colordecider1

[displaypanel\_defs::greenOut: {}, displaypanel\_defs::yellowOut: {}, displaypanel\_defs::redOut: {1}, displaypanel\_defs::DPLevel: {2}] generated by model displaypanel1

[buzzer\_defs::buzzOut: {1}] generated by model buzzer1

00:00:52:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<int>::out: {4}] generated by model input\_reader

00:00:55:000

[colordecider\_defs::CDOut: {1}, colordecider\_defs::L\_Out: {2}] generated by model colordecider1

[displaypanel\_defs::greenOut: {}, displaypanel\_defs::yellowOut: {}, displaypanel\_defs::redOut: {1}, displaypanel\_defs::DPLevel: {2}] generated by model displaypanel1

……

The behavior of the water level indicator is described in the corresponding range of outputs produced at the different inputs taken into the model.

As we can see that at time 30sec , Input to top model is 2 is given,

1)So in 40 sec we can see state of buzzer = 0 (which as we described for 0<Sin<3 , then DigOut =0; )

2)Output from Display panel is coming as redOut is coming 1 which is expected after certain delay at 50 sec.