

PORTFOLIO

KAMPANATH KAPPAGO



TEST PROGRAM

Automated Optical Test (LCD panel product, Rosemount)

Test Program Porting From VB6 To Labview (Ultrasonic Liquid Sensor, Mobrey)

Automated Optical Test (Face Cleaner Product, Clarisonic)

Test Program For Handheld Medical Device

Automated Optical Inspection (Medical Module, Trimble)

Automated Optical Inspection (SSD, Seagate)

Test Automated Improvement(Network Module, Emerson)

Test Program For Semiconductor Device (MCU, ICs, Microchip, UTAC, Toshiba)

Water Temperature Control

GAME DEVELOPMENT

Marble Ball Racing 2022 (PC - Steam, Android Phone)

Marble Ball Racing (PC - Steam, Android Phone, Apple iPhone)

Space Rush (Android Phone, Apple iPhone)

Ball Sort Puzzle (Android Phone, Apple iPhone)

Tic Tac Toe (Android Phone)

WEB APPLICATION

Web App For Restaurant

Web App For Support Test System (Show And Edit Test Parameter)

Web App For Online Shopping

Web App For Online Food

EMBEDDED SYSTEM

Survey Robot (Senior Project)

Simple Web Interfacing To Embedded Device

Basic Image Processing With Raspberry Pi And OpenCV (Object Detection)

TEST PROGRAM

Test software is used in electronic factory, to test electrical function of finish assemble of electronic PCBA board or ICs. This software will provide GUI for user, can communicate with test equipment like oscilloscope, digital multi meter, send log of test result to local drive and database, some need to have image processing function to inspect product appearance.

Software Tools And Programming Language

Labview
NI Vision Development
NI Vision Builder
Visual Basic
C#
AutoIt
Arduino, C/C++
TestStand

Example of My Project

- 1.Vision Inspection Test Program (LCD panel product)
- 2.Test Program Porting From VB6 To Labview (Ultrasonic Liquid Sensor)
- 3.Test Program For Handheld Medical Device
- 4.Automated Optical Inspection For Trimble Product (Medical Module)
5. Water Temperature Control

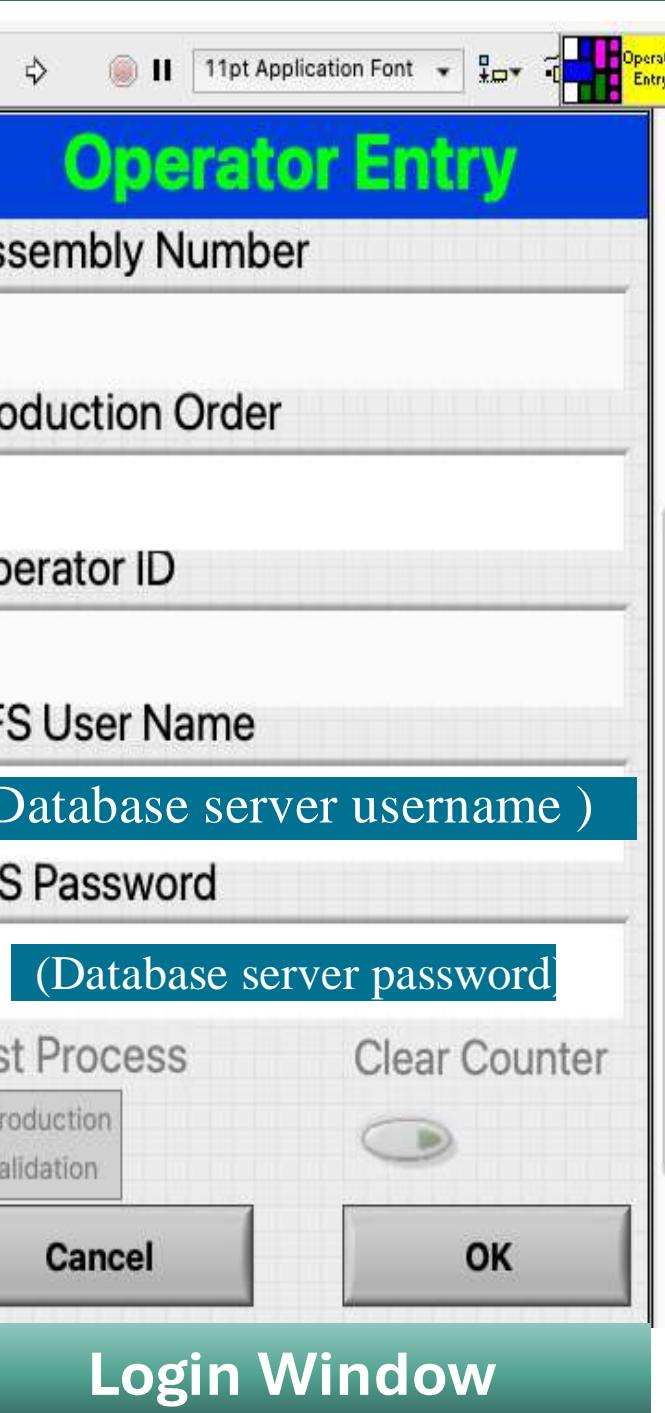
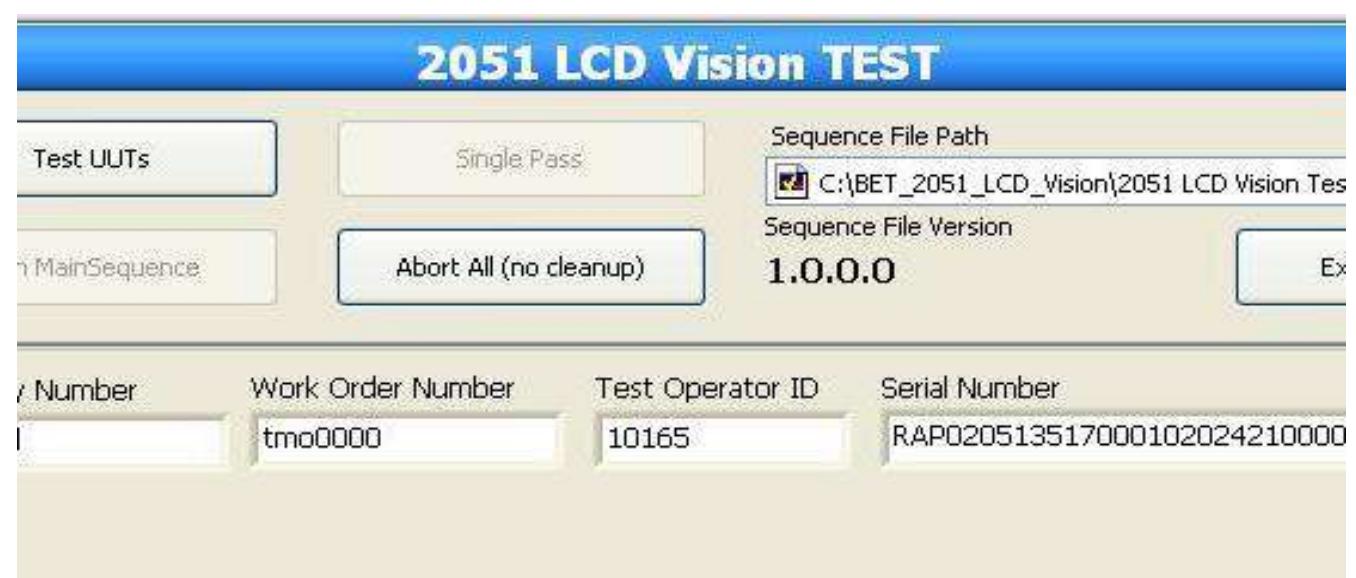
Automated Optical Test (LCD panel product, Rosemount)

Software Tools and Programming Language : Labview, NI vision development, NI Vision Builder and TestStand

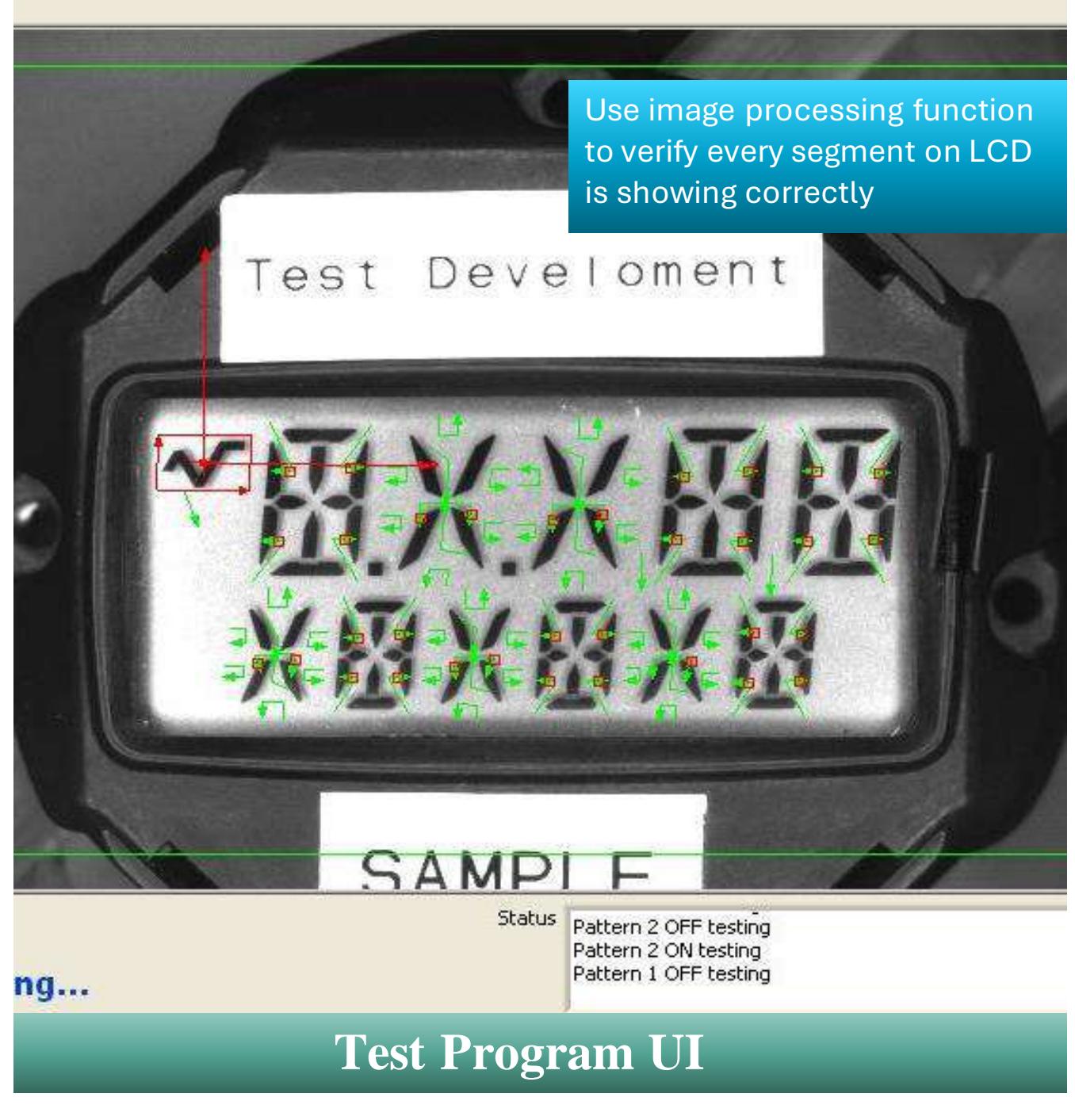
- Measure voltage and current with DMM via GPIB (NI-488)
- Communicate to unit via I2C (USB-8451) to drive LCD to show specific message
- General IO with USB-6501 (turn ON/OFF light source for camera, detect start test button)
- Acquire unit image from camera
- Verify message on LCD with NI vision to ensure there is no defect on LCD's segment
- Automatic read 2D barcode on unit
- Save result to database server



LCD Panel Product

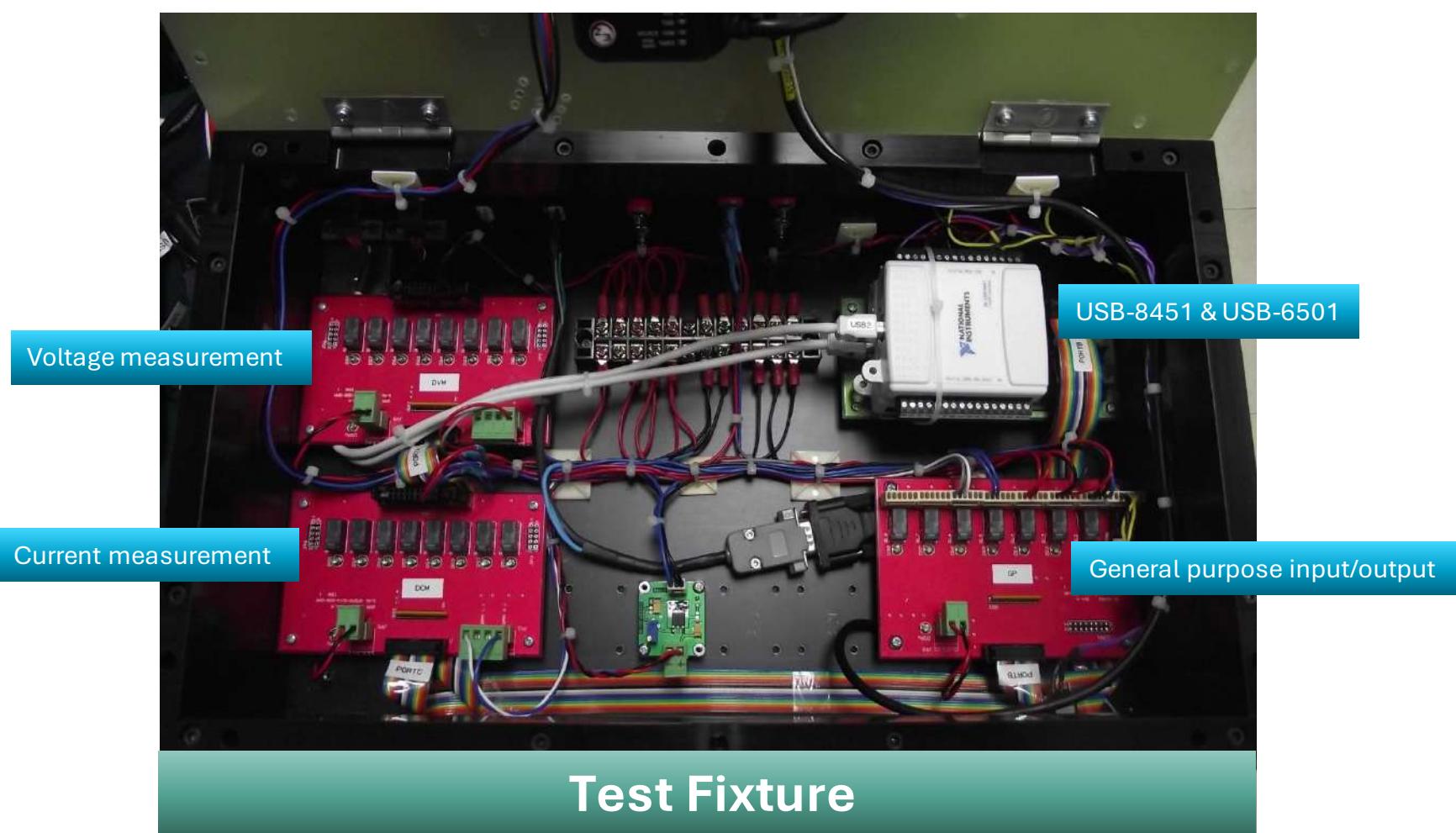


Login Window

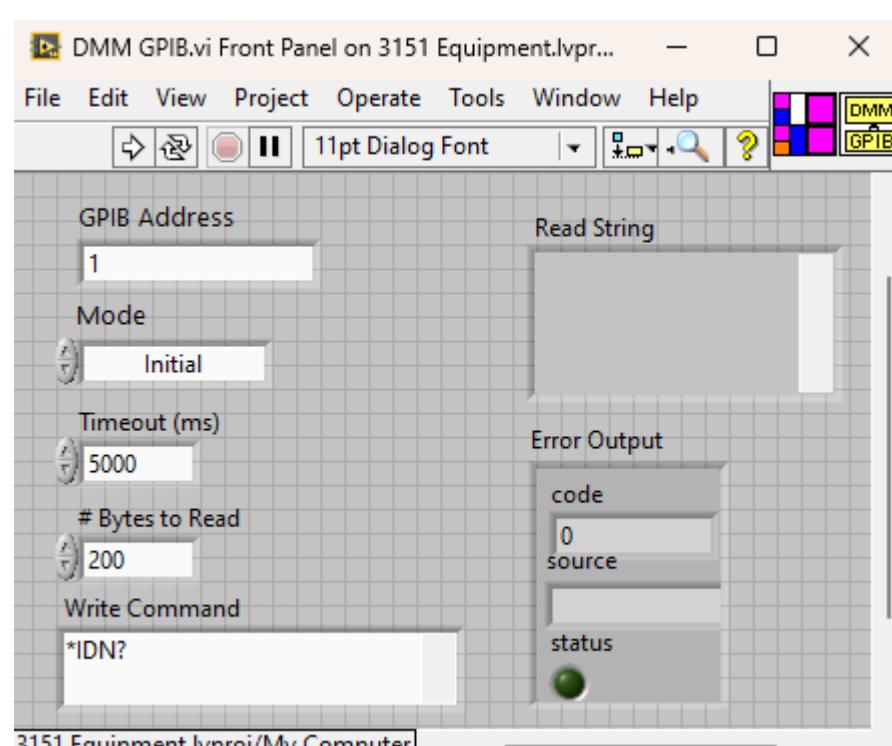
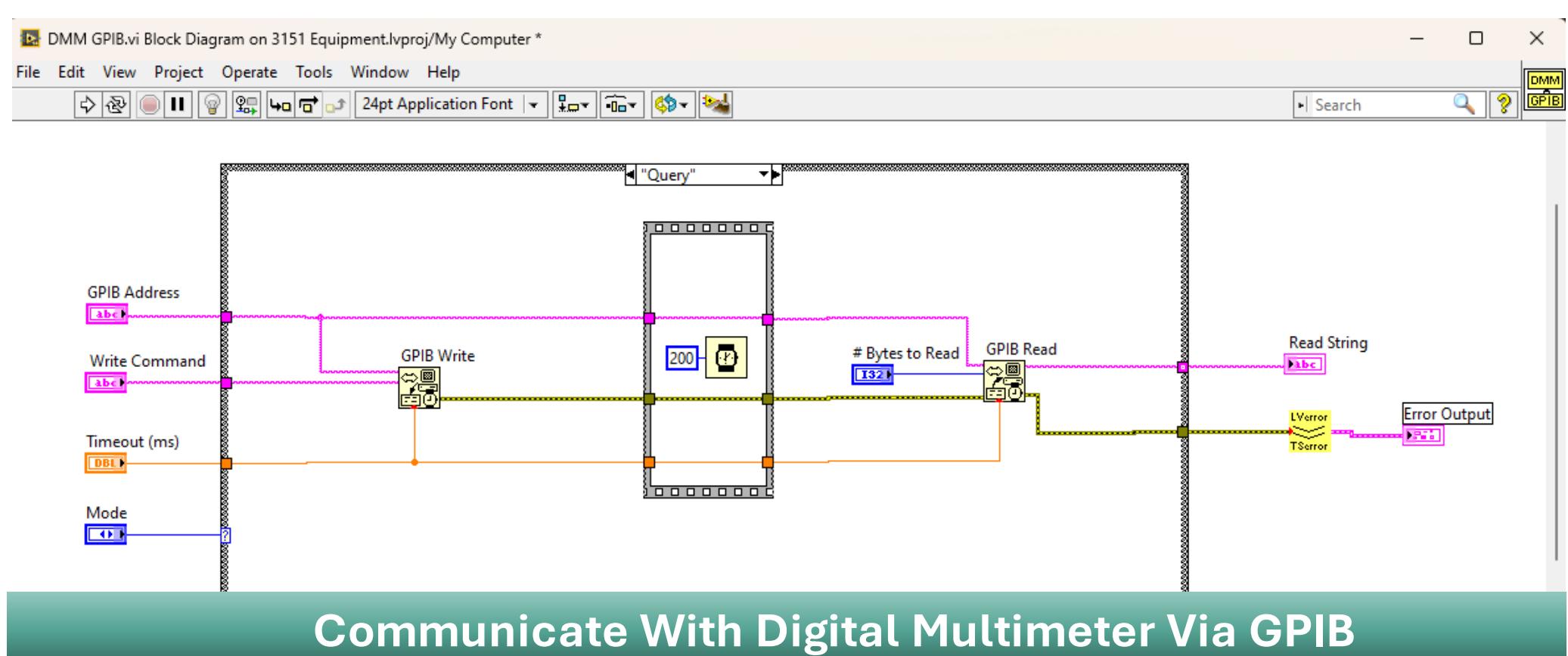
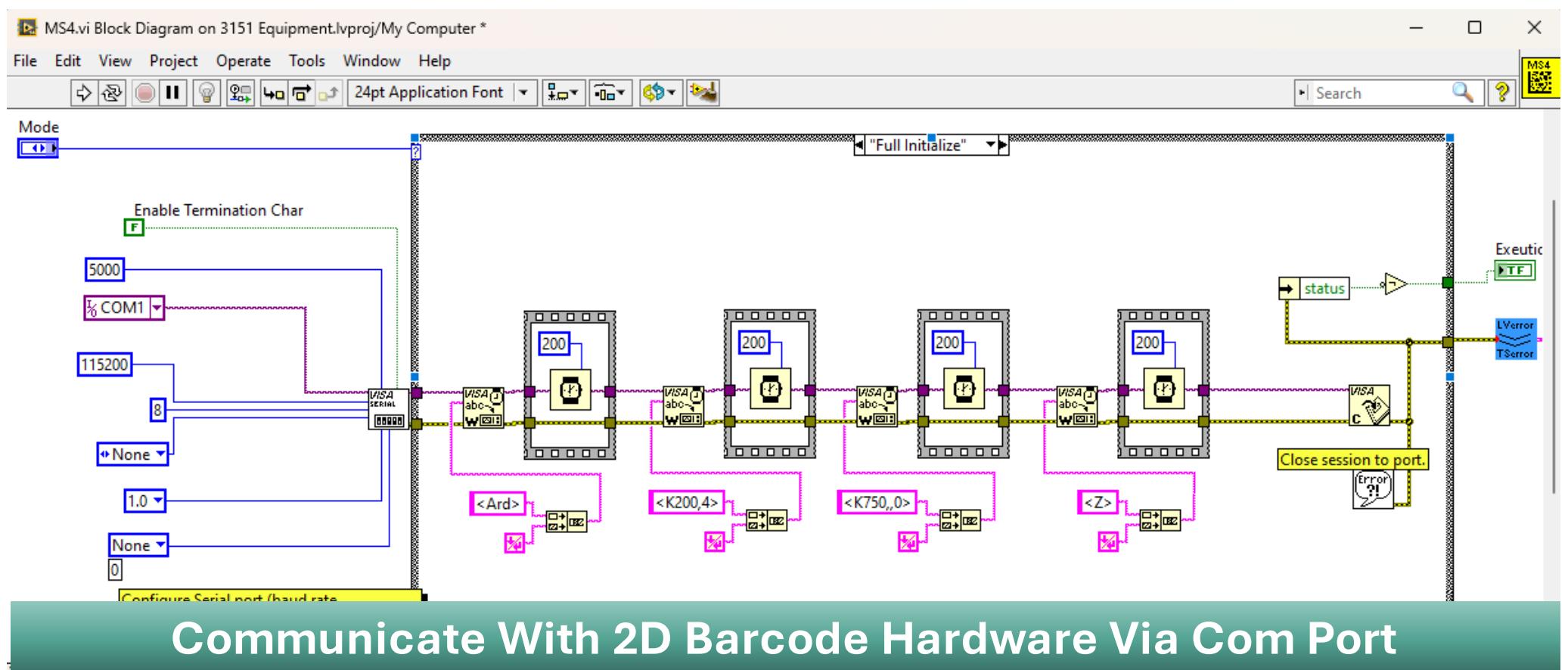


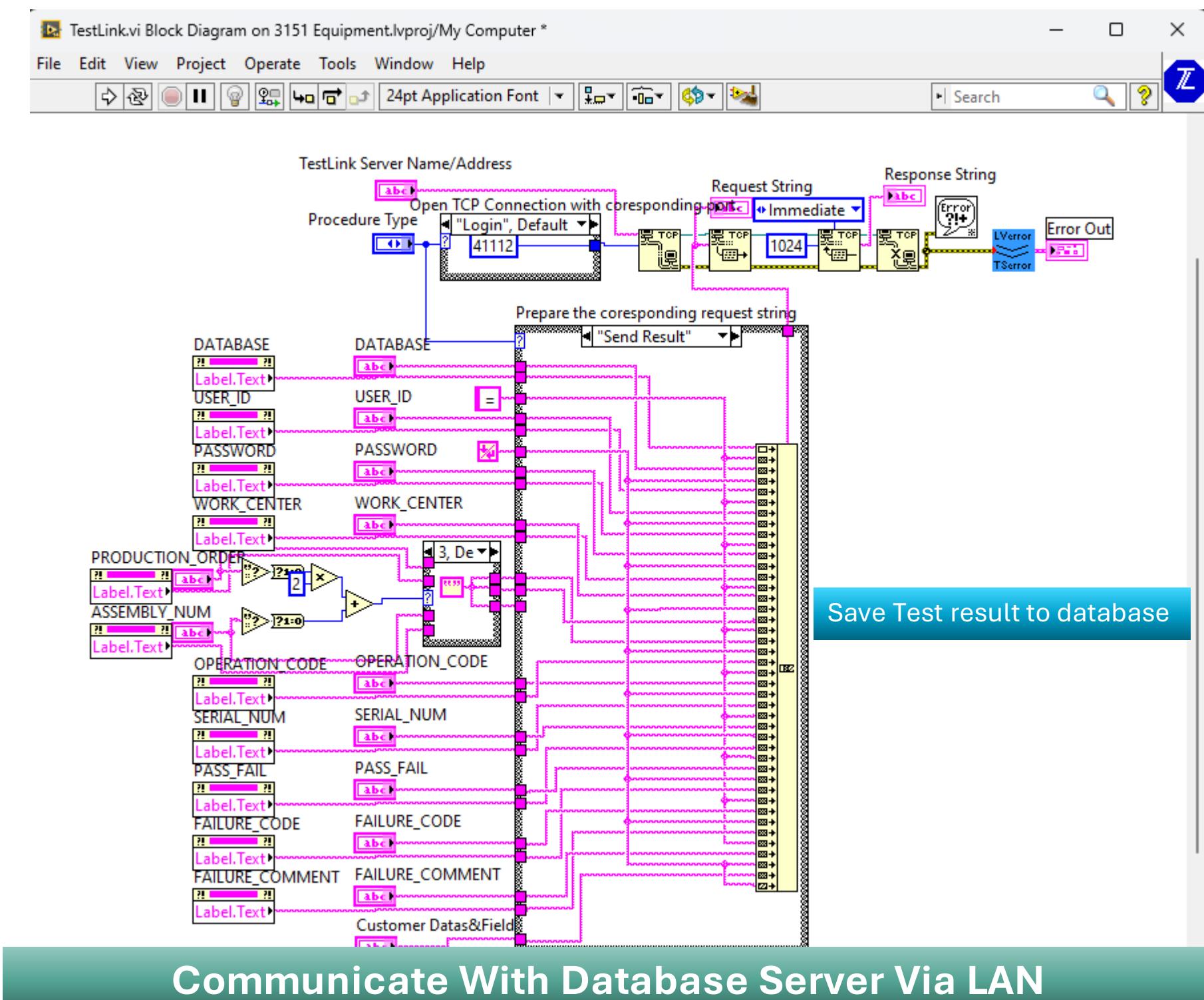


Test Station

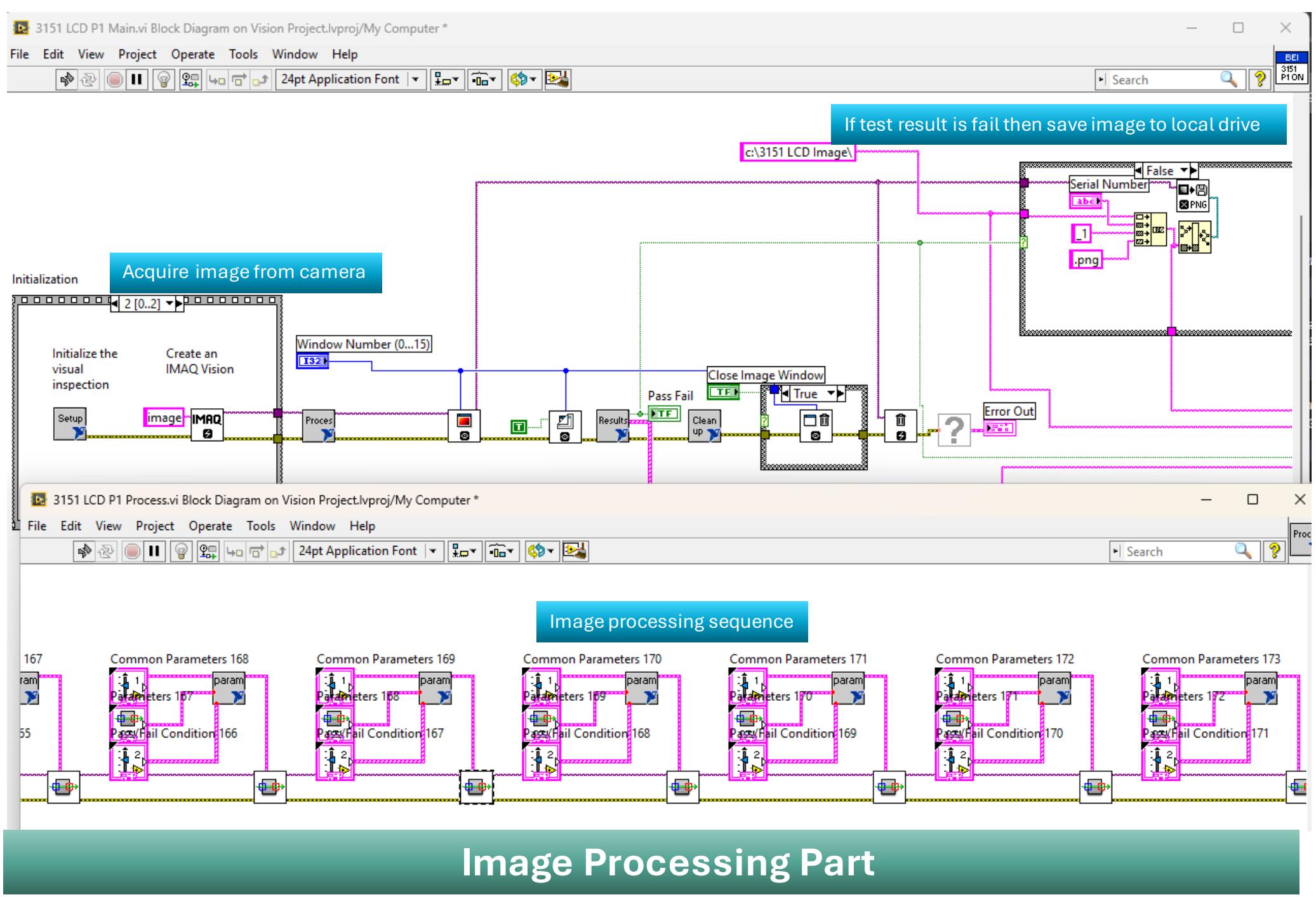


Test Fixture





Communicate With Database Server Via LAN



Test Plan



BEI-Thailand Form
Title: 8800-7611 LCD Test Plan



9.11 Test Software Procedure

9.11.1 Check UUT present

9.11.1.1 Turn light source on by turn GP-RL1 on.

9.11.1.2 Measure contrast on UUT LCD display by camera.

9.11.2 Check serial number

9.11.2.1 Read serial number by 2D scanner and verify the character number.

9.11.3 Query PFS

9.11.3.1 Query PFS with current serial number.

9.11.4 Check PSU 3.3V

9.11.4.1 Set DMM to voltmeter.

9.11.4.2 Connect DMM to PSU3.1V by turn DVM-RL1 on.

9.11.4.3 Measure the voltage must be in limit.

9.11.4.4 Remove DMM from PSU3.1V by turn DVM-RL1 off.

9.11.4.5 Set DMM to Ammeter.

9.11.4.6 Apply PSU3.1V to UUT by turn GP-RL2 on.

9.11.4.7 Set DMM to current circuit by turn GP-RL6 on.

9.11.4.8 Connect Ammeter to VDD path.

9.11.4.8.1 Turn GP-RL4 off.

9.11.4.8.2 Turn DCM-RL1 on.

9.11.5 Send pattern 1 ON

9.11.5.1 Communicate to UUT by NI USB-8451 for set the LCD segment to pattern 1 ON. (see item 9.7.5.1)

9.11.6 Check VDD current consumption on LCD pattern 1 ON

9.11.6.1 Measure the current must be in limit.

9.11.7 Check LCD segment pattern 1 ON

9.11.7.1 Check all segments that define in pattern 1 must be present.

9.11.8 Check LCD segment pattern 2 OFF

9.11.8.1 Check all segments that define in pattern 2 must be not present.

9.11.9 Send pattern 2 ON

9.11.9.1 Communicate to UUT by NI USB-8451 for set the LCD segment to pattern2 ON. (see item 9.7.5.2)

9.11.10 Check VDD current consumption on LCD pattern 2 ON

9.11.10.1 Measure the current must be in limit.

9.11.11 Check LCD segment pattern2 ON

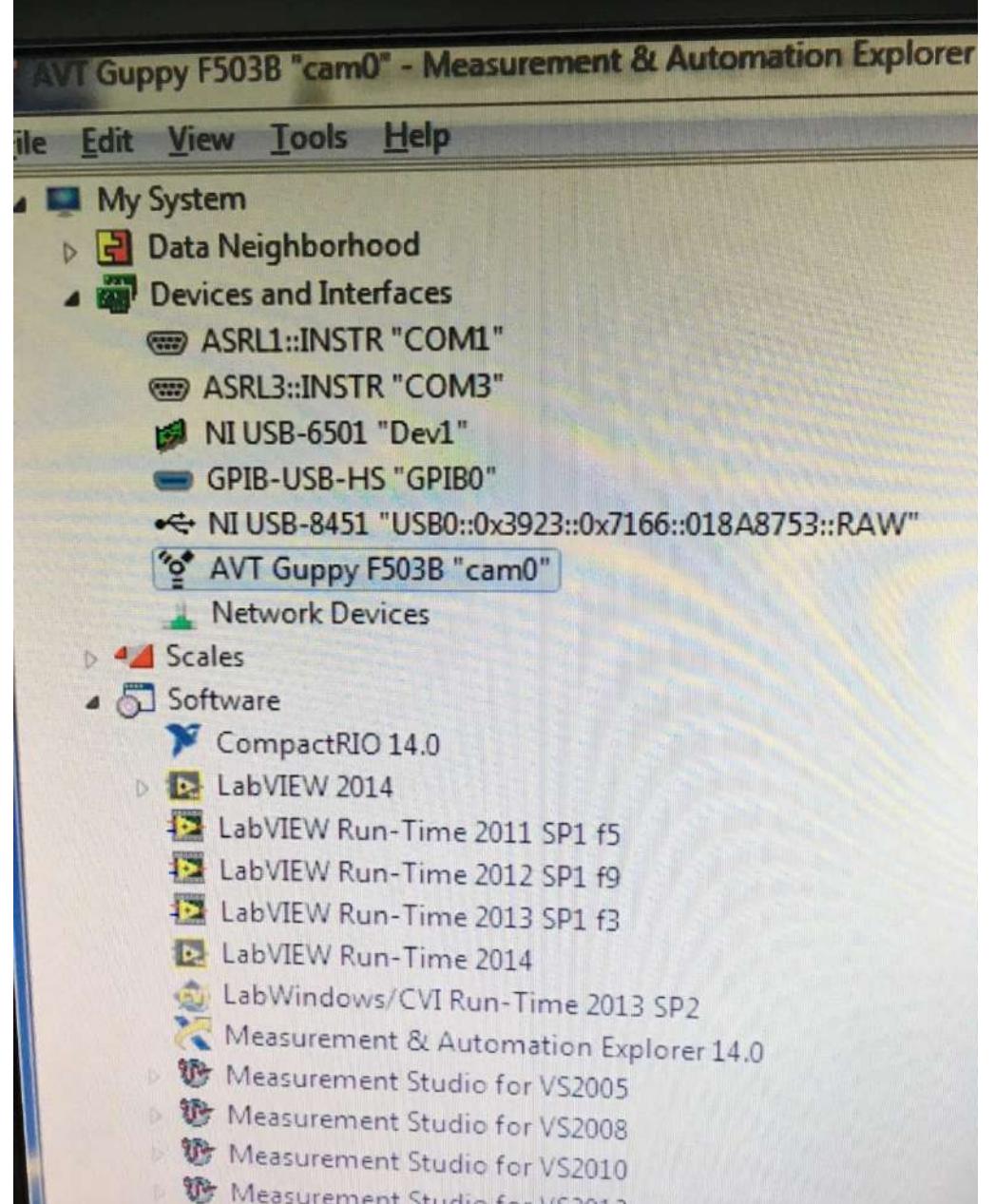
9.11.11.1 Check all segments that define in pattern 2 must be present.

9.11.12 Check LCD segment pattern1 OFF

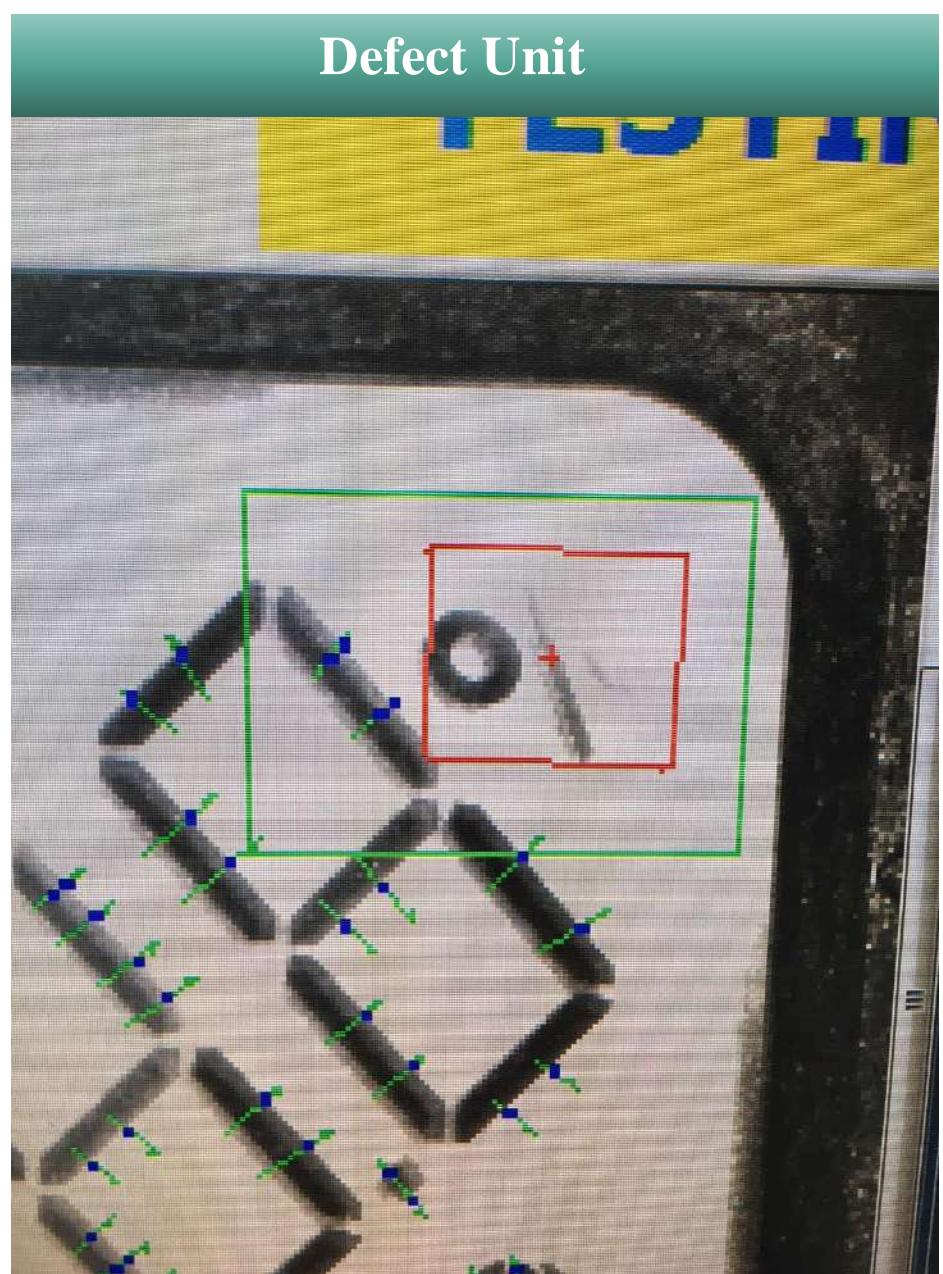
NI-Max Software List



NI-Max Device And Interface List



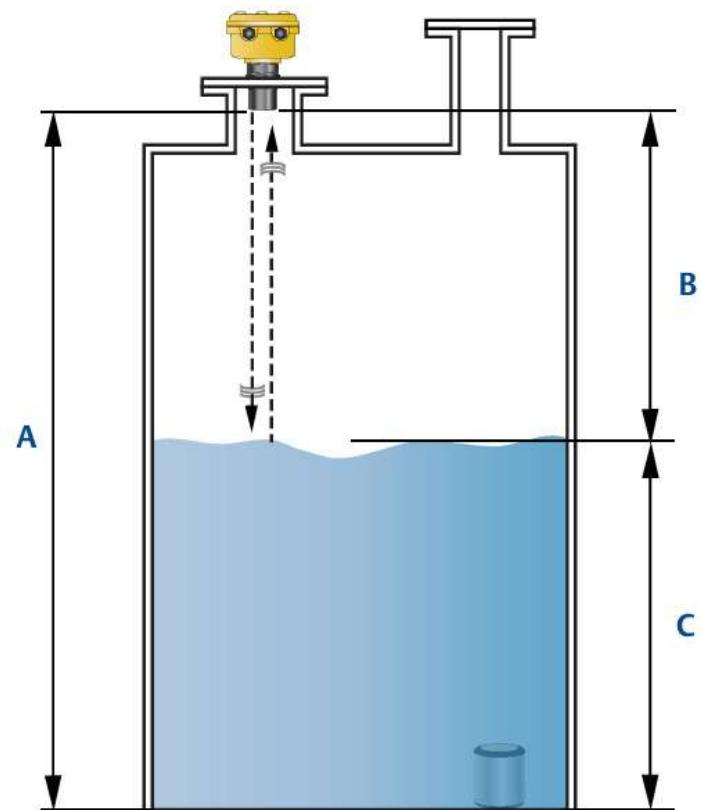
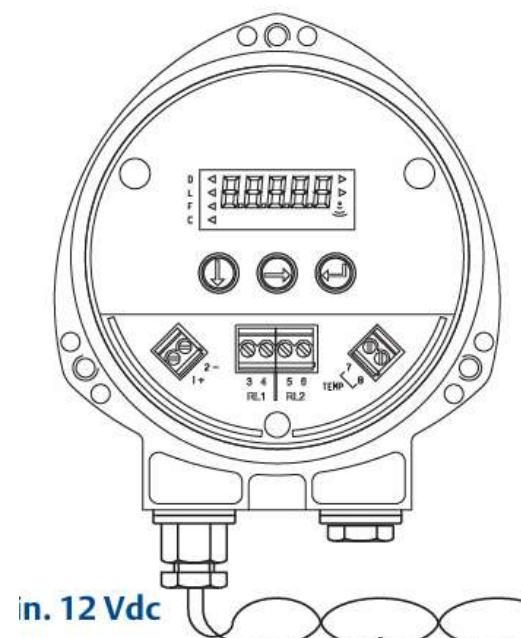
Defect Unit



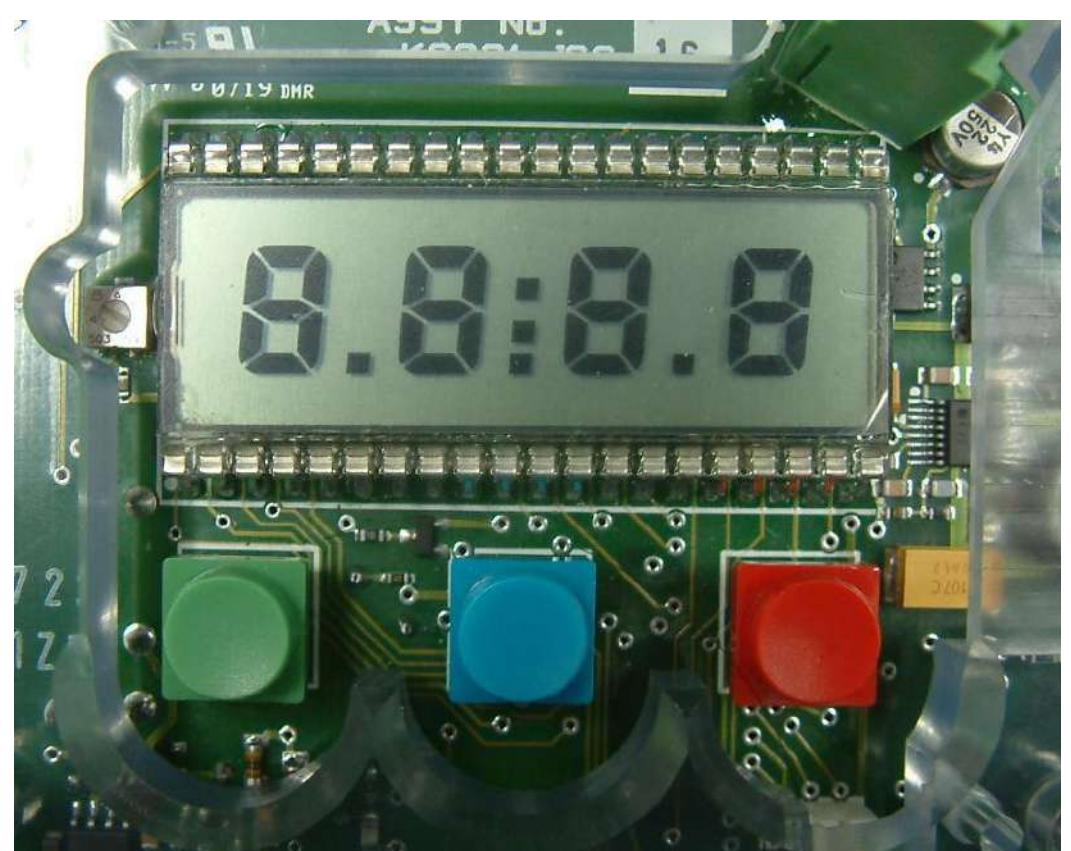
Test Program Porting From VB6 To Labview (Ultrasonic Liquid Sensor, Mobrey)

Software Tools and Programming Language : Labview, VB6

- Porting existing test program(VB6) to labview
- Measure signal frequency, AC/DC voltage, current, amplitude, vpp, pulse width and pulse count with oscilloscope
- Signal generator, AC/DC Power supply, usb-6501, digital multimeter, thermometer, attenuator, flash programmer, HART protocol
- Connect to database server

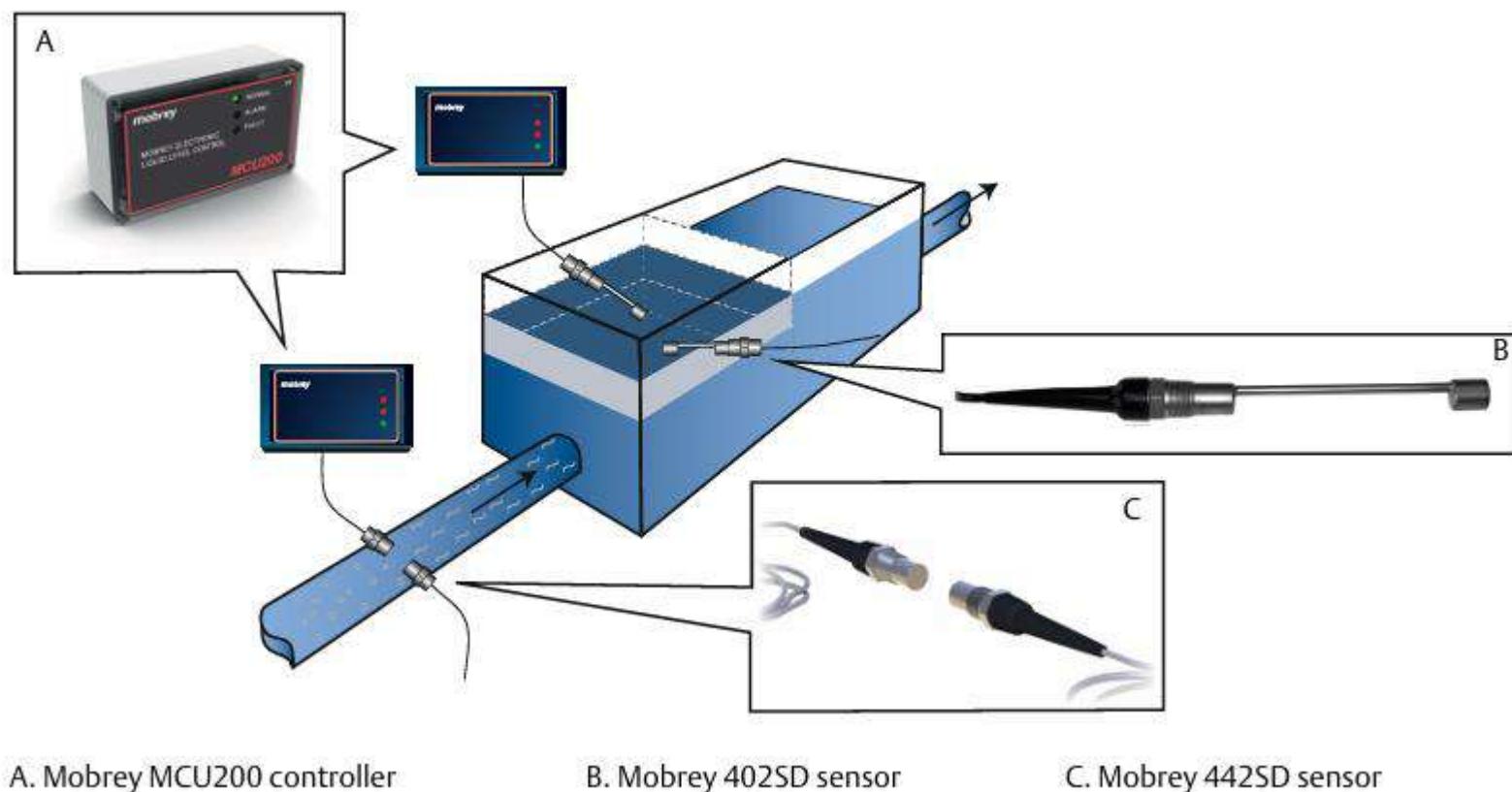


Ultrasonic Level Transmitter



PCBA

Figure 1. Typical application with Mobrey 442SD and 402SD ultrasonic Gap Sensors

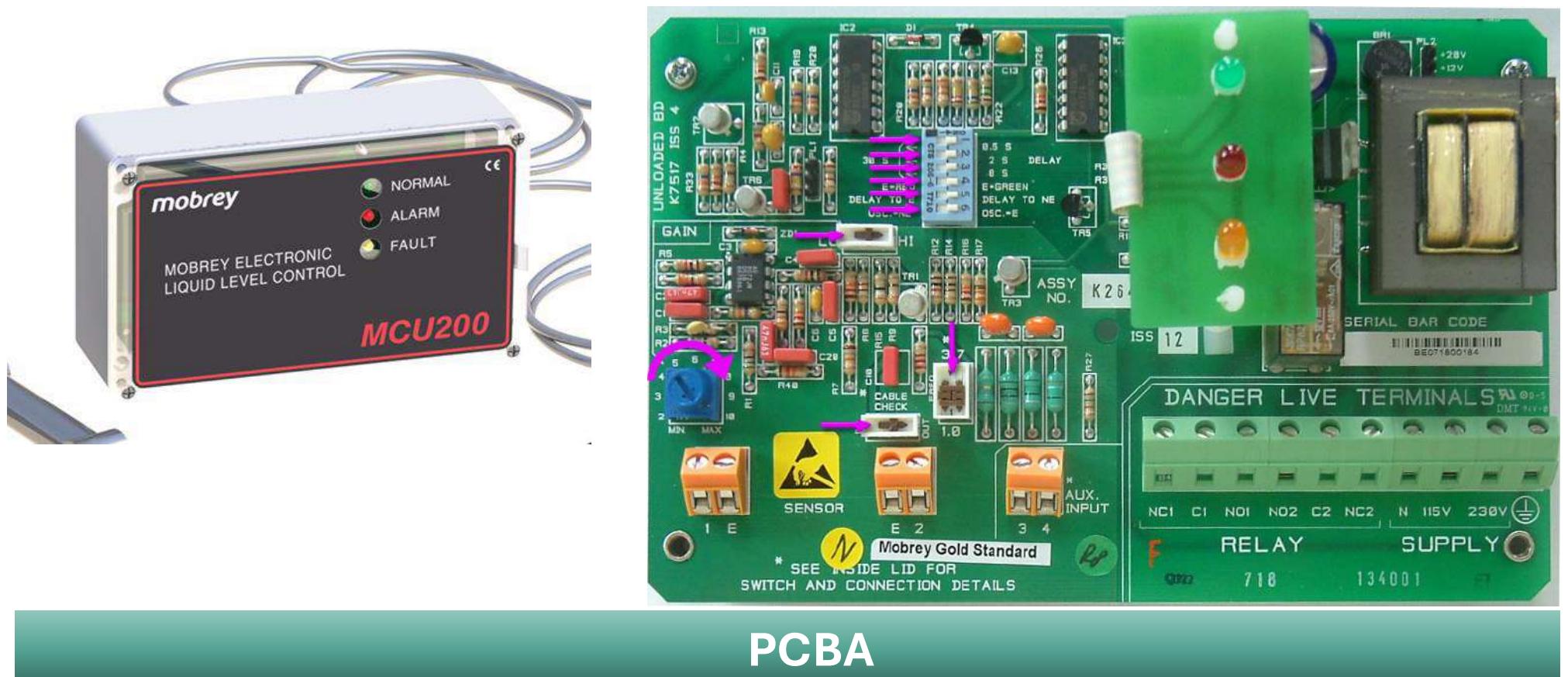


A. Mobrey MCU200 controller

B. Mobrey 402SD sensor

C. Mobrey 442SD sensor

Liquid Level Control



PCBA

	A	B	C	D	E	F	G	H
1	AgileReference	DDS-K8021_FinalTest_LV	Rows	42	ATE Type	Mobrey	Validation Status	Not Validated
2	AgileDescription	Assembly Test and Calibration						
3	Test Name	Measurement Name	Expected String	Low Limit	High Limit	Units	CompStr	TestType
4	Initialize	Init instruments					EQ	Pass/Fail
5	PowerOnUUT	Power On UUT					EQ	Action
6	PowerRails10Volt	Power Rails 10Volt		10.07	10.43	Volt	GELE	MultipleNumeric
7	PowerRails5Volt	Power Rails 5Volt		4.65	5.05	Volt	GELE	MultipleNumeric
8	LoadDefaultDisplay	Load Default Display					EQ	Pass/Fail
9	VoltR500	Current Output at Volt R500				2	Volt	GE
10	Trim4mA	Calibrae Current 4 mA		3.995	4.005	mV	GELE	MultipleNumeric
11	Trim20mA	Calibrae Current 20 mA		19.995	20.005	mV	GELE	MultipleNumeric
12	SetCurrent12mA	Check linearity at 12 mA		11.99	12.01	mV	GELE	MultipleNumeric
13	TE Voltage	Volt Transmit Enable		4.5	5.5	Volt	GELE	MultipleNumeric
14	TE Pulse Width	Pulse Width Transmit Enable		100	300	uS	GELE	MultipleNumeric
15	TE Frequency	Frequency Transmit Enable		0.9	1.1	Hz	GELE	MultipleNumeric
16	NumOfPulse PL3/1&PL3/2	Number of Pulse at PL3/1&PL32		10		Number	EQ	MultipleNumeric
17	FreqAtPL3/1&PL3/2	Frequency at PL3/1&PL3/2		49	59	KHz	GELE	MultipleNumeric
18	VoltageRefIC3-52	Voltage Reference IC3-52		2.43	2.63	Volt	GELE	MultipleNumeric
19	Ambient Temperature	Ambient Temperature				degreeC	LOG	MultipleNumeric
20	Difference Temperature	Difference Temperature		-0.2	0.2	degreeC	GELE	MultipleNumeric
21	Button1 Test	Button1 Test					EQ	Pass/Fail
22	Button2 Test	Button2 Test					EQ	Pass/Fail
23	Button3 Test	Button3 Test					EQ	Pass/Fail
24	Voltage 5V Rail	Power Fail Detection		3.9	4.56	Volt	GELE	MultipleNumeric
25	Power On UUT	Power On UUT					EQ	Action
26	EchoSize80	EchoSize80		79	81	Percent	GELE	MultipleNumeric
27	GenPulse7.6Vpp	GenPulse 7.6Vpp		7.30	7.90	Vpp	GELE	MultipleNumeric
28	Volt Threshold 900mV	Volt Threshold 900mV		800	1080	mV	GELE	MultipleNumeric

Test Step

191								
192	NumOfPulse PL3/1&PL3/2							
193	NumOfPulse PL3/1&PL3/2_Start							
194	InputParameters.Control_PulseGen					5 0:Meas Voltage,1:Meas Puls		
195	InputParameters.Osc_Setting.Vertical Scale (V)					200		
196	InputParameters.Osc_Setting.Vertical Offset					0		
197	InputParameters.Osc_Setting.Time per Division					0.000050		
198	InputParameters.Osc_Setting.Time Base Delay					0.000100		
199	InputParameters.Osc_Setting.Probe Attenuation (10)					100		
200	InputParameters.Osc_Setting.Trig_CH (0: Ch 1)					1		
201	InputParameters.Osc_Setting.Trigger Level (V)					2		
202	InputParameters.Osc_Setting.Acquisition Type (0: Normal)					0		
203	InputParameters.Osc_Setting.Acquisition Mode					0		
204	InputParameters.Osc_Setting.Measure_Ch					1		
205	InputParameters.Pulse_Threshold.Threshold_High					120		
206	InputParameters.Pulse_Threshold.Threshold_Low					-150		
207	InputParameters.Relays.Relay_1st					Dev0/line12 OSC_CH2_TE_0V = RL13		
208	InputParameters.Relays.Relay_2nd					Dev0/line1 PL3_1_2 = RL2		
209	InputParameters.Relays.Relay_3rd					Dev0/line10 OSC_CH1_X100 = RL11		
210	InputParameters.Relays.Relay_4th					Dev1/line7 PROBE_SEL = RL32		
211	InputParameters.Convert Unit					0 0;Don't,1:Do		
212	NumOfPulse PL3/1&PL3/2_End							
213								
214	FreqAtPL3/1&PL3/2							
235								
236	VoltR500							
251								
252	Trim4mA							

Test Parameter

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	MOBREY TEST	FAIL											
2	Model Number	02120-1060-0001	Test Start Date	10/18/2016	User	29078 Local Pa	C:\Rosemount						
3	Serial Number		12 Test Start Time	9:50 AM	DDS	DDS-J3 NetWor	C:\Rosemount						
4	Station ID	ATE-MB004.bench.com	Cycle Time(Sec)	280.4	DPARM	DDS-J3206_LV							
5													
6	Test Name	Measurement Name	Measurement	Min	Max	Unit	Result	TestTime_Sec	DLL Name	I Function Name	Function R	Validated?	
7	PCB_Issue	PCB Issue		3	3		PASS	2.201126	Mobrey	StringVerify		YES	
8	SoftwareVerify	Software Version		2.1	2.1		PASS	1.863106	Mobrey	StringVerify		YES	
9	TP7ResetLow	Reset Low at TP7		0		0.1 Volt	PASS	2.843163	Mobrey	Meas_DMM	V 1.0.0.0	YES	
10	TP5+4VSWSupply	TP5 +4V SW Supply		0.96		1.1 Volt	PASS	1.066061	Mobrey	Meas_DMM	V 1.0.0.0	YES	
11	TP7ResetHigh	Reset High at TP7		2.86	2.7	3.1 Volt	PASS	2.064118	Mobrey	Meas_DMM	V 1.0.0.0	YES	
12	TP8OscillatorFreq	Meas Osc Frequency at TP8		500.48	475	525 KHz	PASS	6.445369	Mobrey	Meas_Scope	V 1.0.0.0	YES	
13	TP8OscillatorAmpl	Meas Osc Amplitude at TP8		974.37	600	mVpp	PASS	3.199183	Mobrey	Meas_Scope	V 1.0.0.0	YES	
14	PL1/7At500Hz	Meas Voltage at PL1/7 500Hz		532.7	490	590 mVpp	PASS	4.791274	Mobrey	Meas_Scope	V 1.0.0.0	YES	
15	PL1/7At700Hz	Meas Voltage at PL1/7 700Hz		480	420	500 mVpp	PASS	2.533145	Mobrey	Meas_Scope	V 1.0.0.0	YES	
16	PL1/7At1200Hz	Meas Voltage at PL1/7 1200Hz		450	380	460 mVpp	PASS	4.450254	Mobrey	Meas_Scope	V 1.0.0.0	YES	
17	TP6SensorAmpl	Meas Sensor Amplitude at TP6		3.22	2.5	Vpp	PASS	2.884165	Mobrey	Meas_Scope	V 1.0.0.0	YES	
18	TP6SensorFreq	Meas Sensor Frequency at TP6		1200.1	1125	1275 Hz	PASS	4.127236	Mobrey	Meas_Scope	V 1.0.0.0	YES	
19	Quiescent Current	Quiescent Current		0.95	0.92	0.98 mA	PASS	2.267129	Mobrey	Meas_DMM	V 1.0.0.0	YES	
20	TP9 VoltageAtLoadOff	TP9 Voltage at Load off		4.02	3.9	4.1 Volt	PASS	3.32419	Mobrey	Meas_Scope	V 1.0.0.0	YES	
21	TP9 RippleAtLoadOff	TP9 Ripple at Load off		3.01		25 mV	PASS	5.237299	Mobrey	Meas_Scope	V 1.0.0.0	YES	
22	TP11 VoltageAtLoadOff	TP11 Voltage at Load off		3.29	3.23	3.37 Volt	PASS	3.656209	Mobrey	Meas_Scope	V 1.0.0.0	YES	
23	TP11 RippleAtLoadOff	TP11 Ripple at Load off		28.74		40 mV	PASS	5.157295	Mobrey	Meas_Scope	V 1.0.0.0	YES	
24	Red LED flash rate	Red LED flash rate		0.49	0.45	0.55 Hz	PASS	10.853621	Mobrey	Meas_Scope	V 1.0.0.0	YES	
25	DelayLoadOnAt1300Hz	Delay Load On at1300Hz		0.13	0.9	1.2 second	FAIL	8.171468	Mobrey	Meas_Delay	V 1.0.0.0	YES	
26	CurrentLoadOnAt1300Hz	Current load On at 1300Hz		2.37	2.2	2.5 mA	PASS	1.124064	Mobrey	Meas_DMM	V 1.0.0.0	YES	
27	TP9 VoltageAtLoadOn	TP9 Voltage at Load on		4.02	3.9	4.1 Volt	PASS	4.276244	Mobrey	Meas_Scope	V 1.0.0.0	YES	
28	TP9 RippleAtLoadOn	TP9 Ripple at Load on		3.52		25 mV	PASS	4.971284	Mobrey	Meas Scope	V 1.0.0.0	YES	

Datalog

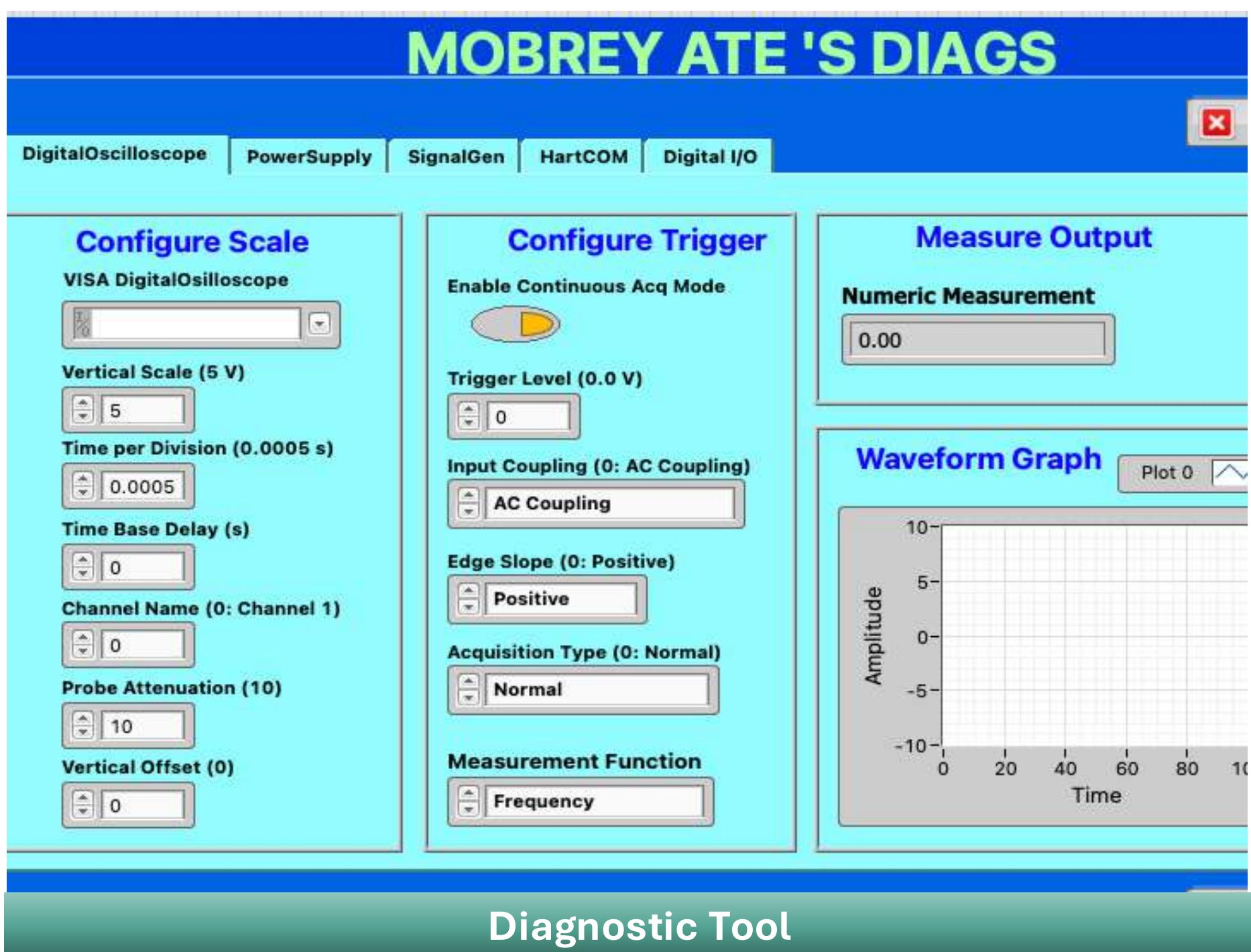
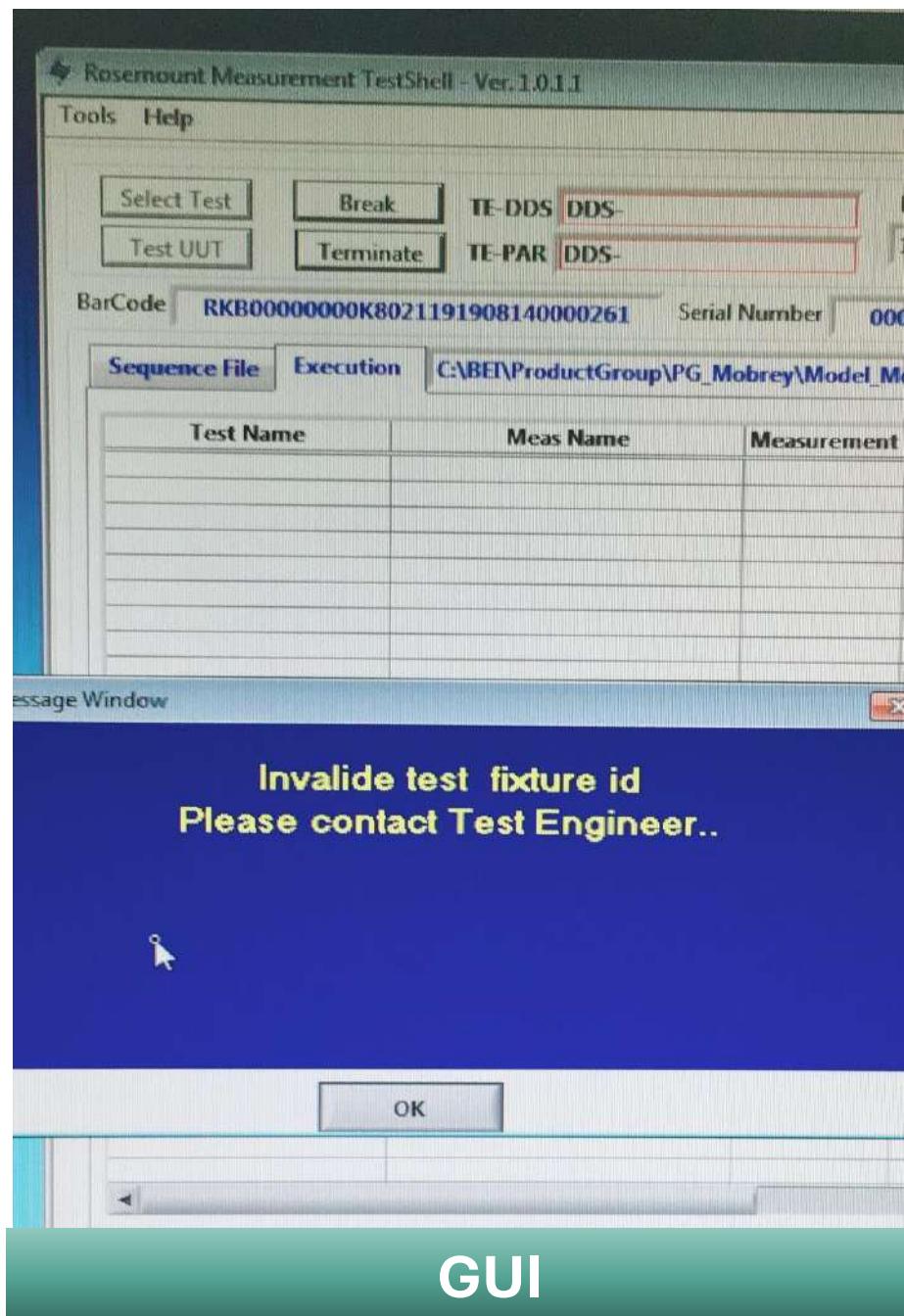
mobrey		PROCEDURE		
TITLE	ASSEMBLY TEST & CALIBRATION	DOCUMENT NUMBER	TS89006/480	
12	600/11	CHANGE NOTE	NAME	REVISED CHECKED
				K9658 269 DIC
3.2 Load Default Parameters and Check Display				
3.2.1 Connect Modem or Programmer as per connections above. 3.2.2 Unlock the UUT using Command 130, data 2, to set to 'fully open'. 3.2.3 Reset UUT using Command 42.				
3.3 Check and Calibrate Temperature Measurement				
Note: The UUT will warm up over a period of time. An offset of 1.3°C is used to compensate for this self heating but to ensure accuracy this test should be performed within 1 minute of power being applied.				
3.3.1 Check that the reference voltage on IC3 Pin 52 with respect to 0V is 2.63V ±0.1V. 3.3.2 Use Command 162, data 4..7, to read temperature (D25). 3.3.3 Measure the ambient temperature in °C from the thermometer 3.3.4 Subtract 1.3°C from this value to give T 3.3.5 Calculate temperature difference T1 using T1 = T - D25 (ignore any minus sign) 3.3.6 If T1 is greater than 9°C reject the UUT. 3.3.7 Calculate T2 using T2 = (D25 - (T1/1.587)) 3.3.8 Enter the value T2 into the UUT using Command 126. 3.3.9 Check reading D25 is equal to 3.3.8 ±0.2 °C				
3.4 Check Current Output				
3.4.1 Use Command 40 to select current output 3.4.2 Trigger oscilloscope from Transmit Enable, PL2.7, and observe waveform across the 500 Ω resistor with time base set to 0.1s/div. 3.4.3 Check that the voltage across to 500 Ω resistor does not exceed 2.0V (3.6mA) during the 1s cycle 3.4.4 Use Command 40 to set current to 4mA. Measure using DMM 3.4.5 Set current output to 4.000mA ±0.0005mA using Command 45 3.4.6 Use Command 40 to set current to 20mA. Measure using DMM 3.4.7 Set current output to 20.000mA ±0.005mA using Command 46 3.4.8 Use Command 40 to set current to 12mA. Measure using DMM 3.4.9 Check that the current drawn is 12mA ±0.01mA.				
3.5 Check Pulse Generation				
3.5.1 Check that a positive 5V pulse typically 100µs to 300µs long, appear on Transmit Enable, PL2.7, every 1s ±0.1s. 3.5.2 Using Transmit Enable, PL2.7, as a trigger check that a number of pulses, typically 10, appear between PL3.1 and PL3.2 with a frequency typically between 49 kHz and 59 kHz.				

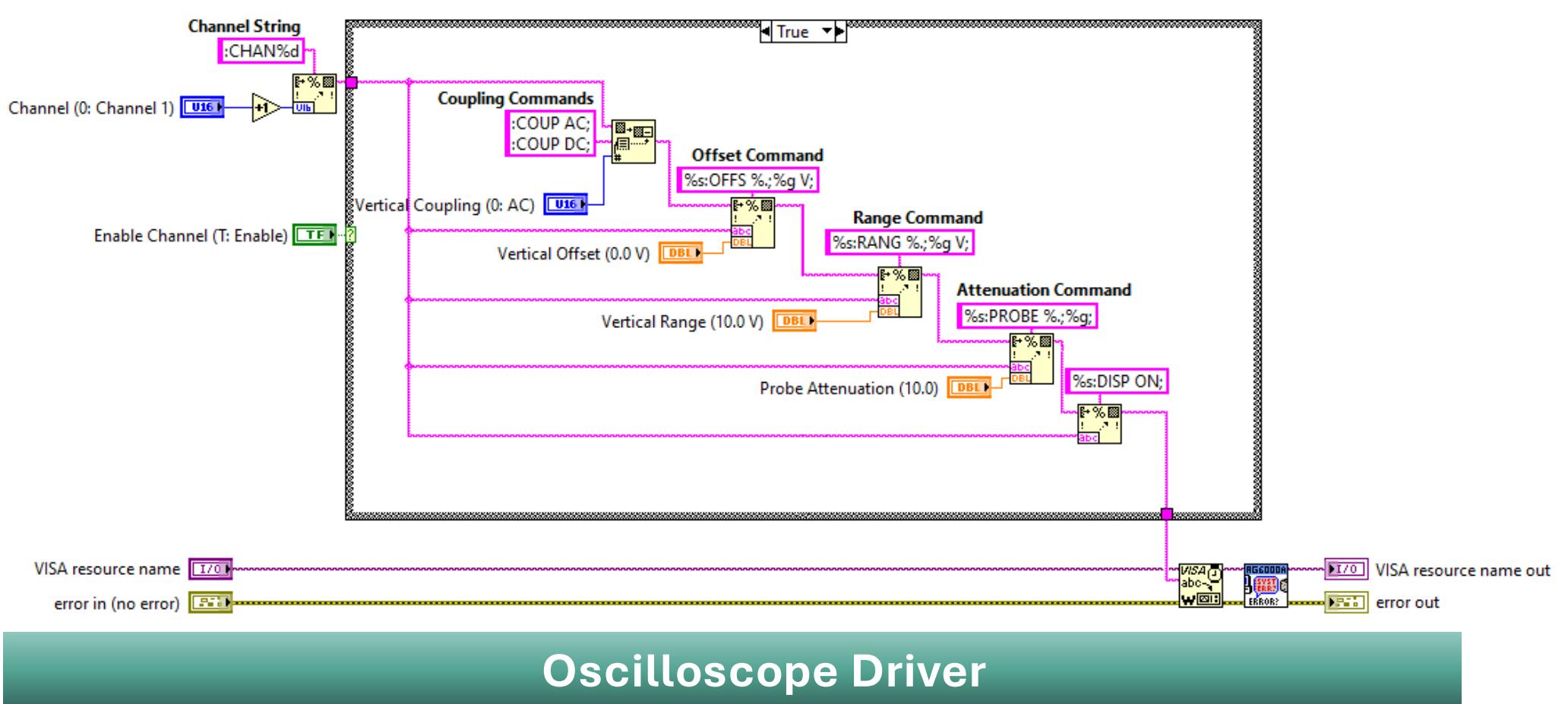
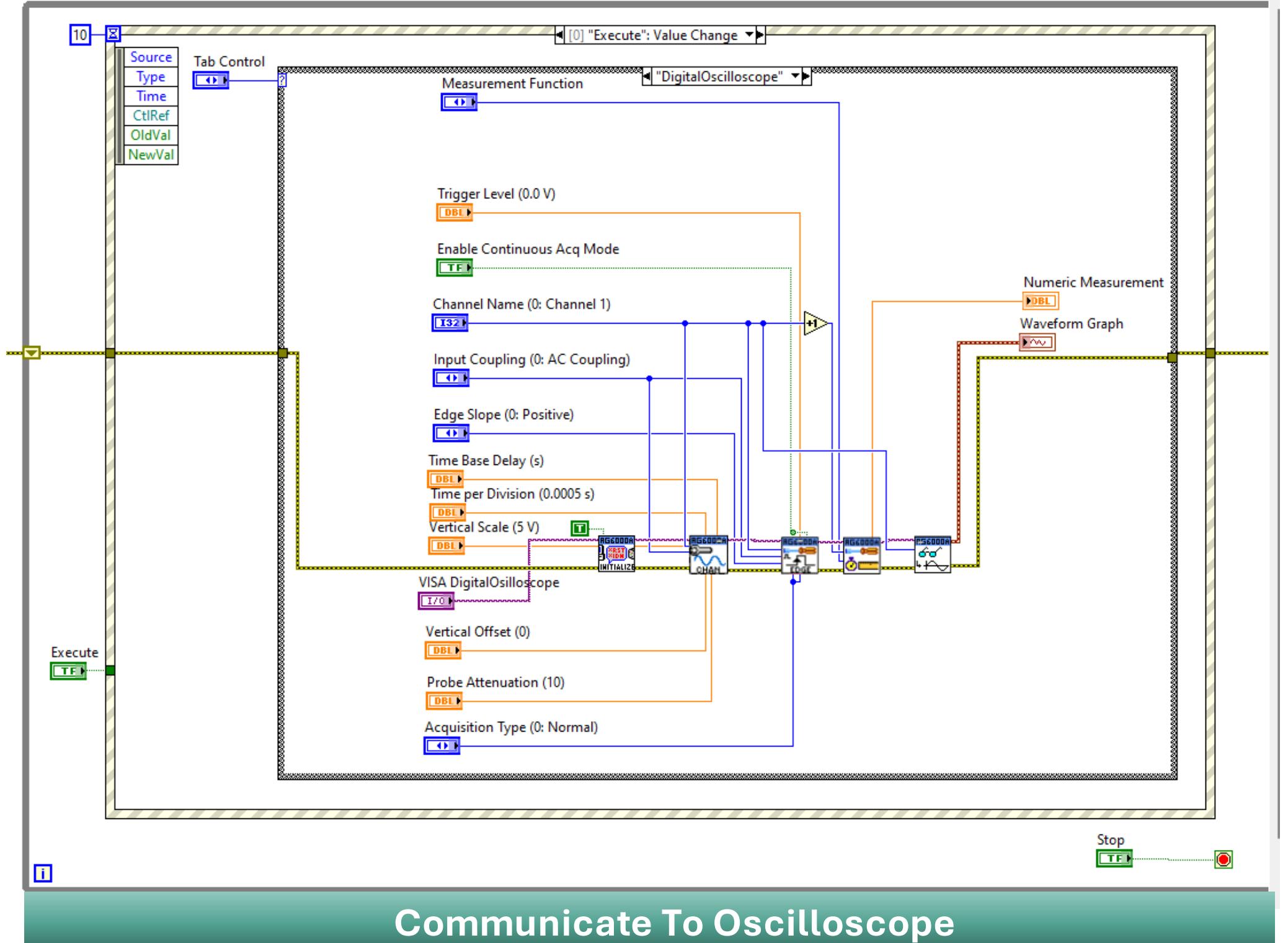
Dev0/line12 PL2.7 - CH2

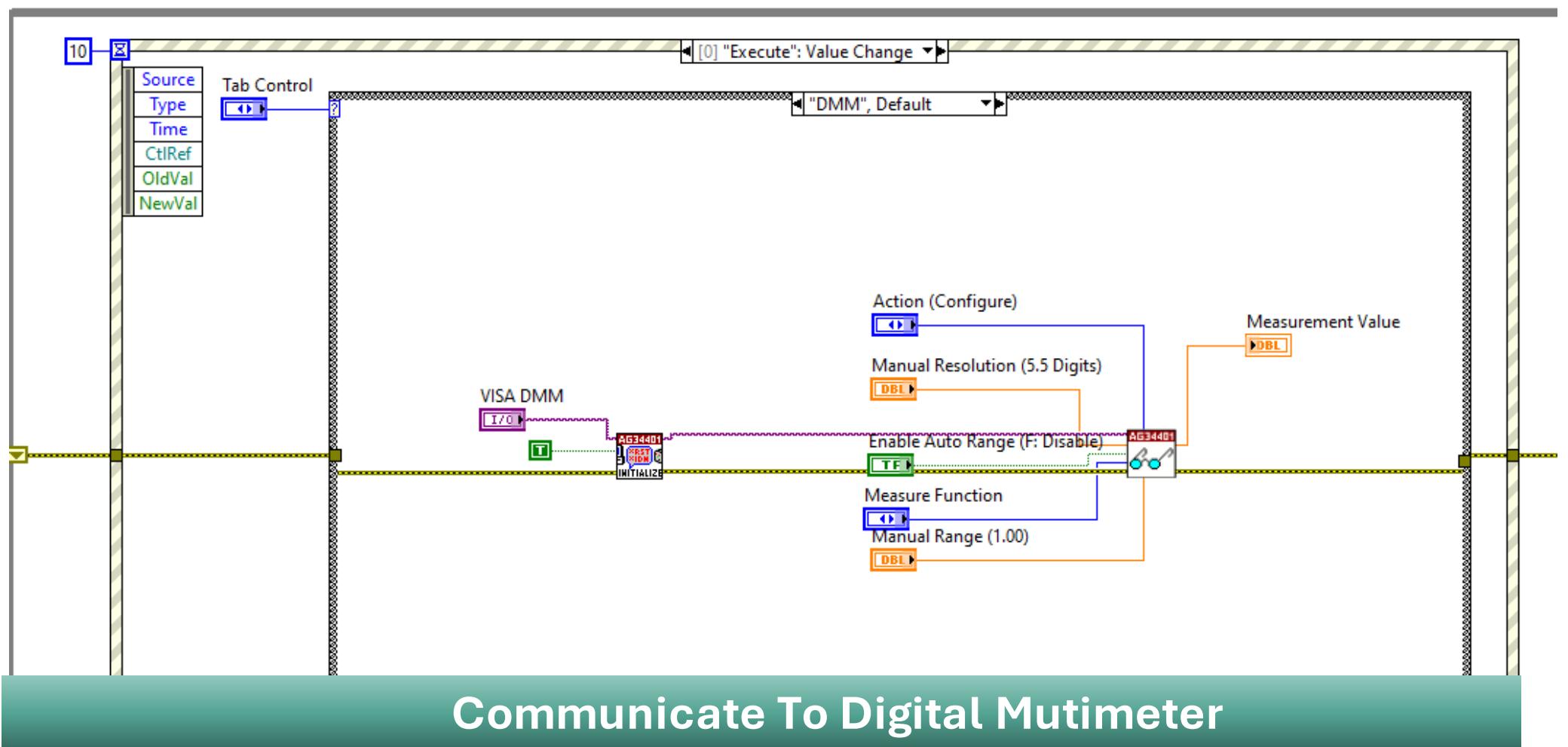
Dev0/line10 PL3.2 - CH1 - X100

Dev1/line7 Probe Select

Test Plan







Communicate To Digital Multimeter

```

frmMain.OSC_Config_CH CH2, COUPLING, "DC"
frmMain.OSC_Trig_Edge SOURCE_T, CH2
frmMain.OSC_Config_Timebase "100us"
frmMain.OSC_Config_CH CH2, SCALECH, "2V"
frmMain.OSC_Trig_Edge Slope, CH9, "POS"
frmMain.OSC_Trig_Edge Level, CH2, "1"
frmMain.OSC_Config_Timebase_delay "17.3mS"
frmMain.Relay "OSC_CH2_TE_0V", L_ON      'Connect CH2 to TE
frmMain.Relay "OSC_CH1_PL2_4", L_ON      'Connect CH1 to TP2
frmMain.Set_Generator_Output R_500HM
frmMain.Set_Generator "SIN", "54000", "0.1", 0, ON_   'Enable Output of sig gen

'Adjust the pot on the dummy transducer until the burst of pulse appear at 17.5 mS
Utility.InfoBox vbCrLf & "Adjust knob until the waveform on oscilloscope(Yellow)"

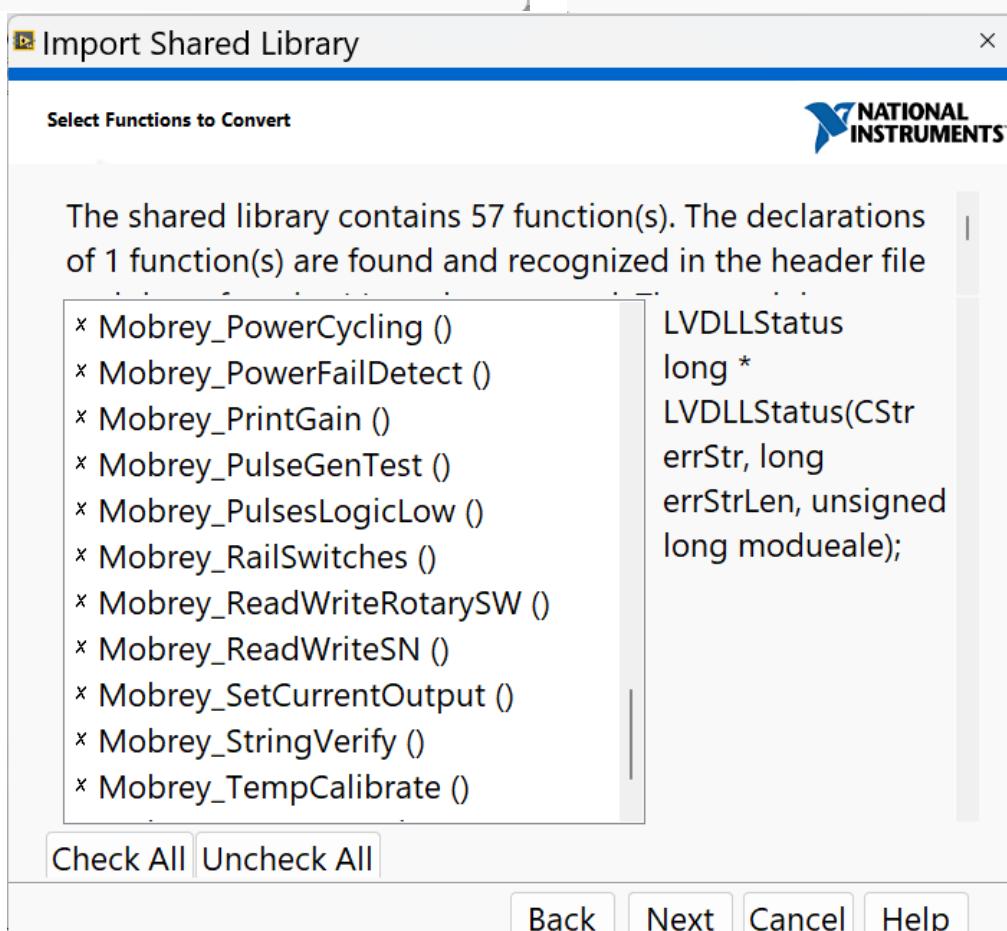
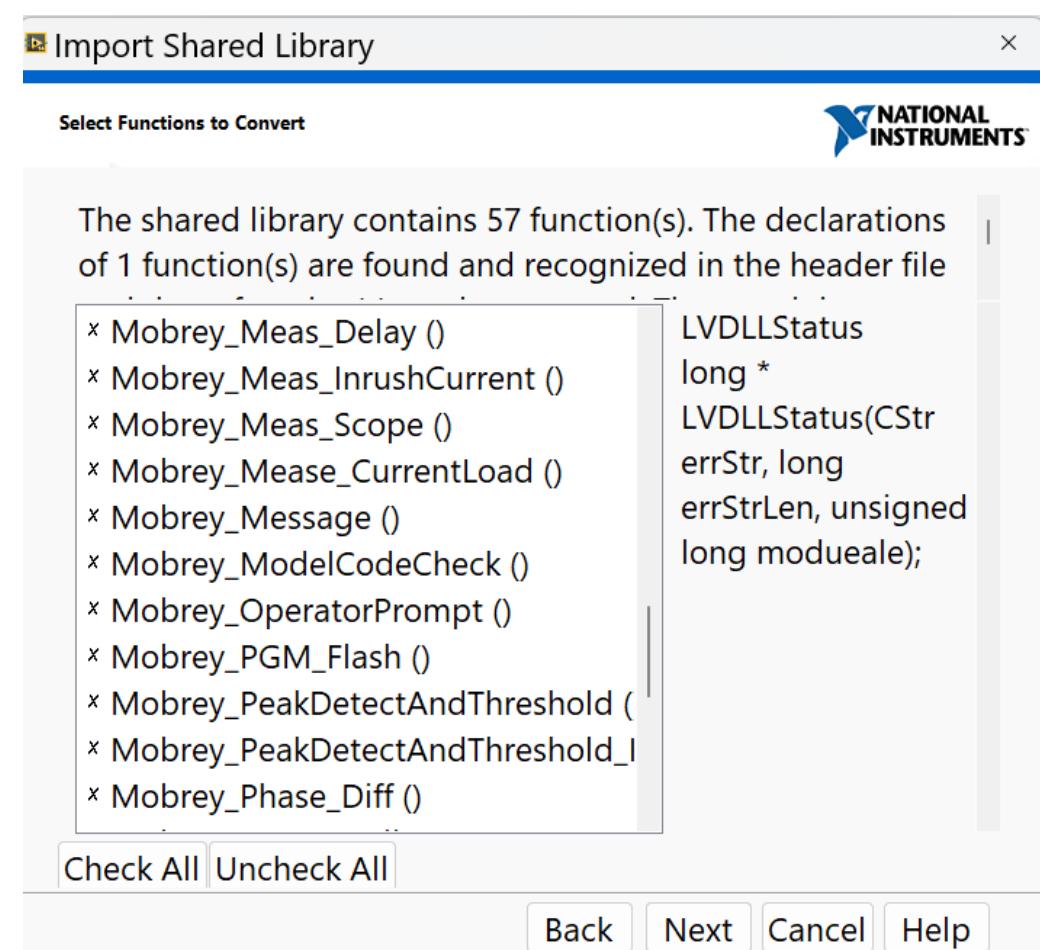
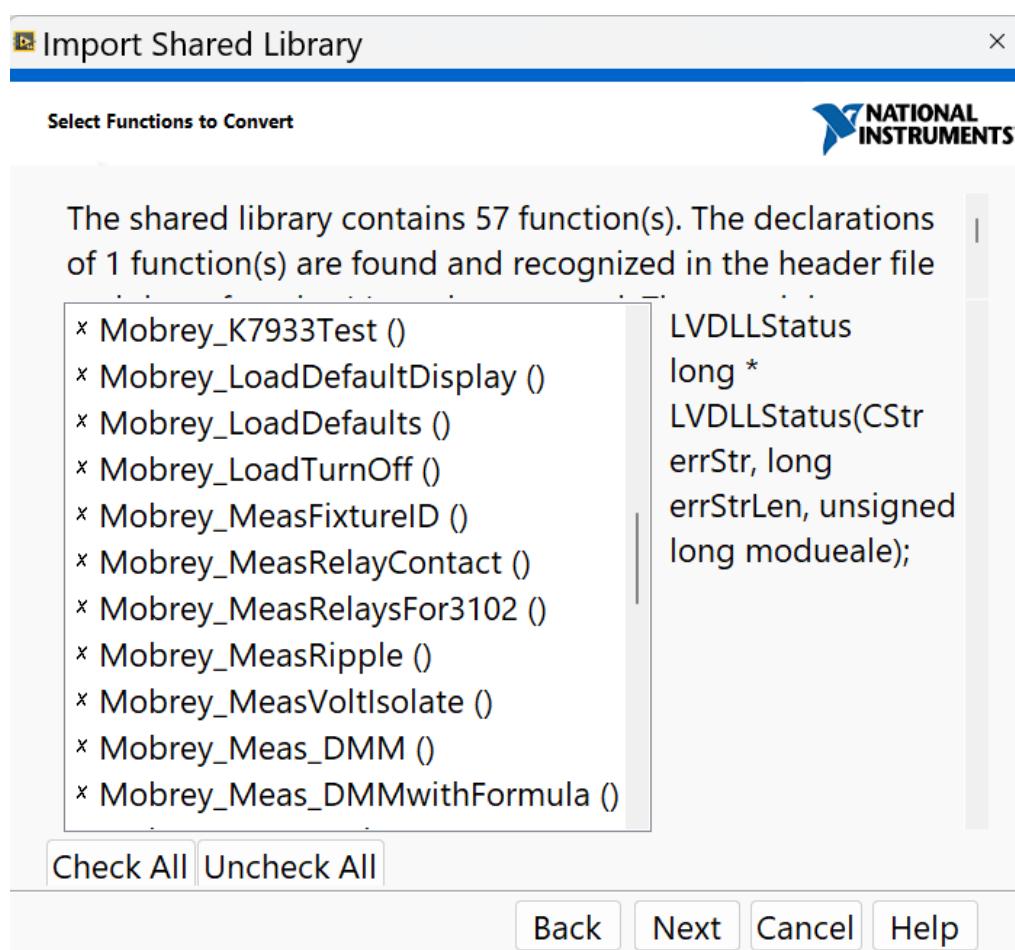
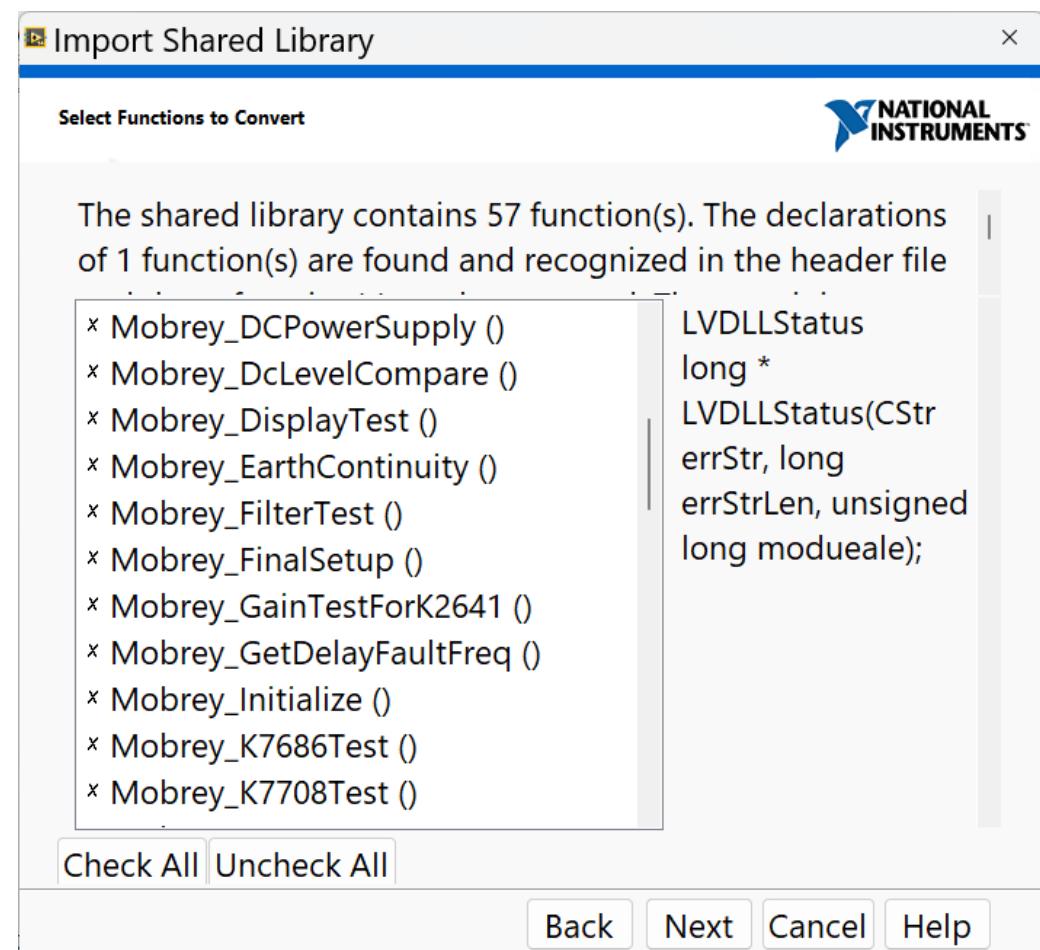
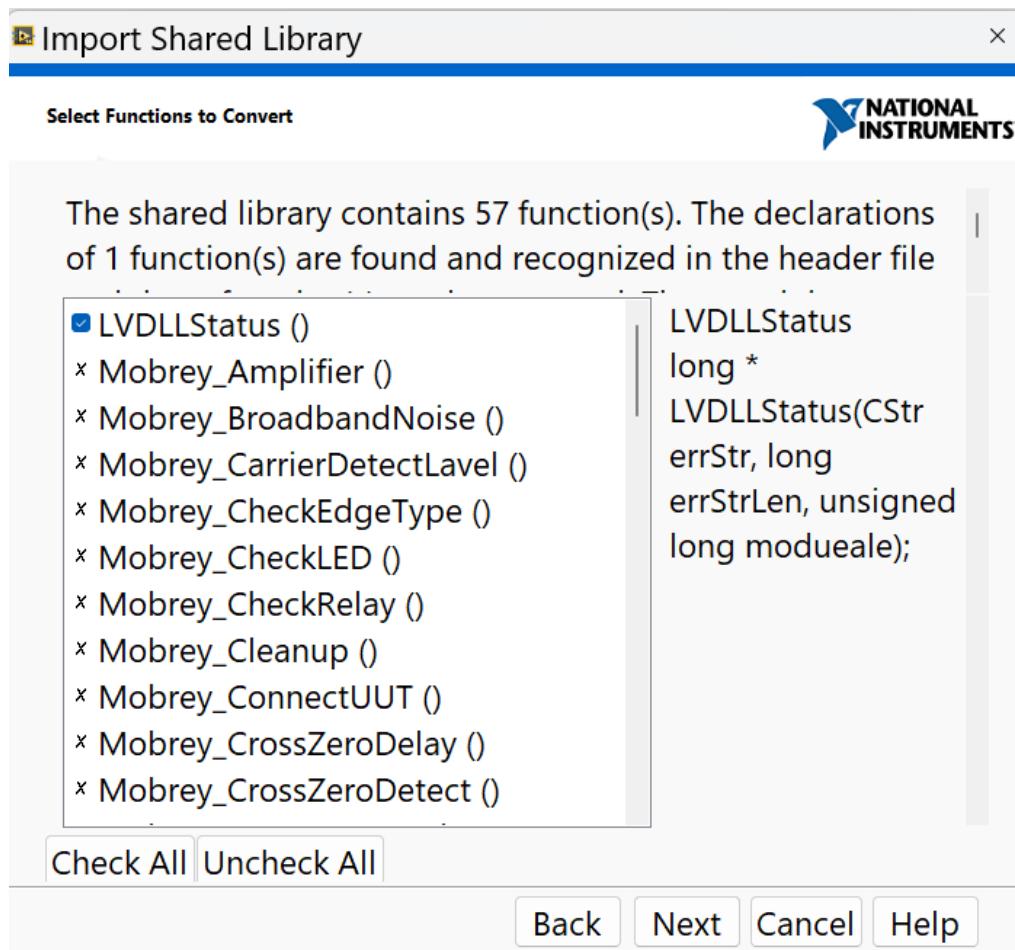
I

Finetune_Echo_Size dblEcho, 80
If frmMain.mblnTestAbort Then Exit Sub
For i = 1 To 3
    If ATEscope.Model = "DSO6012A" Then
        dblTemp = frmMain.OSC_Measurement(Vpp, CH1)
    Else
        dblTemp = frmMain.OSC_Measurement(Vpp, CH3)
    End If

    If dblTemp >= frmMain.mdblSpecMin And dblTemp <= frmMain.mdblSpecMax Then
        'Measurement successful
    Else
        'Measurement failed or out of range
    End If
Next i

```

Visual Basic Code



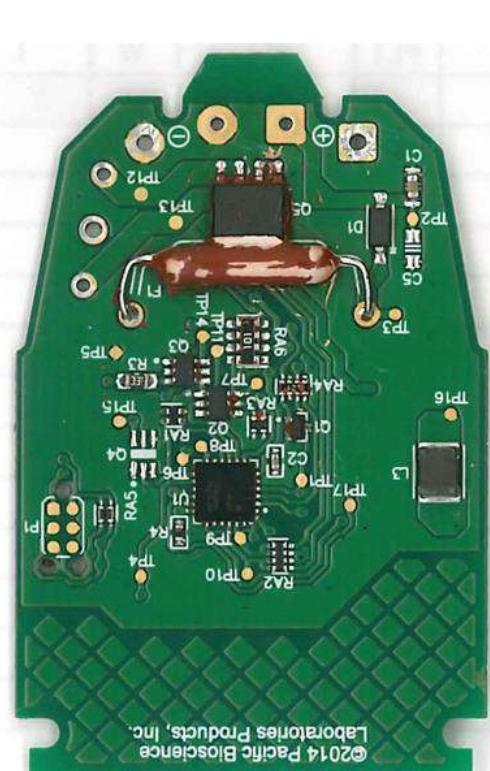
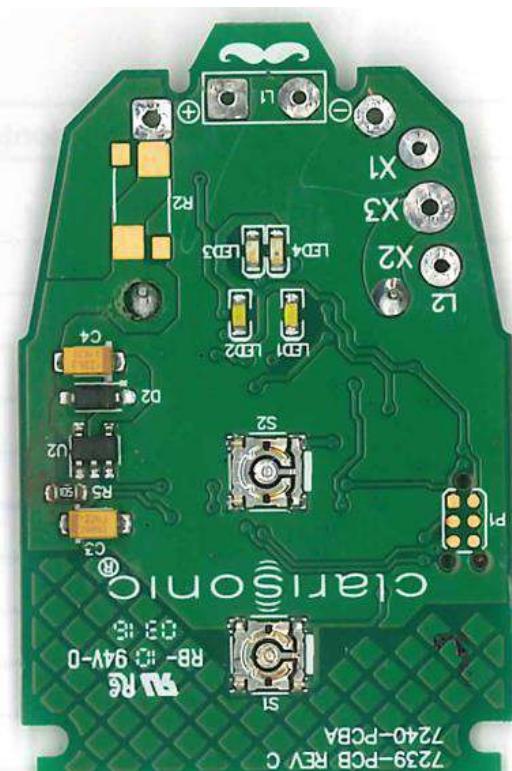
Automated Optical Test (Face Cleaner Product, Clarisonic)

Software Tools and Programming Language : Labview, OpenCV, and TestStand

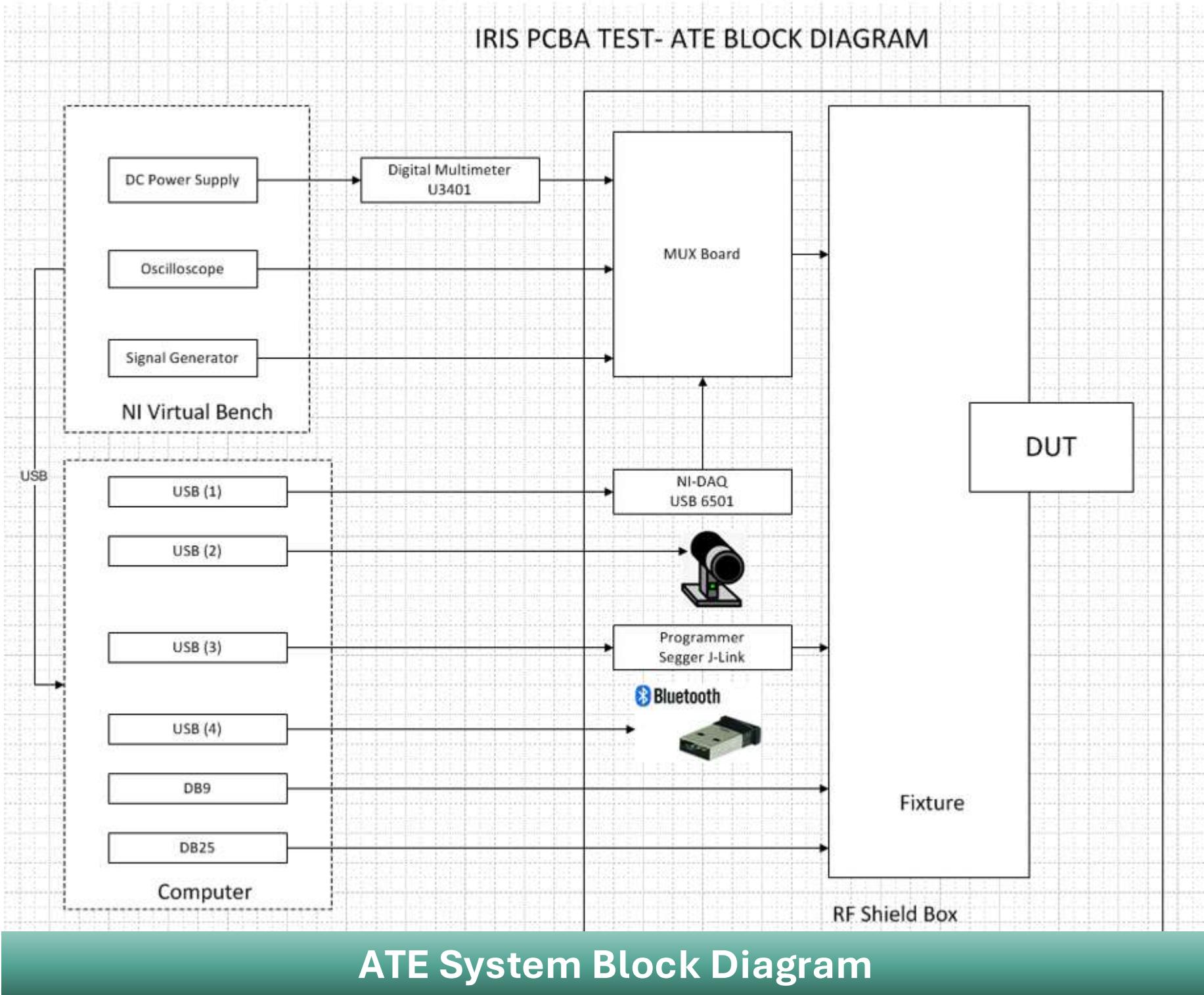
- Measure voltage and current with DMM
- Measure motor drive signal with 2 channel oscilloscope
- Communicate flash programmer to reprogram microcontroller Renesas 16-bit 16KB
- Measure capacitance with LCR meter
- Verify LED working with camera and OpenCV (open computer vision library)
- Verify bluetooth RSSI in shield test box (product can connect from smart phone application)



Face Cleaner



PCBA

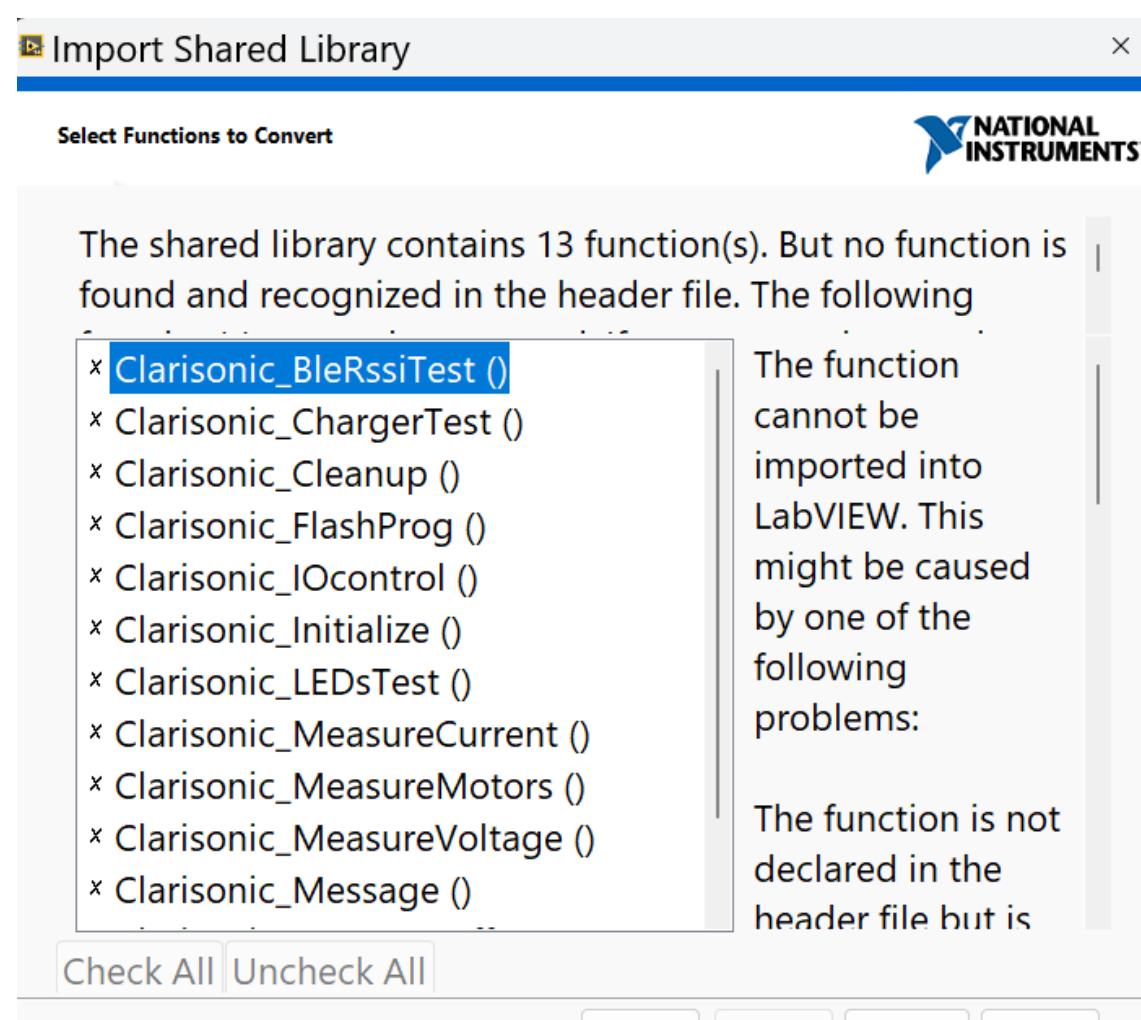


A	B	C	D	E	F	G	H
AgileReference	DDS-IRIS_5000484_LV	Rows	33	ATE Type	Clarisonic	Validation Status	Validated
AgileDescription	IRIS PCBA Test						
Test Name	Measurement Name	Expected Str	Low Limit	High Limit	Units	CompStr	TestType
Initialize	Init instruments and follow setup 2					EQ	Pass/Fail
PowerOnUut	Power On					EQ	Pass/Fail
VCC IN	Check Power Input		2.25	2.55	Volt	GELE	MultipleNumeric
PwsCurrent	Check Power Current		18	30	mA	LEH	MultipleNumeric
FlashProg	1. Program the microprocessor					EQ	Pass/Fail
PWS Current	3.(a) Less than TBD mA from power supply			100	mA	LEH	MultipleNumeric
T7 Volt	3.(b) Power Supply voltage measured on X3		2.375	2.625	Volt	GELE	MultipleNumeric
T4 Volt	3.(c) 5V 5% on T4		4.75	5.25	Volt	GELE	MultipleNumeric
P1/1 Volt	3.(d) 3.3V 5% on pin 1 of P1		3.135	3.465	Volt	GELE	MultipleNumeric
Battery_Connect	Enable Battery					EQ	Pass/Fail
Verify D5	5.(a) LEDs D5A, D5B and D5C all activate for 200ms					EQ	Pass/Fail
PWS Current_S1	5.(b) Less than TBD mA from power supply			350	mA	LEH	MultipleNumeric
VerifyLEDsOn	6.(a) LEDs D1, D2, D3, D4 and D8 all activate for 200ms					EQ	Pass/Fail
X1_Off	6.(b) Motor is OFF			0.3	Vpp	LEH	MultipleNumeric
X2_Off	6.(b) Motor is OFF			0.3	Vpp	LEH	MultipleNumeric
PWS Current_S2	6.(c) Less than TBD mA from power supply			350	mA	LEH	MultipleNumeric
X1_Frequency	7.(b) Motor activates at 170Hz with 14.1% duty cycle for 400ms		161.5	178.5	Hz	GELE	MultipleNumeric
X1_DutyCycle	7.(b) Motor activates at 170Hz with 14.1% duty cycle for 400ms		13.395	14.805	Percent	GELE	MultipleNumeric
X2_Frequency	7.(b) Motor activates at 170Hz with 14.1% duty cycle for 400ms		161.5	178.5	Hz	GELE	MultipleNumeric
X2_DutyCycle	7.(b) Motor activates at 170Hz with 14.1% duty cycle for 400ms		13.395	14.805	Percent	GELE	MultipleNumeric
VerifyLEDsOff	7.(a) All LEDs OFF					EQ	Pass/Fail
D7 Flashing	8.(a) LEDs D6 and D7 are flashing, 200ms ON, 200ms OFF		150	250	ms	GELE	MultipleNumeric
D6 Flashing	8.(a) LEDs D6 and D7 are flashing, 200ms ON, 200ms OFF		150	250	ms	GELE	MultipleNumeric
Remove_R100	9 Remove the 100OHM pull-down					EQ	Pass/Fail
Battery_Disconnect	Disconnect Battery					EQ	Pass/Fail
Sleep Current	10 Sleep current < 10uA			10	uA	LOG	MultipleNumeric
PowerOffUut	12. Turn off the power supply					EQ	Action
M	13. (a) Remove the PCB					EQ	Action

Test Step

A	B	C	D
1 AgileReference	DDS-IRIS_5000484_LV	No Of rows	385
2 AgileDescription	IRIS PCBA Test		
3			
4 TestShell Parameters			
22			
23 Initialize			
24 Initialize_Start			
25 GlobalTestParameters.VirtualBench_ID	VirtualBench		
26 GlobalTestParameters.DMM_ID	DMM		
27 GlobalTestParameters.BLE_ID	BLEdongle		
28 GlobalTestParameters.RelayControl.GP_SetTestMode		1	
29 GlobalTestParameters.RelayControl.GP_Fixture		12	
30 GlobalTestParameters.RelayControl.GP_Battery		13	
31 Initialize_End			
32			
33 VCC IN			
34 VCC IN_Start			
35 InputParameters.TestPoint		13	VBAT
36 VCC IN_End			
37			
38 PwsCurrent			
39 PwsCurrent_Start			
40 InputParameters.Unit		1	0: Original,1:mA,2:uA
41 PwsCurrent_End			
42			
43 FlashProg			
48			
49 PWS Current			
53			
54 T7 Volt			
55 T7 Volt_Start			
56 InputParameters.TestPoint		9	Test point number
57 T7 Volt_End			
58			
59 T4 Volt			
60 T4 Volt_Start			
61			

Test Parameter



Function List From Labview Main DLL



AMI HANDLE PCBA TEST PLAN

TABLE OF CONTENTS

1. SCOPE	3
1.1. INTRODUCTION.....	3
1.2. APPLICABLE & RELATED DOCUMENTS	3
2. TEST SETUP	3
2.1. REQUIRED TEST POINTS FOR SPRING FIXTURE.....	3
2.2. EQUIPMENT.....	3
2.3. PROCEDURE.....	4

1. SCOPE

1.1. INTRODUCTION

APLUS/OSIE will build PCBAs for Ami. This is a test plan for these units. The tests listed below should be done before conformal coating.

1.2. APPLICABLE & RELATED DOCUMENTS

DOCUMENT TITLE	DOC # (IF USED)	DOCUMENT TYPE	COMMENT
Ami FRS	6718	MS Word, FRS	Rev B

2. TEST SETUP

2.1. REQUIRED TEST POINTS FOR SPRING FIXTURE

Reference schematic and PCB layout

- On P1 (6 thru-holes): VCC, RESET, GND, TOOL0
- Vbat+ and Vbat- (GND) at the battery solder thru-holes
- X1, X2, X3 solder thru-holes for stator coil connection
- L2 (charging coil) soldering points- connected to a charger coil or signal generator

2.2. EQUIPMENT

- Programmer for RL78-G12 (PGFP5)
- DC supply to power the UUT (with output current measurement or meter)
- 2-channel Oscilloscope for Motor drive signals viewing
- Multimeter
- LCR Meter, Capable of measuring capacitance at 200kHz (Agilent 4825A or equivalent)
- Ami charger and associated USB AC/DC supply or appropriate signal generator
- Spring-pin fixture
- To source the proper signal into the charging circuit, use an Ami charger, placed over and aligned with an Ami charger coil. Alternatively, use a signal generator set to the Ami charger frequency of ~60KHz and that is capable of sourcing ~3V peak-to-peak. One of these two signal sources should be attached to the Ami charger coil wiring points during the charger circuit test.

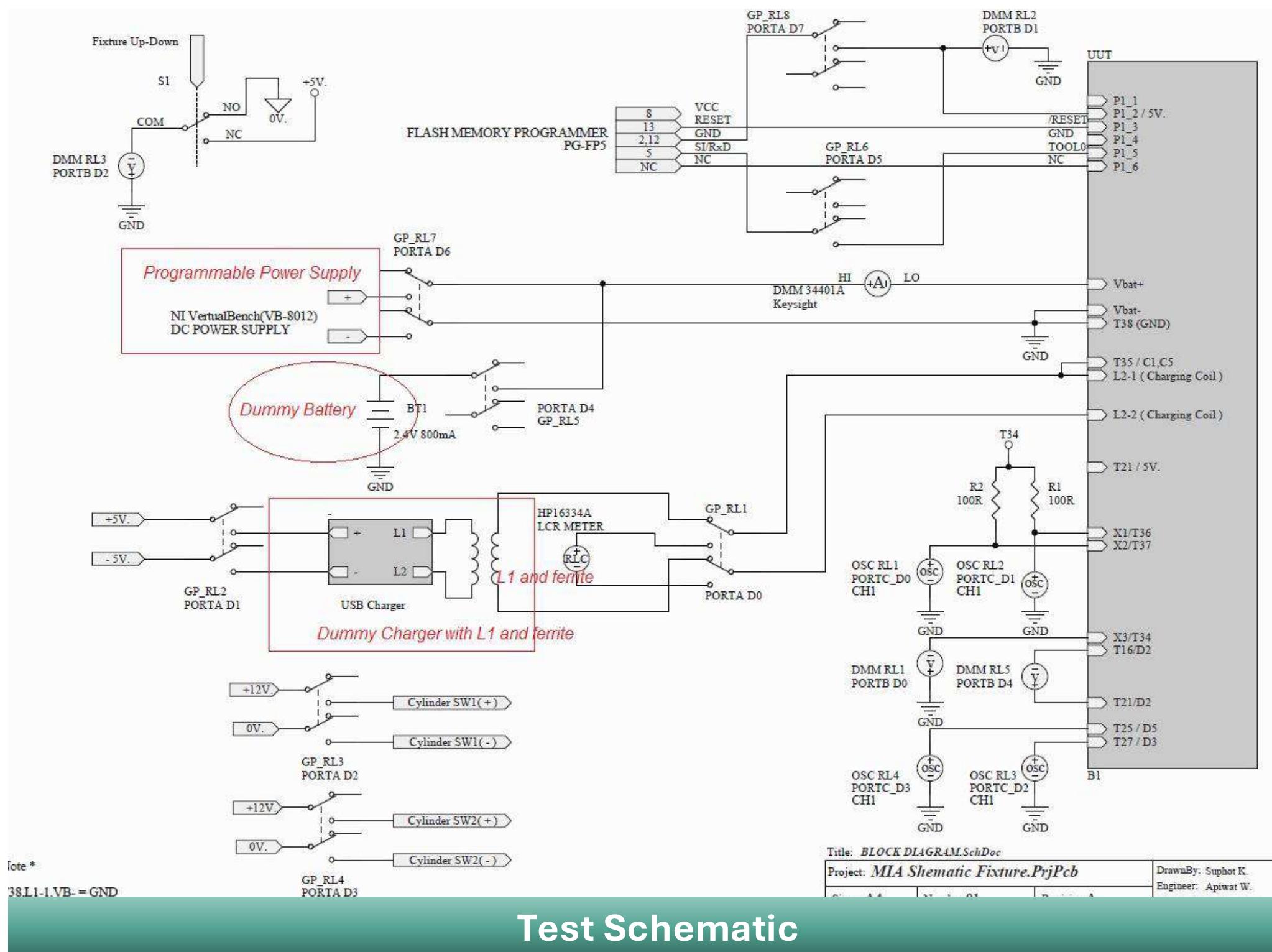
Test Plan



AMI HANDLE PCBA TEST PLAN

2.3. PROCEDURE

1. Program the microprocessor using P1 and an E1 or PGFP5 and with the turn-on test code, PBL 7142.
2. Place UUT onto spring pin fixture and:
 - a. Apply 100 ohm pull-up between the UUT power supply and X1 and X2.
 - b. Attach Oscilloscope probes to X1 and X2
 - c. Attach Multimeter probe to X3
3. With the DC power supply, apply a 0.5A current limited 2.5VDC signal to Vbat+, Vbat- and observe that:
 - a. No LEDs on, no motor drive signals
 - b. Less than 8mA current from the power supply.
 - c. Power supply voltage measured on X3
 - d. 5V ±5% on pin 2 of P1
4. Press SPEED button (S1) and observe that:
 - a. All LEDs are on (including LED5.)
 - b. Less than 100mA discharge current from DC power supply
5. Press SPEED button (S1) again and observe that:
 - a. All LEDs off
 - b. 168Hz, 14% duty factor, square wave motor drive signals appear at X1 and X2
 - c. Less than 40mA current from the power supply.
6. Press SPEED button (S1) again and observe that:
 - a. Green LED is flashing
 - b. No motor drive signals
 - c. Current between 40mA and 60mA from the power supply (bleed circuit is activated)
7. Press Speed button (S1) again and observe that:
 - a. All LEDs are off
 - b. No motor drive signals.
 - c. Less than 8mA current from the power supply (bleed circuit is off)
8. Press ON/OFF button (S2).
 - a. All LEDs are off



Test Schematic



LED Verify With OpenCV

Test Program (Handheld Medical Device)

Software Tools and Programming Language : VB, Visual Studio, Arduino

- Measure voltage and current with handheld digital multimeter
- PC interface with Arduino(RS232) to control relay board to connect to specific test point
- Verify that LED control button can turn on/off and dimming LED
- Use solenoid to push the button
- Tester have 2 head test that can work completely separate
- Connect to company database server



MedLight Functional Test Version 1.0.0

MedLight Functional tester

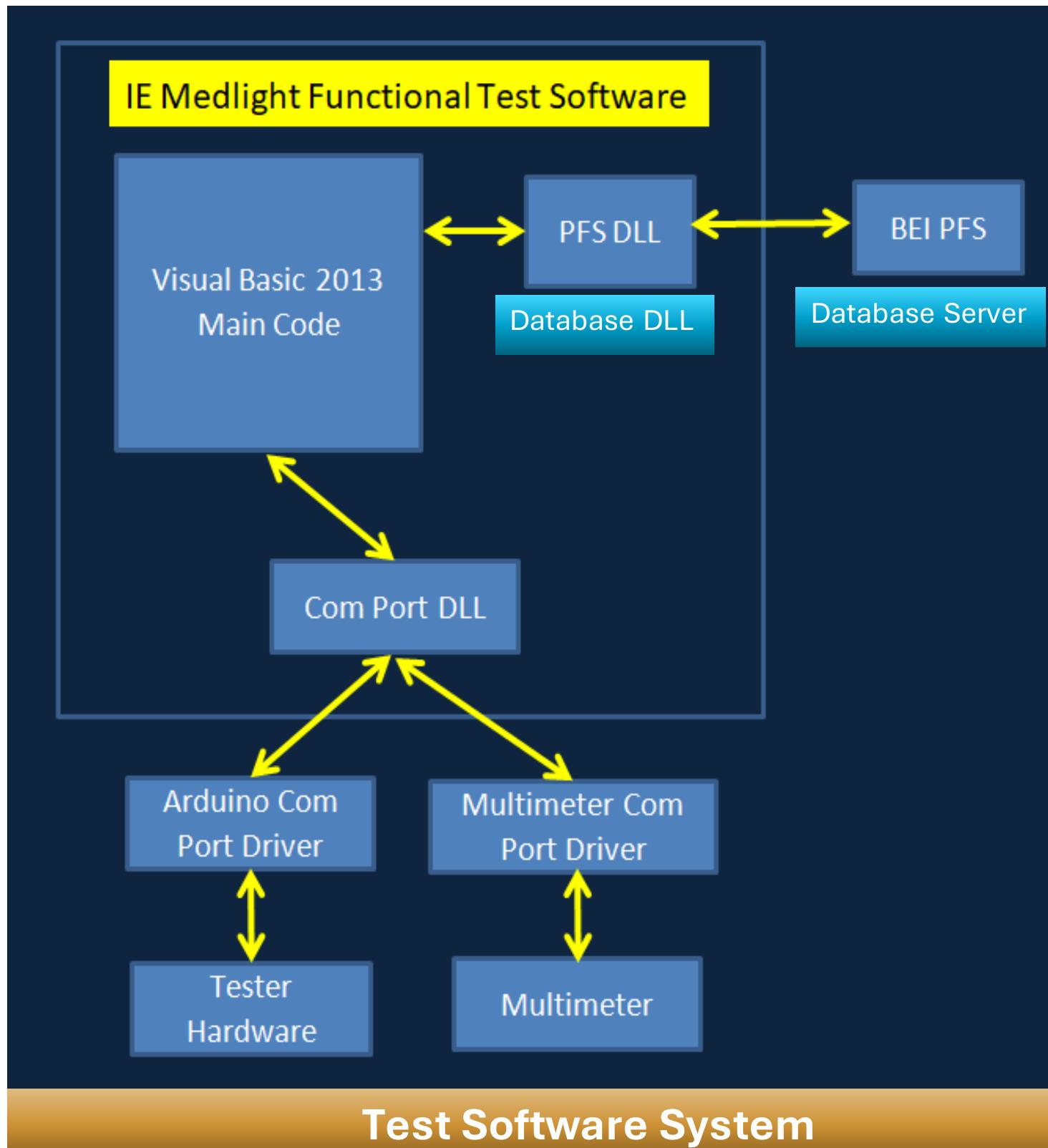
Head 1 Head 2

Serial Number	<input type="text"/>
PASS	
<button>Clear Error Head 1</button> <button>Clear Error Head 2</button>	

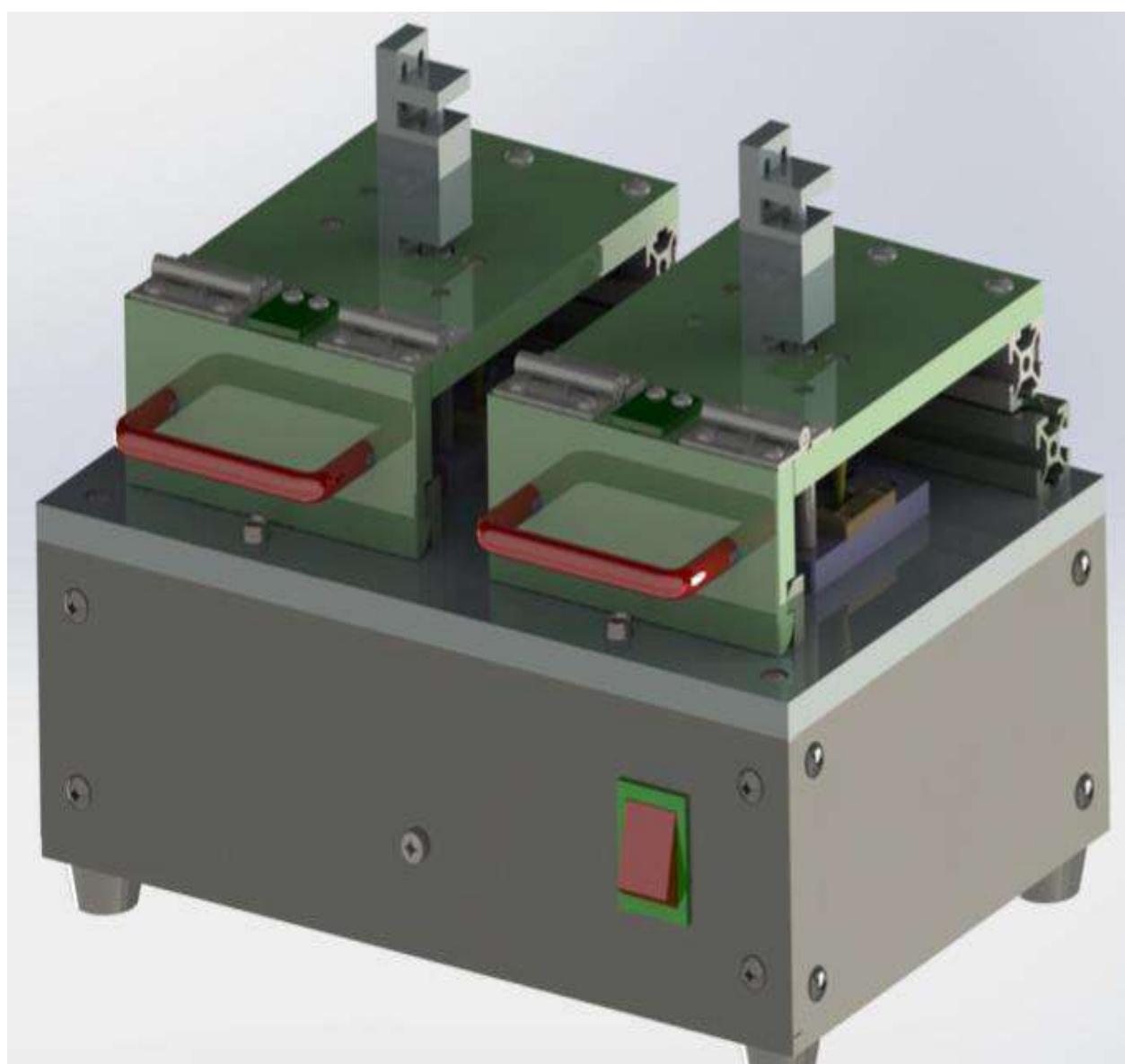
Update CSV File Start 16-May-2020 15:44:20.727
Create Summary Local Log_Local = C:\Temp_LOG\MedLight Functional Test_2563.csv
Create Summary Server Log_Local = C:\Temp_LOG\MedLight Functional Test_2563.csv
Update CSV File Done 16-May-2020 15:44:20.750
=====

16

User Interface

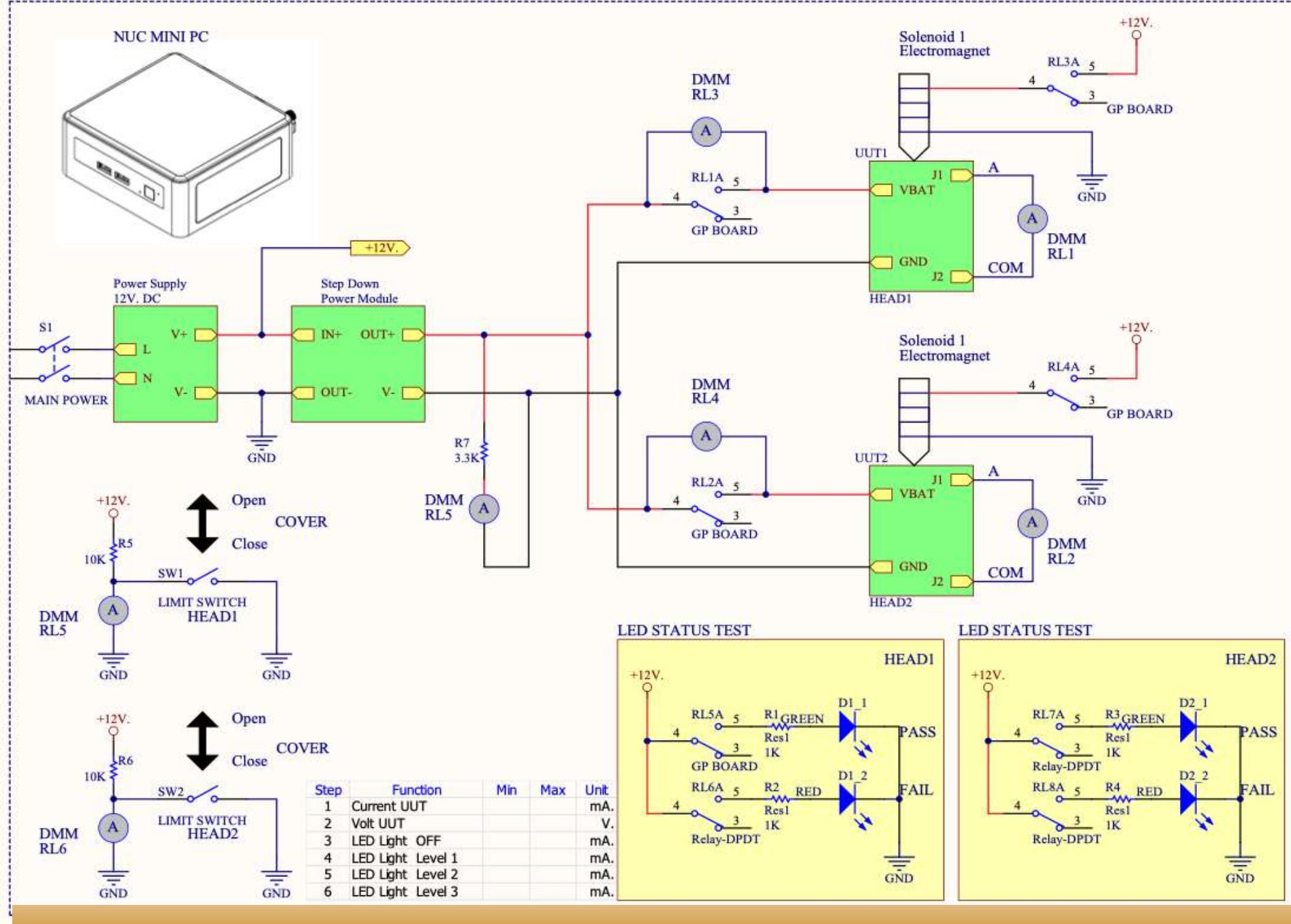


Test Software System



Test Fixture

IE MEDLIGHT FUNCTION TEST FIXTURE



Test Schematic

ht Function Test/ie_medlight

≡ Diagnostic.Designer.vb /Volumes/.../02 IE Medlight Function Test/ie_medlight

≡ Main.Designer.vb

Volumes > Untitled > Users > User > Desktop > My Own Business > Job > Benchmark > IE > 02 IE Medlight Function Test > ie_medlight > ie_n

```

1 Imports System.Data
2 Imports System.Data.SqlClient
3 Imports VB = Microsoft.VisualBasic
4 Imports System.IO.Ports
5 Imports System.Threading
6 Imports System.ComponentModel
7 Imports System
8 Public Class Main
9
10    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
11        ' Load Config.ini File
12        Variable.Dmm_comport = Utility.Readini("DMM", "Comport", "Can't Fine Config.ini", "\config.ini")
13        Variable.Dmm_Baud_Rate = Utility.Readini("DMM", "Baud_Rate", "Can't Fine Config.ini", "\config.ini")
14        Variable.Dmm_Data_bits = Utility.Readini("DMM", "Data_Bits", "Can't Fine Config.ini", "\config.ini")
15        Variable.Dmm_Parity = Utility.Readini("DMM", "Parity", "Can't Fine Config.ini", "\config.ini")
16
17        Variable.IO_Comport = Utility.Readini("IO", "Comport", "Can't Fine Config.ini", "\config.ini")
18        Variable.IO_RL1 = Utility.Readini("IO", "RL1", "Can't Fine Config.ini", "\config.ini")
19        Variable.IO_RL2 = Utility.Readini("IO", "RL2", "Can't Fine Config.ini", "\config.ini")
20        Variable.IO_RL3 = Utility.Readini("IO", "RL3", "Can't Fine Config.ini", "\config.ini")
21        Variable.IO_RL4 = Utility.Readini("IO", "RL4", "Can't Fine Config.ini", "\config.ini")
22        Variable.IO_RL5 = Utility.Readini("IO", "RL5", "Can't Fine Config.ini", "\config.ini")
23        Variable.IO_RL6 = Utility.Readini("IO", "RL6", "Can't Fine Config.ini", "\config.ini")
24        Variable.IO_RL7 = Utility.Readini("IO", "RL7", "Can't Fine Config.ini", "\config.ini")
25        Variable.IO_RL8 = Utility.Readini("IO", "RL8", "Can't Fine Config.ini", "\config.ini")
26
27    End Sub

```

Visual Basic

RelayConfig.h X **IO.ino**

D: > Benchmark > IE > 02 IE Medlight Function Test > Arduino > IO > **RelayConfig.h**

```

1  unsigned char relayPin[16] = {2,3,4,5,6,7,8,9,10,16,14,15,A0,A1,A2,A3};
2  int iTotalRelay = 16;
3  char InString[20]; // Allocate some space for the string
4  char inChar = -1; // Where to store the character read
5  byte Temp1, Temp2;
6  String inData;
7  int i;
8  //-----
9  void Relay_ON()
10 {
11     int i;
12     for(i = 0; i < iTotalRelay; i++) {
13         digitalWrite(relayPin[i],LOW);
14     }
15 }
16 }
17 //-----
18 void Relay_OFF()
19 {
20     int i;
21     for(i = 0; i < iTotalRelay; i++) {
22         digitalWrite(relayPin[i],HIGH);
23     }
24 }
25 //-----
26 void printMenu(void)
27 {
28     Serial.print("\n");
29     Serial.print(" My Arduino Version 1.00 !!!\n");
30     Serial.print("\n");
31 }
```

Arduino code (control relay)

RelayConfig.h X **IO.ino**

D: > Benchmark > IE > 02 IE Medlight Function Test > Arduino > IO > **IO.ino**

```

12 //-----
13 void loop(void) {
14     unsigned long currentMillis = millis();
15
16     while (Serial.available() > 0)
17     {
18         char received = Serial.read();
19         inData += received;
20
21         // Process message when new line character is received
22         if (received == '\n')
23         {
24             // Serial.print("\n");
25             // Serial.print("Arduino Received: ");
26             // Serial.print(inData);
27         }
28         if (inData == "r1 off\r\n") { digitalWrite(relayPin[0],HIGH); Serial.println("RELAY 1 OFF"); }
29         if (inData == "r1 on\r\n") { digitalWrite(relayPin[0],LOW); Serial.println("RELAY 1 ON"); }
30         if (inData == "r2 off\r\n") { digitalWrite(relayPin[1],HIGH); Serial.println("RELAY 2 OFF"); }
31         if (inData == "r2 on\r\n") { digitalWrite(relayPin[1],LOW); Serial.println("RELAY 2 ON"); }
32         if (inData == "r3 off\r\n") { digitalWrite(relayPin[2],HIGH); Serial.println("RELAY 3 OFF"); }
33         if (inData == "r3 on\r\n") { digitalWrite(relayPin[2],LOW); Serial.println("RELAY 3 ON"); }
34         if (inData == "r4 off\r\n") { digitalWrite(relayPin[3],HIGH); Serial.println("RELAY 4 OFF"); }
35         if (inData == "r4 on\r\n") { digitalWrite(relayPin[3],LOW); Serial.println("RELAY 4 ON"); }
36         if (inData == "r5 off\r\n") { digitalWrite(relayPin[4],HIGH); Serial.println("RELAY 5 OFF"); }
37         if (inData == "r5 on\r\n") { digitalWrite(relayPin[4],LOW); Serial.println("RELAY 5 ON"); }
38         if (inData == "r6 off\r\n") { digitalWrite(relayPin[5],HIGH); Serial.println("RELAY 6 OFF"); }
39         if (inData == "r6 on\r\n") { digitalWrite(relayPin[5],LOW); Serial.println("RELAY 6 ON"); }
40         if (inData == "r7 off\r\n") { digitalWrite(relayPin[6],HIGH); Serial.println("RELAY 7 OFF"); }
41         if (inData == "r7 on\r\n") { digitalWrite(relayPin[6],LOW); Serial.println("RELAY 7 ON"); }
42         if (inData == "r8 off\r\n") { digitalWrite(relayPin[7],HIGH); Serial.println("RELAY 8 OFF"); }
```

Arduino code (receive command from PC)

```

[SERVER]
----- PFS Server
PFS Server = pfs-gw-thp4.corp.bench.com
PFS Database = pfsthp4
PFS Port = 41123
----- PFS Option
PFS Enable = no ; yes/no
PFS Enable Confirm Send Fail = yes ; yes/no
PFS Control Certify = yes ; yes/no
PFS_COMMENT = HISTORY_COMMENT
PFS OperationCode = TLA_FCT1
PFS Work Center = TLAFACT

-----
[SW Option]
Log Local Folder = C:\Temp_LOG\
Log Server Folder = C:\Temp_LOG\
S/N Format = B#####

[ EQUIPMENT ]
Arduino Com Port = 3
Comport Setting = 9600,n,8,1
Comport TimeOutReceive = 1000 ;Time out read data(second)
Comport RThreshold = 1

[ DMM ]
Multimeter Comport = 8

[TEST SPEC]

currentUUT_lowLimit = -2 ;mA
currentUUT_highLimit = 2 ;mA

voltUUT_lowLimit = 1.2 ;V
voltUUT_highLimit = 1.7 ;V

```

User Config File

Test Name	VoltUUT
Test Description	Measure voltage that apply to UUT
High Limit	1.7 V
Low Limit	1.2 V
Step1	Connect DMM at RL5
Step2	Read current from DMM and convert to voltage
Step3	Disconnect DMM at RL5

Test Name	currentUUT
Test Description	Measure current that flow into UUT when output off
High Limit	2 mA
Low Limit	-2mA
Step1	Connect DMM at RL3 (head1) or RL4 (head2)
Step2	Read current from DMM
Step3	On relay RL1A(head1) or RL2A(head2) to apply power to UUT for next test
Step4	Disconnect DMM at RL3 (head1) or RL4 (head2)

Test Name	LEDOFF
Test Description	Measure current at UUT 's LED drive pin when output off
High Limit	2 mA
Low Limit	-2mA
Step1	Connect DMM at RL1 (head1) or RL2 (head2)
Step2	Read current from DMM

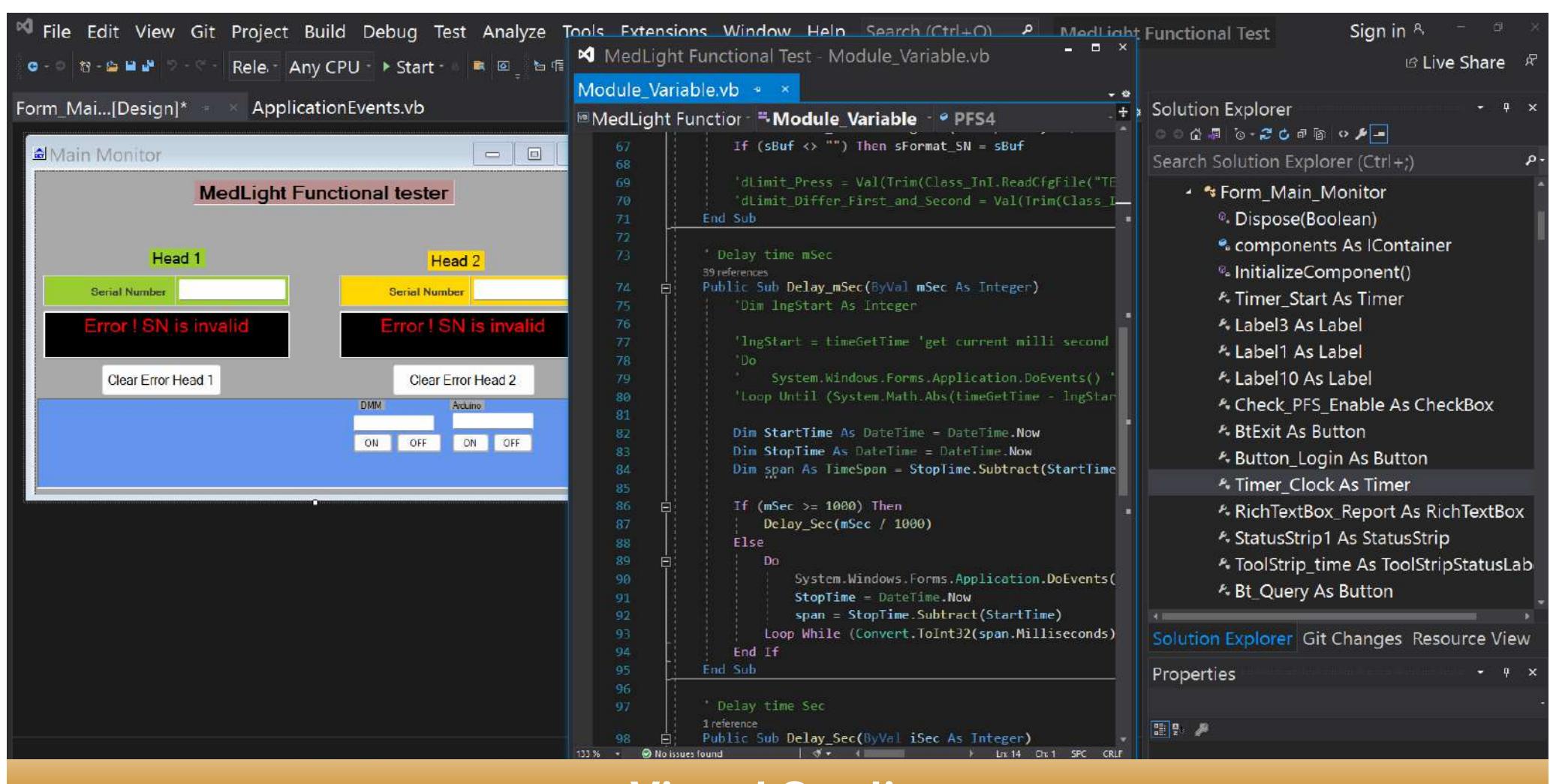
Test Plan

Test Name	LEDlv1
Test Description	Measure current that flow into UUT when output current is lv1
High Limit	40 mA
Low Limit	20 mA
Step1	Turn on relay at RL3A (head1) or RL4A (head2) to push the button
Step2	Turn off relay at RL3A (head1) or RL4A (head2)
Step3	Read current from DMM

Test Name	voltOut_lv1
Test Description	Measure voltage at UUT's output pin when output current is lv1
High Limit	4 V
Low Limit	3 V
Step1	On relay RL6A(head1) or RL8A(head2)
Step2	Disconnect DMM at RL1 (head1) or RL2 (head2)
Step3	On relay RL1A(head1) or RL2A(head2)
Step4	Disconnect DMM at RL3 (head1) or RL4 (head2)
Step5	Read current from DMM then converrt to voltage
Step6	Disconnect DMM at RL1 (head1) or RL2 (head2)

Test Name	LEDlv2
Test Description	Measure current that flow into UUT when output current is lv1
High Limit	70 mA
Low Limit	50 mA
Step1	Turn on relay at RL3A (head1) or RL4A (head2) to push the button
Step2	Turn off relay at RL3A (head1) or RL4A (head2)
Step3	Read current from DMM

Test Plan



Visual Studio

Automated Optical Inspection (Medical Module, Trimble)

Software Tools and Programming Language : Labview, NI Vision Builder

- Build to operate at Benchmark China
- Capture unit image from camera
- Visual inspect for label presence, read 1D/2D barcode, logo, glue presence
- Measure distant from label to other object edge
- Check mismatch and duplicate serial number with database server
- Read part number text and serial number text then compare with barcode(OCR)
- Check connector pin count
- Save test result to database and local
- Save image log to local



USER: Pfstesterlink , W/O: xxxxxx

PfsSendResults Failure:
Production order 'XXXXXX' was not found.

Auto Program Selection : Program 2

Step	Condition	Result	Status
Acquire Image (1394, GigE, or Vision Assistant 2		Pass	Frame Index = 0
Convert to 8bit grayscale		Pass	
Symmetry		Pass	
Rotate 180 degree		Pass	
FC Sign Reference		Pass	# Matches = 1
FC Coordinate		Pass	
Model Text Highlight		Pass	
Read Model		Pass	
Custom Overlay 1		Pass	
Calculate Program To Use		Pass	Model name = 926:
Check_board		Pass	
Set Variable 1		Pass	Program = 2
Program To Use		Pass	Program_To_Use =
Set Inspection Status 1		Pass	Inspection Status =
PFS connect		Pass	
PFS Read Info		Pass	Database = pfsthp
Set Variable Pfs		Pass	OpCode = AOI3
PFS_SEND_PACKAGE		Pass	Data_Send = REQL
Set Variable 1		Pass	Work_Order = xxxx
Set Variable 2		Pass	Loop = 1
User Input 1		Pass	OK Pressed
PFS Sending		Pass	
PFS_SEND		Pass	Data_Send = REQL
Calculator 1		Pass	PFS_STRING_OUT
Custom Overlay 1		Pass	
Light On		Pass	
Light Turn ON		Pass	standard output =
wait_light_turn_on		Pass	Delay = 500 (ms)
set_Light_Status		Pass	Light_Status = True
Online/Offline		Pass	
Select online/offline		Pass	Yes Pressed
Offline mode		Pass	
Set Variable 1		Pass	
check_first_time		Pass	Count_Retest = 0
Set Variable 1		Pass	OK Pressed
User Input 1		Pass	
Logging		Pass	
Copy of Custom Overlay 2		Pass	
Set Variable 1		Pass	
Model manual key		Pass	
Manual_model_key		Pass	
Calculate Program Manual Key		Pass	
Set Variable 1		Pass	
Model not match		Pass	
Model Not Match		Pass	
Model3		Pass	
Select Image 1		Pass	
Read ID Barcode 1		Pass	
OCR Read Serial Number		Pass	
Manual Key SN		Pass	
Compare SN Barcode And OCF		Pass	
Vision Assistant 1		Pass	
sign10		Pass	
Set Inspection Status 1		Pass	
Copy (3) of Set Variable 1		Pass	

Display All Images 2592x1944 0.41X 8-bit image 6 (0,0)

Active vs. Idle Time (s.)

Yield (%) 100.00

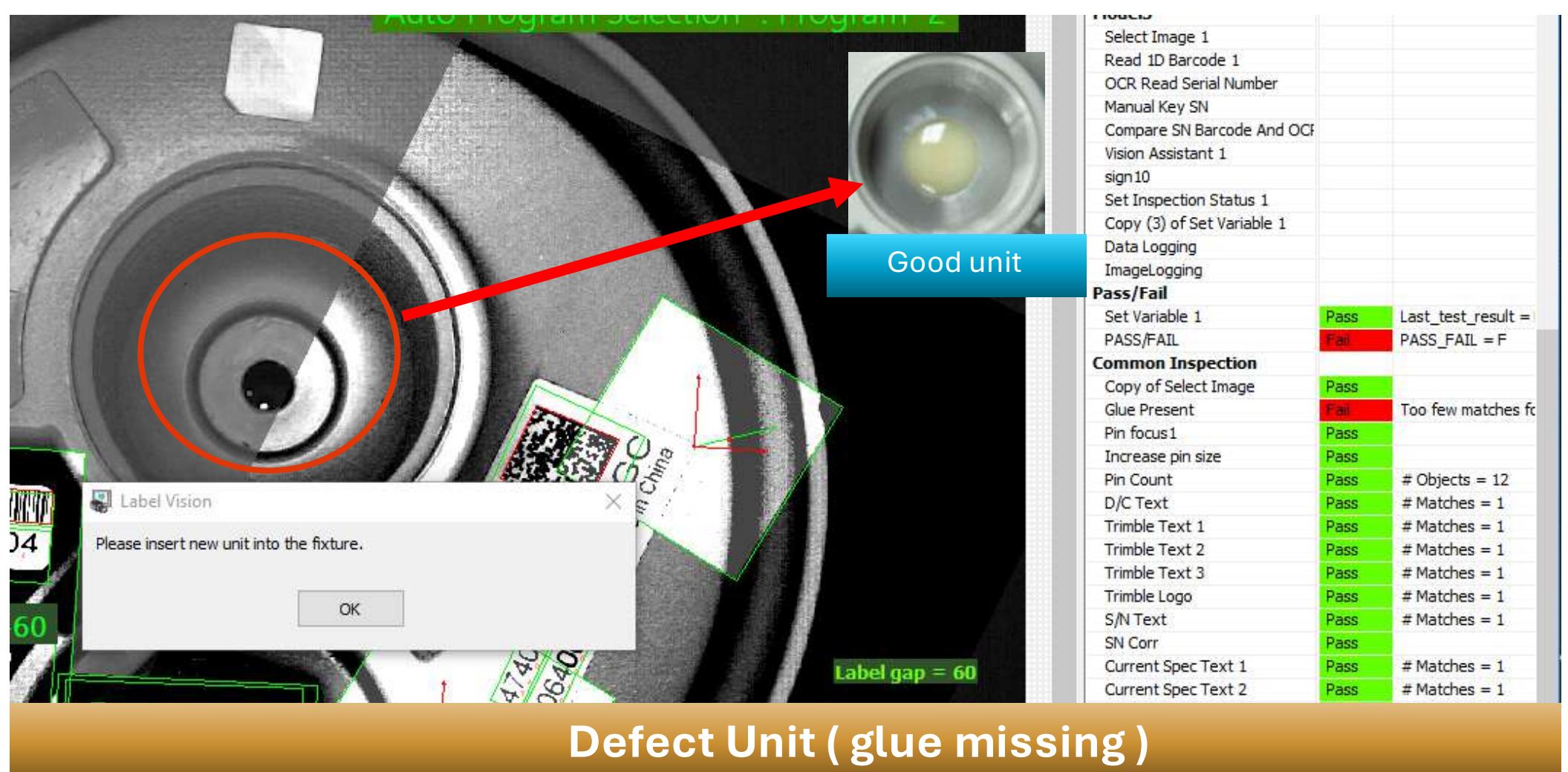
Pass 1 # Fail 0

Distribution of iteration time for the last 25 inspection iterations Parts/s

0.00

50 -

User Interface



Defect Unit (glue missing)



W A Y B X X X - 0 0 X X X X X

Revision:

A

1. PURPOSE

The purpose of this document describe how to operate and maintenance Trimble AOI software.

2. SCOPE

- 2.1** This document establishes software/tool detail and requirement of installation software for new Trimble AOI tester.

3. REFERENCE DOCUMENTS AND FORMS

4. EQUIPMENT

HARDWARE REQUIREMENT

- 4.1** Trimble AOI Tester
4.2 Dell Computer with MAC address 8C-EC-4B-6F-8C-DB

Software Requirement

- 4.3** Microsoft Window 10 x64bit
4.4 NI Software
- 4.4.1** Measurement & Automation Explorer 17.0.0
 - 4.4.2** NI Vision Builder AI 2014
 - 4.4.3** NI-VISA Version 16.0
 - 4.4.4** NI-488.2 16.0
 - 4.4.5** NI-DAQmx 17.1.0
 - 4.4.6** NI LabVIEW Runtime 2014 SP1 f11
 - 4.4.7** NI-IMAQ 16.0
 - 4.4.8** NI-IMAQ Runtime 16.0
 - 4.4.9** NI Vision Builder Runtime license, use with PC in 4.2 only
 - 4.4.10** IDS uEye Camera Driver (Full) For Window 64 bit

5. SAFETY

- 5.1** The PC must have antivirus software. Window defender security center already installed.

Manual Document



W A Y B X X X - O O X X X X X X X

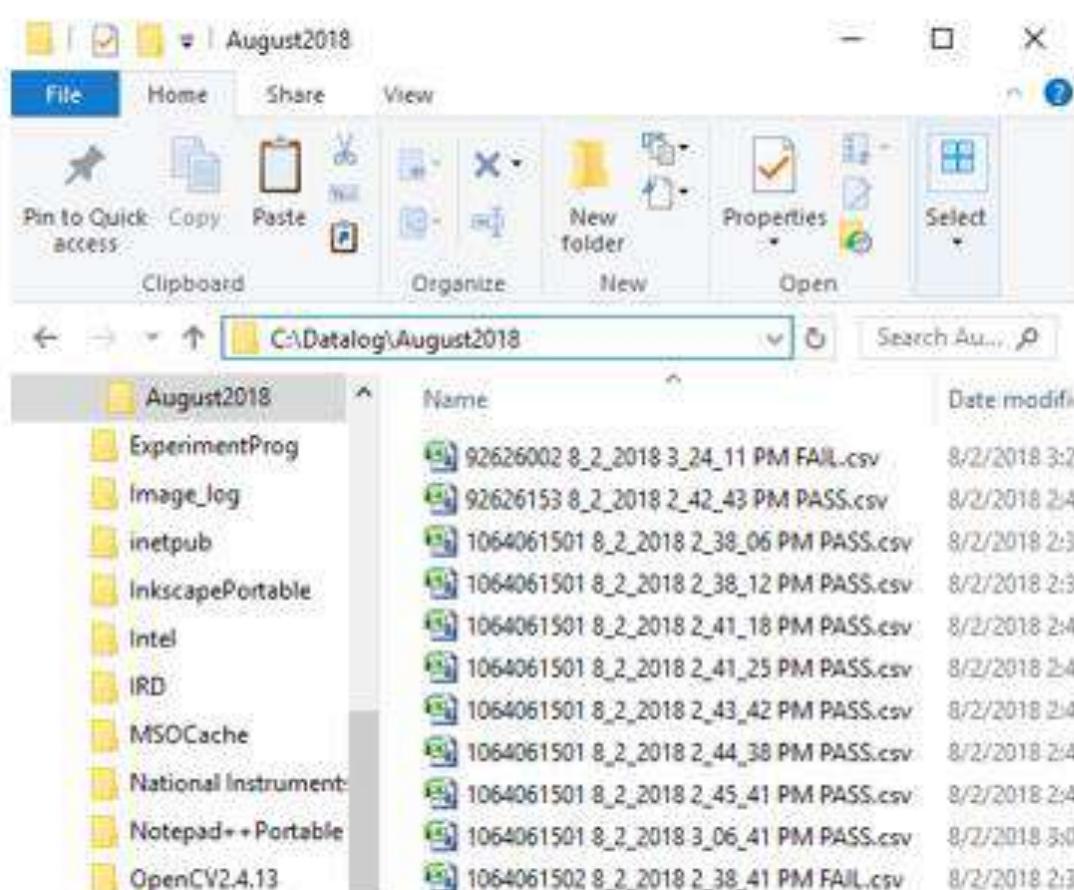
Revision:

A

8.2 Software Maintenance

8.2.1 Log Files

The datalog will keep in C:\Datalog\MonthYear .Month and year will change automatically .The file name will begin with serial number follow by date time and pass/fail result.



File type is .csv .The datalog will look like image below .

A	B	C	D	E	F	G	H	I	J
State Name				Model_Selectic	Model_Selectic	Common Inspe	Common Inspe	Common Inspe	Common Inspe
Step Name				Read Model	Program To Use D/C Text		Trimble Text 1	Trimble Text 2	
Result Name				Read Text	Step Result	Step Status	Step Status	Step Status	Step Status
Units									
Iteration	Date	Time	Time (mil)	Status					
2	8/2/2018	2:41:25 PM	65	Pass	106406-15	TRUE	Pass	Pass	Pass

Example :

- Logo Inspection → Trimble logo, D/C and FC



Logo Inspection

Example:

- 1D Bar code Reader

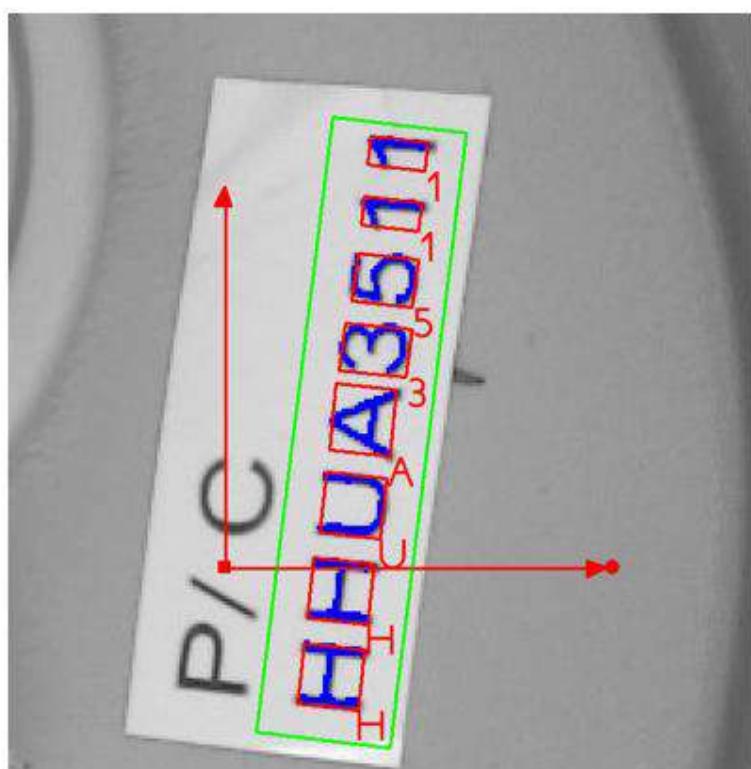
Barcode Reading



Character Reading (OCR)

Capabilities:

- Optical character recognition (OCR)



Compare SN text with barcode

Capabilities:

- Mismatch and duplicate serial number.

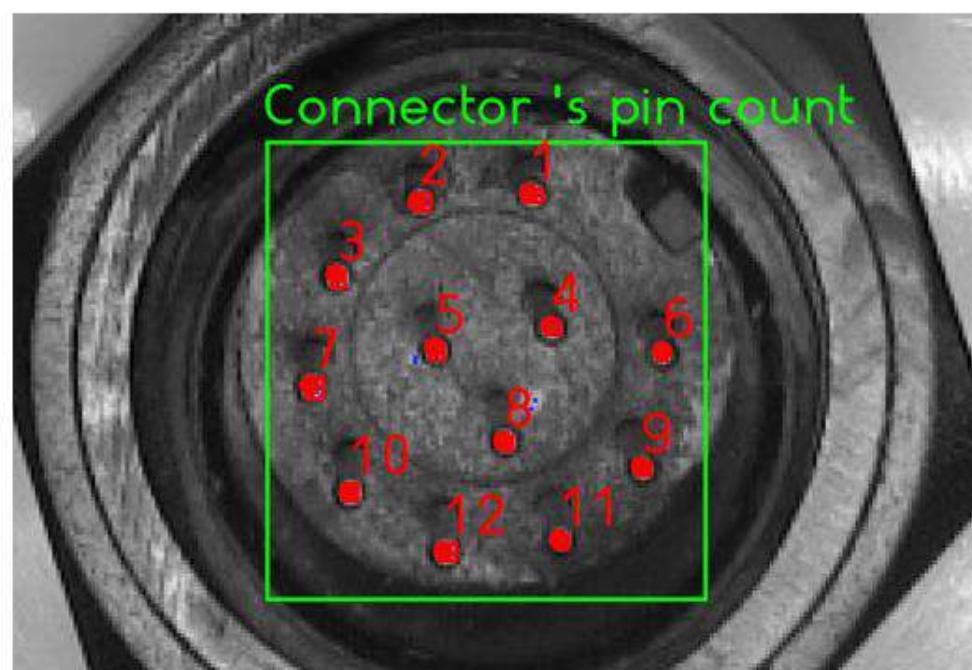


Enable to compare SN OCR with SN barcode

Connector Pin Count

Capabilities:

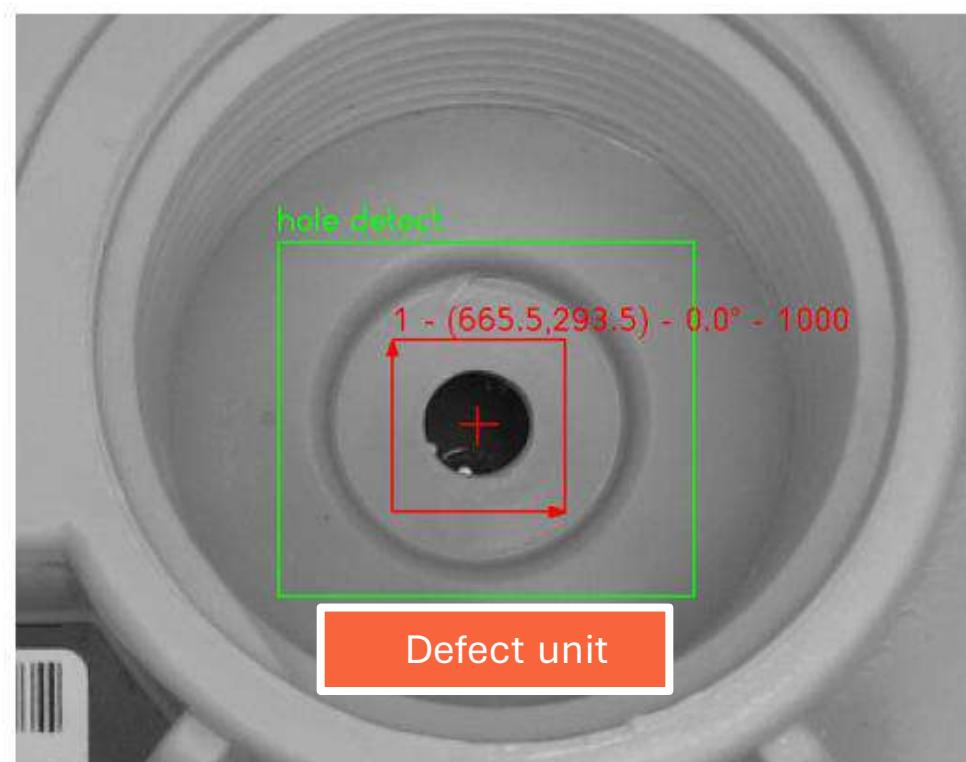
- Contact pins count → The contact pins count must be 12 pins



Check Glue Presence

Capabilities:

- Yellow glue detection → Must be present the yellow glue

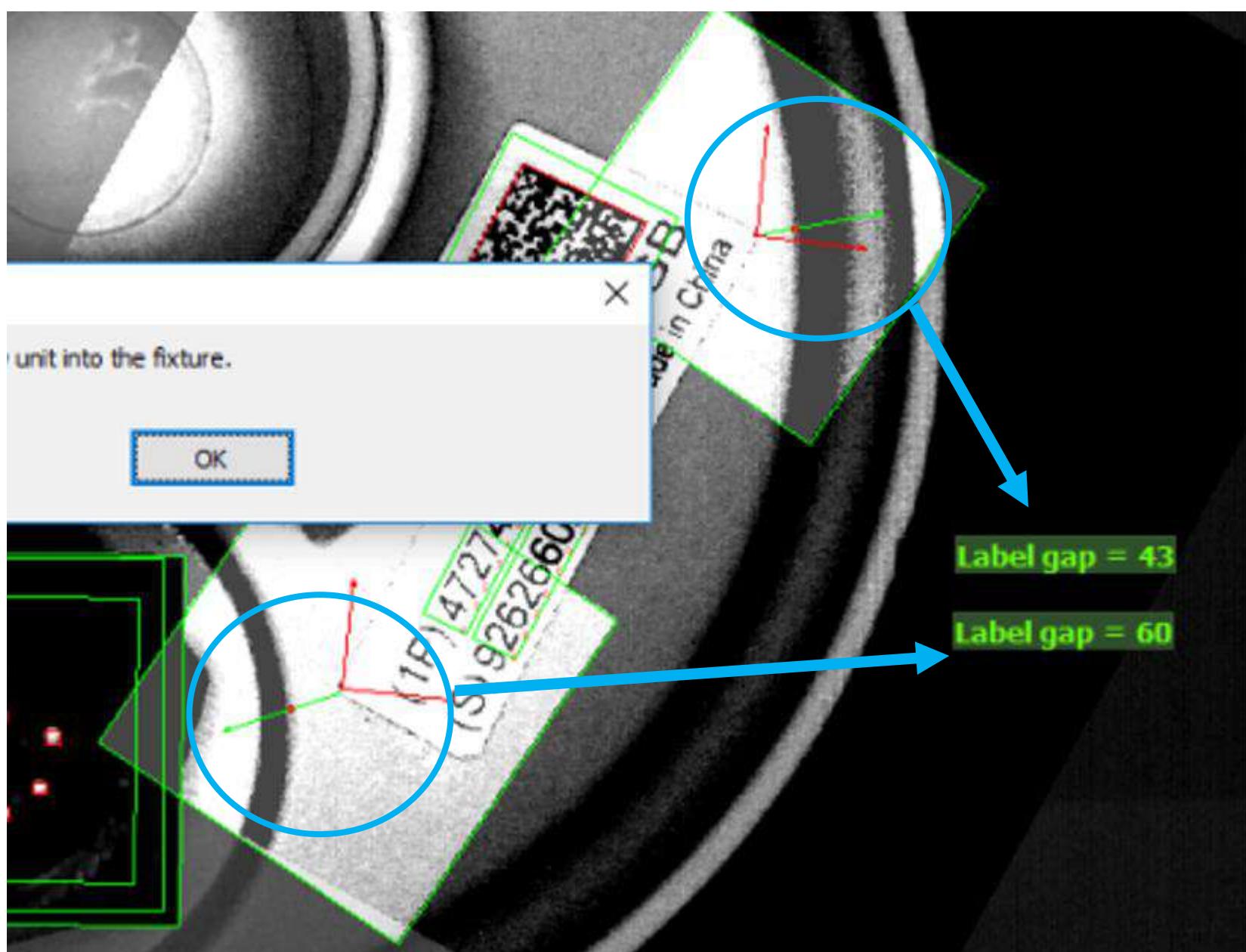


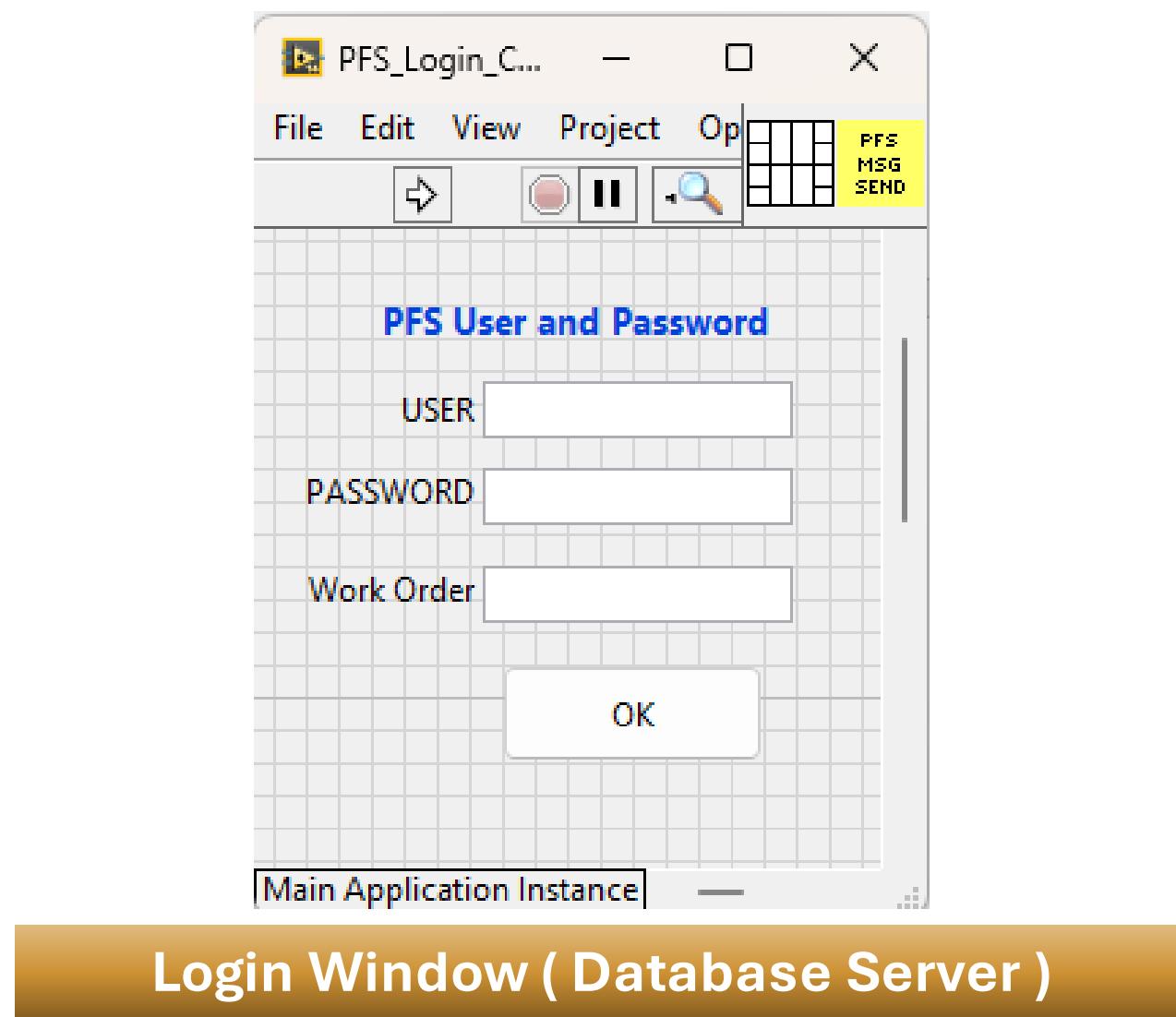
Capabilities:

- P/C Label position → Need to discuss with BEC, how to measure the distance 1-3 mm.

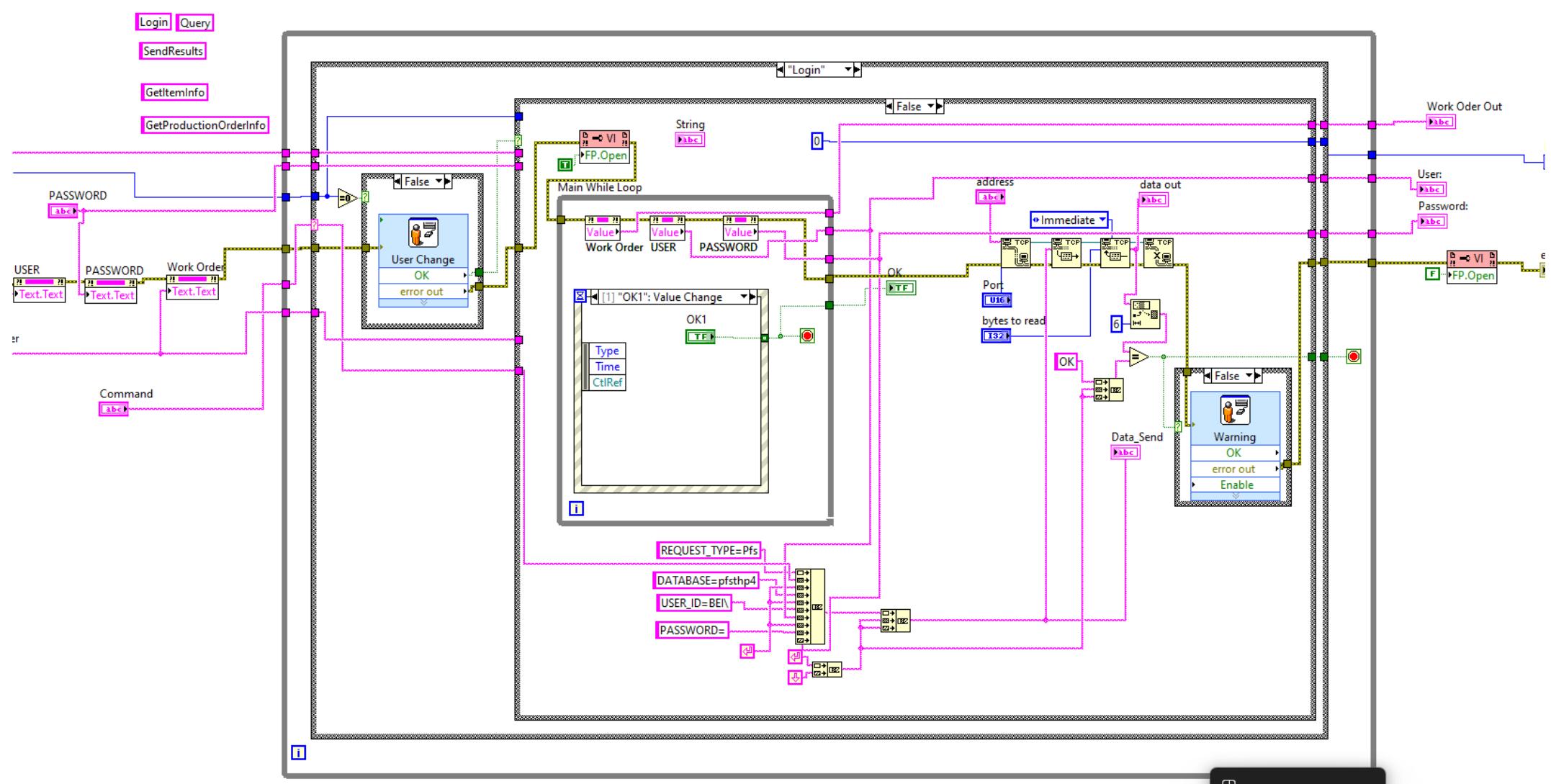


Measure Distance To Verify Label Position





Login Window (Database Server)



Login Window Labview Code

Automated Optical Inspection (SSD, Seagate)

Software Tools and Programming Language : Labview, NI Vision Builder

- Capture unit image from camera
- Visual inspect for label presence, read 1D/2D barcode, logo
- Check mismatch and duplicate serial number with database server
- Read part number text and serial number text then compare with barcode(OCR)
- Save test result to database and local
- Save image log to local



Product



User Interface

USER: pfstesterlink , W/O: TMN153322 S/N: ZE601SV6

FAILED

PfsSendResults Failure:

Serial number ZE601SV6 is not on this production order.

It is on production order(s) TMN154248.



Fail – This SN Is Not In This Production Order

State: Inspect

Display Result Image for this State

Buttons: Select Image (PASS), Match Pattern 1 (FAIL), Pre_ref, Mid1_ref, Find_Angle2, Mid1, Barcode_SN, Barcode_SN#2, SN ID Barcode, Calculator 1, Barcode_PN, Barcode_PN#2, IMAQ Match Pattern 4.

Inspection Status: PASS

State: Inspect

Display Result Image for this State

Buttons: PN 1D Barcode, Hi-Light, Read/Verify_SN, SN OCR Read, Sign1, Sign2, Sign3, Sign4, Sign5, Sign6, Sign7, Sign8.

Inspection Status: PASS

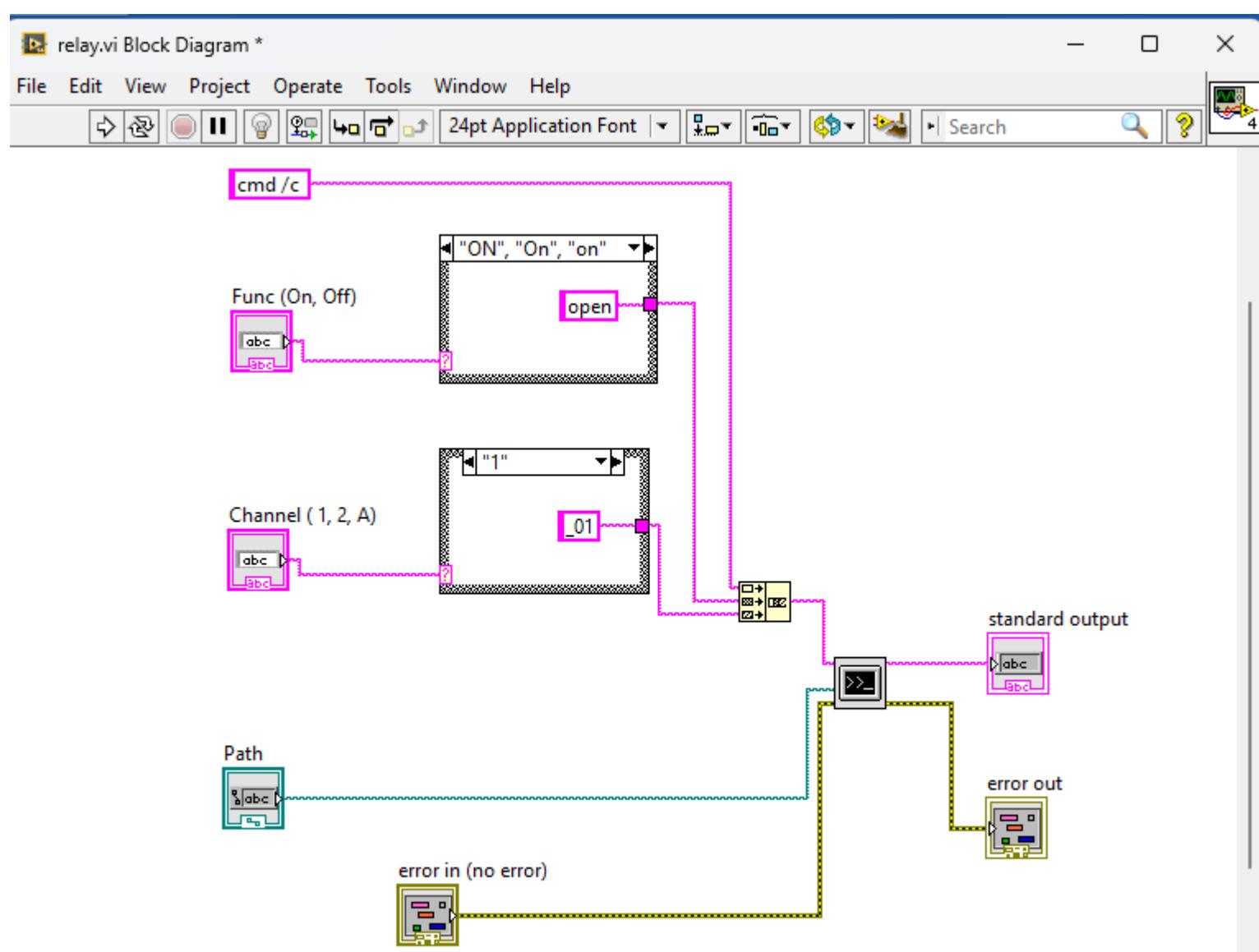
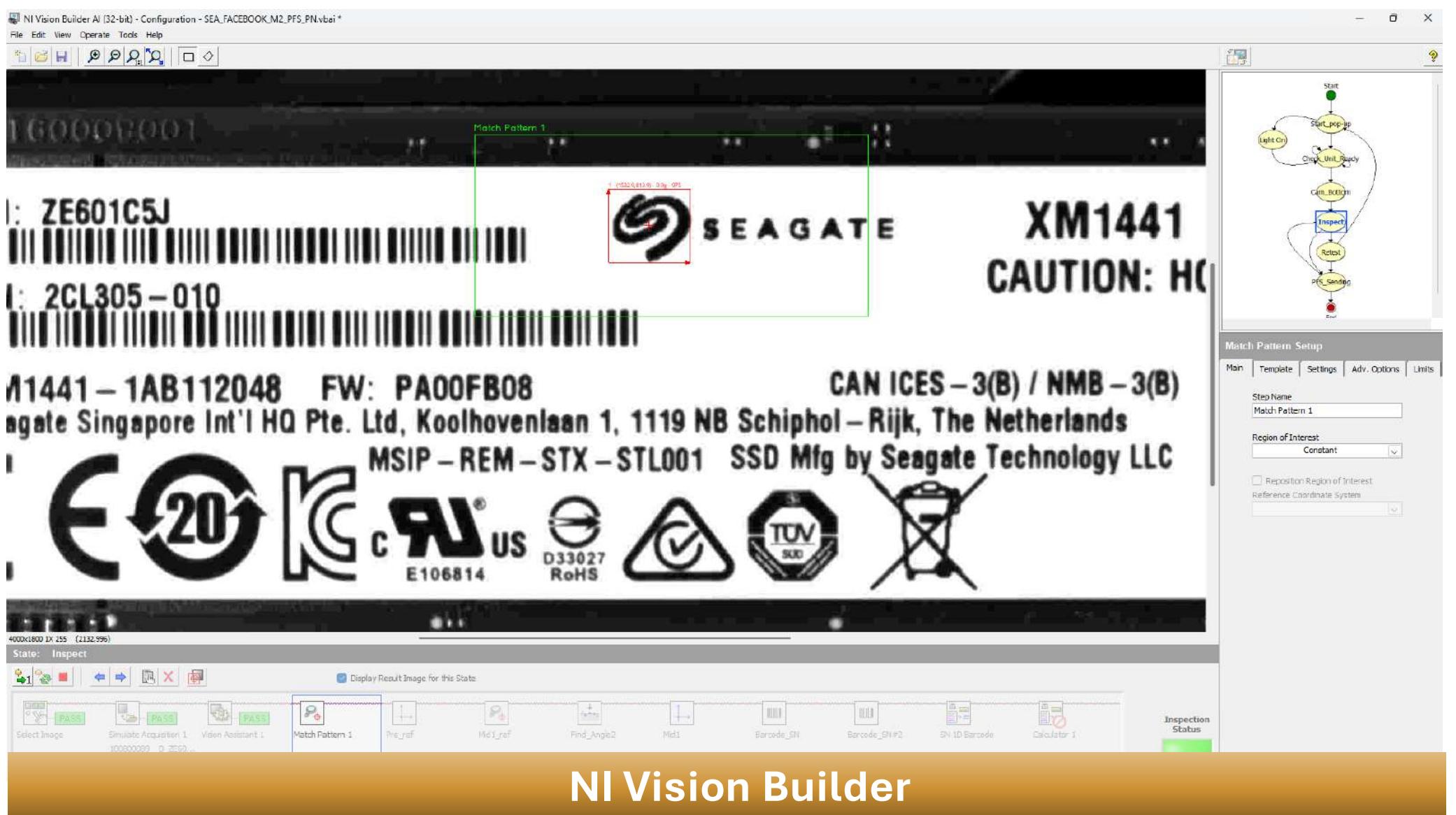
State: Inspect

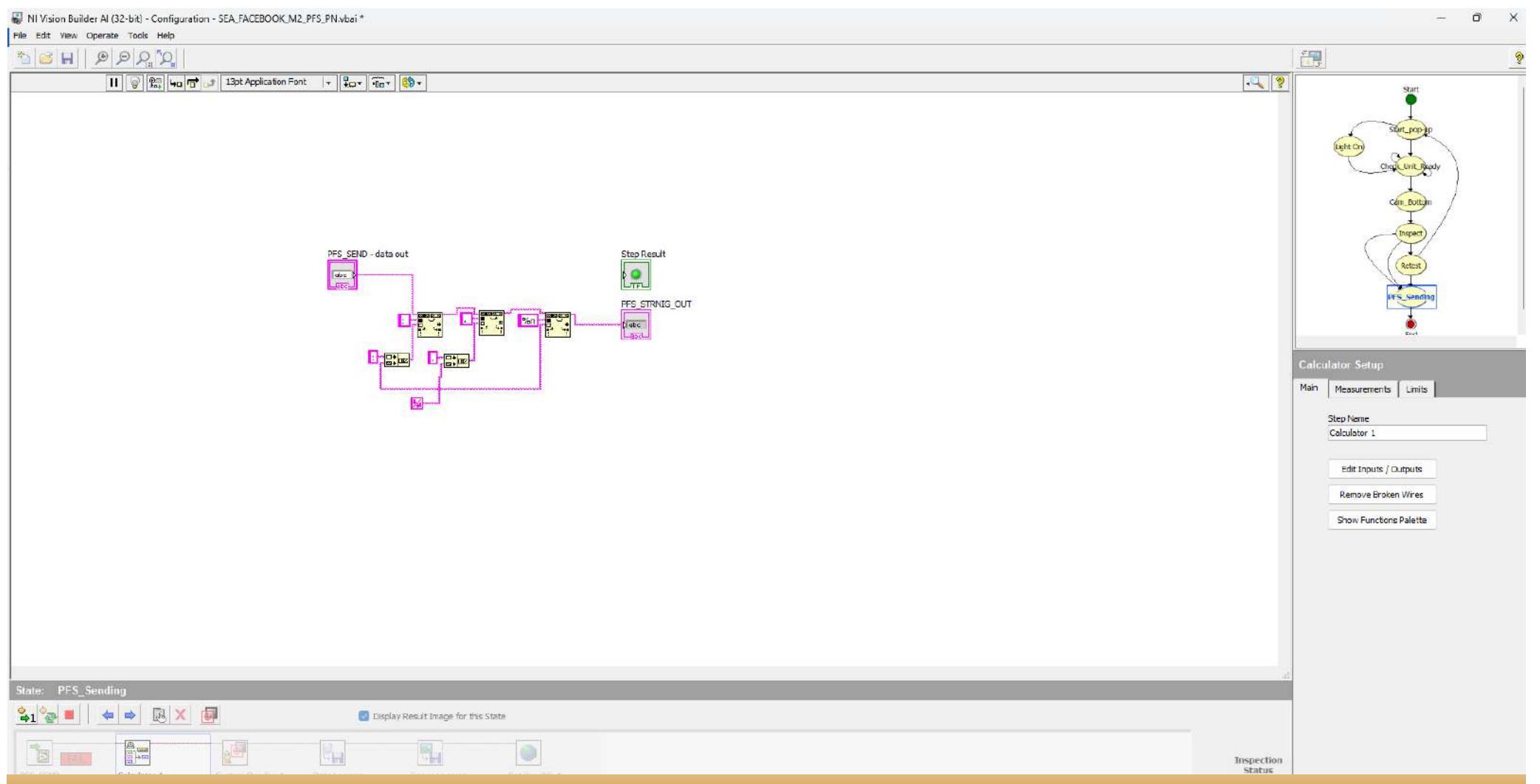
Display Result Image for this State

Buttons: Sign9, Sign Check Result, Select Image 1, Match Pattern 3, Find Straight Edge 1, Set Coordinate Syste..., SN 2D Barcode, 2D Barcode ID, Match Pattern 2, Set Coordinate Syste..., Facebook 2D Barcode, Comp_SN.

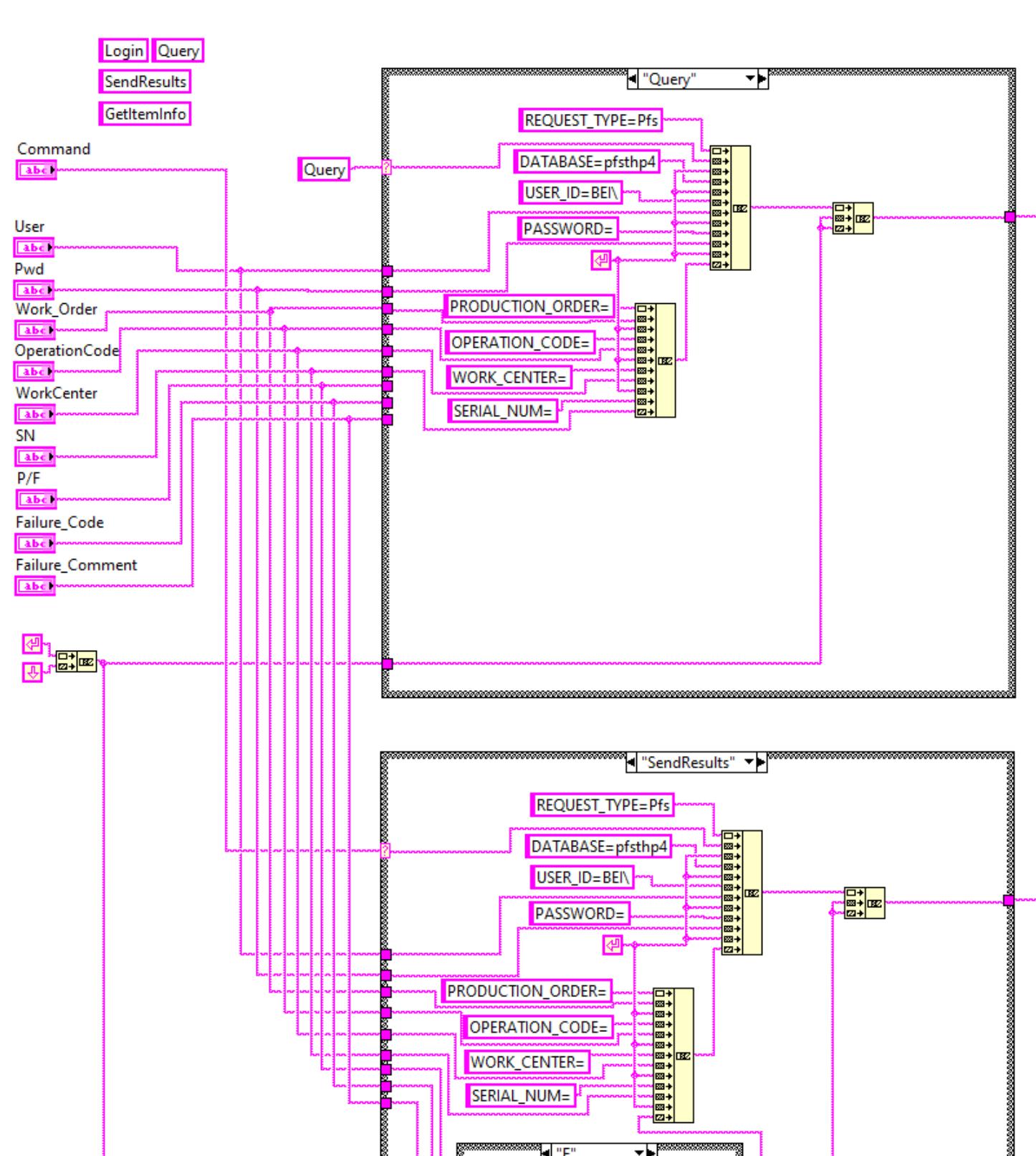
Inspection Status: PASS

Test Step In Vision Builder





Labview Code In Vision Builder

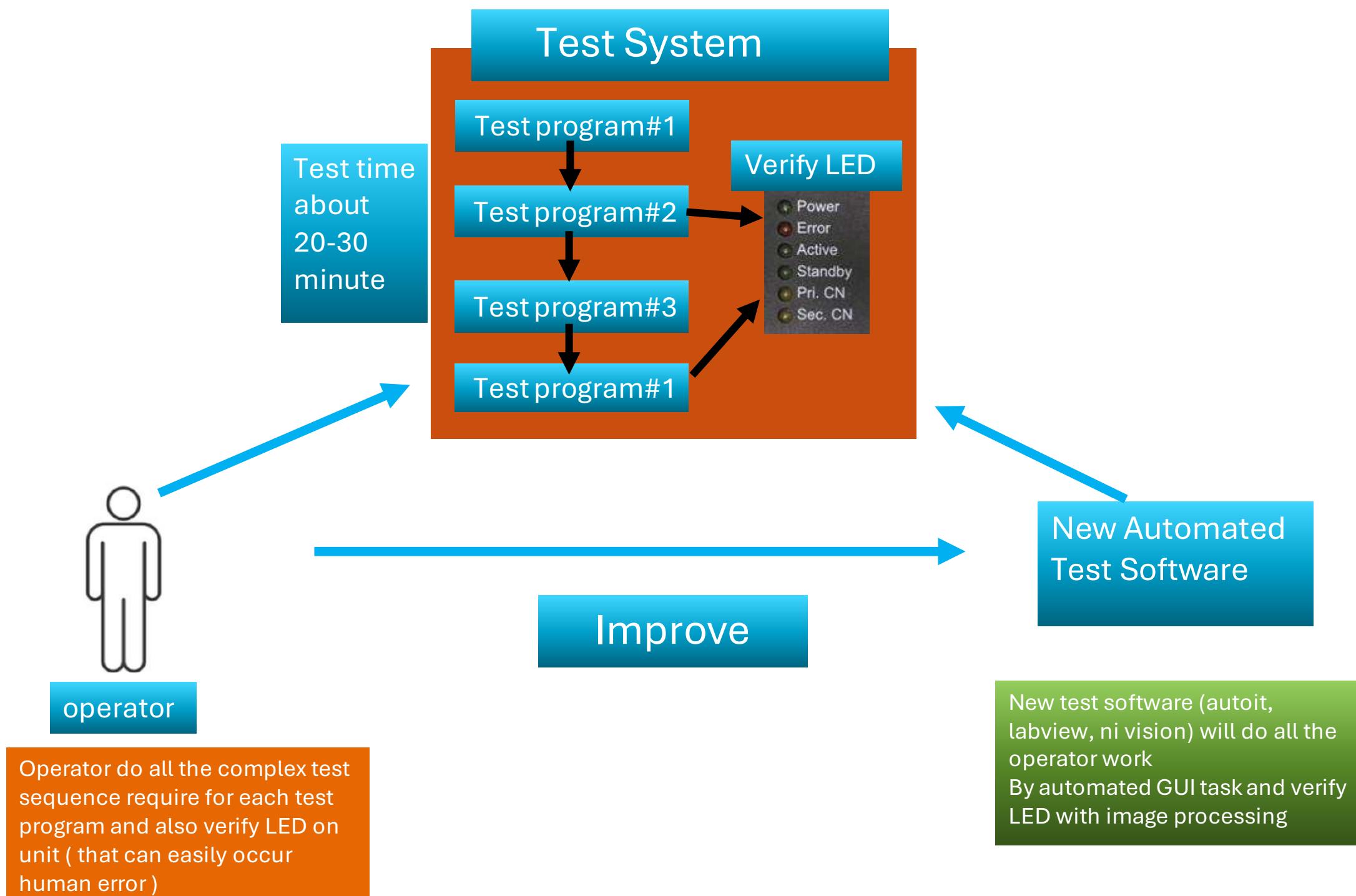


Labview Code (Communicate With Database Server)

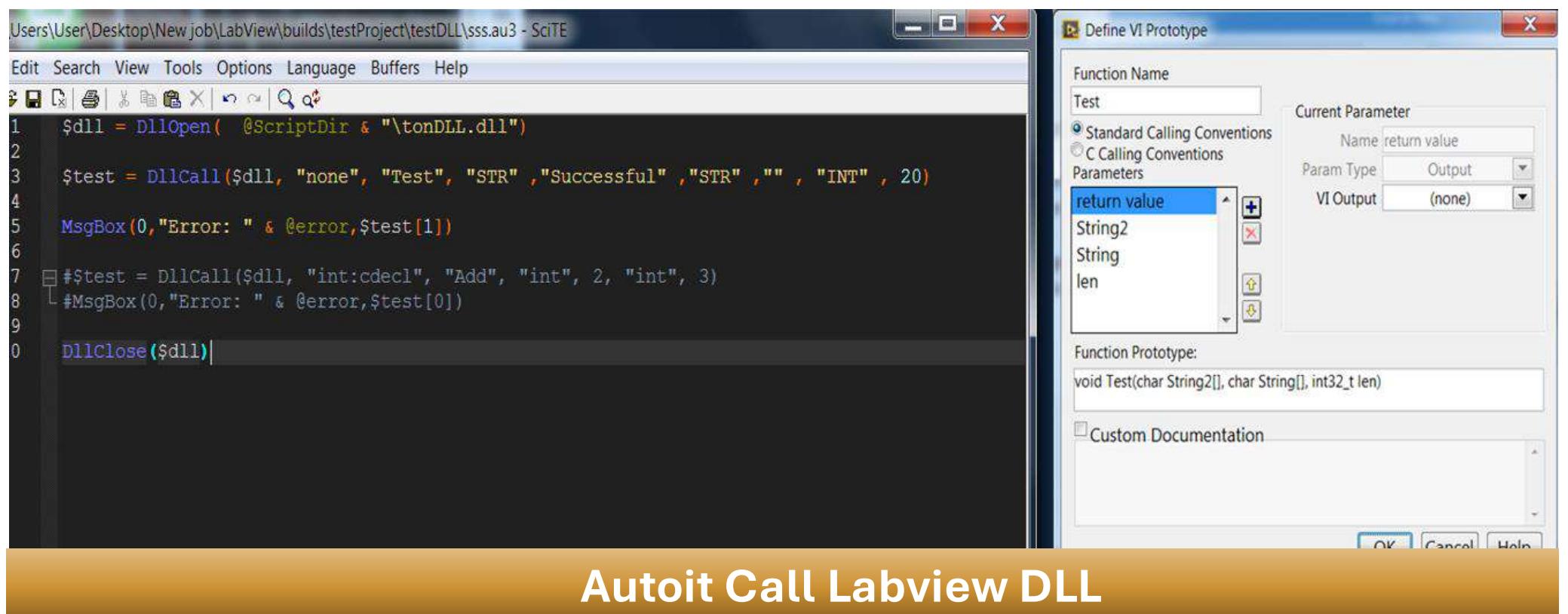
Test Automation Improvement (Network Module, Emerson)

Software Tools and Programming Language : Autoit, Labview, NI Vision Builder

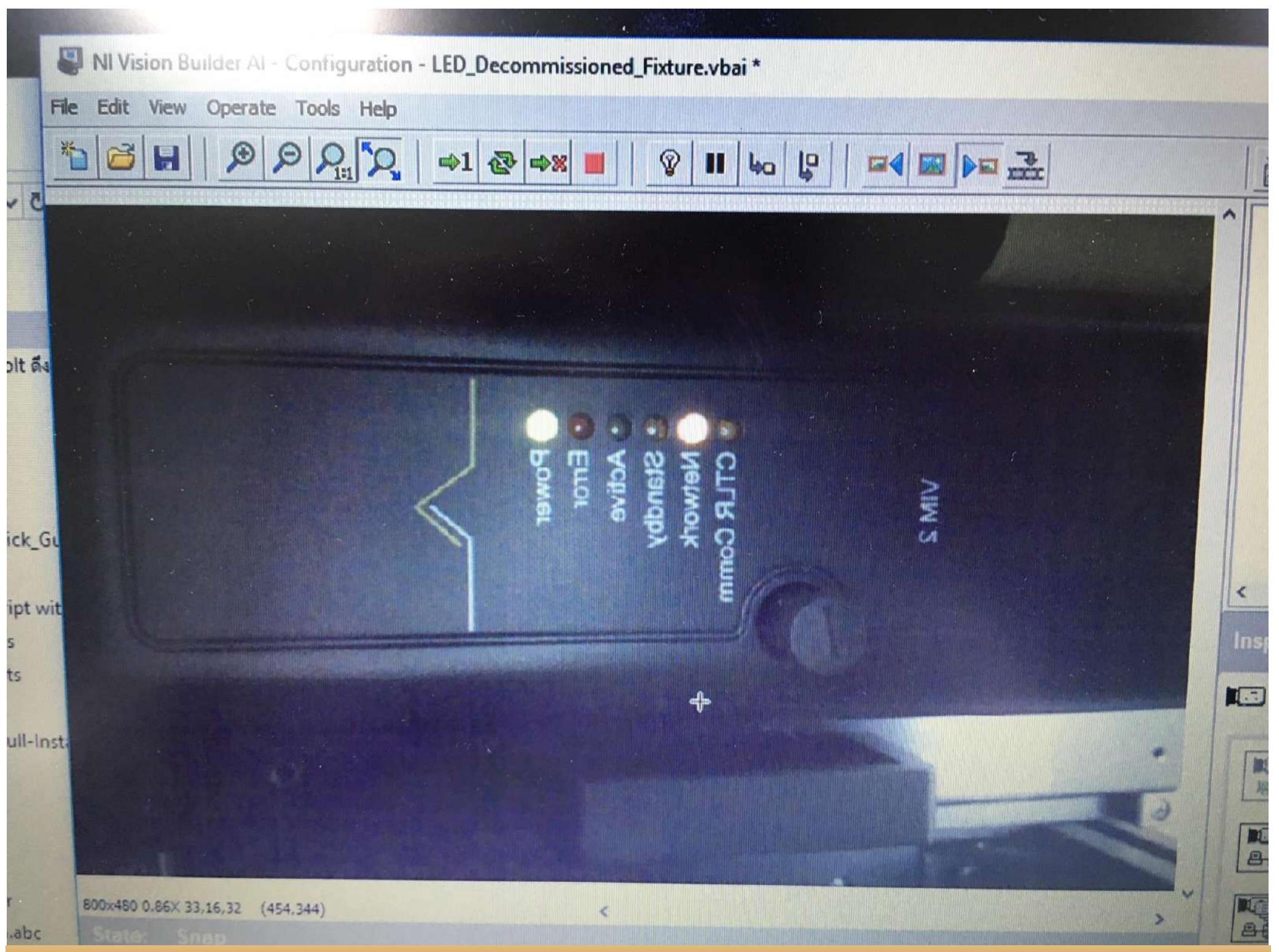
- Use autoit for automating the Windows GUI task (replace operator work)
- Use Labview, NI vision builder, NI vision development to verify LED pattern (replace operator work)
- Reduce human error (complex test sequence, long test time about 20 minutes)
- Reduce human error from vision inspection







Autoit Call Labview DLL



NI Vision Builder

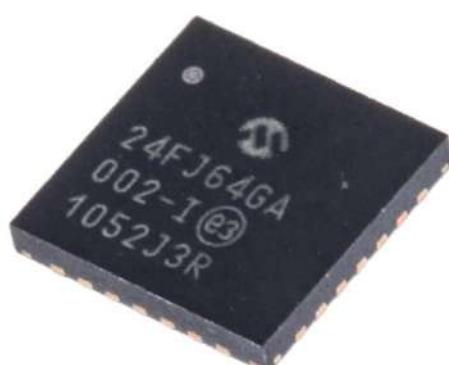
Test Program For Semiconductor (MCU, ICs, Microchip, UTAC, Toshiba)

Software Tools and Programming Language : Visual C++, Visual Basic 6, Visual ATE, IG-XL

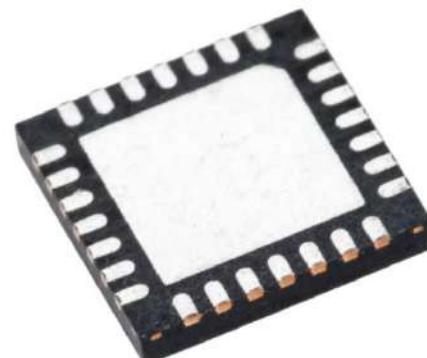
- Teradyne J750 Tester, Credence TMT ASL-1000 Tester
- UF3000 prober for wafer test (room, hot and cold test)
- Multitest MT9928 for MSOP, SOIC, QFN (room, hot and cold test)
- Aetrium 5050 for SOIC, QSOP (room test)
- MCT SH-5300 for strip test (room, hot and cold test)
- RASCO-1000 for SOIC, SSOP, TSSOP, MSOP, and SOT (room, hot and cold test)
- Temptronic thermostream for temperature setting for hot and cold manual test
- Loadboard design



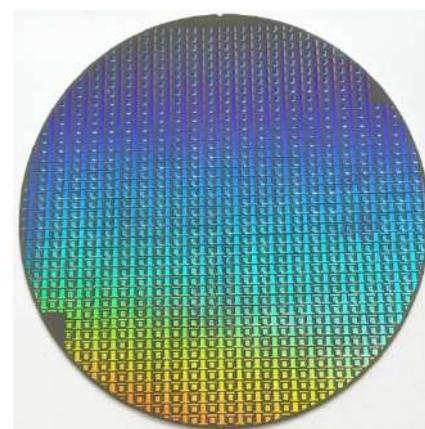
TQFP packages



QFN packages



SOIC packages



Wafer

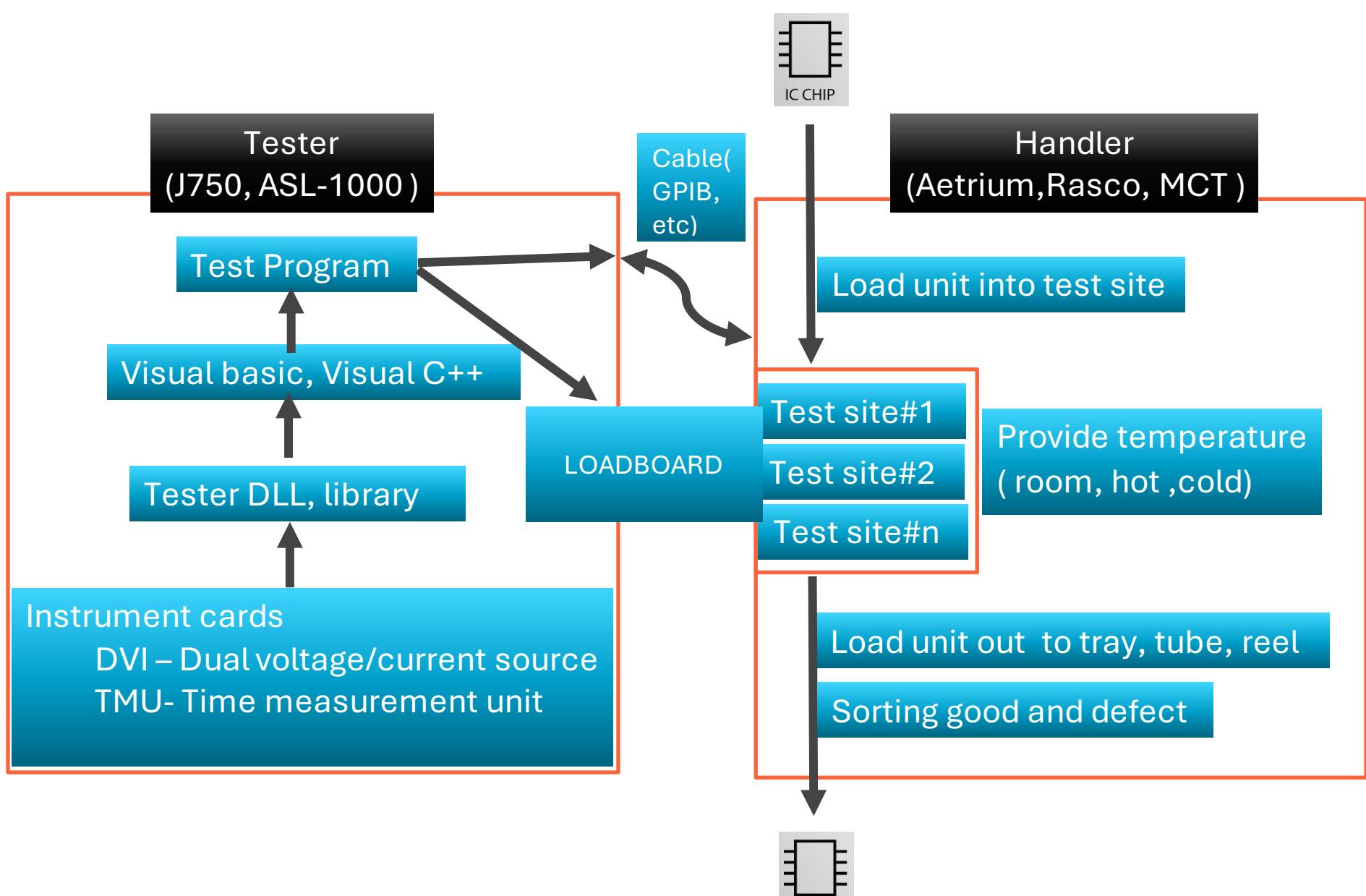


strip

16bit Microcontroller Product (Microchip)



Test Station (engineering setup)

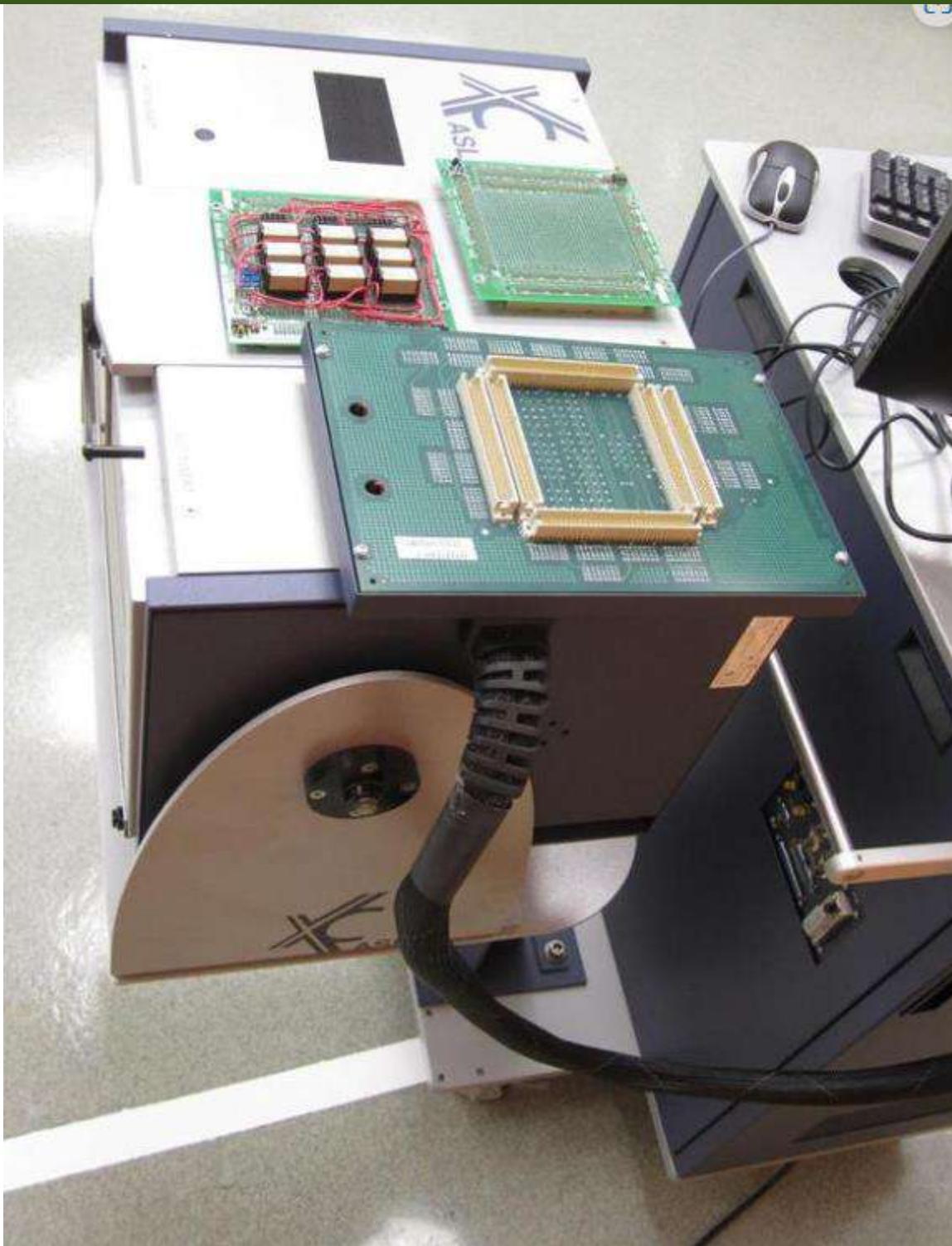


Test System (Tester + Handler)

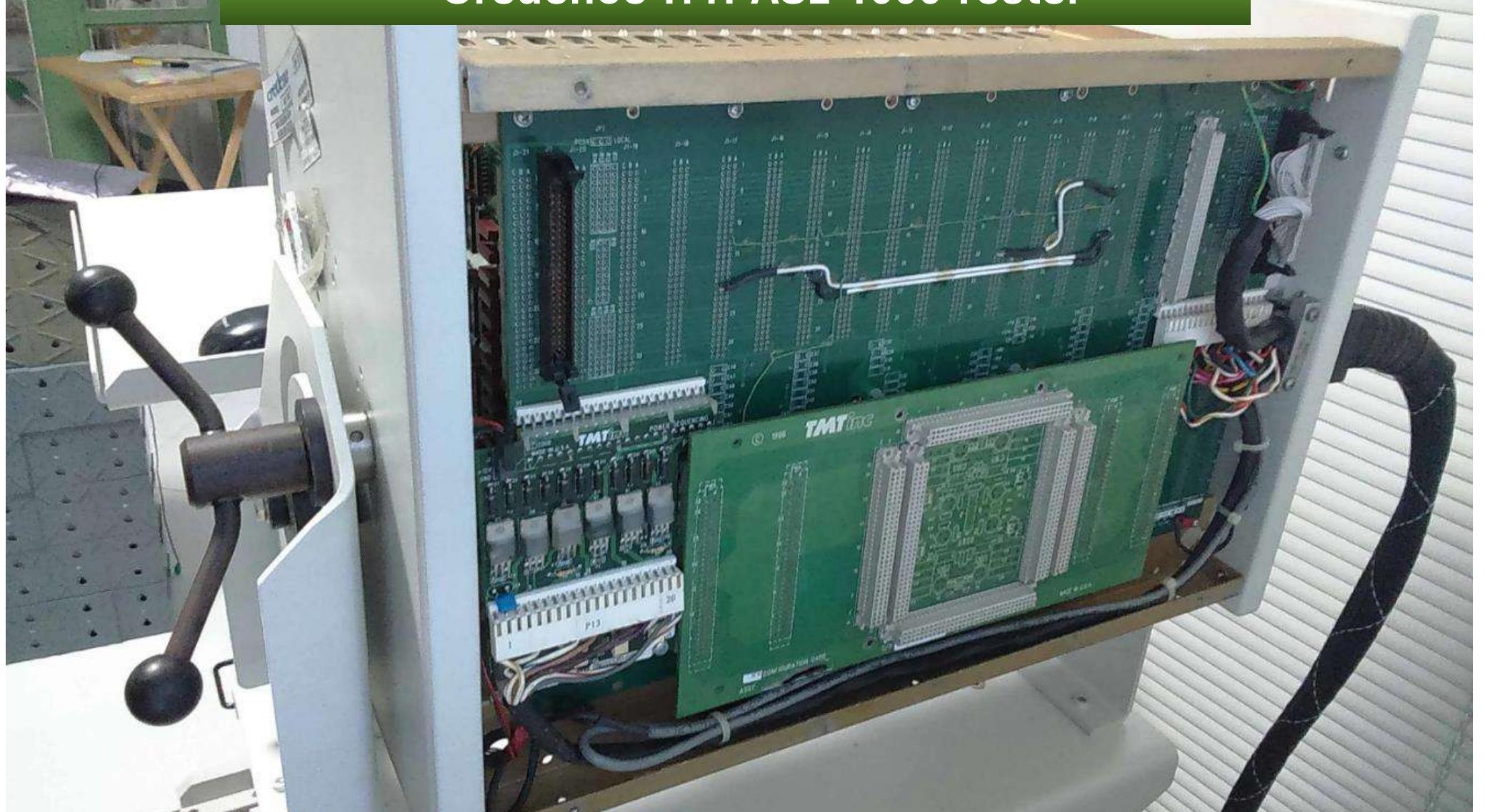
Credence TMT ASL-1000 Tester



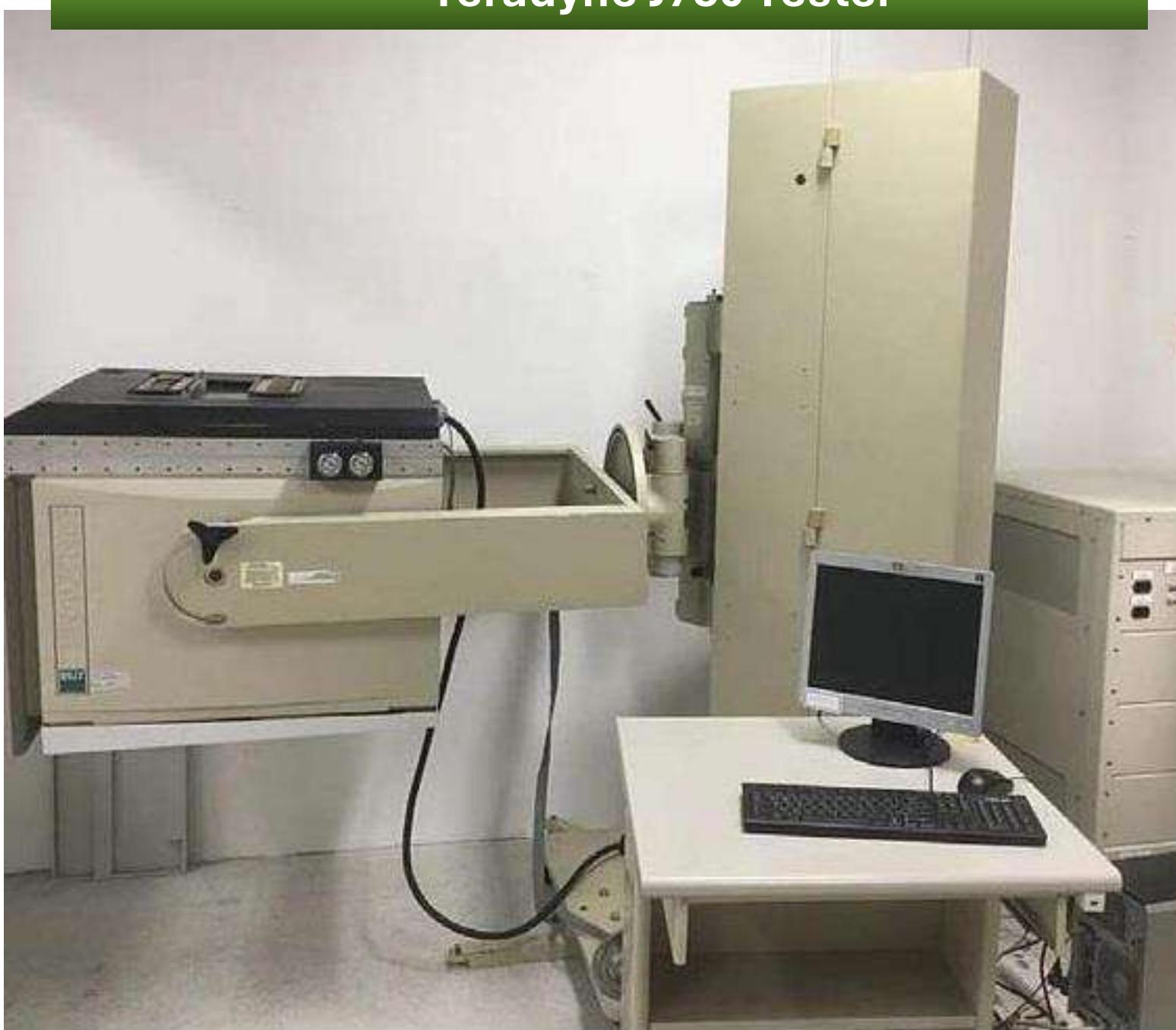
Credence TMT ASL-1000 Tester



Credence TMT ASL-1000 Tester



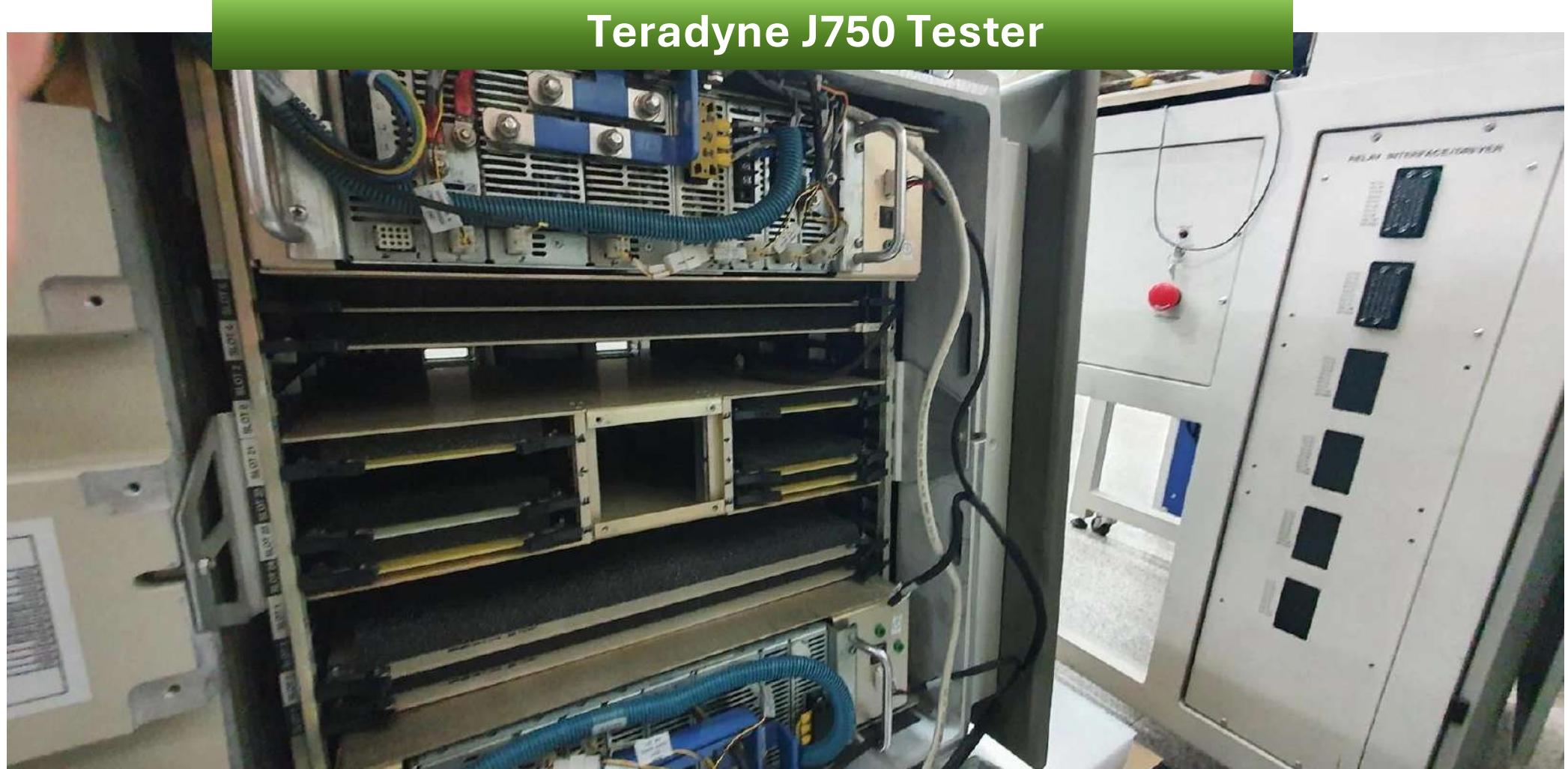
Teradyne J750 Tester



Teradyne J750 Tester

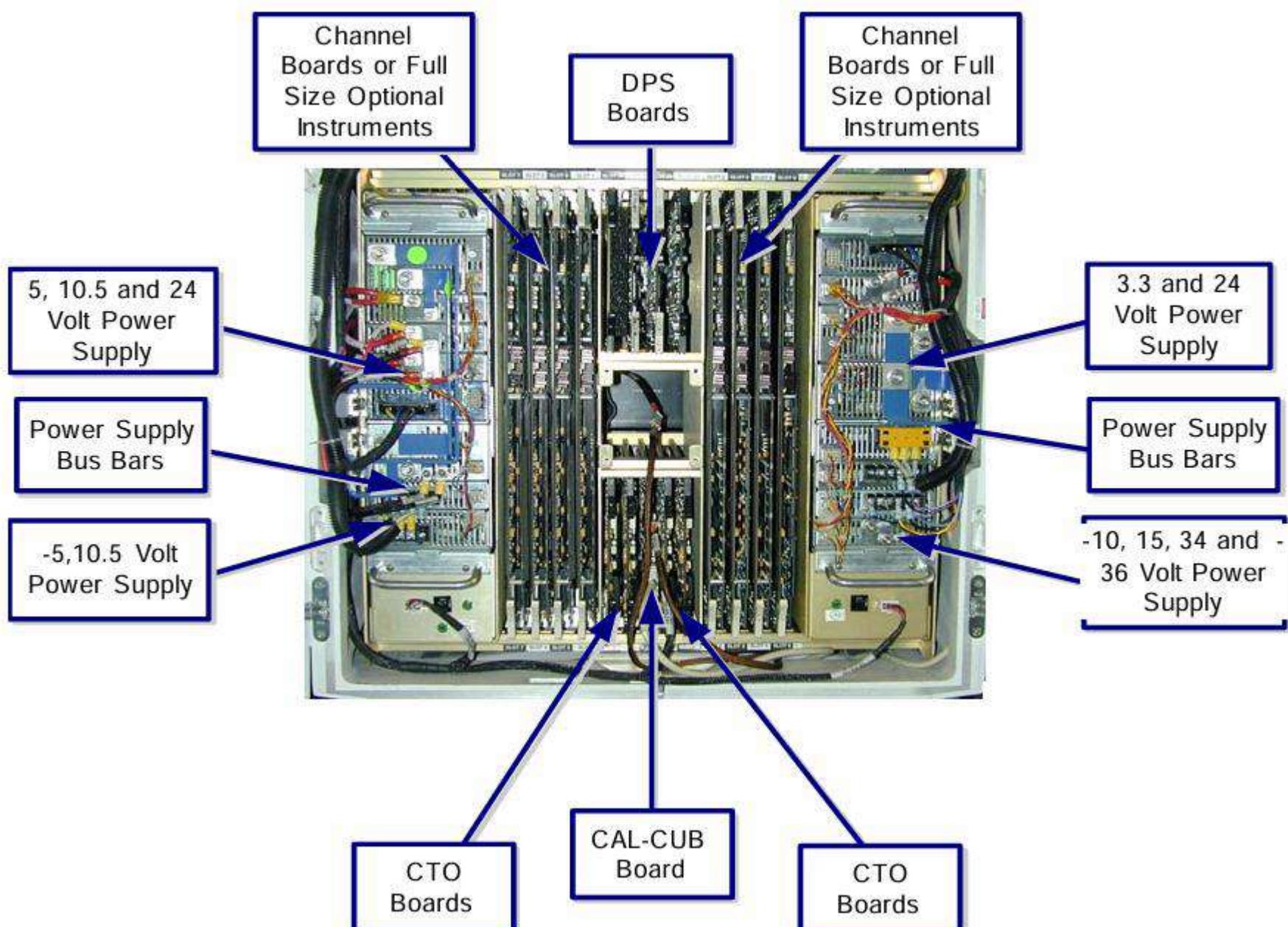


Teradyne J750 Tester

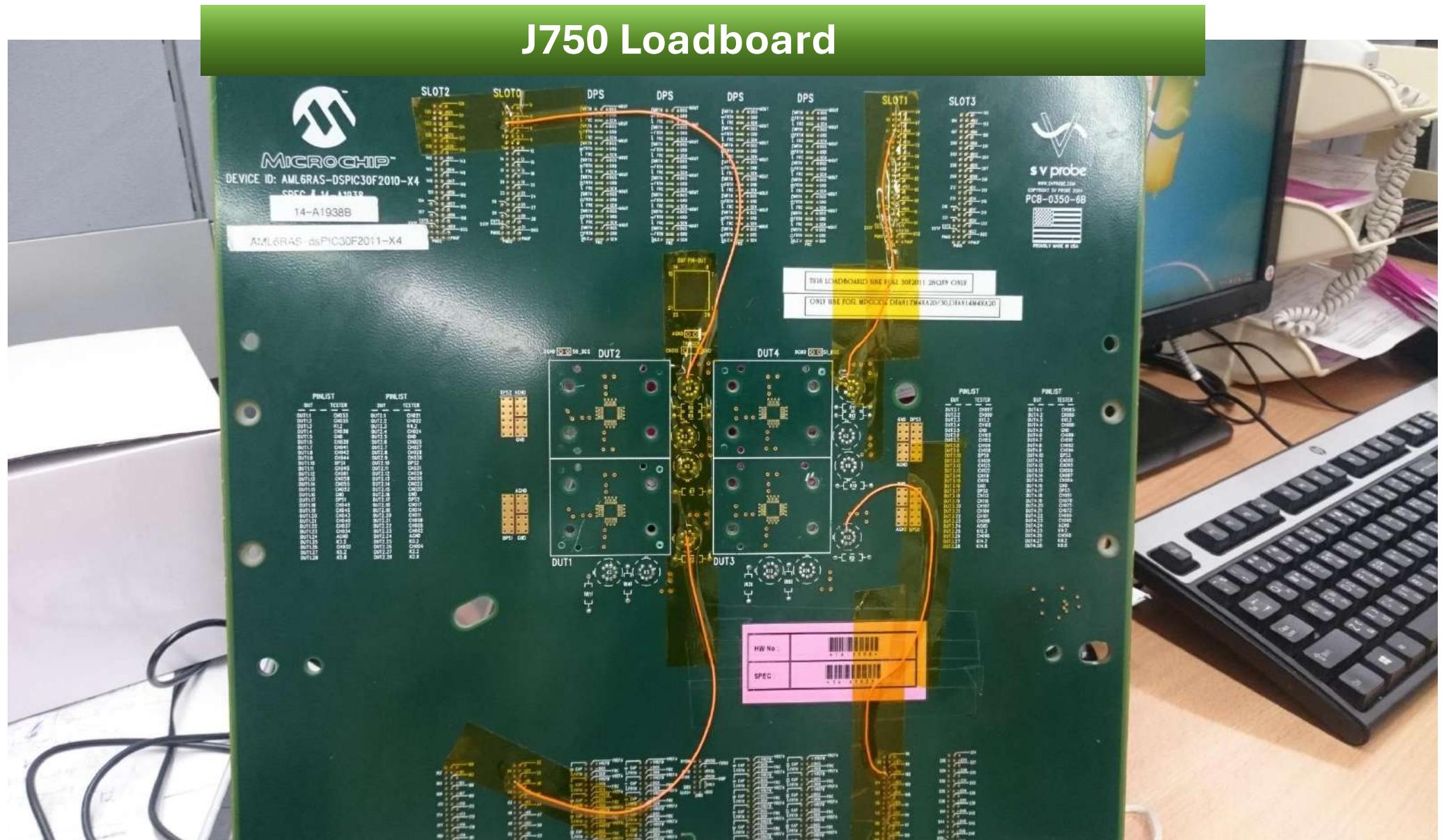


J750 Instrument Card

Rear of Tester (Fan Plate Assembly End)



J750 Loadboard



Visual Basic

```

Microsoft Visual Basic for Applications
File Edit View Insert Tools Window Help
Project - Template
(General)
Dim lngNChans As Long
Dim lngNSites As Long
Dim strErr As String
Dim lngX As Long
Dim lngY As Long
Dim lngOutSel As Long
Dim varForceV As Variant
Dim dblForceV As Double
Dim lngThisSite As Long
Dim lngCh As Long

If MainOrAlt = tl_tm_GetIndexOf(TL_C_MOALSTR) Then
    'when in Ambient Mode, record the voltage for each pin
    lngX = 0

    Call TheExec.DataManager.GetChanList(DpsPinList, lngThisSite,
                                         chDPS, lngChans, lngNChans, lngNSites, strErr)
    For lngY = 0 To lngNChans - 1
        lngCh = lngChans(lngY)

        For lngThisSite = 0 To TheExec.Sites.ExistingCount - 1
            If TheExec.Sites.Site(lngThisSite).Selected Then

                lngCh = lngChans(lngY)

                With TheHdw.DPS.chans(lngCh)
                    lngOutSel = .OutputSource
                    If lngOutSel <> dpsVSourceUnknown Then
                        varForceV = .forceValue(lngOutSel)
                        dblForceV = CDbl(varForceV(0))
                    Else
                        dblForceV = .HWProgrammedValue
                    End If
                End With

                ForceChan(lngX) = lngCh
                ForceV(lngX) = dblForceV
                lngX = lngX + 1
            End If

            lngChans(lngY) = lngChans(lngY) + 1
        Next lngThisSite

        Next lngY
        Exit Function
    End If

    ' set DPS force value for the Main or Alternate Programmed Value, or the Levels value
    If MainOrAlt = tl_tm_GetIndexOf(TL_C_MOAMSTR) Then

```

Visual Basic

The screenshot shows the Microsoft Visual Basic IDE interface. The title bar reads "Visual Basic". The menu bar includes File, Edit, View, Insert, Format, Debug, Run, Tools, Window, Help. The toolbar has icons for New, Open, Save, Print, Find, Replace, Copy, Paste, Cut, Undo, Redo, and others. The Project Explorer on the left shows a solution named "J750Maint" with a file "DataTool.x" containing a module "J750Main". The code editor window displays VBScript code for initializing constants and variables. The Properties window and Alphabetic/Categorized list are also visible.

```

Option Explicit

Private Const CubPin As Long = TL_CAL_CHANNEL1
Private Const CubSli As Long = TL_CAL_CHANNEL2
Private Const CubSlot As Long = TL_CAL_CHANNEL1 / TL_DIGITAL_CHANNELS_PER_BOARD

Private Const TimeSet1 As Long = 1      ' selects "EdgeSet0" or "EdgeSet1"
Private Const TimeSet2 As Long = 2      ' selects "EdgeSet2"
Private Const EdgeSet0 As Long = 0      ' disables all edges
Private Const EdgeSet1 As Long = 1      ' enables all edges, disables R0 & R1
Private Const EdgeSet2 As Long = 2      ' enables all edges
Private Const Format0 = 0
Private Const Period0 As Long = 0

Private Const cpp As Long = 1

Private Const DriveHi As Double = 3#
Private Const DriveLo As Double = 0#
Private Const Vt As Double = ((DriveHi - DriveLo) / 2# + DriveLo)
Private Const ILoad = (Vt / 50#)          ' unterminated current load
Private Const ILoadHi As Double = -(ILoad)
Private Const ILoadLo As Double = ILoad

Private SystemFrequency As Double
Private SystemClock As Double

```

Single Site Datalog

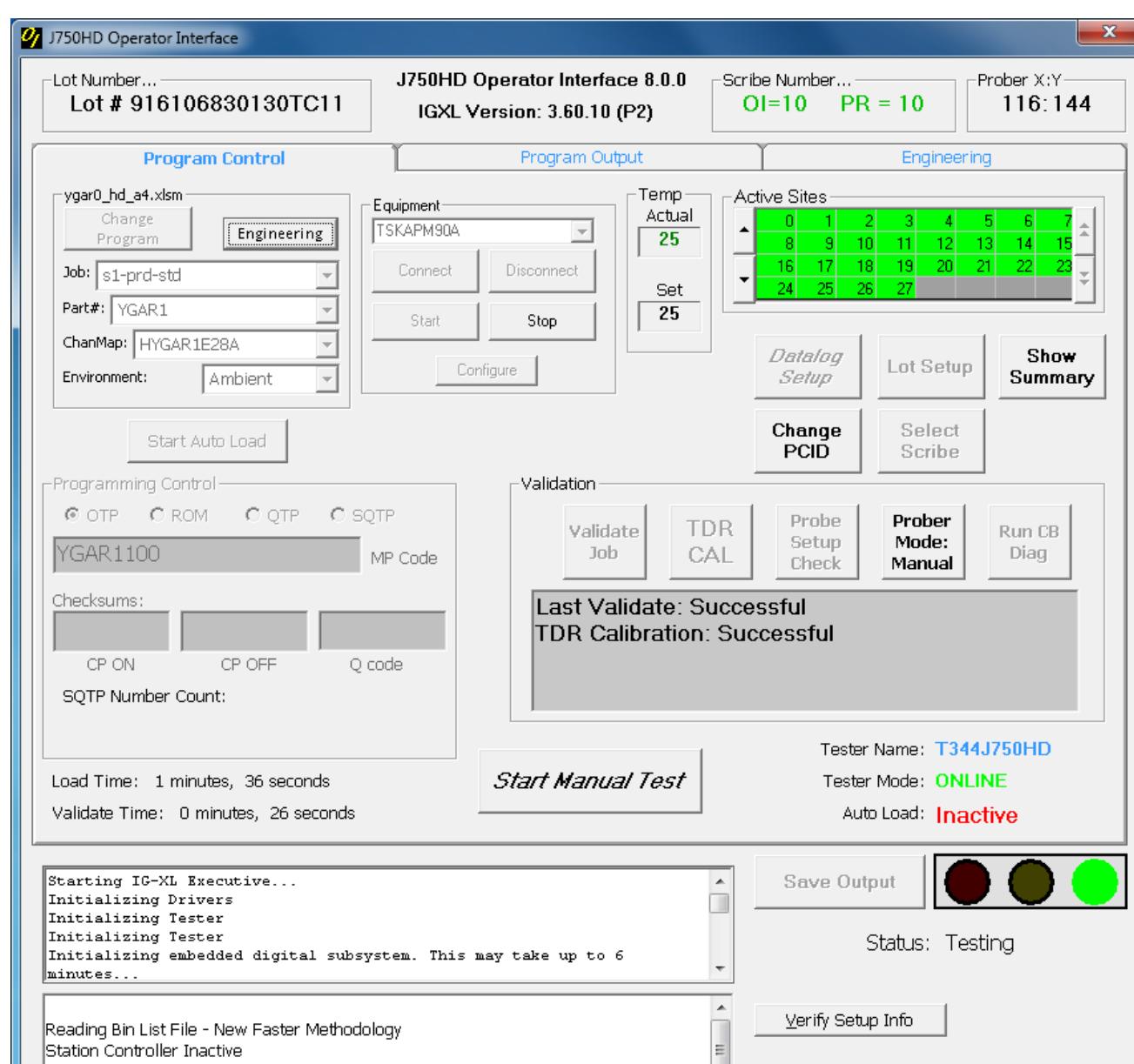
Number	Site	Result	Test Name	Pin	Channel	Low	Measured	High	For
169									
170	9301	0	PASS	f_FRC_Tune	an3	5.ch69	7.2253 Mhz	7.3840 Mhz	7.5203 Mhz
171	9400	0	PASS	f_FRC_Final_Cal	fuse_FRCTRIM_Cal				3.2
172	9500	0	PASS	f_FRC_Read_3v	FRC_Freq_Read				
173	9501	0	PASS	f_FRC_Read_3v	an3	5.ch69	7.2268 Mhz	7.3880 Mhz	7.5188 Mhz
174	9600	0	PASS	f_LPRC_Tune	fuse_TFFID1_Tune				3.2
175	# MinFreq: 26214.4 MaxFreq: 39321.6 TargFreq: 32768 IdealStepFreq: 3666 GlobalOverrides: 0 Interger for Center Freq:4								
176	# PH1 Site: 0 I: 0 Alarm: 0 MThis: 14063 MPrev: 0 CNext: 5 CThis: 0 CDelt: 5 StpSz: 3666 WBits:0011								
177	# PH1 Site: 0 I: 1 Alarm: 0 MThis: 34375 MPrev: 14063 CNext: 5 CThis: 5 CDelt: 0 StpSz: 4063 WBits:0011								
178	# PH2 Site: 0 MThis: 30313 MPrev: 34375 CNext: 4 CThis: 5 MFINL: 34375 CFINL: 5 WBits:0011								
179	9601	0	PASS	f_LPRC_Tune	an3	5.ch69	26.2144 Khz	34.3750 Khz	39.3216 Khz
180	9700	0	PASS	f_LPRC_Final_Cal	fuse_FFFID1_Cal				3.2
181	9800	0	PASS	f_LPRC_Read_3v	LPRC_Freq_Read				
182	9801	0	PASS	f_LPRC_Read_3v	an3	5.ch69	26.5421 Khz	34.3750 Khz	38.9939 Khz
183	2300	0	PASS	DevRev_A4_veri	fuse_DEVREV_a4_3004_veri				3.2
184	18800	0	PASS	f_cmp_min	pt_cmpx_t11_var3				
185	18800	0	PASS	f_cmp_min	pt_cmpx_an6_t1_extref				
186	18801	0	PASS	f_cmp_min	pt_cmpx_an6_t1_vref				
187	14900	0	PASS	f_cmp_max	pt_cmpx_t11_var3				
188	14900	0	PASS	f_cmp_max	pt_cmpx_an6_t1_extref				
189	14901	0	PASS	f_cmp_max	pt_cmpx_an6_t1_vref				
190	38900	0	PASS	m_F1PageEr_t11	f1page_t11				
191	9200	0	PASS	fuse_lptrim_t11	fuse_LPTRIM_prsrv_f32t_pllcfg_0C				
192	37800	0	PASS	f_auxpll_EC_prog_load	Keboda_APPL_EC_fuse_prog				
193	37801	0	PASS	f_auxpll_EC_prog_load	Keboda_APPL_EC_fuse_veri				
194	37802	0	PASS	f_auxpll_EC_prog_load	apll_screen_v1_prog				
195	46100	0	PASS	f_lvhf_min	auxpll_pwm1h_exec_ws				
196	46101	0	PASS	f_lvhf_min	pwm1h	5.ch20	3.4000 Mhz	3.9345 Mhz	4.6000 Mhz
197	39000	0	PASS	m_F2PageEr_t11	f2page_t11				2.9
198	41200	0	PASS	m_AdcfrcPLLFuse_max	adcfrcpll_fuse_prog				
199	41201	0	PASS	m_AdcfrcPLLFuse_max	adcfrcpll_fuse_veri				
200	45900	0	PASS	m_Keboda_Headlight_prog	Keboda_RP3_Headlight_prog				
201	45901	0	PASS	m_Keboda_Headlight_prog	Keboda_RP3_Headlight_veri				
202	46000	0	PASS	f_Keboda_Headlight_CAPA	Keboda_Headlight_Func				
203	46001	0	PASS	f_Keboda_Headlight_CAPA	pwm1h	5.ch20	25.0000	41.0000	0.0000
204	9900	0	PASS	a_vregulator_meas	vcap_vddcore	7.ch71	2.3000 V	2.5866 V	2.7000 V
205	22400	0	PASS	IPD_ipd_test_peri_off	Vdd	23.dps12	1.0000 uA	126.8635 uA	497.0000 uA

28 Site Datalog

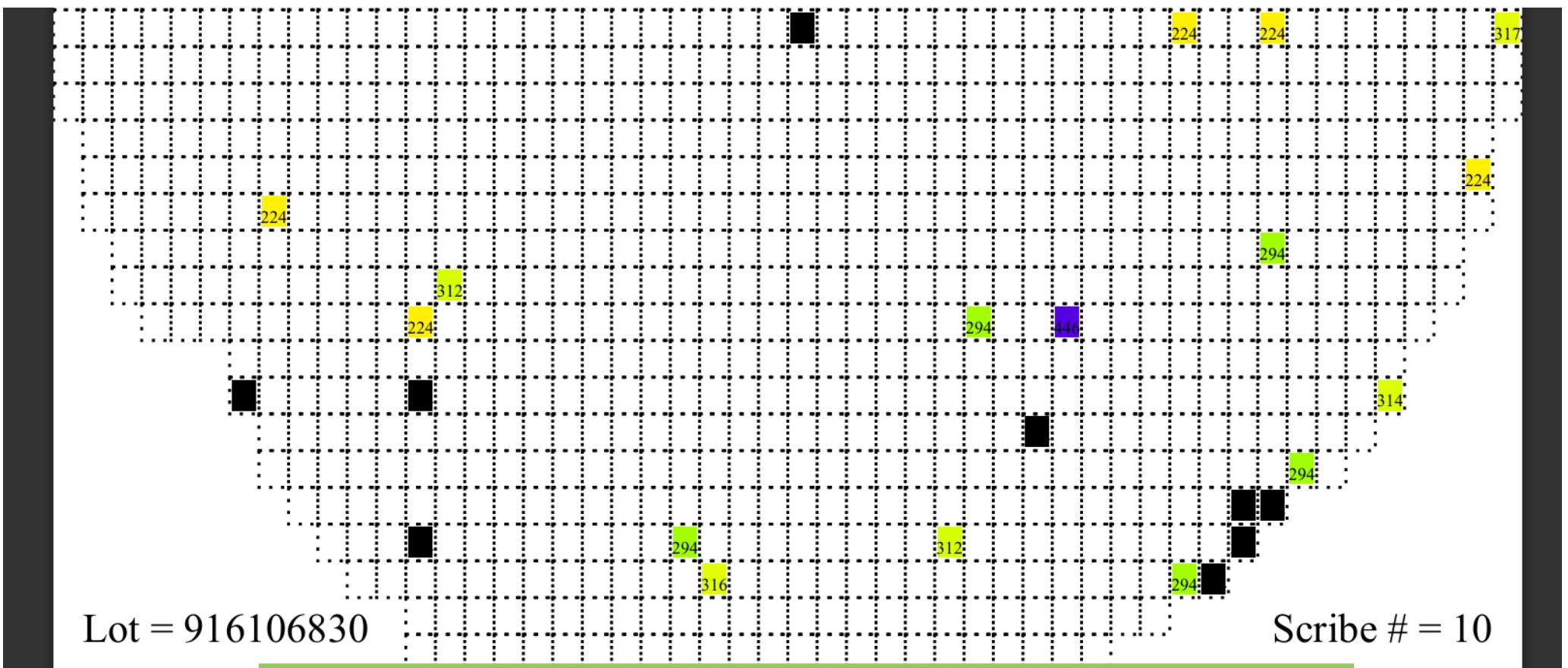
49	2000	27	PASS	m_GlobalSubr_t11	AA_GlobalSubr_YGAR_t11
50	38400	0	PASS	m_passcode_erase	passcode_erase
51	38400	1	PASS	m_passcode_erase	passcode_erase
52	38400	2	PASS	m_passcode_erase	passcode_erase
53	38400	3	PASS	m_passcode_erase	passcode_erase
54	38400	4	PASS	m_passcode_erase	passcode_erase
55	38400	5	PASS	m_passcode_erase	passcode_erase
56	38400	6	PASS	m_passcode_erase	passcode_erase
57	38400	7	PASS	m_passcode_erase	passcode_erase
58	38400	8	PASS	m_passcode_erase	passcode_erase
59	38400	9	PASS	m_passcode_erase	passcode_erase
60	38400	10	PASS	m_passcode_erase	passcode_erase
61	38400	11	PASS	m_passcode_erase	passcode_erase
62	38400	12	PASS	m_passcode_erase	passcode_erase
63	38400	13	PASS	m_passcode_erase	passcode_erase
64	38400	14	PASS	m_passcode_erase	passcode_erase
65	38400	15	PASS	m_passcode_erase	passcode_erase
66	38400	16	PASS	m_passcode_erase	passcode_erase
67	38400	17	PASS	m_passcode_erase	passcode_erase
68	38400	18	PASS	m_passcode_erase	passcode_erase
69	38400	19	PASS	m_passcode_erase	passcode_erase
70	38400	20	PASS	m_passcode_erase	passcode_erase
71	38400	21	PASS	m_passcode_erase	passcode_erase
72	38400	22	PASS	m_passcode_erase	passcode_erase
73	38400	23	PASS	m_passcode_erase	passcode_erase
74	38400	24	PASS	m_passcode_erase	passcode_erase
75	38400	25	PASS	m_passcode_erase	passcode_erase
76	38400	26	PASS	m_passcode_erase	passcode_erase
77	38400	27	PASS	m_passcode_erase	passcode_erase
78	1200	0	PASS	c_cont_vss	cont_vss
79	1200	1	PASS	c_cont_vss	cont_vss
80	1200	2	PASS	c_cont_vss	cont_vss
81	1200	3	PASS	c_cont_vss	cont_vss
82	1200	4	PASS	c_cont_vss	cont_vss
83	1200	5	PASS	c_cont_vss	cont_vss
84	1200	6	PASS	c_cont_vss	cont_vss
85	1200	7	PASS	c_cont_vss	cont_vss
86	1200	8	PASS	c_cont_vss	cont_vss
87	1200	9	PASS	c_cont_vss	cont_vss
88	1200	10	PASS	c_cont_vss	cont_vss



UF3000 Prober (Wafer Test)



Tester – Prober Interface UI



Test Result Map

Bin	Test Name	Cnt
1024	SITE_00 FAILS	2
1025	SITE_01 FAILS	0
1026	SITE_02 FAILS	2
1027	SITE_03 FAILS	0
1028	SITE_04 FAILS	1
1029	SITE_05 FAILS	3
1030	SITE_06 FAILS	2
1031	SITE_07 FAILS	1
1032	SITE_08 FAILS	0
1033	SITE_09 FAILS	0
1034	SITE_10 FAILS	2
1035	SITE_11 FAILS	0
1036	SITE_12 FAILS	1

Bin	Test Name	Cnt
11	C_CONT_PMU_VSS	0
188	F_CMP_MIN	0
149	F_CMP_MAX	1
94	F_FRC_FINAL_CAL	0
153	F_RAM_RETENTION	0
99	A_VREGULATOR_MEAS	0
97	F_LPRC_FINAL_CAL	0
298	READ_SORT1_RECORD	0
303	MASS_ERASE_25V	0
305	MRG1_25V	0
316	MRG0_28V	1
95	F_FRC_READ_3V	0
98	F_LPRC_READ_3V	0

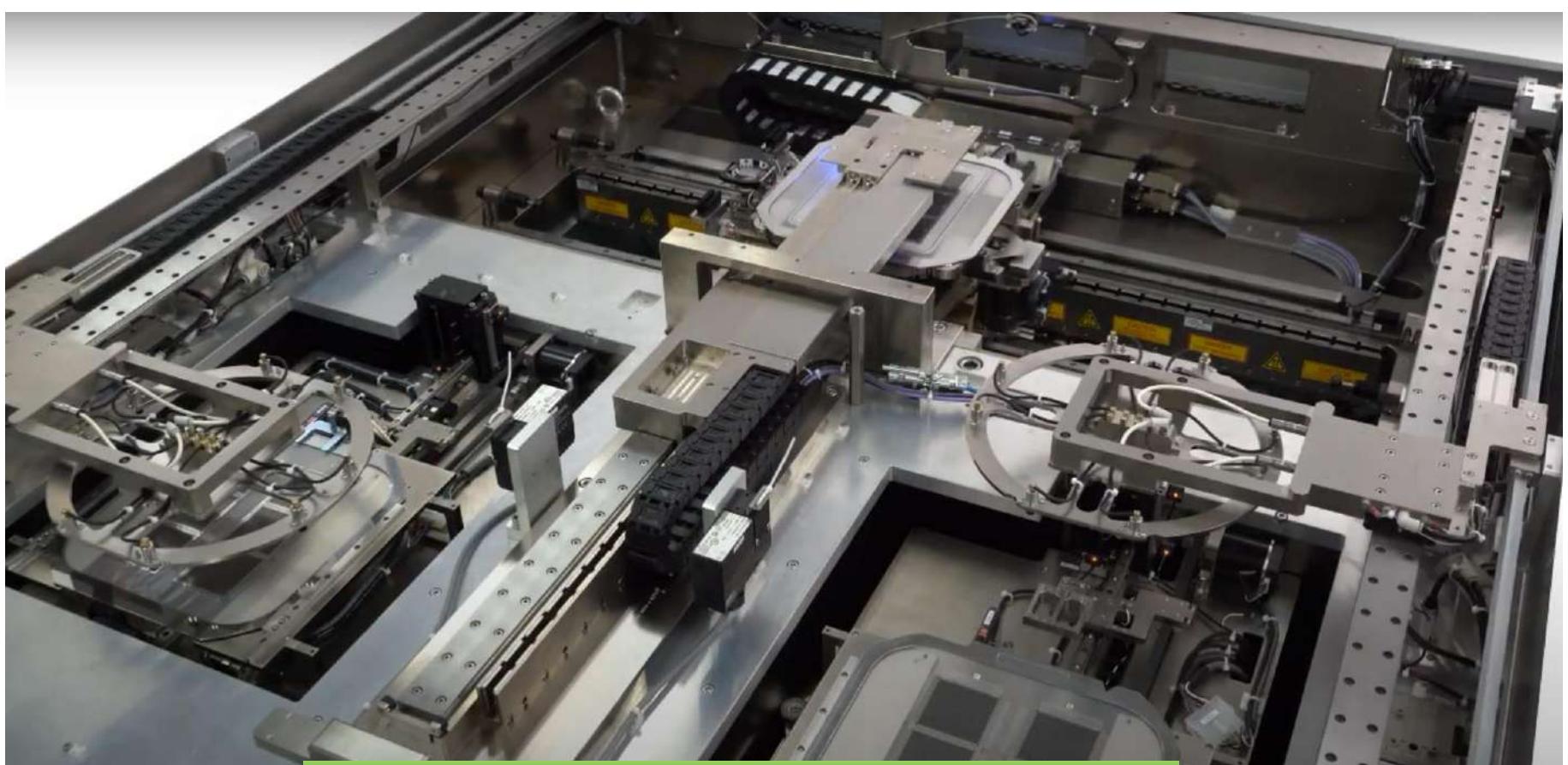
Bin	Test Name	Cnt
274	TSMC_GLOBAL_SUBR	0
294	READ_FF_ALL_22V	5
295	READ_FF_ALL_28V	0
304	MASS_ERASE_22V	0
288	PAGE_ERASE_2MS_22V	0
289	PAGE_ERASE_10MS_22V	0
275	BLOCK_ERASE_ALL_25V	0
309	MRG1_ODD_ROW_25V_LONG	0
311	MRG1_EVEN_ROW_25V_LONG	0
278	PROG_ALL_ROWS_FF_25V	0
312	MRG1_CKBD_22V	2
317	MRG0_25V	3
306	MRG1_ALL_25V	0

Test Bin





MCT SH-5300 Handler



MCT SH-5300 Handler

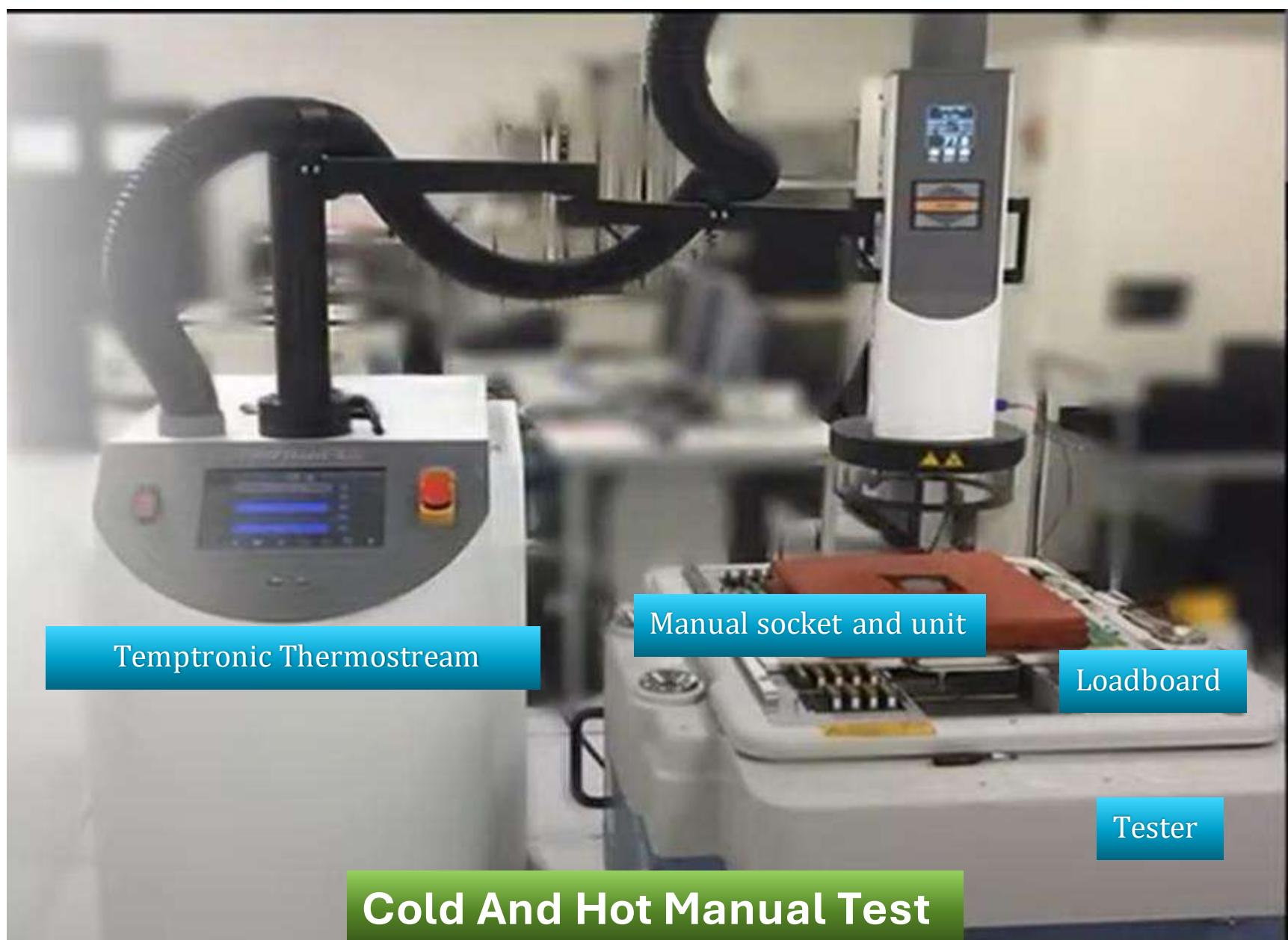
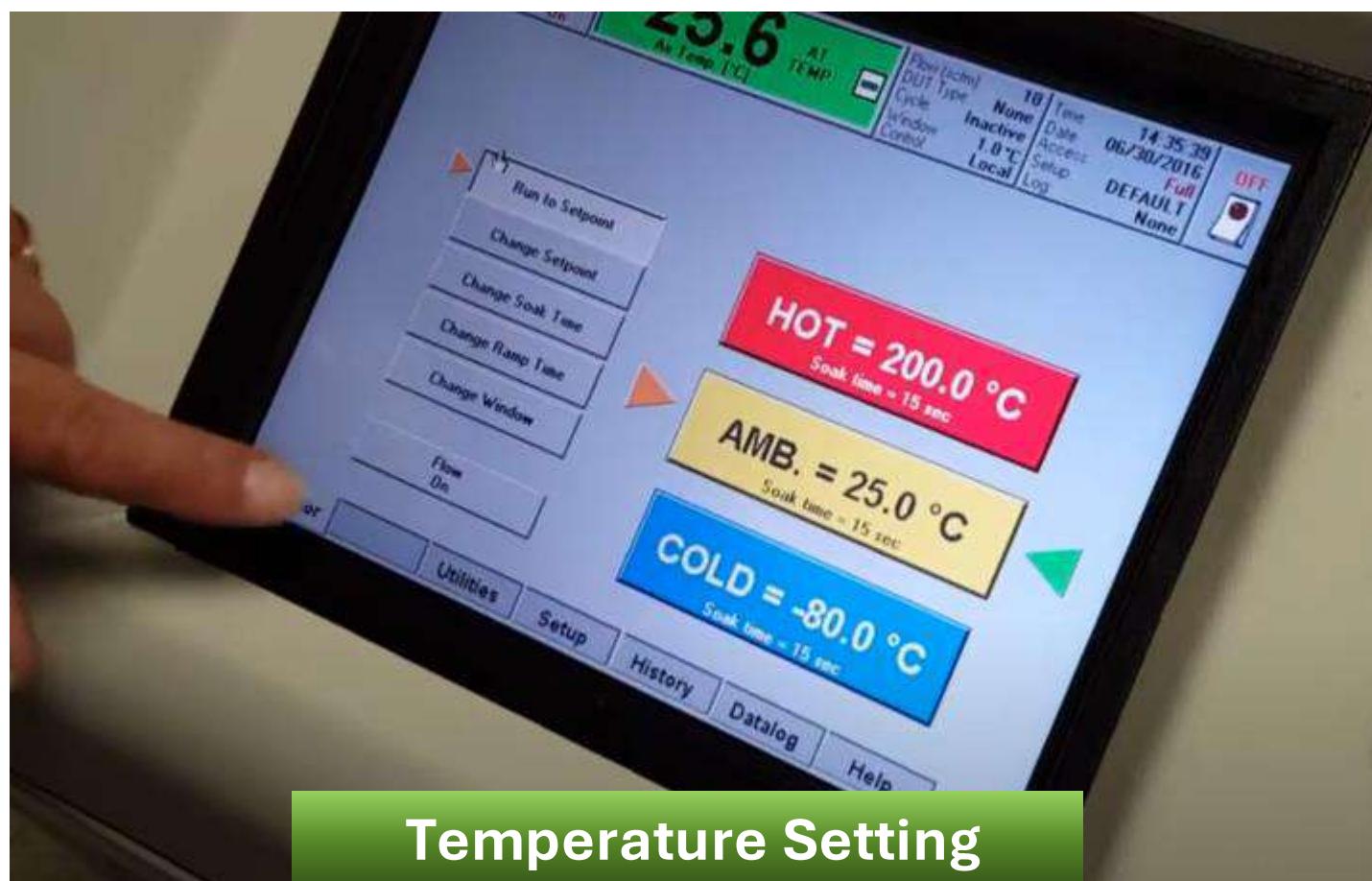


Strip test manual socket



Provide temperature to unit
For hot and cold test

Temptronic Thermostream



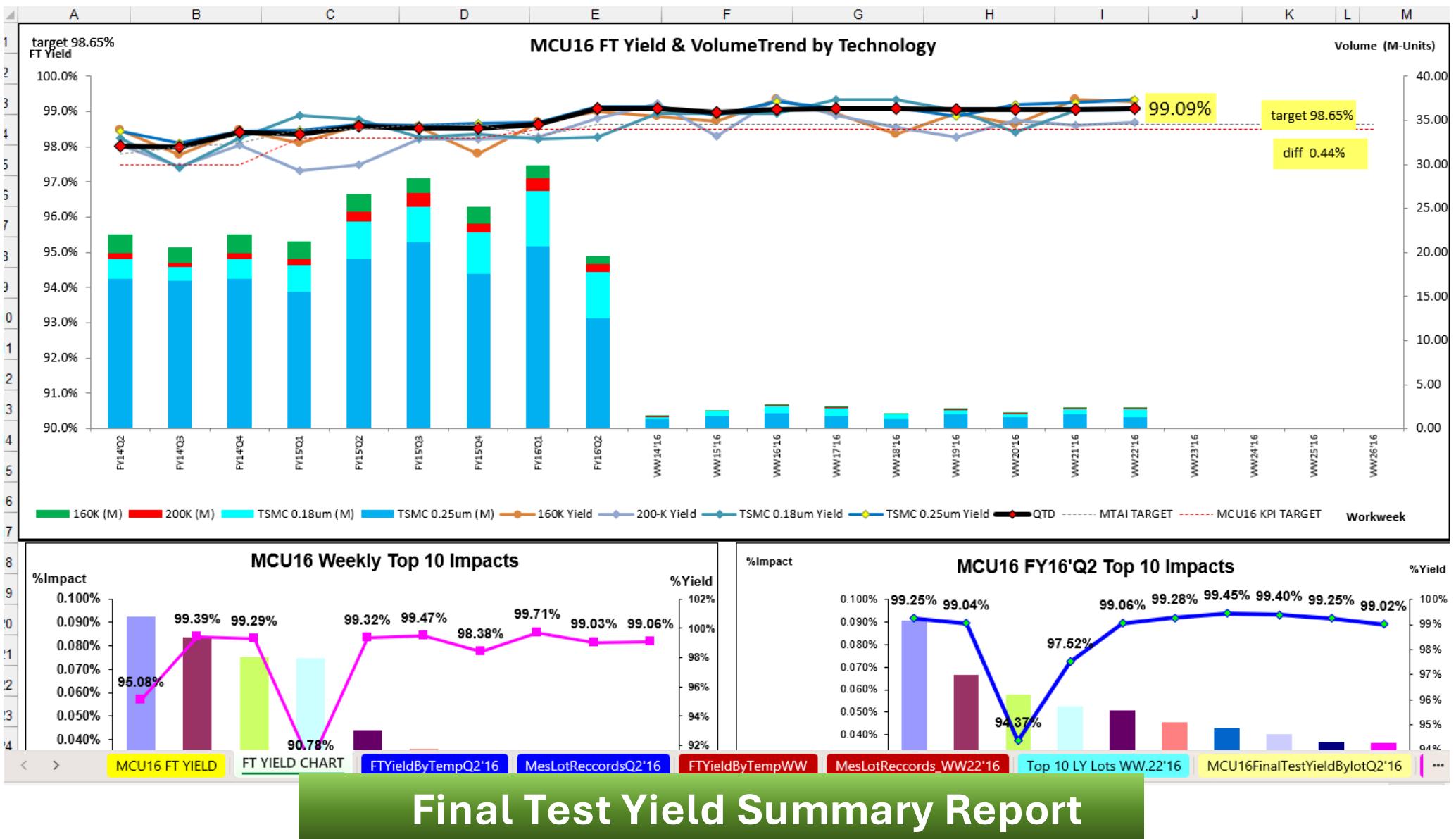


Rasco-1000 Handler

	A	B	C	CZ	DA	DB	DC	EF	EG	EH	EI	EJ	EK	EL	EM
1	MCU16 FINAL TEST YIELD SUMMARY														
2	MASK	TECHNOLOGY	RTP DATE	FY16'Q2				WW21'16				WW22'16			
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
4	MASK	TECHNOLOGY	RTP DATE	IN (EA)	OUT (EA)	YIELD	% IMPACT	IN (EA)	OUT (EA)	YIELD	% IMPACT	IN (EA)	OUT (EA)	YIELD	% IMPACT
5	DFAA1	160K	28/4/2004	5,504	5,174	94.00%	0.00%			-	-			-	-
6	DFAB1	160K	6/5/2004	146,013	144,407	98.90%	0.01%	1,734	1,724	99.42%	0.00%	38,061	37,897	99.57%	0.01%
7	DFAC1	160K	5/7/2004	84,374	82,900	98.25%	0.01%	19,634	19,334	98.47%	0.01%	8,786	8,575	97.60%	0.01%
8	DFAD1	160K	3/9/2004	123,272	122,672	99.51%	0.00%	14,996	14,947	99.67%	0.00%	5,087	5,072	99.71%	0.00%
9	DFAE1	160K	27/1/2005	31,870	31,476	98.76%	0.00%			-	-	10,351	10,189	98.43%	0.01%
10	DFAF1	160K	23/12/2004	91,381	90,824	99.39%	0.00%	8,965	8,896	99.23%	0.003%	16,103	16,056	99.71%	0.00%
11	DFAH1	160K	15/12/2004	232,698	230,594	99.10%	0.01%	31,653	31,549	99.67%	0.004%	52,215	51,896	99.39%	0.01%
12	DFAJ1	160K	14/7/2005	78,335	77,532	98.97%	0.00%	12,750	12,680	99.45%	0.00%	2,326	2,304	99.05%	0.00%
13	DFAK1	160K	27/1/2006	41,667	41,464	99.51%	0.00%	7,175	7,145	99.58%	0.00%	1,266	1,242	98.10%	0.00%
14	DFAM1	160K	22/12/2005	62,271	62,018	99.59%	0.00%	5,536	5,491	99.19%	0.00%	5,200	5,190	99.81%	0.00%
15	DFAN1	160K	14/3/2005	12,017	11,965	99.57%	0.00%	7,029	6,996	99.53%	0.00%	1,752	1,737	99.14%	0.00%
16	DFAR1	160K	13/10/2006	37,218	36,063	96.90%	0.01%	1,119	1,090	97.41%	0.00%	1,254	1,234	98.41%	0.00%
17	LEAK1	200K	31/3/2009	269,252	265,248	98.51%	0.02%	71,712	70,723	98.62%	0.04%	48,849	48,056	98.38%	0.03%
18	LEAK2	200K	31/3/2009	22,609	22,165	98.04%	0.00%	9,871	9,637	97.63%	0.01%			-	-
19	LEAT1	200K	9/6/2009	53,351	52,312	98.05%	0.01%			-	-	3,389	3,348	98.79%	0.00%
20	LEBE1	200K	18/3/2011	76,329	75,148	98.45%	0.01%	20,410	19,986	97.92%	0.02%	11,749	11,635	99.03%	0.00%
21	LEBE2	200K	18/3/2011	195,952	193,869	98.94%	0.01%	31,423	31,171	99.20%	0.01%	33,341	32,898	98.67%	0.02%
22	LEBY1	200K	12/12/2011	210,057	208,984	99.49%	0.01%	32,100	31,703	98.76%	0.02%	11,531	11,505	99.77%	0.00%
23	LECJ1	200K	23/2/2013	5,403	5,305	98.19%	0.00%	1,807	1,775	98.23%	0.00%	1,766	1,712	96.94%	0.00%
24	LECJ2	200K	23/2/2013	1,566	1,425	91.00%	0.00%			-	-			-	-
25	SAAA1	SFC70	1/2/2016	0	0	-	-			-	-			-	-
26	TBAA1	0.18um	11/7/2011	201,415	190,081	94.37%	0.06%	16,039	14,832	92.47%	0.05%	45,662	43,417	95.08%	0.09%
27	TLAB1	0.18um	6/11/2012	75,205	74,872	99.56%	0.00%	6,982	6,937	99.36%	0.00%	33,841	33,702	99.59%	0.01%
28	TLAC1	0.18um	28/9/2011	946,957	941,685	99.44%	0.03%	189,822	189,173	99.66%	0.03%	146,541	146,145	99.73%	0.02%
29	TLAD1	0.18um	19/7/2012	768,228	763,495	99.38%	0.02%	8,161	8,133	99.66%	0.00%	42,374	42,226	99.65%	0.01%
30	TLAE1	0.18um	31/7/2012	1,234,290	1,225,354	99.28%	0.05%	119,690	118,776	99.24%	0.04%	166,657	165,775	99.47%	0.04%
31	TLAF1	0.18um	8/8/2012	1,247,766	1,242,349	99.57%	0.03%	147,544	146,761	99.47%	0.03%	237,098	236,404	99.71%	0.03%
32	TLAH1	0.18um	8/5/2015	34,240	31,219	91.18%	0.02%	8,424	7,830	92.95%	0.02%	19,756	17,934	90.78%	0.07%

< > MCU16 FT YIELD FT YIELD CHART FTYieldByTempQ2'16 MesLotRecordsQ2'16 FTYieldByTempWW MesLotRecords_WW22'16 Top 10 LY Lots WW.22'16 MCU16FinalTestYieldBylotQ2'16 ... + :

Final Test Yield Summary Report



FY17'Q1 MCU16 Engineering

Project Type: FY17'Q1 QC-LRR Info

- **High Light of the week**

MASK	%LRR	SS Lots	SR Lots	Failure
TQAA1	11.54%	26	3	a_cmpoffset_max , t_FrcTrimMeas_min , vddcore_33v
UGAB1	11.11%	9	1	devid_24FJ64GB004_2V_qc
DFAB1	10.00%	10	1	f_pwm_max
LEBY1	8.33%	24	2	a_compooff_20l , i_ipd_wdt_tmod
DFAC1	7.69%	13	1	m_ram_min

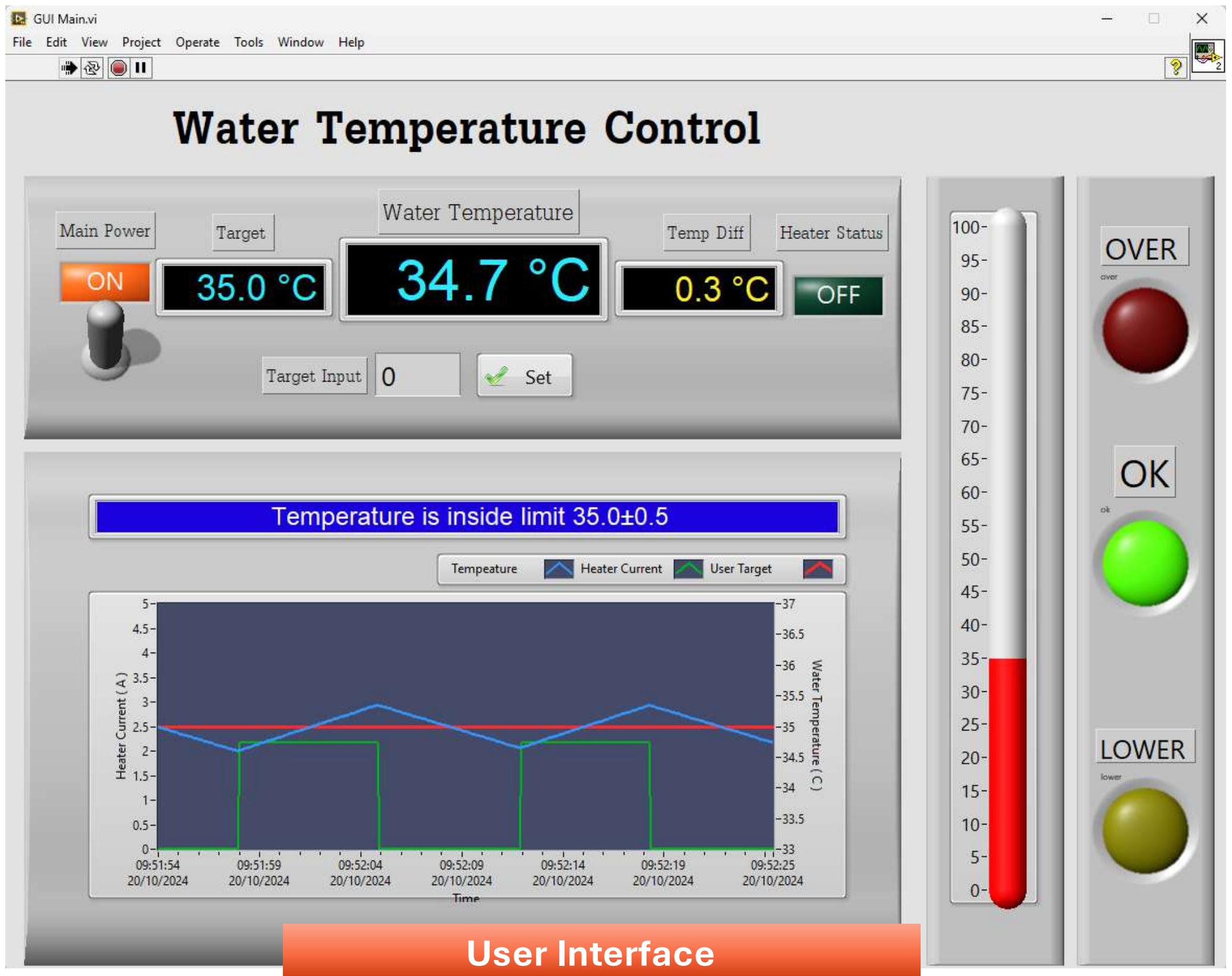
- **Highlight of the Quarter**

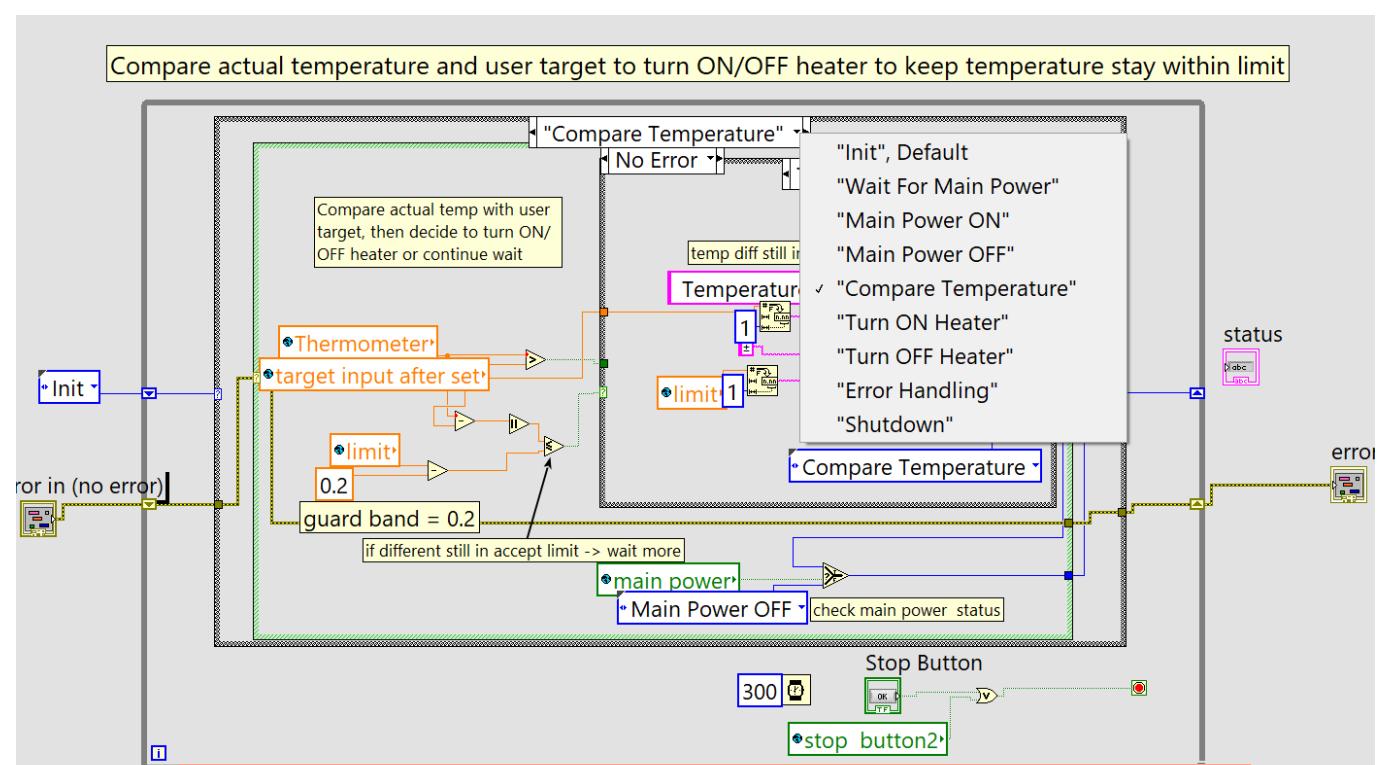
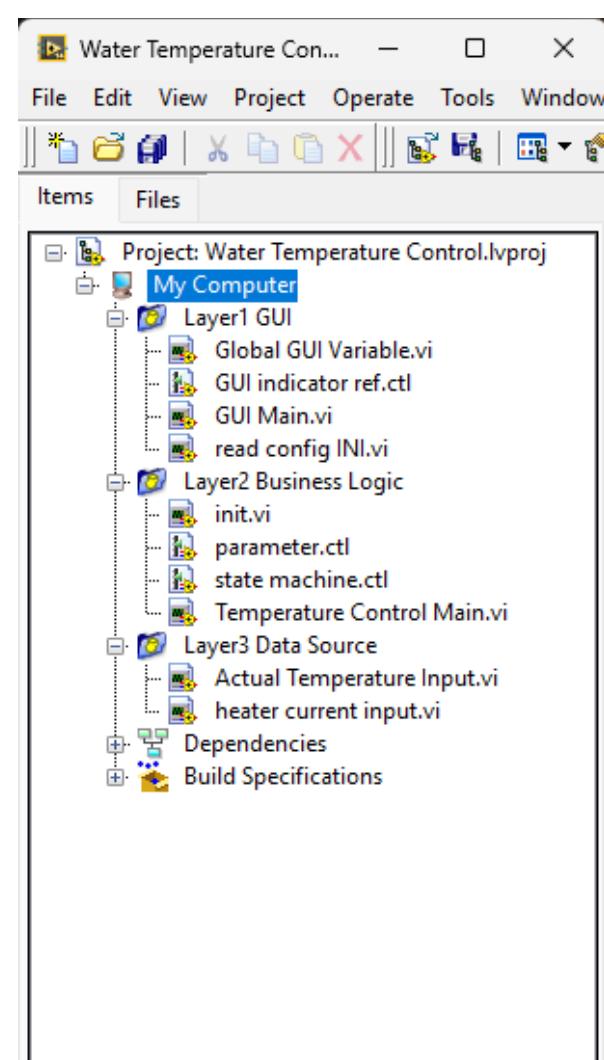
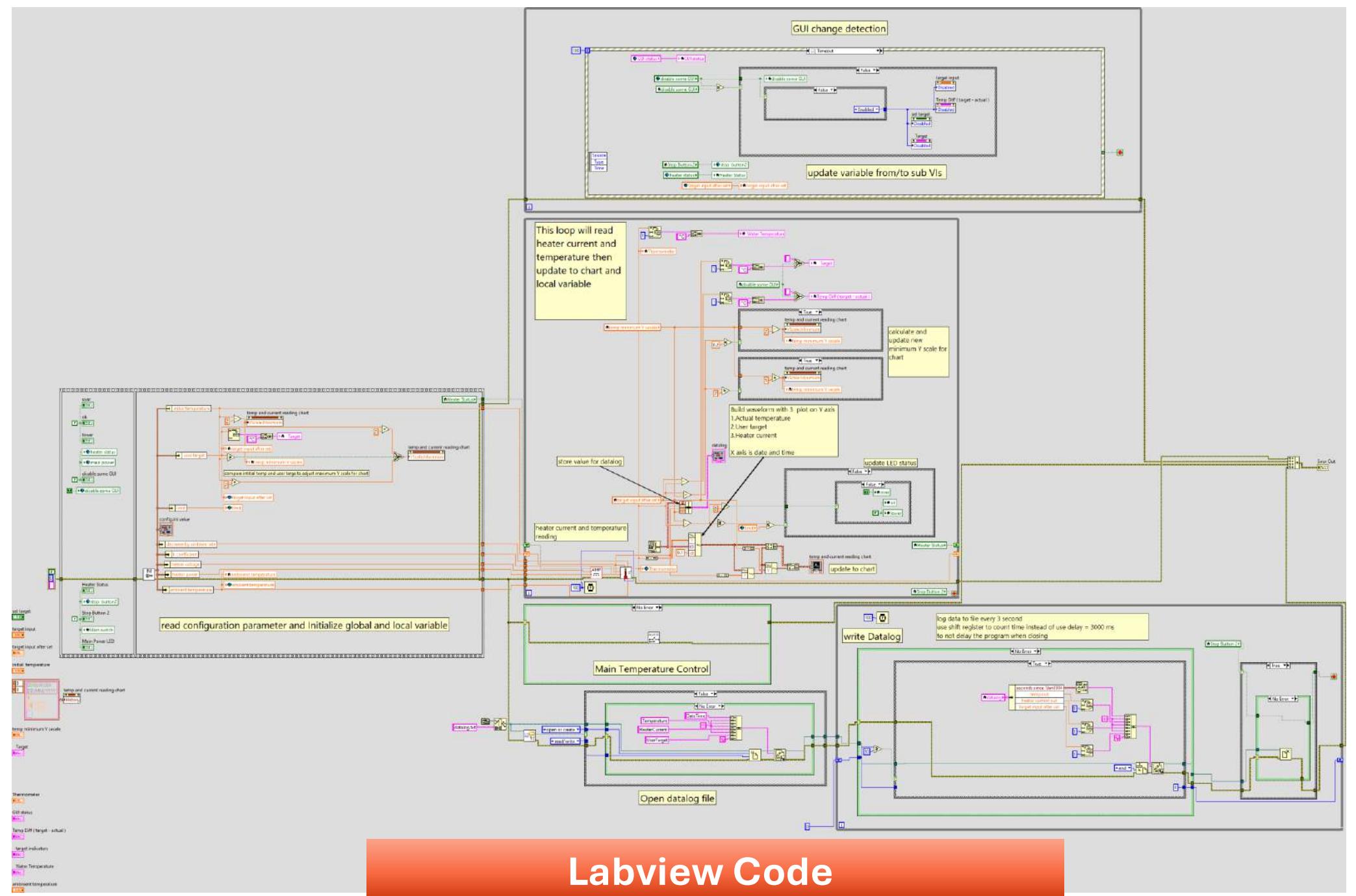
MASK	%LRR	SS Lots	SR Lots
LECJ2	33.33%	3	1
DFAR1	14.29%	7	1
DFAM1	11.11%	18	1
LEBY1	10.00%	50	5
SABE1	9.09%	11	1

Water Temperature Control

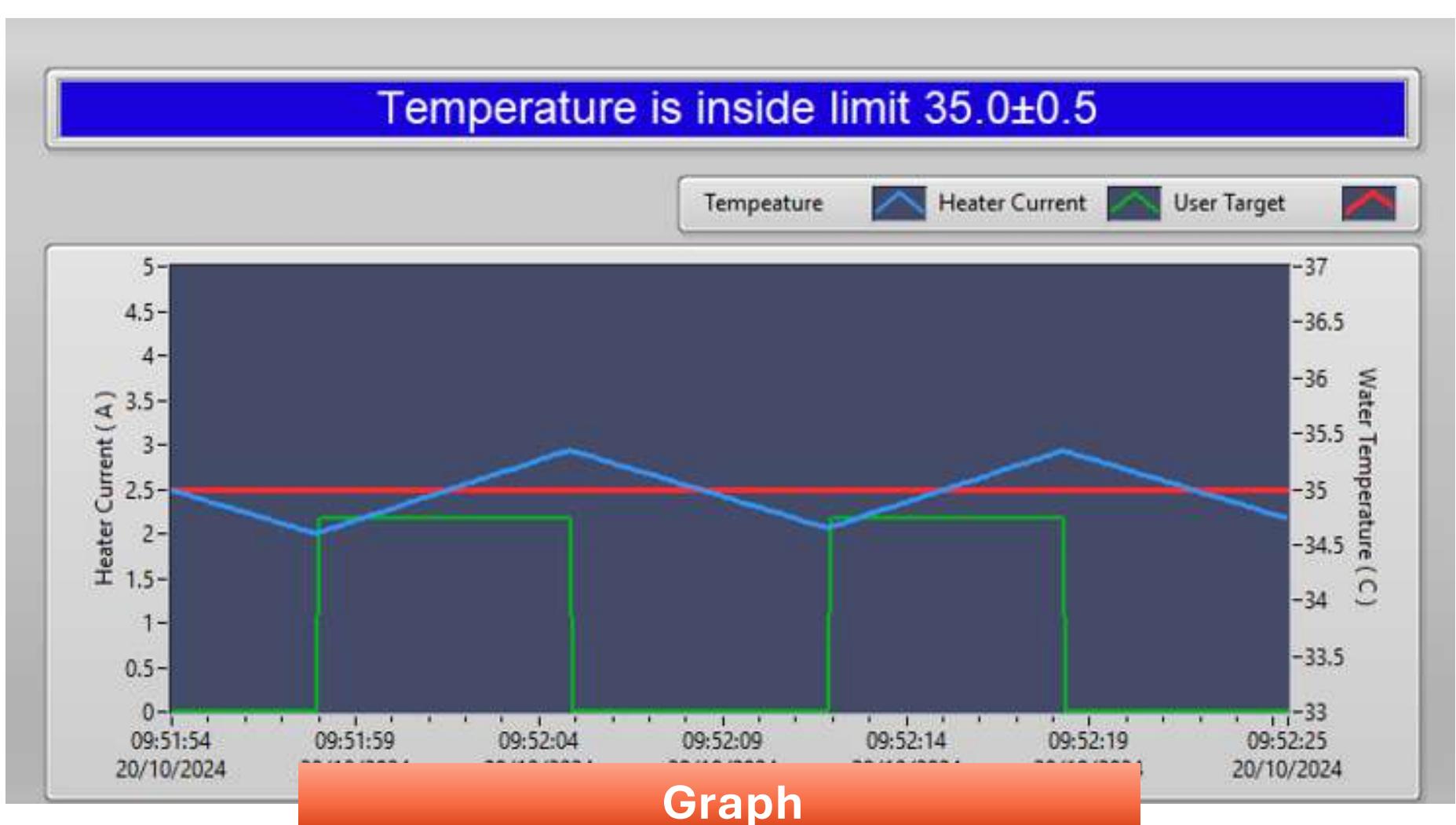
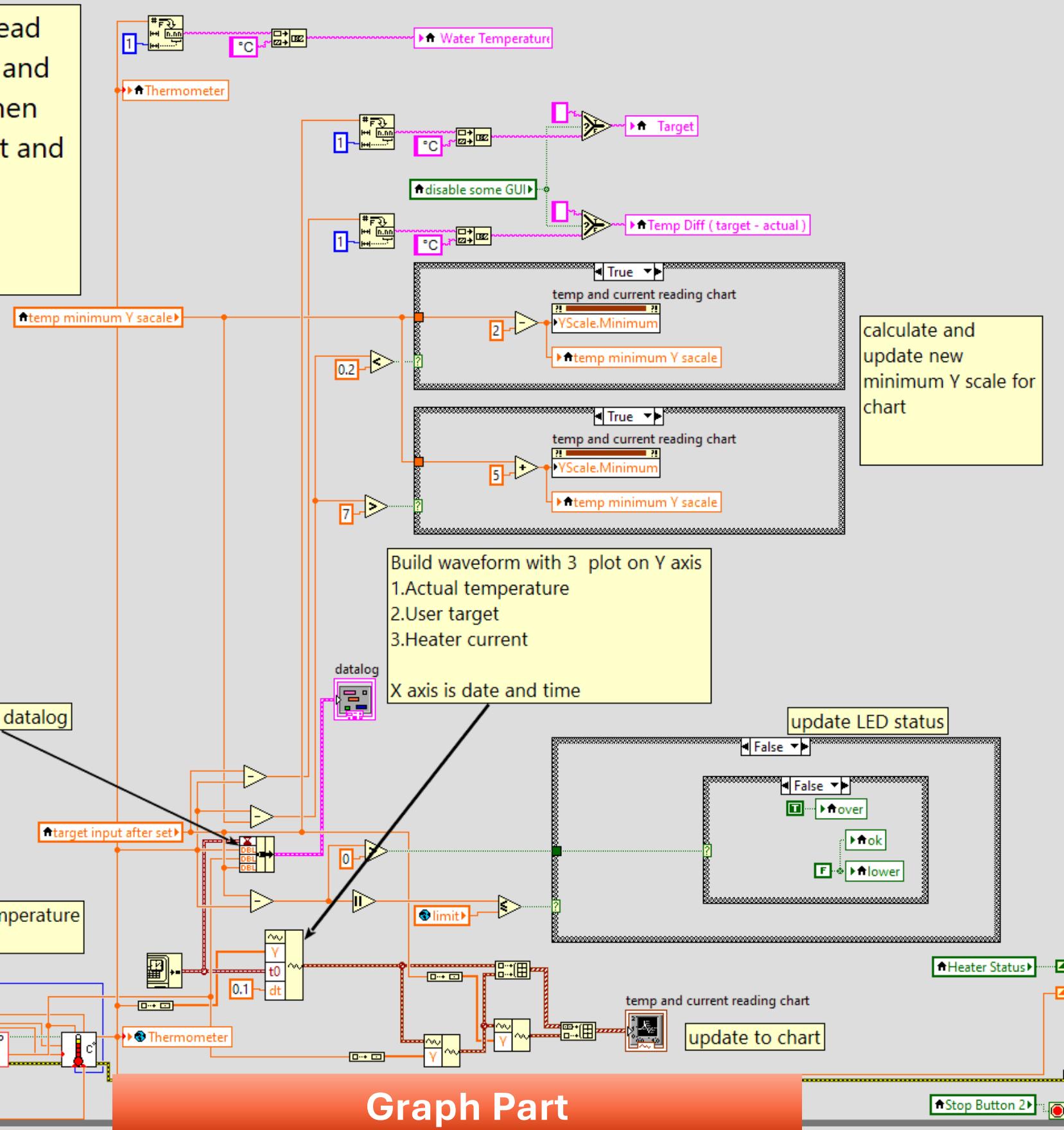
Software Tools and Programming Language : Labview

- Control temperature water within user limit by control heater current
- Show temperature water, heater current, and user limit on graph
- Save datalog to local
- This is only practice project
- There is no real hardware interface (only simulation)





This loop will read heater current and temperature then update to chart and local variable



Game Development

Video game development is the process of creating a video game. It is a multidisciplinary practice, involving programming, design, art, audio, user interface, and writing. Each of those may be made up of more specialized skills; art includes 3D modeling of objects, character modeling, animation, visual effects, and so on.

Software Tools And Programming Language

C#

VS Code (lightweight source code editor)

Unity3D (cross-platform game engine)

Blender (free and open-source 3D computer graphics software)

Audacity (free and open-source digital audio editor)

Inkscape (free and open-source vector graphics editor)

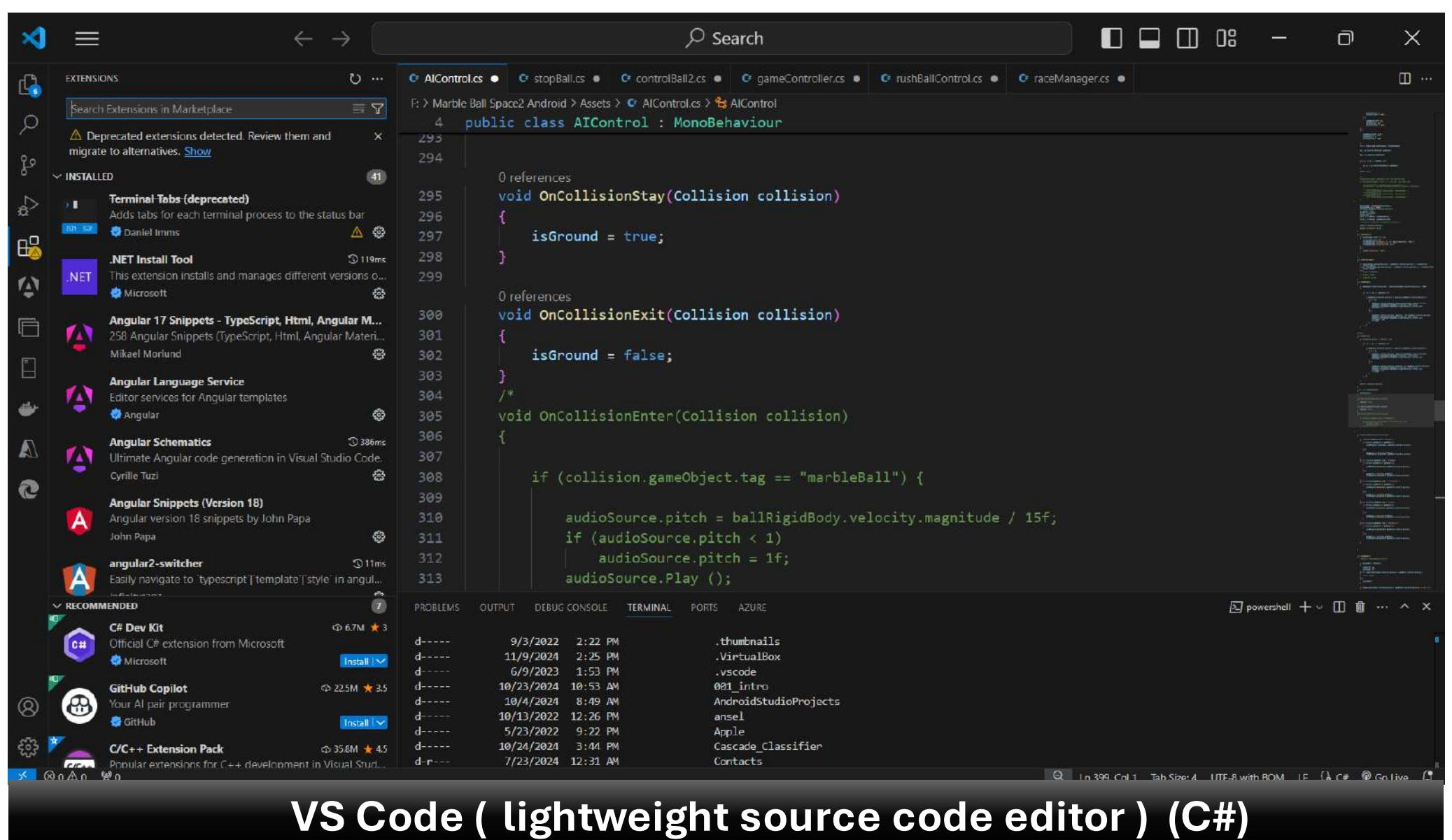
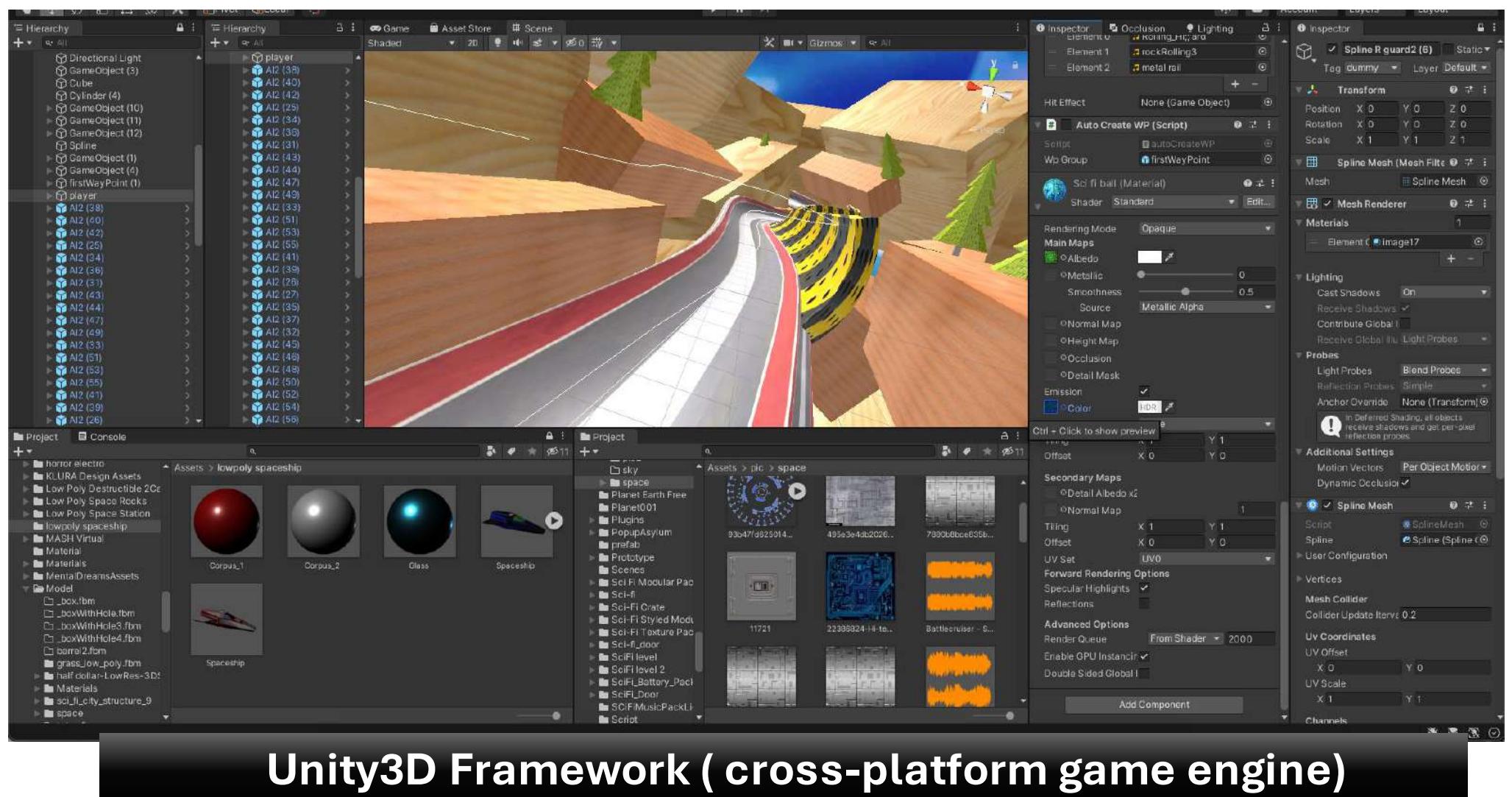
Example of My Project

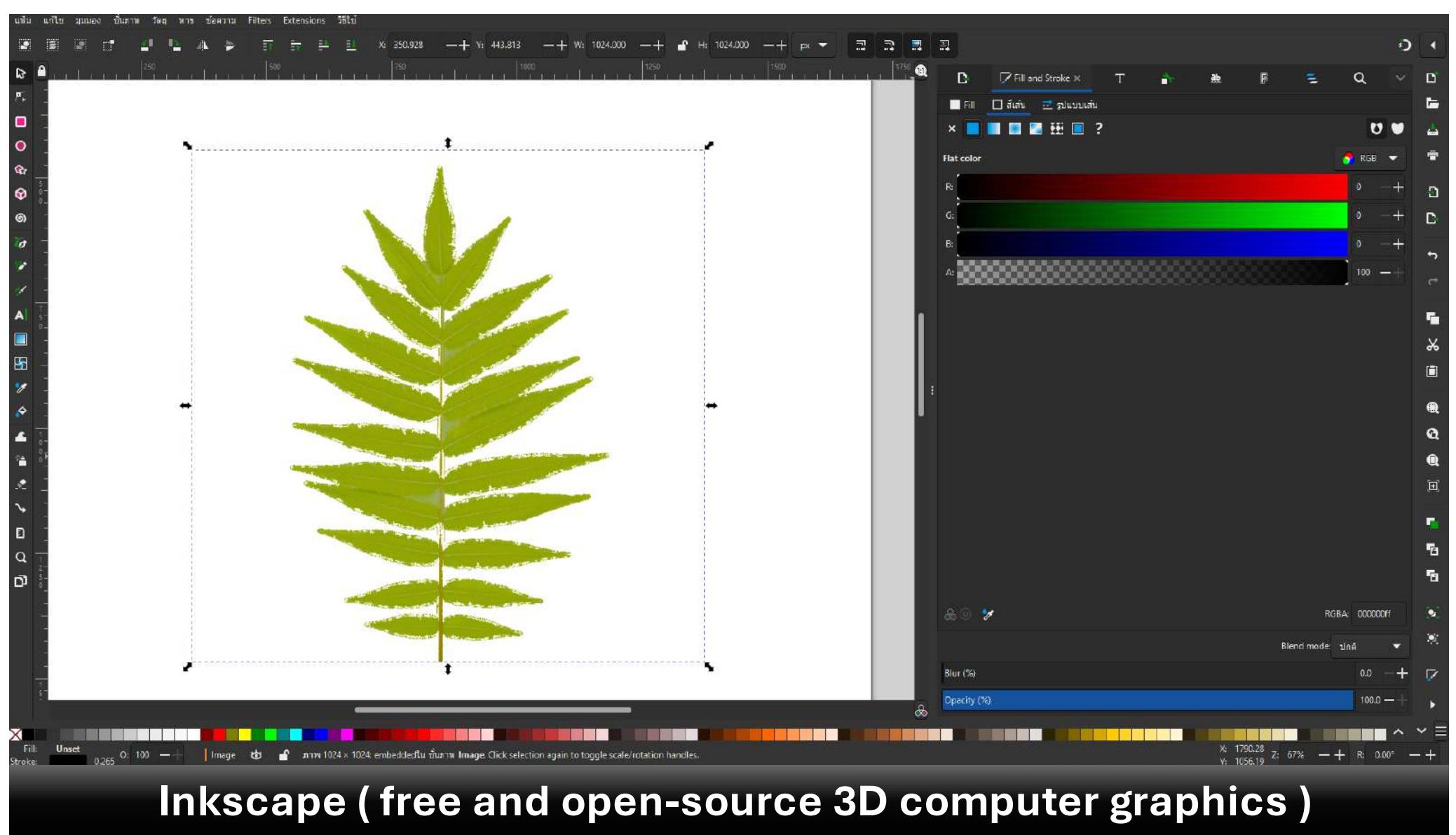
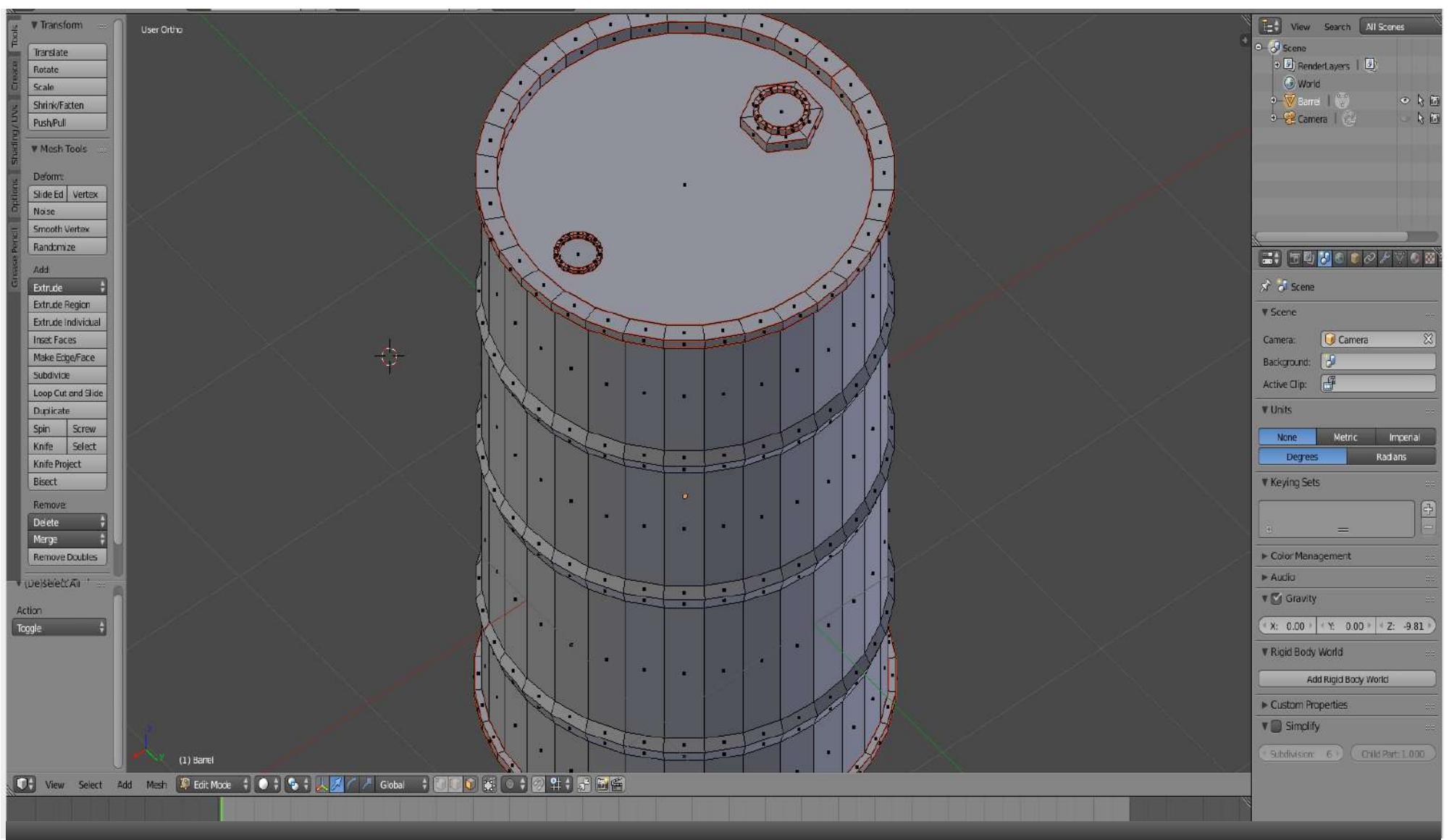
1. Marble Ball Racing 2022 (PC)

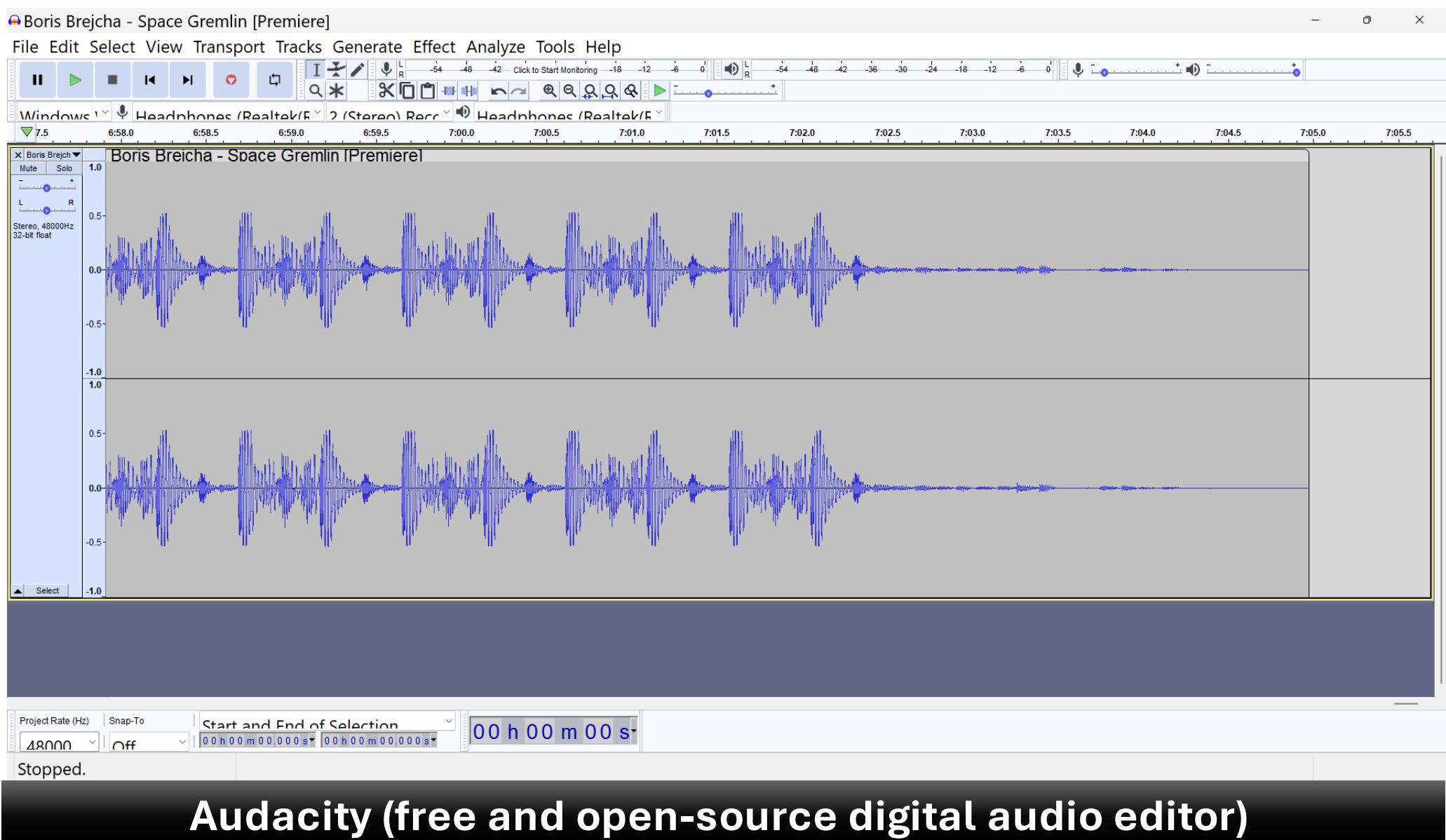
2. Marble Ball Racing (PC)

3. Ball Sort Puzzle 2077 (Android , IOS)

4. Space Rush (Andoird, IOS)

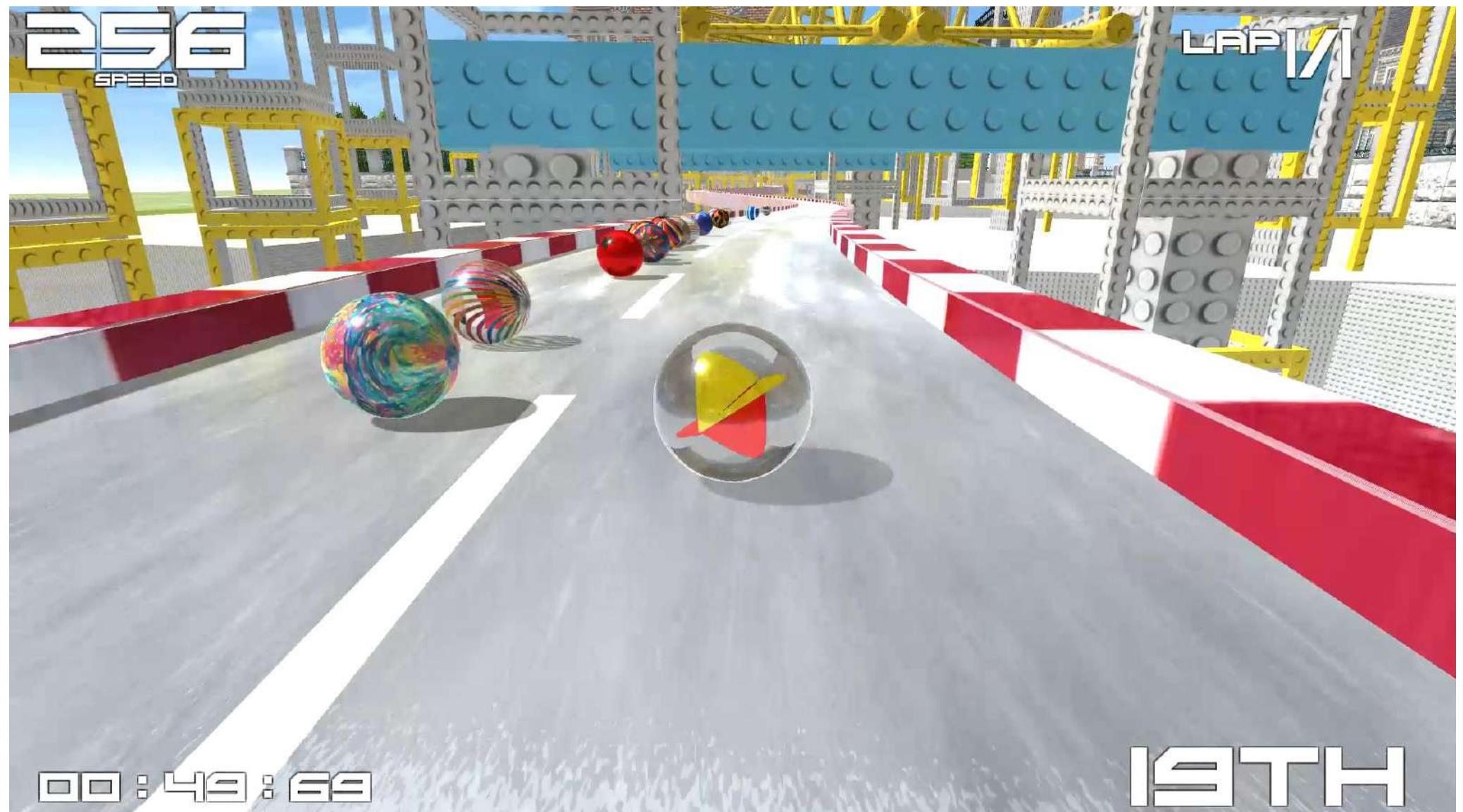






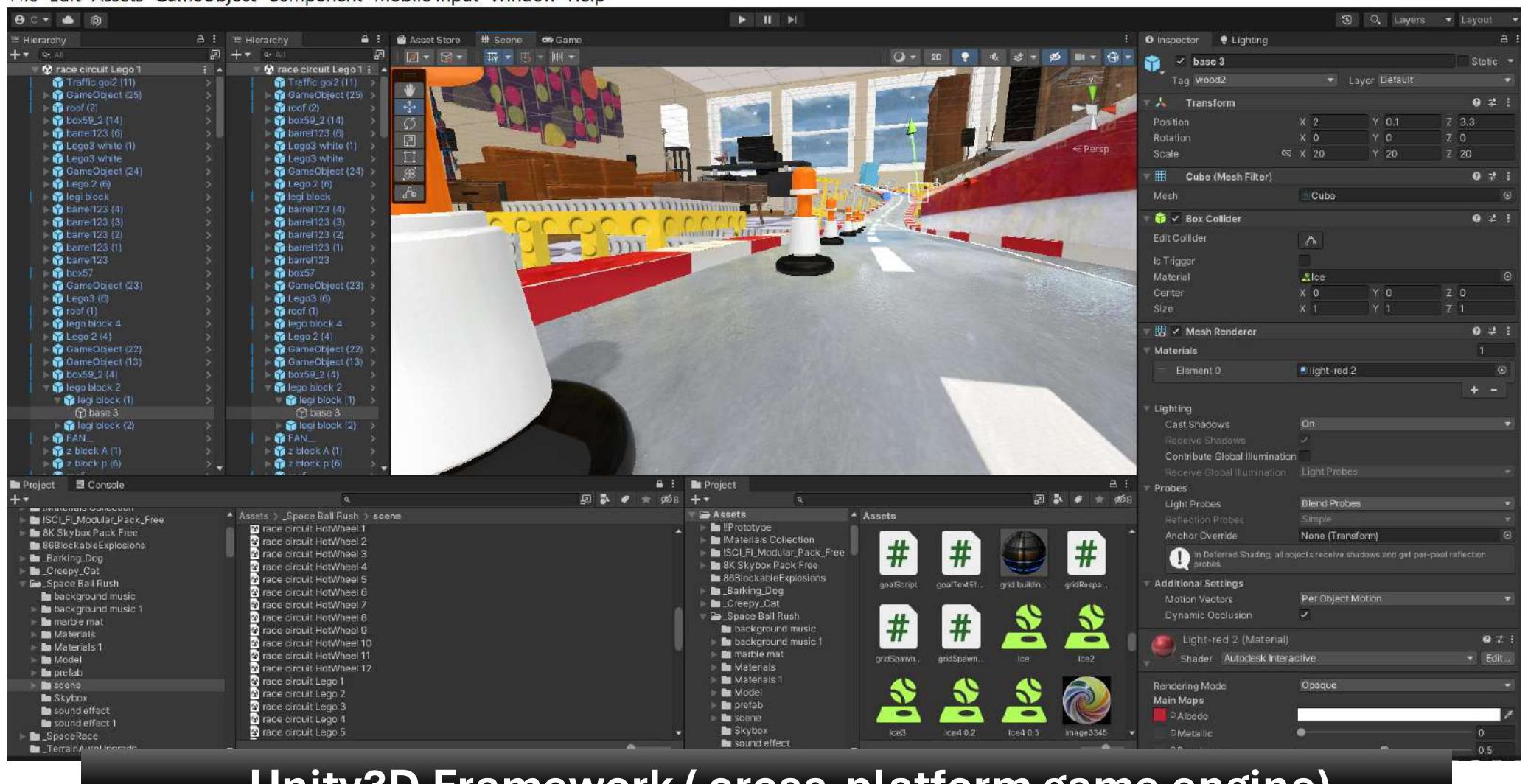
Marble Ball Racing 2022 (PC-steam, Android Phone)

Marble Ball Racing 2022 (Youtube Link)



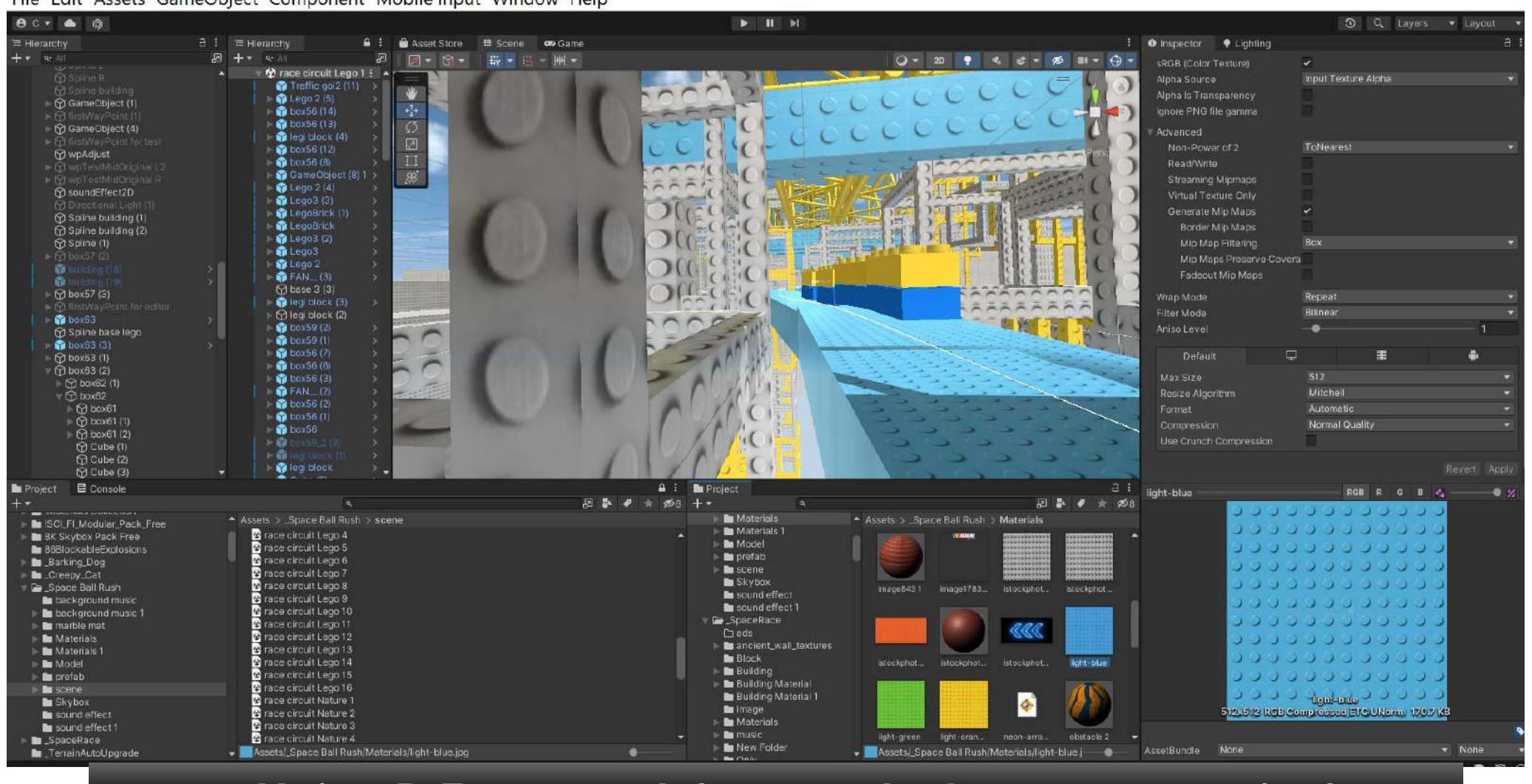
Marble Formula MAC - race circuit Lego 1 - Android - Unity 2021.3.3f1 Personal <DX11>

File Edit Assets GameObject Component Mobile Input Window Help



Marble Formula MAC - race circuit Lego 15 - Android - Unity 2021.3.3f1 Personal <DX11>

File Edit Assets GameObject Component Mobile Input Window Help



A screenshot of the Visual Studio Code interface in dark mode. The main editor window displays C# code for a class named `controlBall2`. The code handles physics logic, including force calculations based on player input and ground status, and rotation. The sidebar shows file navigation and a search bar at the top. The status bar at the bottom indicates the file is `controlBall2.cs`, line 580, column 1, with tab size 4, UTF-8 encoding, and C# language selected.

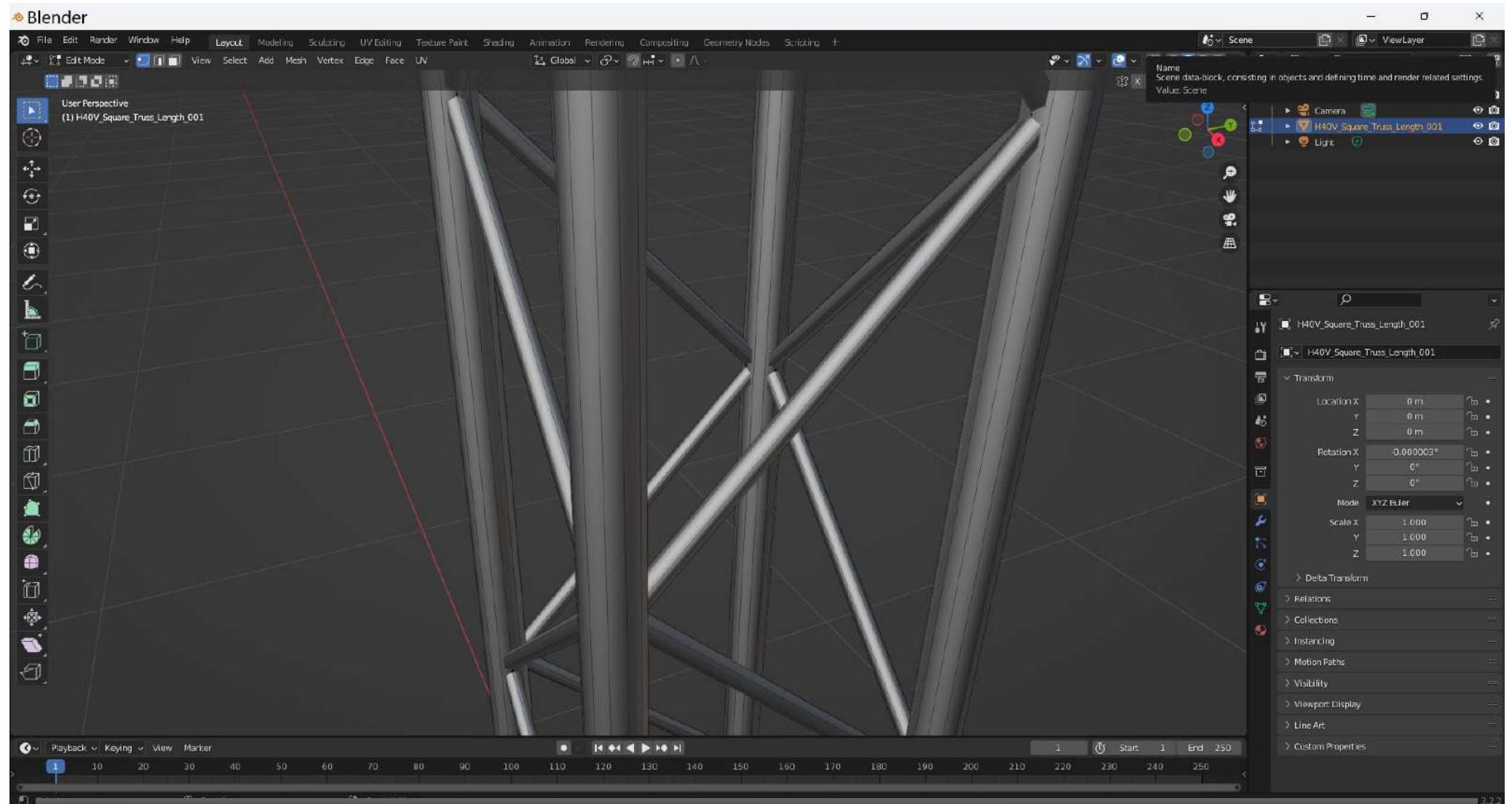
```
8 public class controlBall2 : MonoBehaviour {
488     void FixedUpdate()
575     {
576         right = 0;
577         left = 0;
578     }
579     //comment above when build to mobile
580
581
582
583
584     if (isGround == true)
585     {
586         ballRigidbody.AddForce(transform.forward * forward_back * force);
587         ballRigidbody.AddForce((left + right) * (force * 0f + ballRigidbody.velocity.magnitude * 10f) * transform.right);
588
589
590
591
592     }
593     else if (isGround == false)
594     {
595         ballRigidbody.AddForce(transform.forward * forward_back * force * 0.5f);
596         ballRigidbody.AddForce((left + right) * (force * 0f + ballRigidbody.velocity.magnitude * 5f) * transform.right);
597
598
599     }
600
601     transform.Rotate(0, left + right, 0);
602
603
604 }
```

VS Code (lightweight source code editor) (C#)

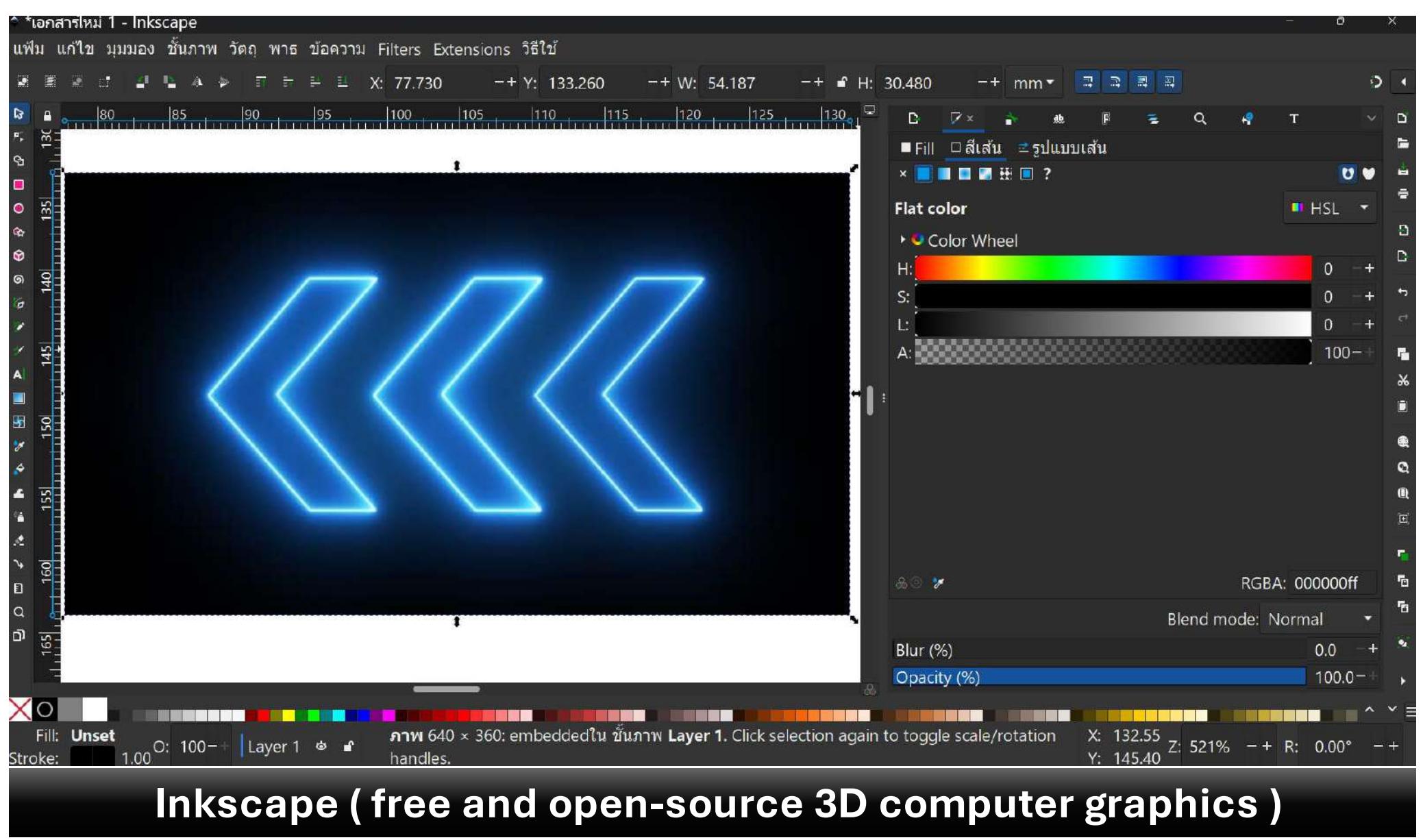
A screenshot of the Visual Studio Code interface in dark mode. The main editor window displays C# code for a class named `raceManager`. The code implements a racing order calculation logic, including clearing a list of racers and sorting them based on their point index float. The sidebar shows file navigation and a search bar at the top. The status bar at the bottom indicates the file is `raceManager.cs`, line 340, column 1, with tab size 4, UTF-8 encoding, and C# language selected.

```
7 public class raceManager : MonoBehaviour
315     // Update is called once per frame
316     void calculateRaceOrder()
317     {
318
319         if (isFinish == false)
320         {
321
322             racerList.Clear();
323
324             // playerZ = playerTransform.gameObject.GetComponent<controlBall2>().pointIndexFloat;
325
326
327
328             for (int i = 0; i < allRacer - 1; i++)
329             {
330                 racerList.Add(AIControlFormular_[i].pointIndexFloat);
331             }
332
333             playerZ = controlBall2_.pointIndexFloat;
334             playerIndex = playerZ;
335
336             racerList.Add(playerZ);
337
338             racerList.Sort();
339
340
341             //Debug error order when lap change
342             /*
343             for(int i=0;i<racerList.Count-1;i++)
344             {
```

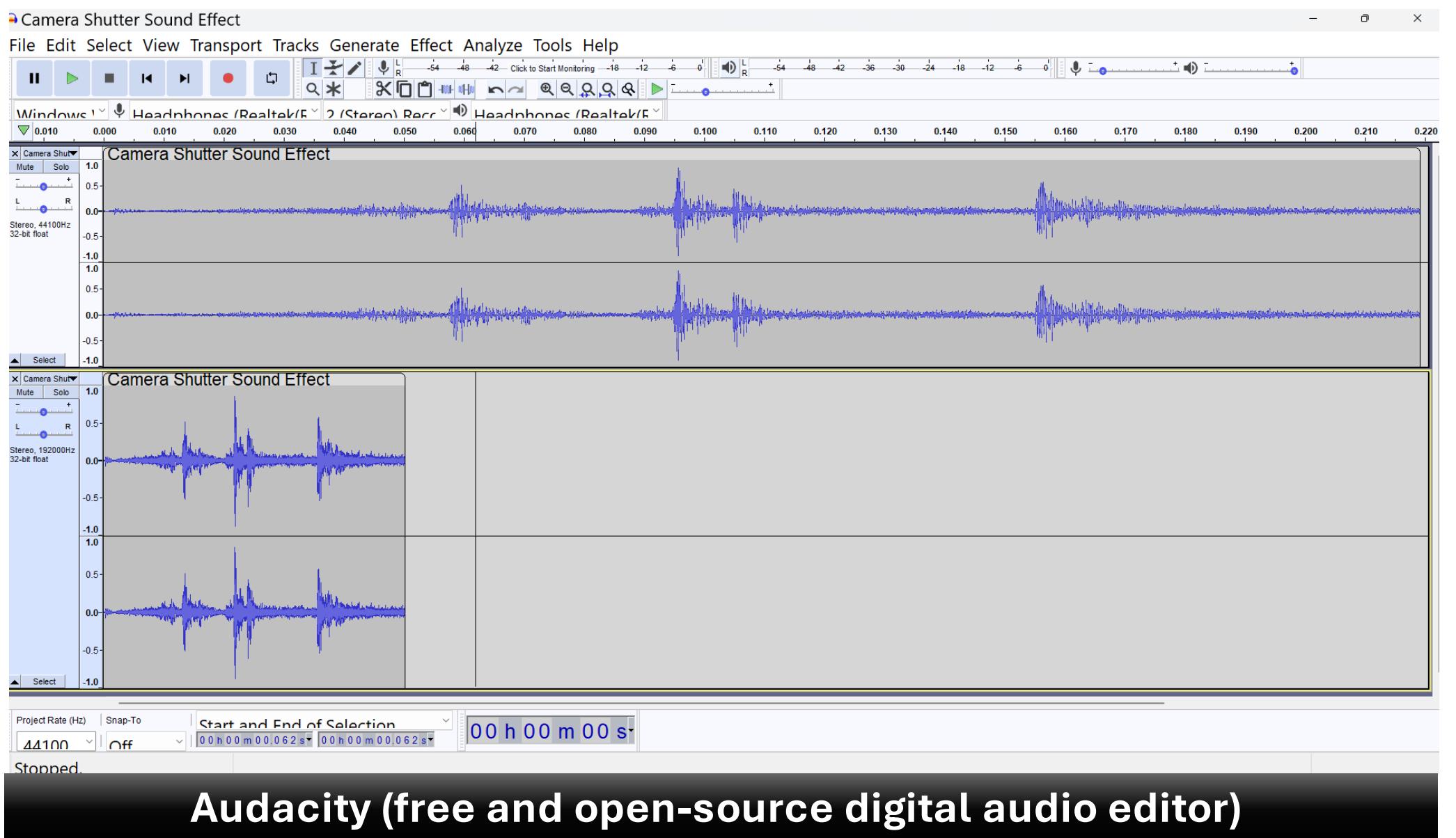
VS Code (lightweight source code editor) (C#)



Blender (free and open-source 3D computer graphics software)



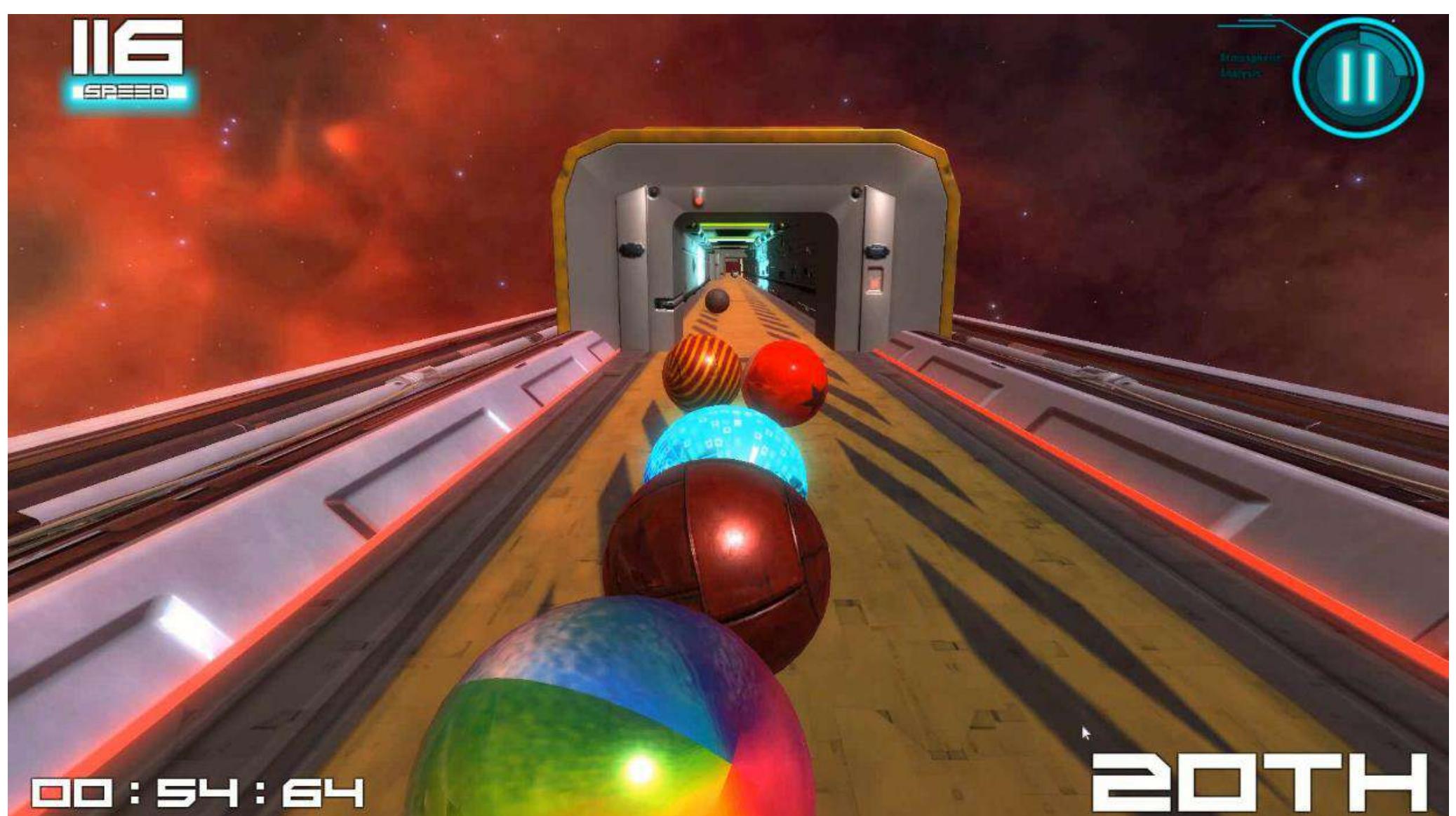
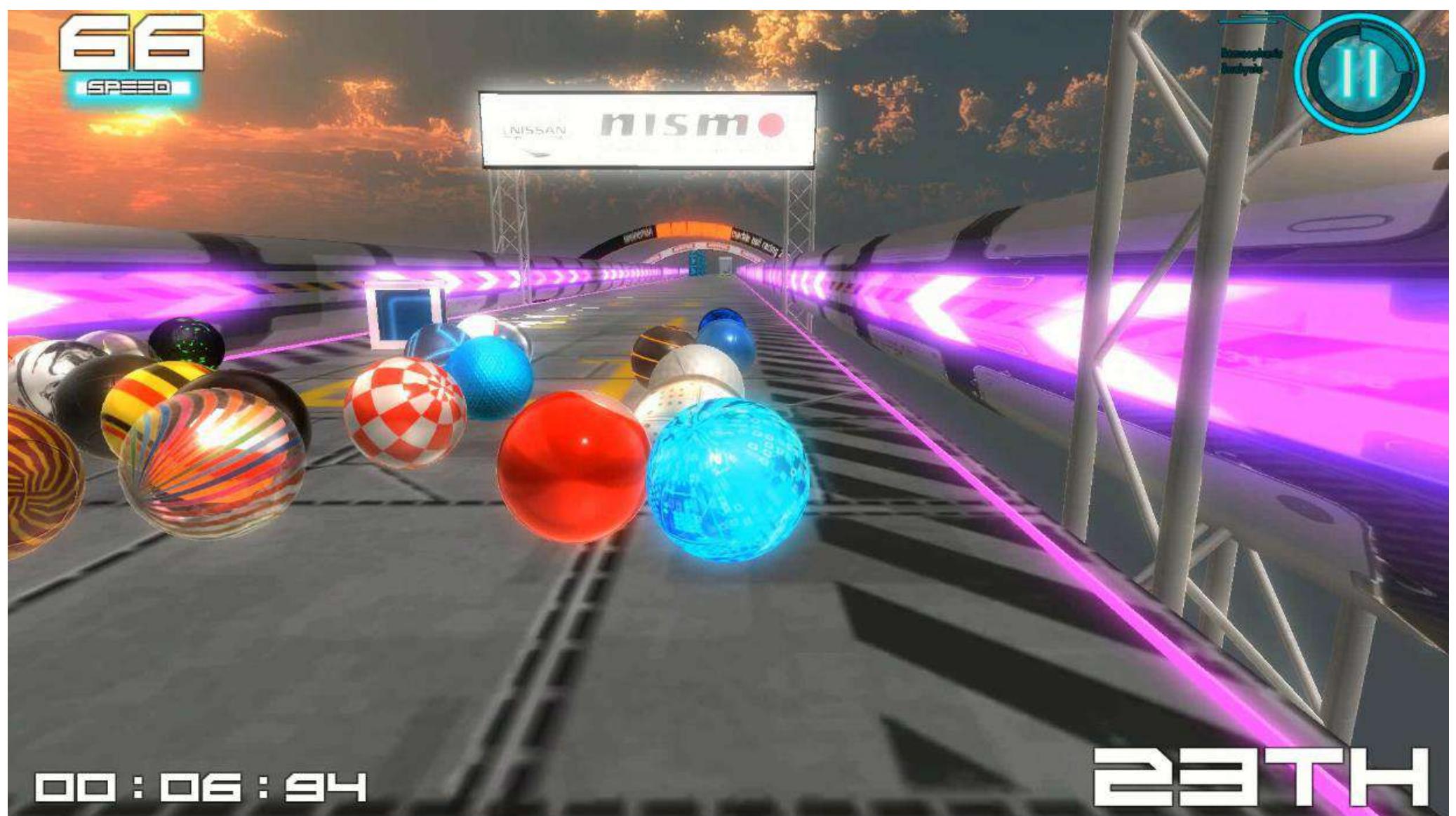
Inkscape (free and open-source 3D computer graphics)

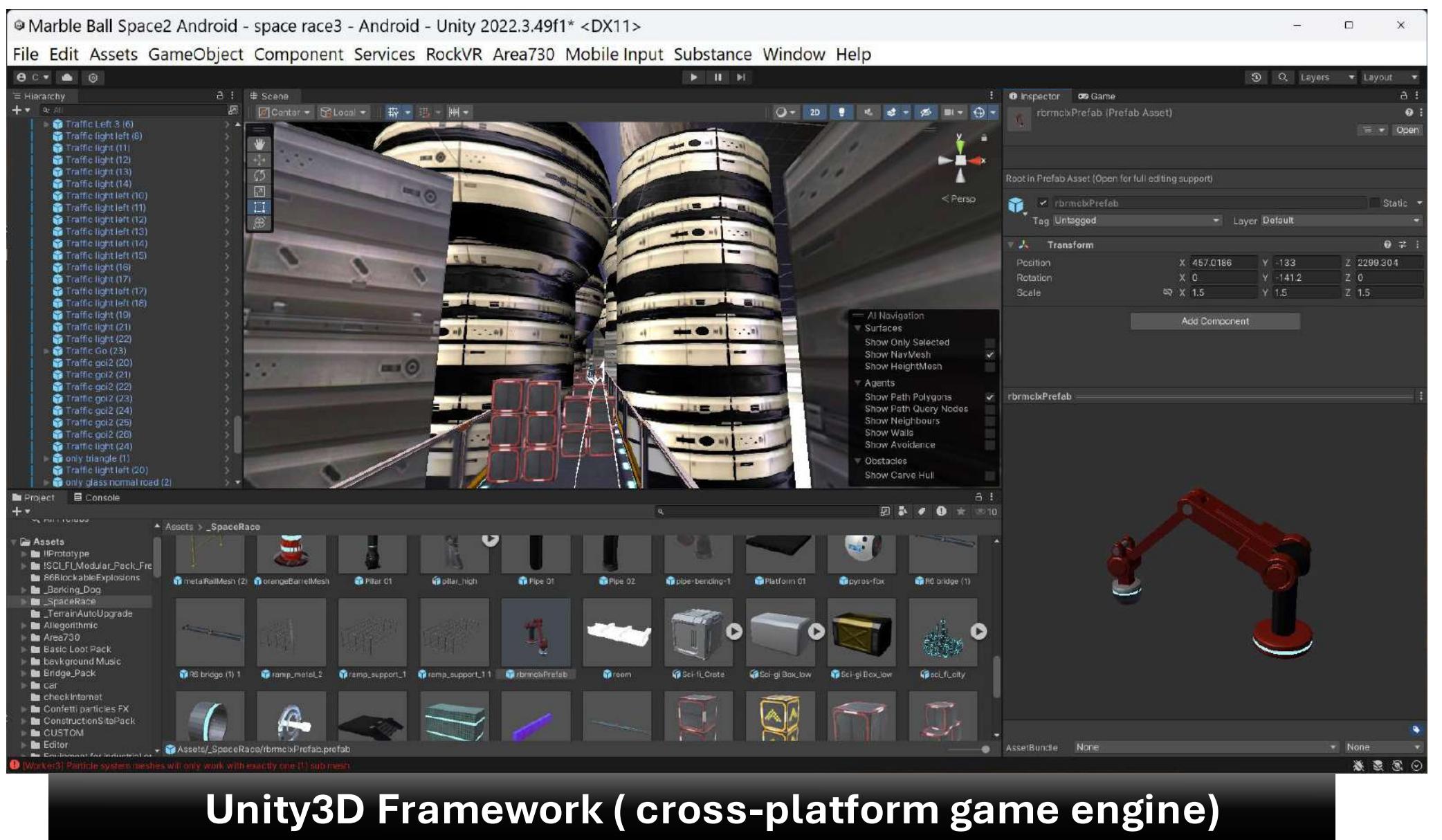
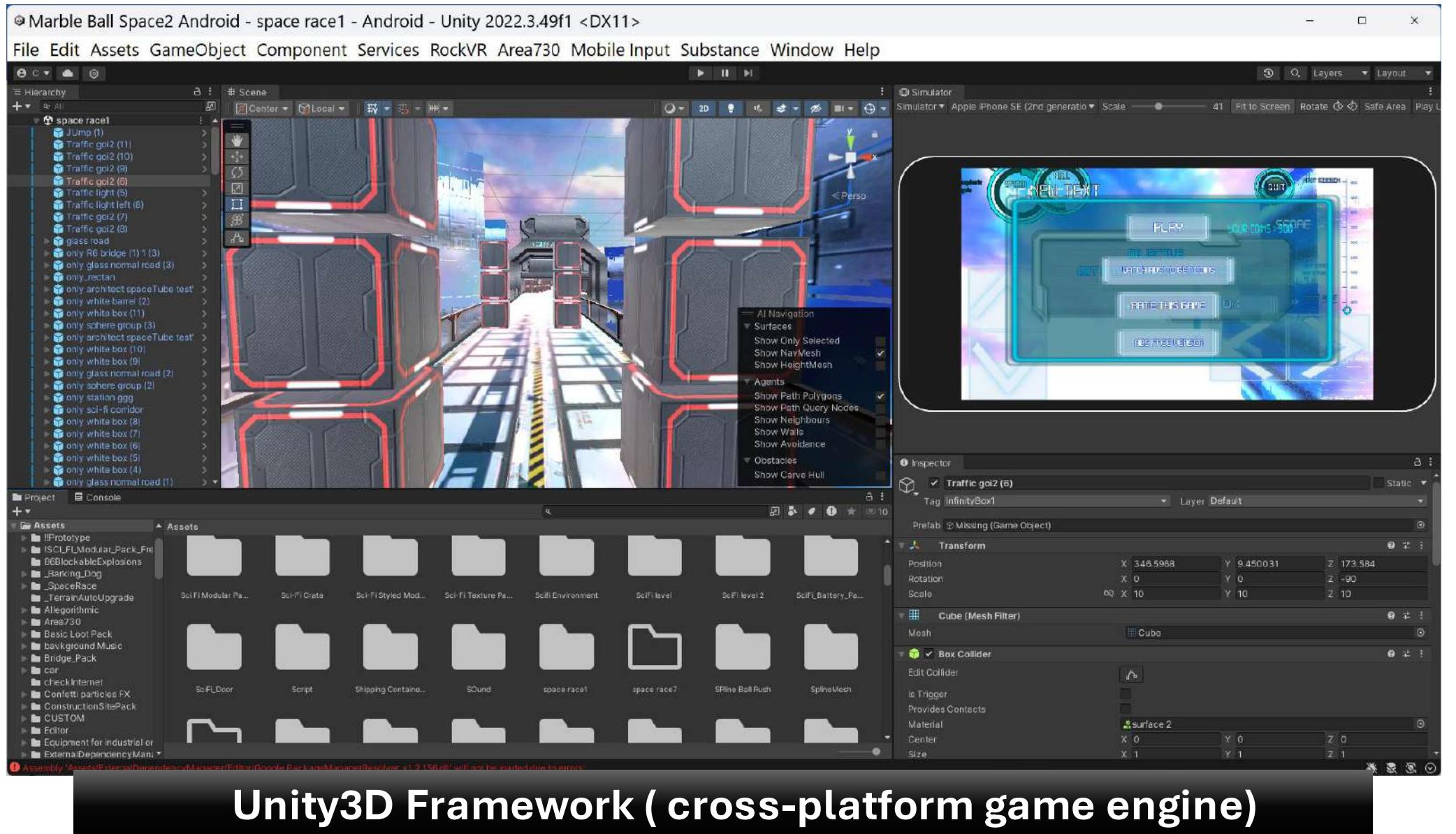


Audacity (free and open-source digital audio editor)

Marble Ball Racing (PC-steam , Android Phone, Apple iPhone)

Marble Ball Racing (Youtube Link)





A screenshot of the Visual Studio Code (VS Code) interface. The main area shows a C# script named `AIControl.cs`. The code implements a `MonoBehaviour` and contains logic for checking object height and updating position. The interface includes a left sidebar with icons for file operations, a top navigation bar with tabs for other files like `stopBall.cs`, `controlBall2.cs`, etc., and a right sidebar with various tool panels.

```
4 public class AIControl : MonoBehaviour
  ...
  void checkHeight()
  {
    if ((gameObject.transform.position.y - AIpoint[pointIndex].transform.position.y) < -100f)
    {
      for (int i = 0; i < lastPoint; i++)
      {
        if (gameObject.transform.position.z < AIpoint[i].gameObject.transform.position.z)
        {
          if (i < 6)
          {
            gameObject.transform.position = AIpoint[3].gameObject.transform.position;
            gameObject.GetComponent<Rigidbody>().velocity = Vector3.zero;
            gameObject.GetComponent<Rigidbody>().angularVelocity = Vector3.zero;
            pointIndex = 3;
          }
          else
          {
            gameObject.transform.position = AIpoint[i - 10].gameObject.transform.position;
            gameObject.GetComponent<Rigidbody>().velocity = Vector3.zero;
            gameObject.GetComponent<Rigidbody>().angularVelocity = Vector3.zero;
            pointIndex -= 10;
            i = 1000;
          }
        }
      }
    }
  }
  ...
  void FixedUpdate()
  {
    //Debug.Log(ballRigidBody.velocity);

    if (pointIndex == lastPoint)
    {
      force = 0f;
      steerForce = 0f;
      pointIndex = 0;
    }
    else if (AIpoint[pointIndex].transform.position.z > gameObject.transform.position.z)
    {
      force = force5;
    }
    else
    {
      pointIndex++;
    }

    if ((AIpoint[pointIndex].transform.position.x - gameObject.transform.position.x) > 0.1f) //0.1
    {
      if ((AIpoint[pointIndex].transform.position.x - gameObject.transform.position.x) < 3f)
      {
        if (ballRigidBody.velocity.x > 1f)
        {
          left_right = -1f * ballRigidBody.velocity.z / 100f;
        }
      }
    }
  }
}
```

VS Code (lightweight source code editor) (C#)

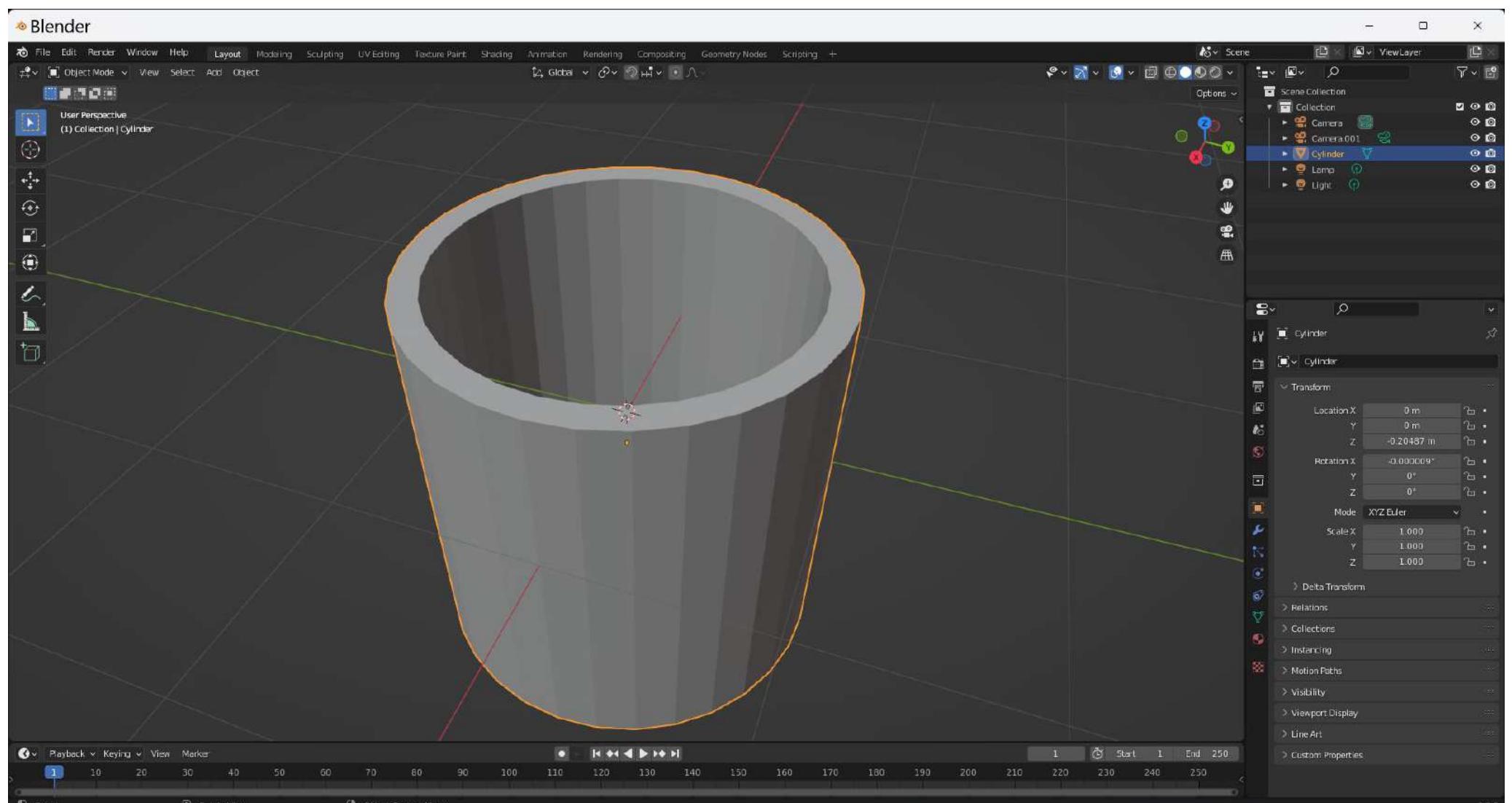
A screenshot of the Visual Studio Code (VS Code) interface, identical to the first one but showing a different section of the `AIControl.cs` script. This part of the code handles the `FixedUpdate()` method, which includes logic for calculating force and steering based on the current point index and target points. The interface elements like the sidebar, tabs, and status bar are consistent with the first screenshot.

```
400 void FixedUpdate()
401 {
402   //Debug.Log(ballRigidBody.velocity);

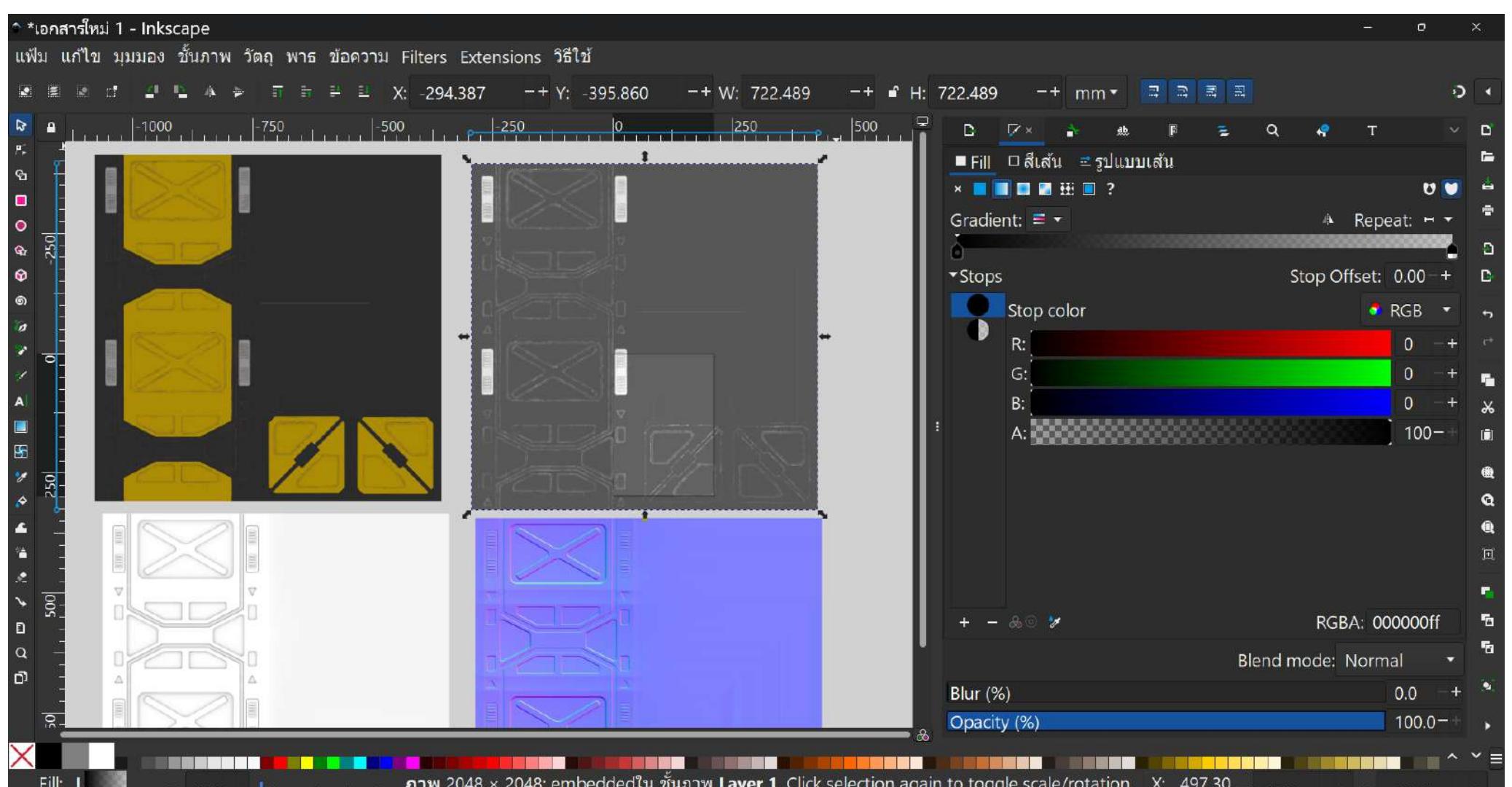
403   if (pointIndex == lastPoint)
404   {
405     force = 0f;
406     steerForce = 0f;
407     pointIndex = 0;
408   }
409   else if (AIpoint[pointIndex].transform.position.z > gameObject.transform.position.z)
410   {
411     force = force5;
412   }
413   else
414   {
415     pointIndex++;
416   }

417   if ((AIpoint[pointIndex].transform.position.x - gameObject.transform.position.x) > 0.1f) //0.1
418   {
419     if ((AIpoint[pointIndex].transform.position.x - gameObject.transform.position.x) < 3f)
420     {
421       if (ballRigidBody.velocity.x > 1f)
422       {
423         left_right = -1f * ballRigidBody.velocity.z / 100f;
424       }
425     }
426   }
427 }
428
429 }
```

VS Code (lightweight source code editor) (C#)

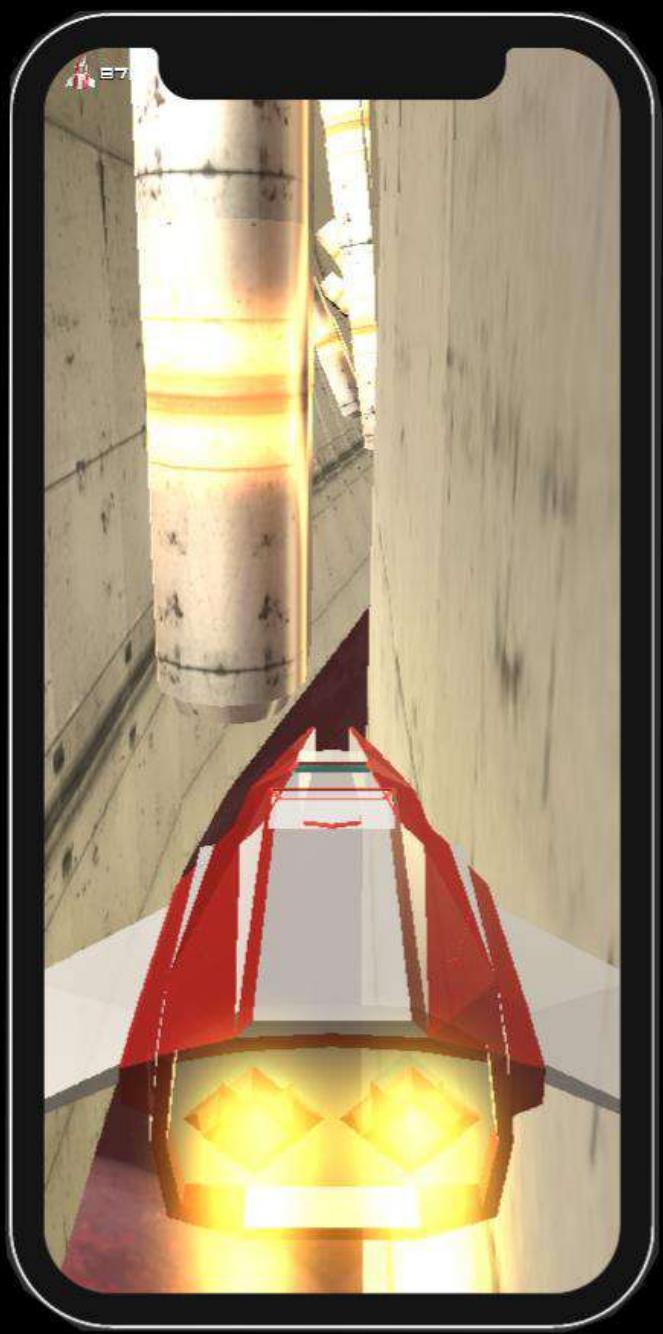
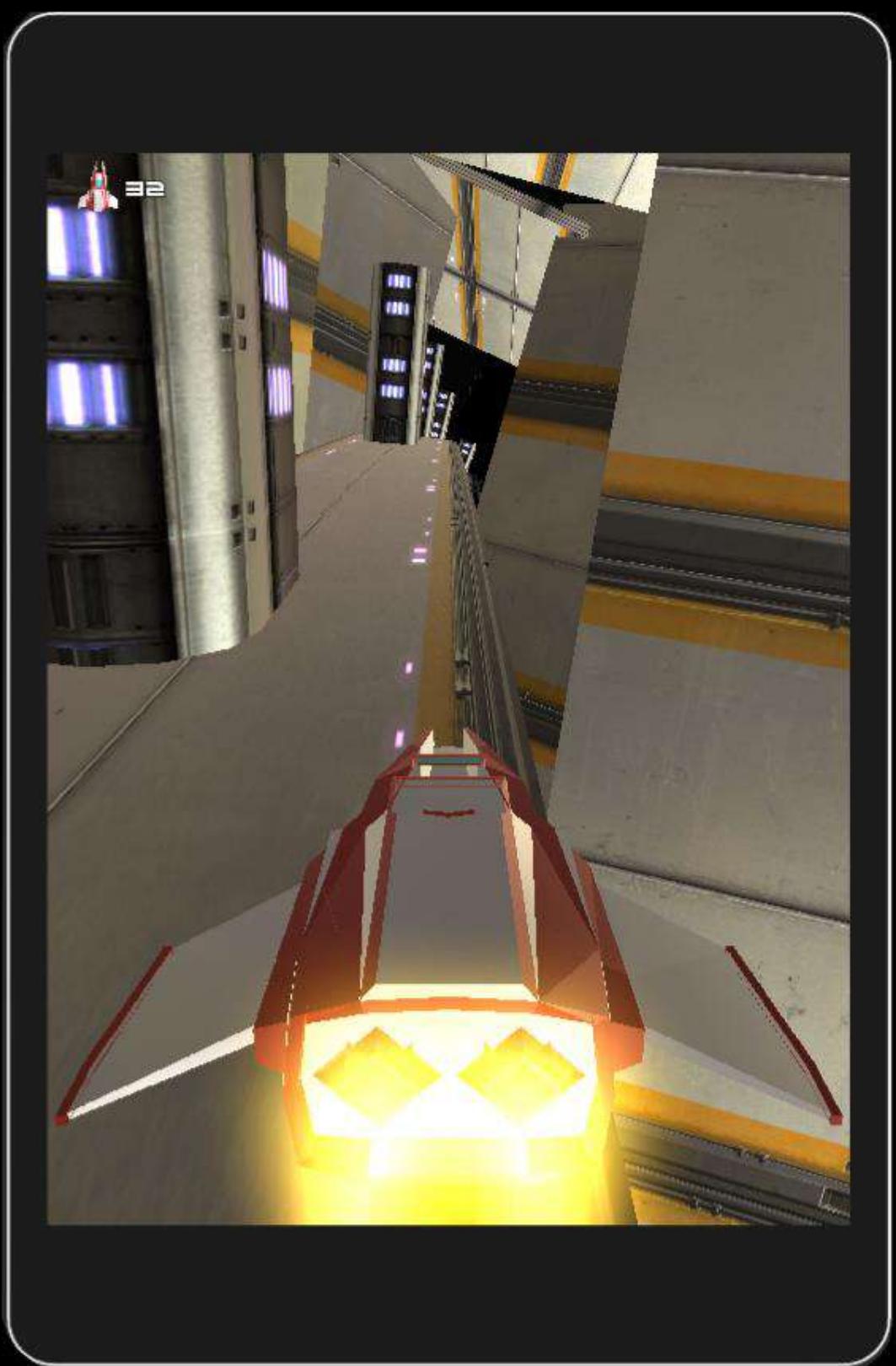
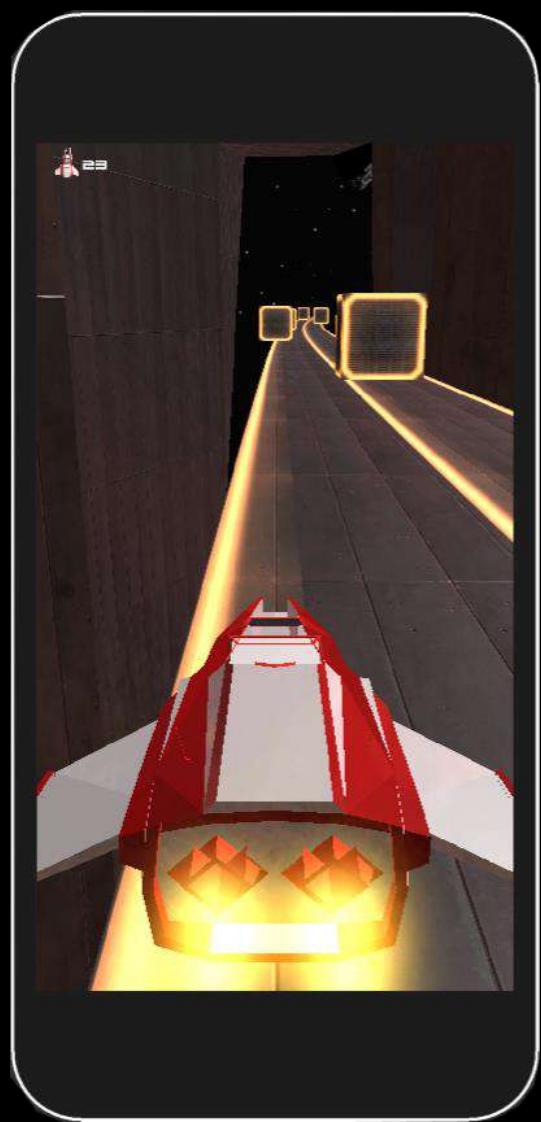


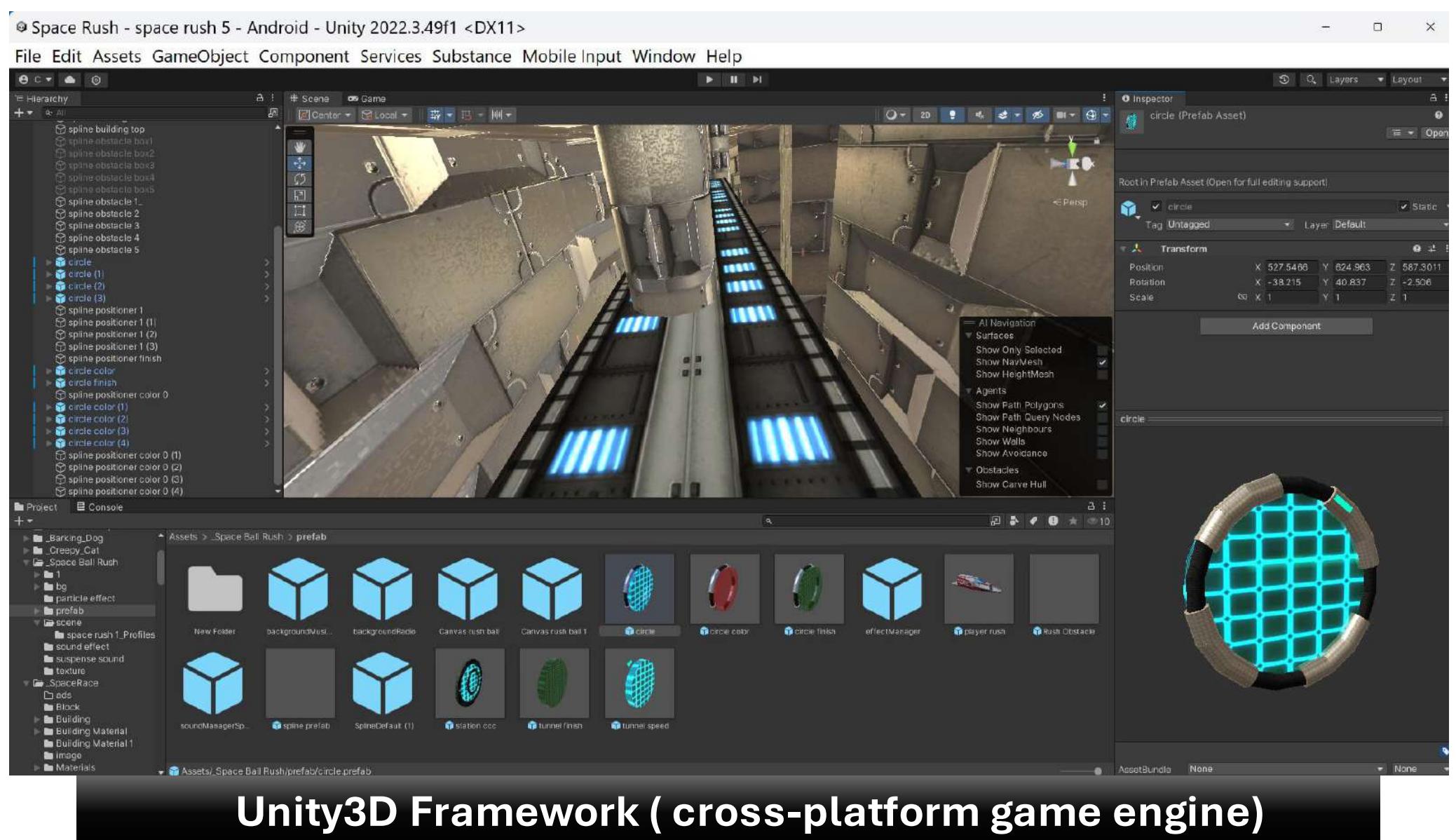
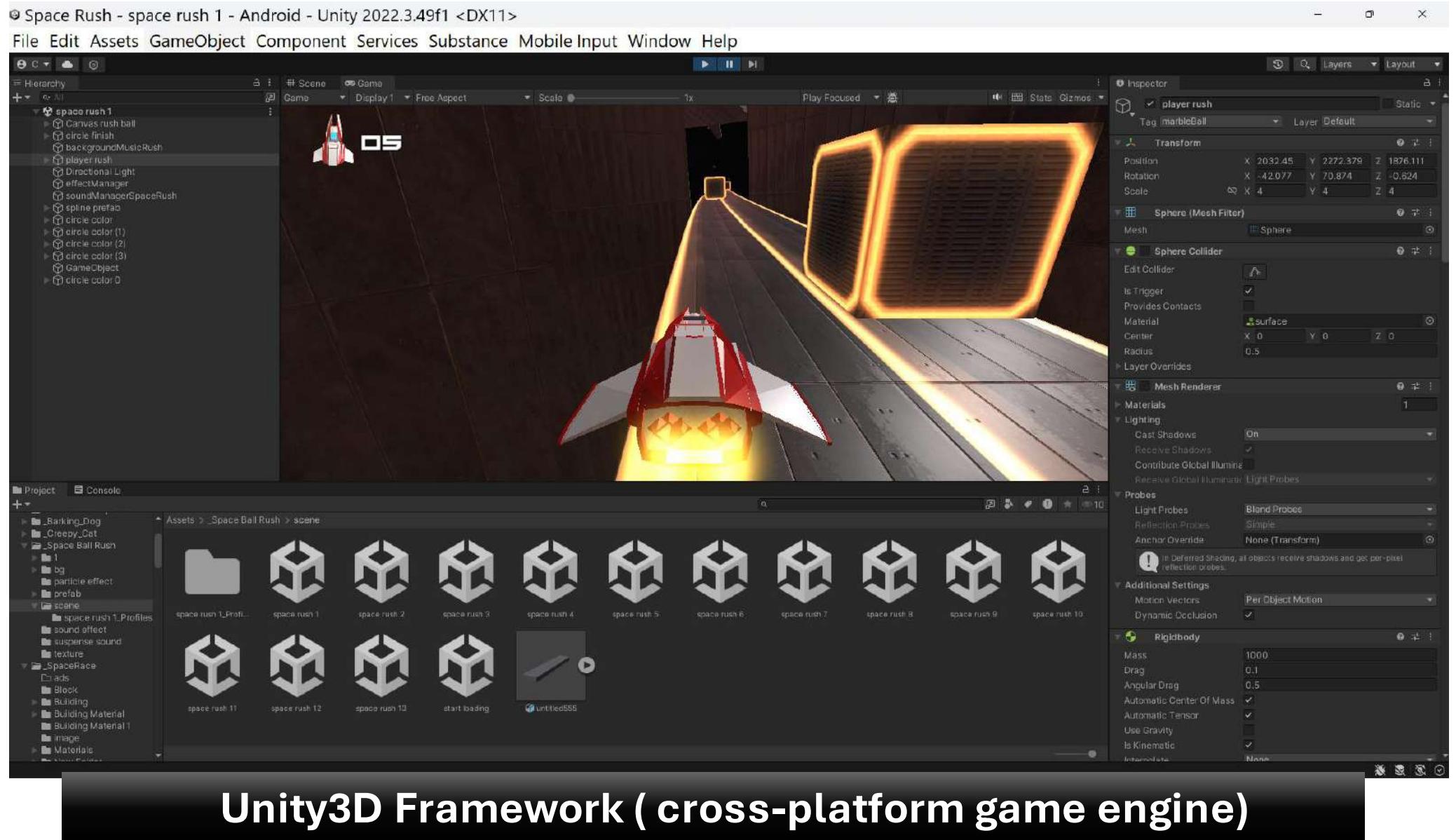
Blender (free and open-source 3D computer graphics software)



Inkscape (free and open-source 3D computer graphics)

Space Rush (Android Phone, Apple iPhone)





A screenshot of the Visual Studio Code interface. The title bar shows the file name "rushBallControl.cs". The left sidebar has icons for file operations like Open, Save, Find, and Delete. The main editor area displays C# code for the "rushBallControl" class. The code includes logic for keydown detection, flag management, and material color interpolation. The status bar at the bottom shows the file path "F:\Space Rush\Assets\rushBallControl.cs", line 403, and column 1.

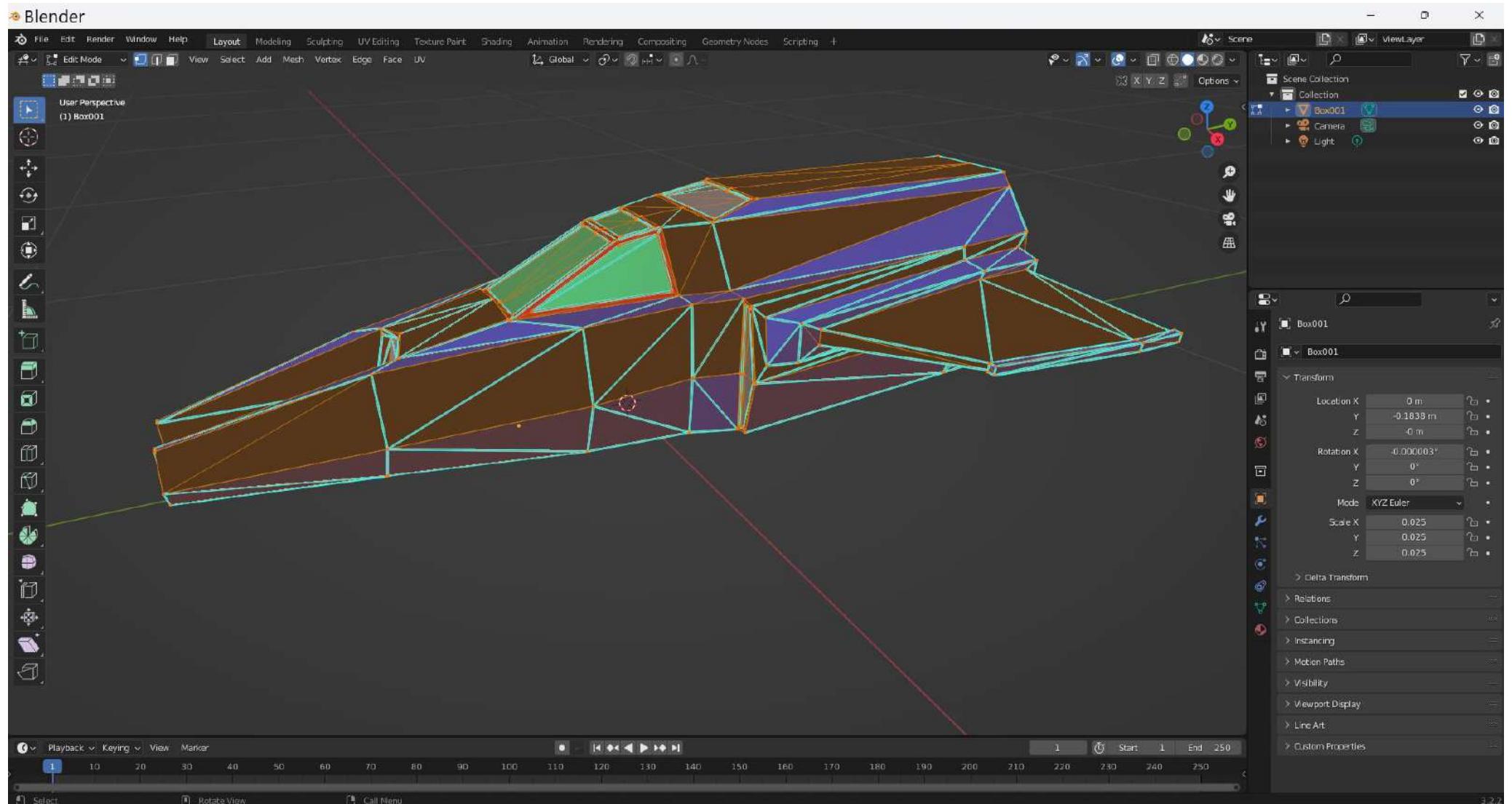
```
11 public class rushBallControl : MonoBehaviour
365     void Update()
374         keyDown = true;
375     }
376     }*/
377
378     if (moveFlag == false || controlFlag == true)
379     {
380         if (keyDown == true)
381         {
382             keyDown = false;
383             offset *= -1f;
384             moveFlag = true;
385             // controlFlag = false;
386             // Invoke("enableControl",0.2f);
387         }
388     }
389 }
390
391 if (blinkMatFlagEnable == true)
392 {
393     float lerp = Mathf.PingPong(Time.time, duration) / duration;
394     m_Material[0].color = Color.Lerp(colorStart[0], colorEnd[0], lerp);
395     m_Material[1].color = Color.Lerp(colorStart[1], colorEnd[1], lerp);
396
397     m_Material[2].color = Color.Lerp(colorStart[2], colorEnd[2], lerp);
398     m_Material[3].color = Color.Lerp(colorStart[3], colorEnd[3], lerp);
399     m_Material[4].color = Color.Lerp(colorStart[4], colorEnd[4], lerp);
400 }
```

VS Code (lightweight source code editor) (C#)

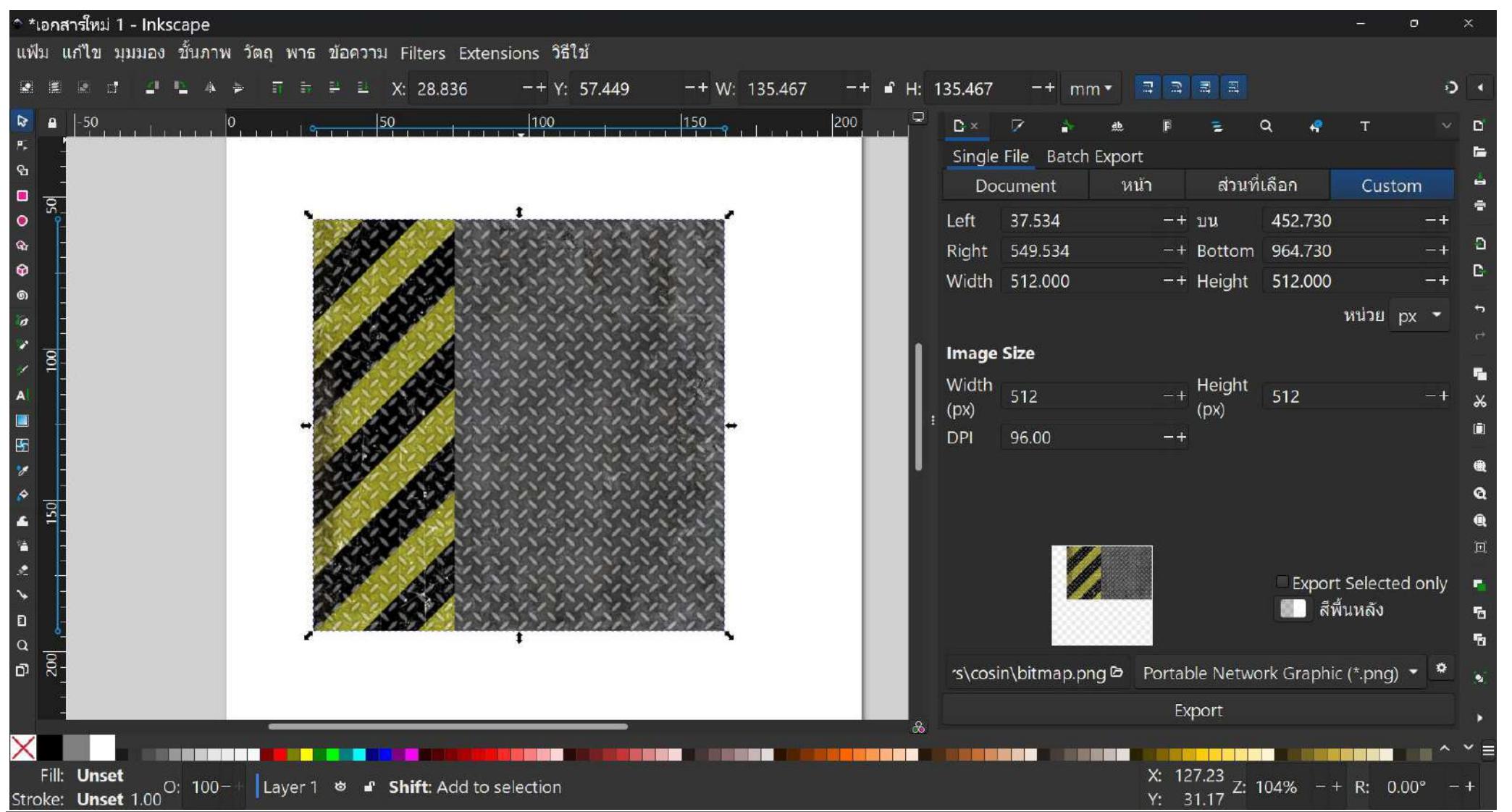
A screenshot of the Visual Studio Code interface. The title bar shows the file name "tunnelSpeed.cs". The left sidebar has icons for file operations like Open, Save, Find, and Delete. The main editor area displays C# code for the "tunnelSpeed" class. It includes an "OnTriggerEnter" method for a specific tag and a "speedUp" method that increases follower speed and camera effects. The status bar at the bottom shows the file path "F:\Space Rush\Assets\tunnelSpeed.cs", line 66, and column 1.

```
7 public class tunnelSpeed : MonoBehaviour
38     void OnTriggerEnter(Collider other)
39     {
40         if (other.tag == "spaceshipzxcv")
41         {
42             tunnelSpeed.cancelFlag = false;
43             soundManagerSpaceRush.playSound(0);
44             //Invoke("speedUp",4f);
45             speedUp();
46         }
47     }
48
49     void speedUp()
50     {
51         if (tunnelSpeed.cancelFlag == false)
52         {
53             follower.followSpeed += 100f;
54
55             cameraRushBallNew.cameraMoveFlag = 1;
56             var main = flame.main;
57             var main2 = flame2.main;
58             main.startSize = 2f;
59             main.startLifetime = 2f;
60             main2.startSize = 2f;
61             main2.startLifetime = 2f;
62
63             Invoke("speedNormal", 5f);
64             Invoke("camFlagNormal", 8f);
65         }
66     }
```

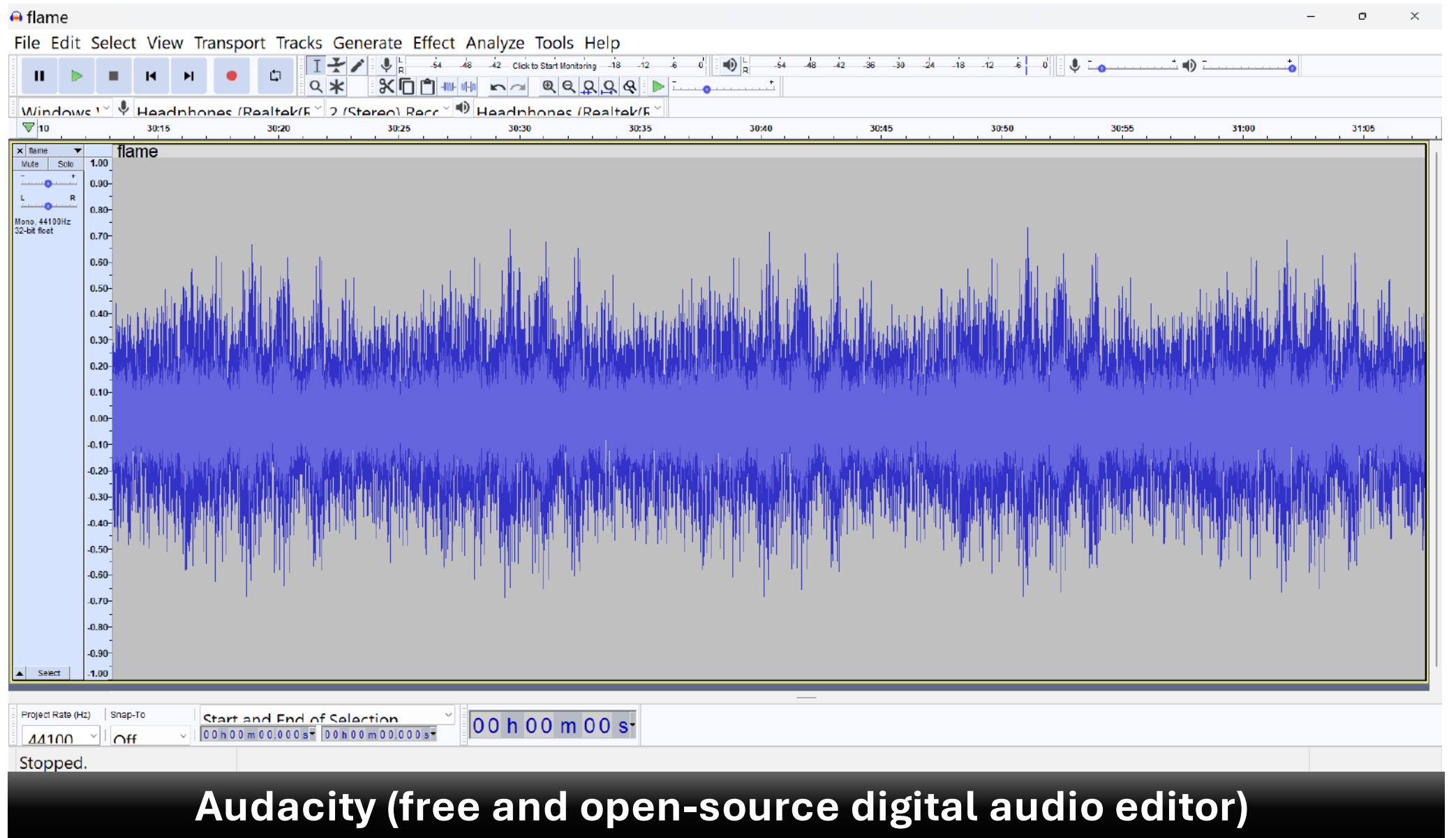
VS Code (lightweight source code editor) (C#)



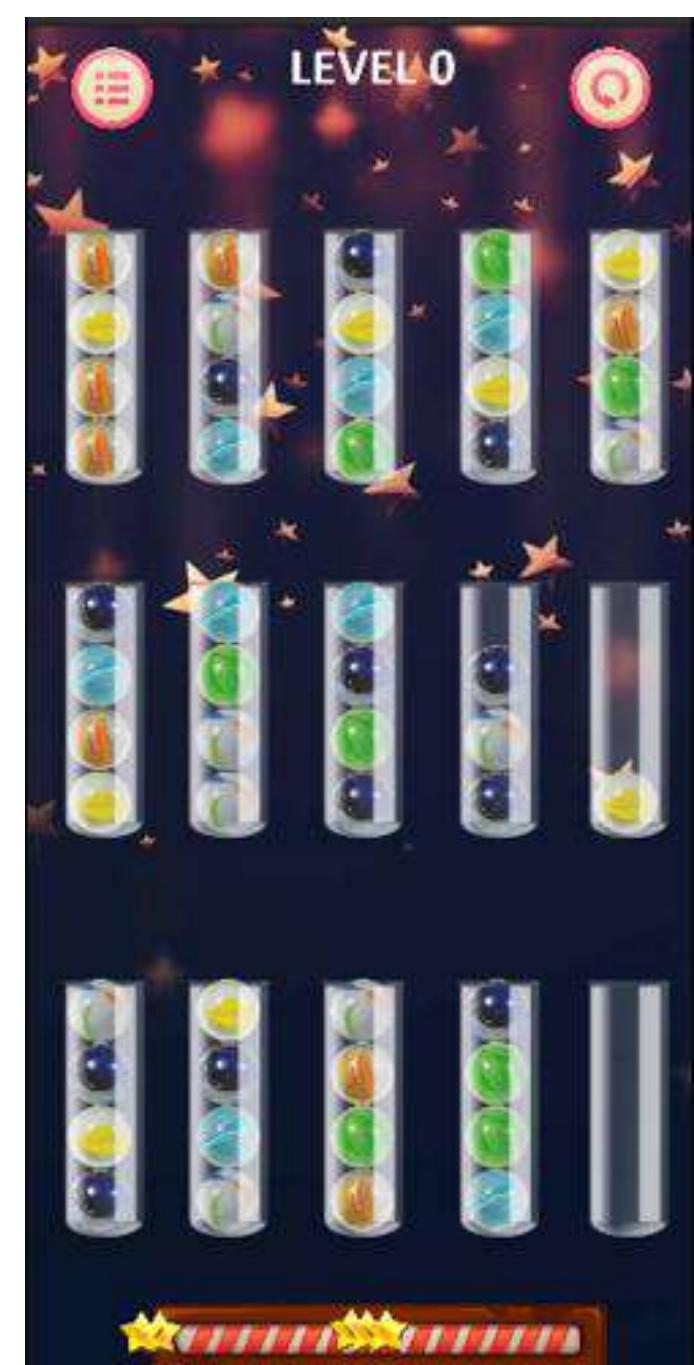
Blender (free and open-source 3D computer graphics software)

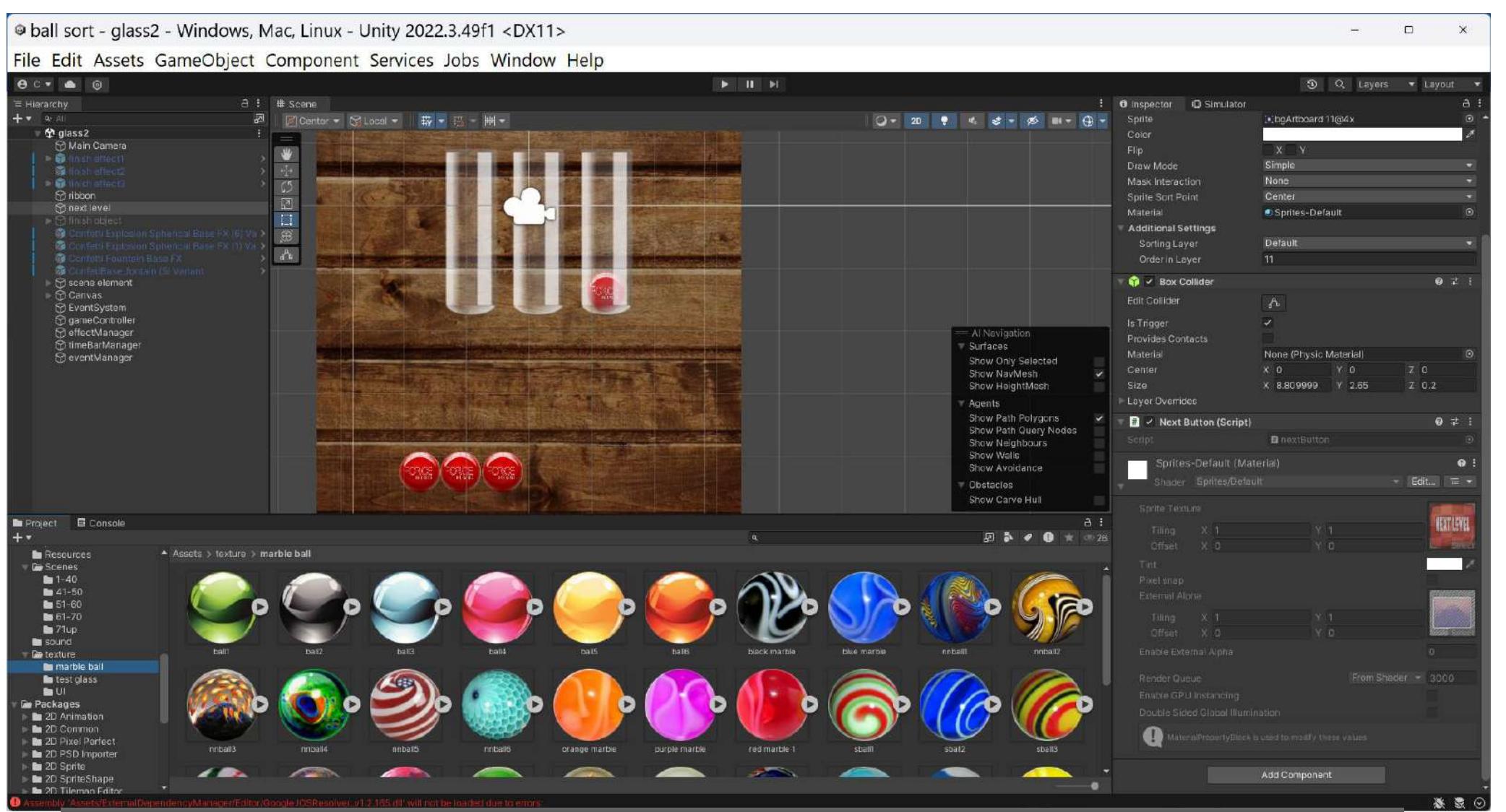
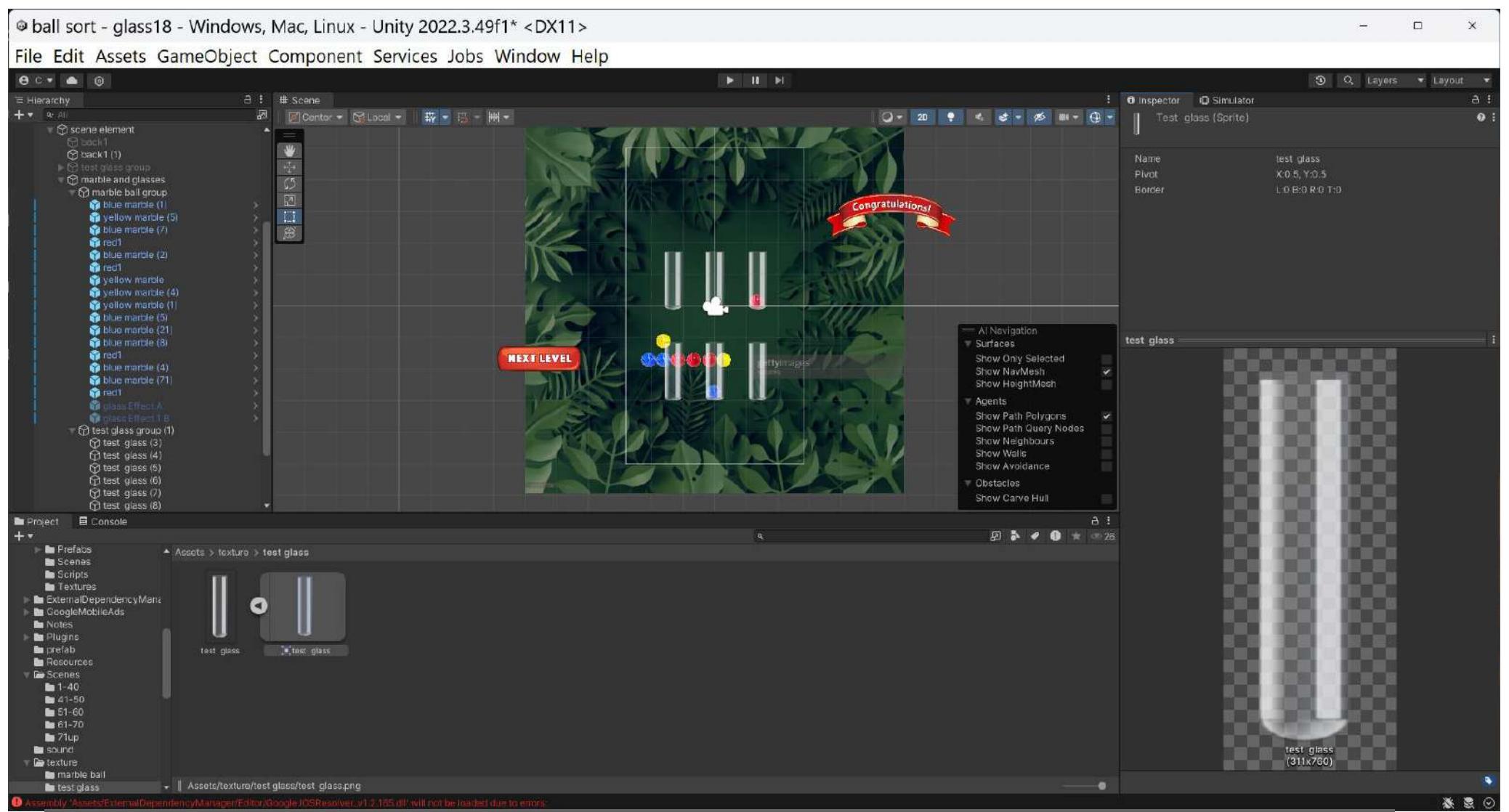


Inkscape (free and open-source 3D computer graphics)

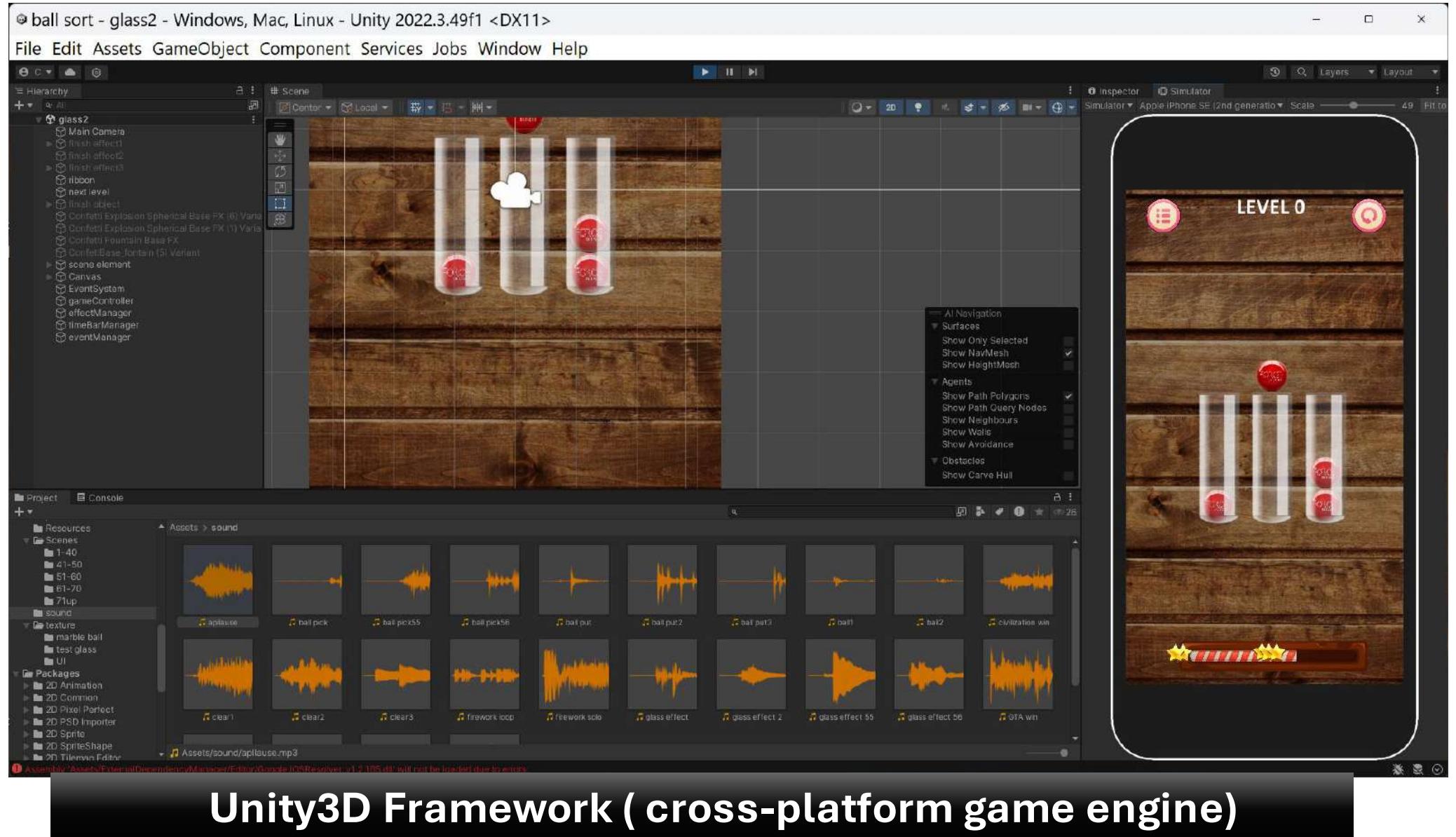


Ball Sort Puzzle (Android phone, Apple iPhone)





Unity3D Framework (cross-platform game engine)



The VS Code interface is shown, displaying the 'gameController.cs' file. The code implements a 'calculateFirstSpawn' method to initialize ball positions. It uses arrays 'glassFirstRandom' and 'manualFirstPlace' to store spawn locations. It calculates a random number of forced spawns ('numberOfForce4') and then iterates from 'glassNumber - 2' down to 0 to set the first spawn location for each glass tube. The code includes comments and debug logging.

```

public class gameController : MonoBehaviour
{
    void calculateFirstSpawn()
    {
        if (IsManualPlace == true)
        {
            glassFirstRandom[glassNumber - 1] = manualFirstPlace[glassNumber - 1];
        }
        else
        {
            glassFirstRandom[glassNumber - 1] = Random.Range(0, 5);
        }

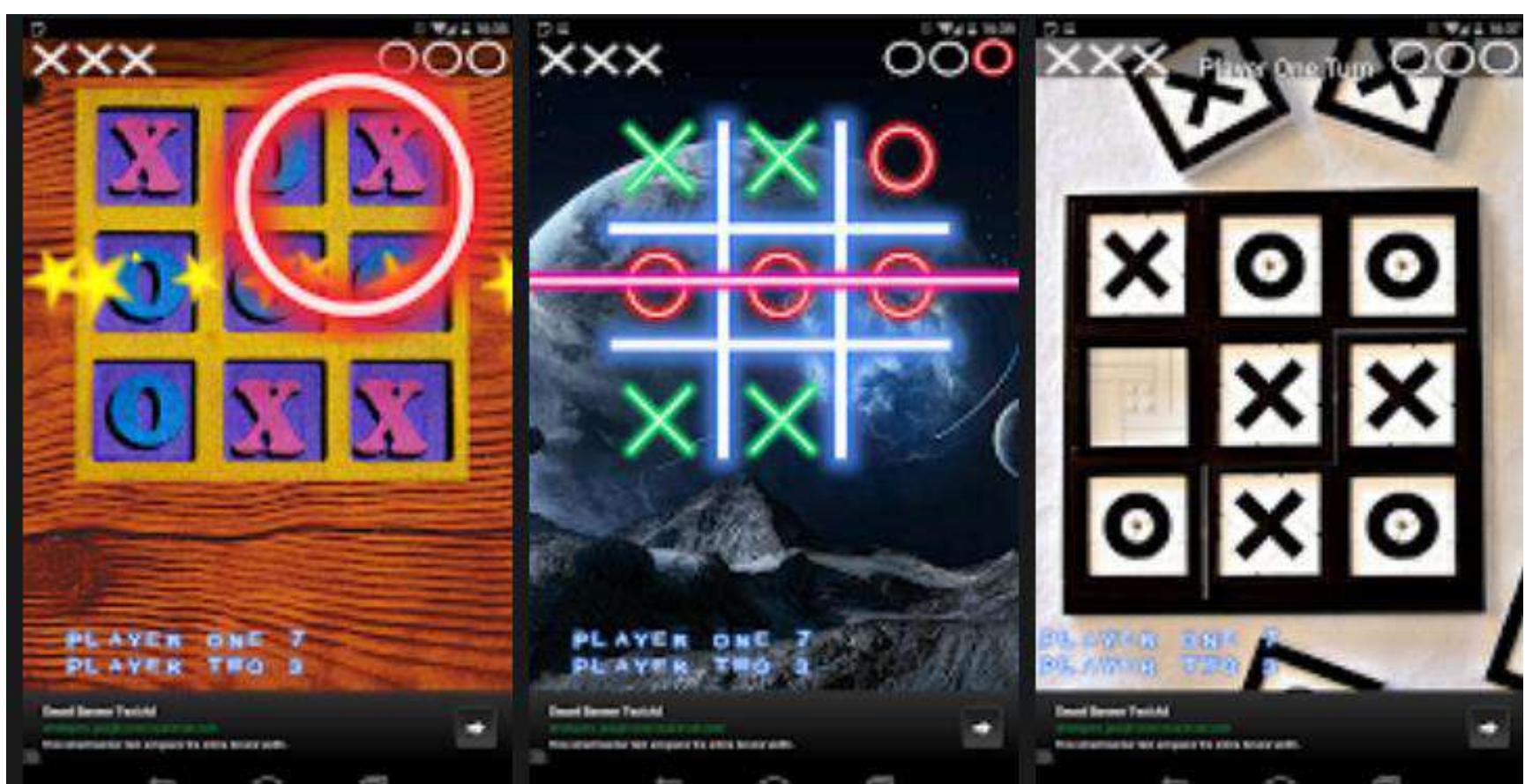
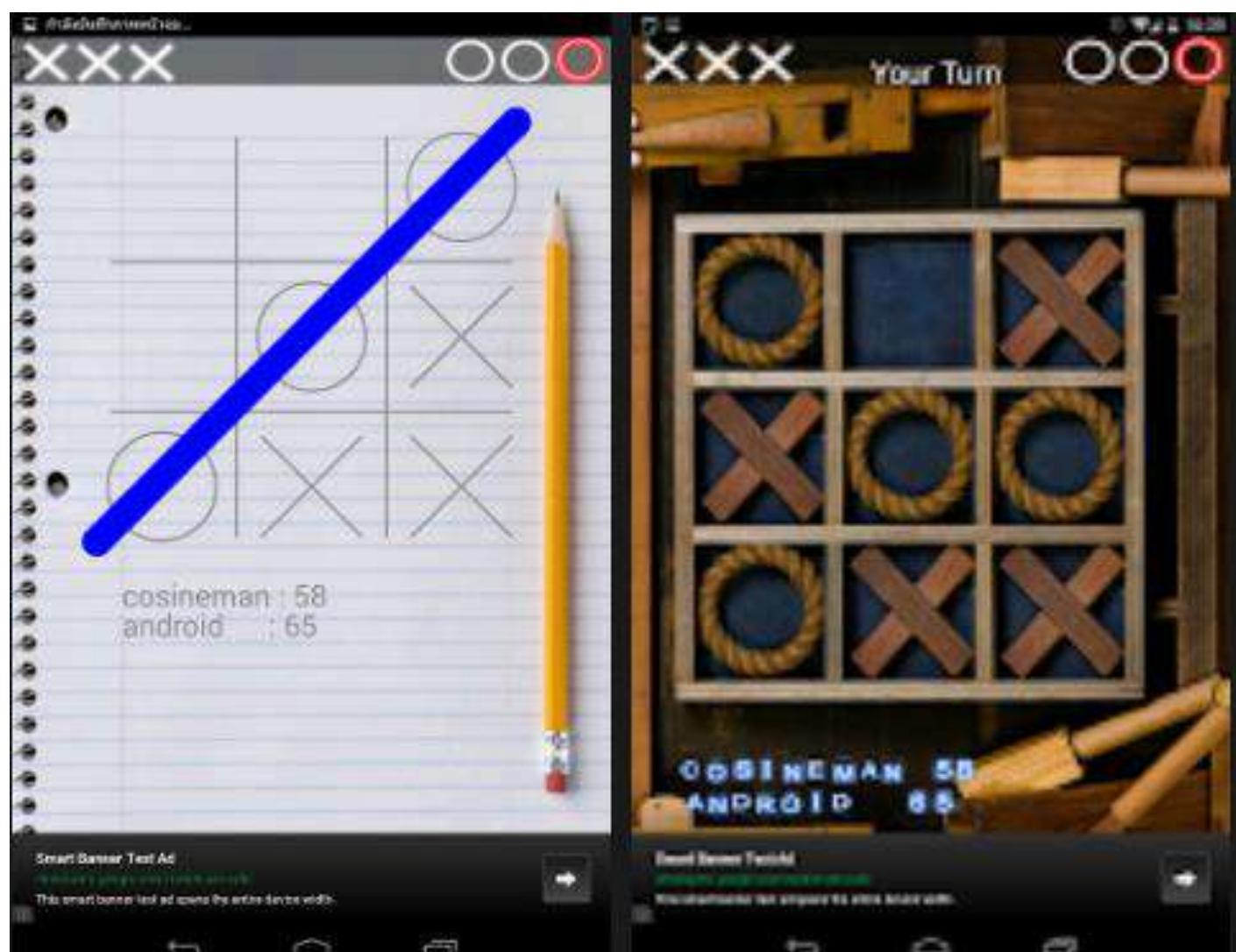
        numberOfForce4 = (allBall - glassFirstRandom[glassNumber - 1]) / 4;
        allRandomBall = glassFirstRandom[glassNumber - 1];

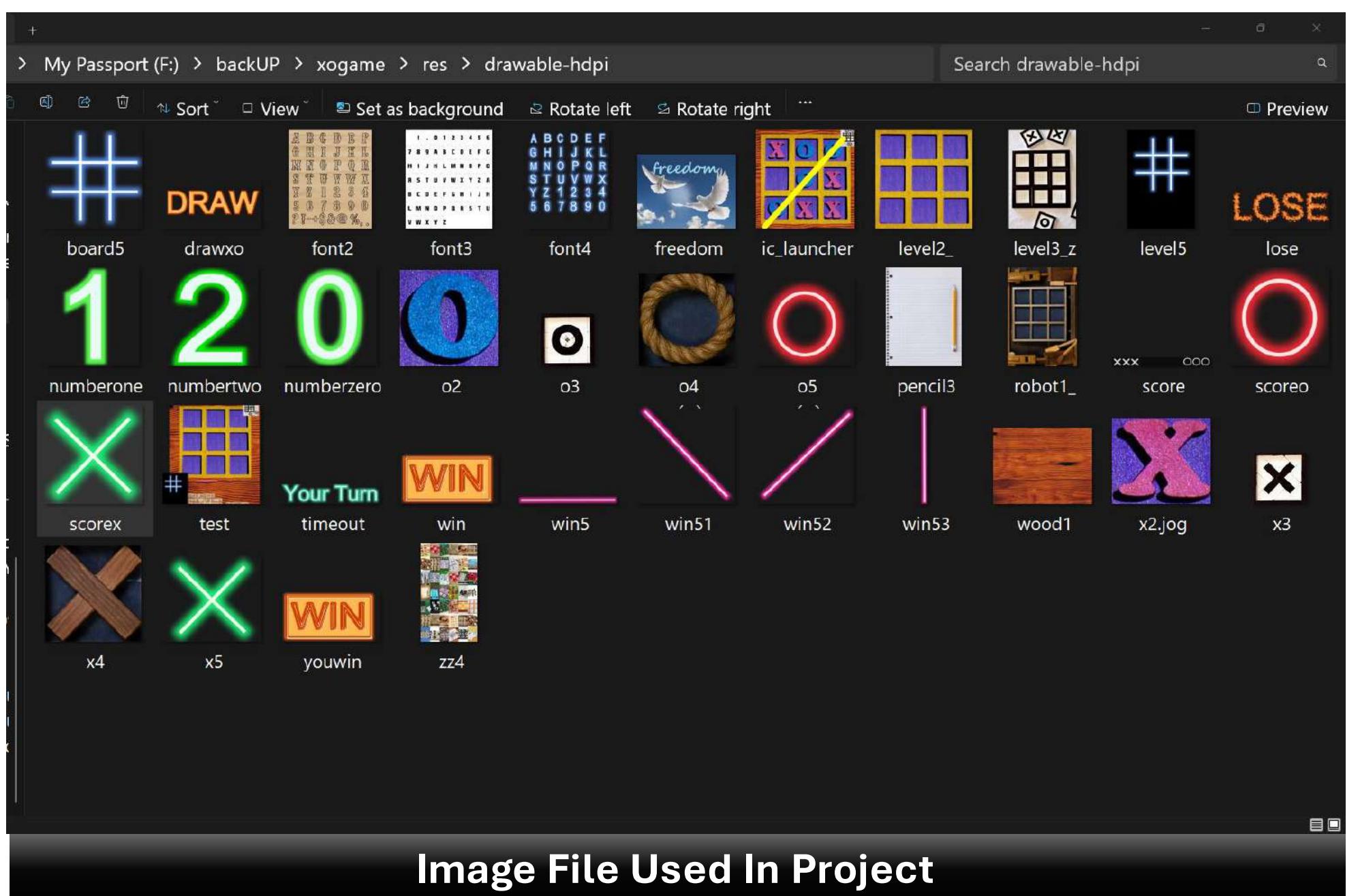
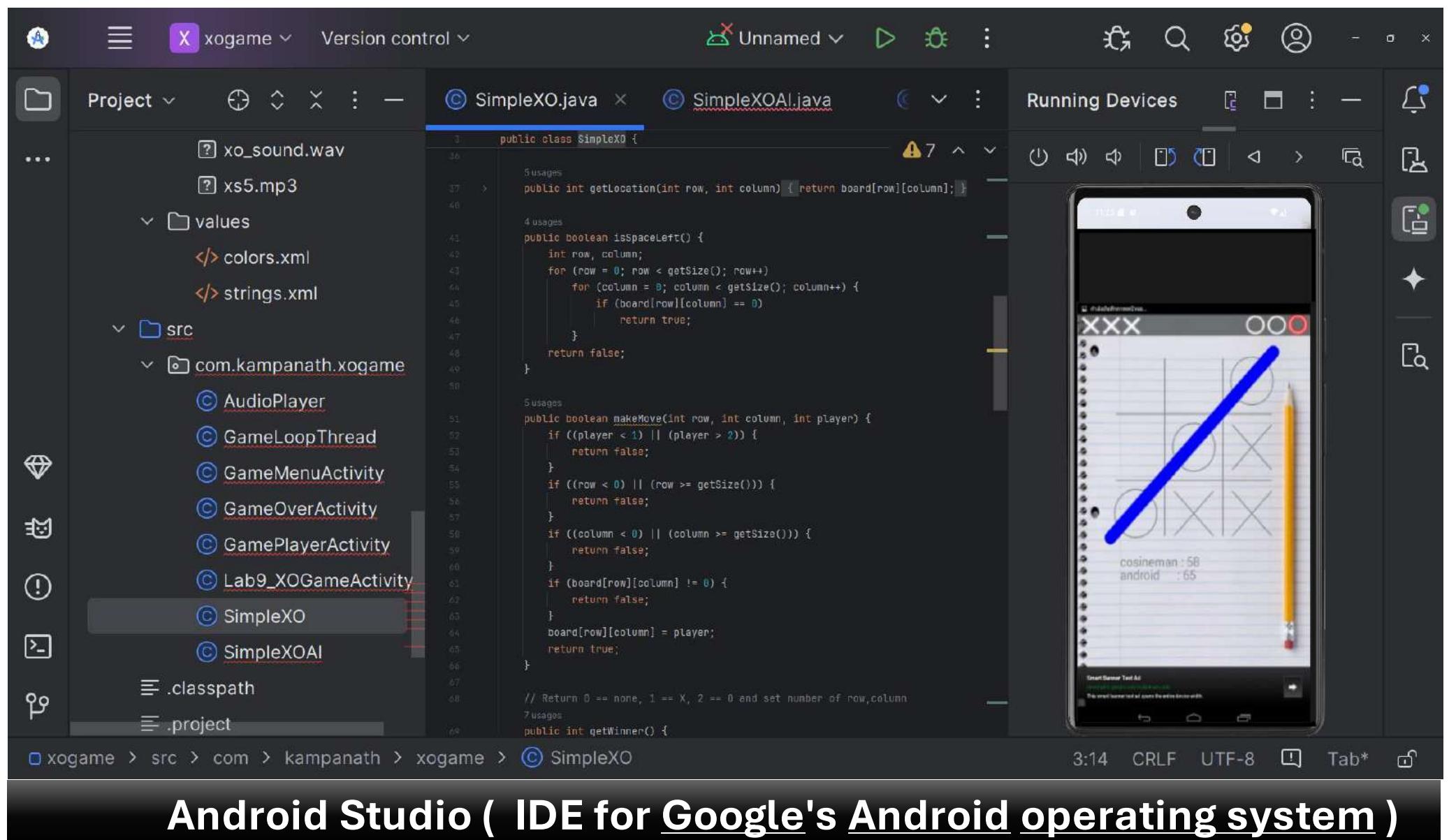
        for (int i = glassNumber - 2; i >= 0; i--)
        {
            // Debug.Log("i=" + i);
            if ((allBall - allRandomBall) == 0)
            {
                glassFirstRandom[i] = 0;
                // Debug.Log("i=" + i + "if1");
            }
            else if (((i + 1) <= numberOfForce4 + 1) && ((allBall - allRandomBall) - (numberOfForce4 * 4) > 0))
            {
                // Debug.Log("i=" + i + "if2");
            }
        }
    }
}

```

VS Code (lightweight source code editor) (C#)

Tic Tac Toe (Android Phone)





Google Play Console

All apps

Dashboard

XO Game Tic Tac Toe
com.kampanath.xogame

Production
Inactive

Show test tracks ▾

Inbox

[View all messages](#)

Changes to taxes for sales to customers in India – Due to recent changes to tax laws in India, beginning August 1, 2024, India... Sep 12

Review changes to Google Play Developer Program Policies – As part of keeping Google Play a safe and trustworthy platform... Nov 6

Protect your developer account from phishing – View resources to help protect your developer account from phishing scams Nov 1

<https://play.google.com/console/u/0/developers/7866042837656215856/app/4976222564038339170/app-dashboard>

Google Play Store Developer Page

WEB APPLICATION

Web application development is the process of creating software applications that run on web servers and are accessed by users through a web browser

Software Tools And Programming Language

Angular

React

Node.js with TypeScript

ASP.NET Core

EF Core

C#

SQL

MSSQL

MySQL

HTML, CSS

Bootstrap

VS Code

Example of My Project

1. Web App For Restaurant (React , BootStrap, Typescript)
2. Web App For Support Test System (Angular ,Asp.Net, C#, Typescript, MSSQL)
3. Web App For Online Shopping (Angular, Asp.Net, C#, Typescript, MSSQL)
4. Web App For Food Online (Angular, Express, Mongo, Nodejs, Typescript)

Web App For Restaurant

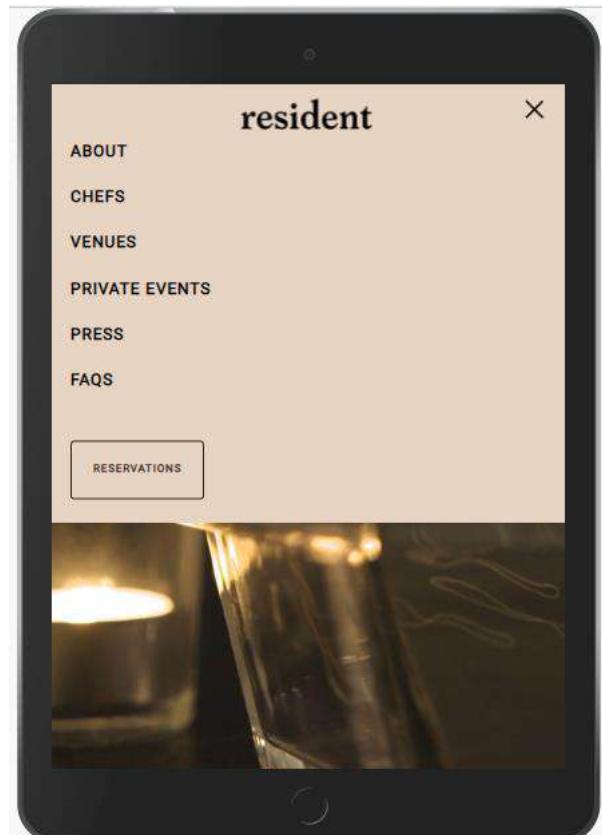
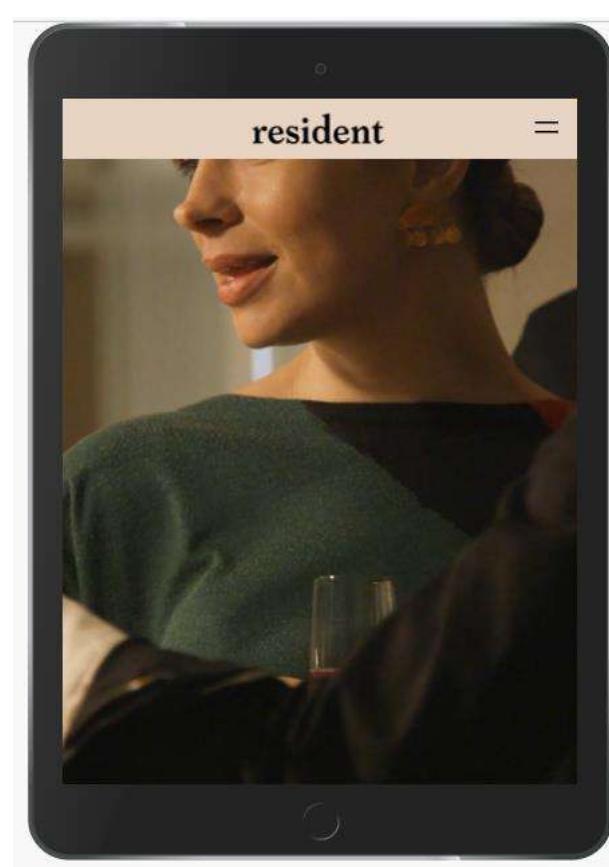
Software Tools And Programming Language

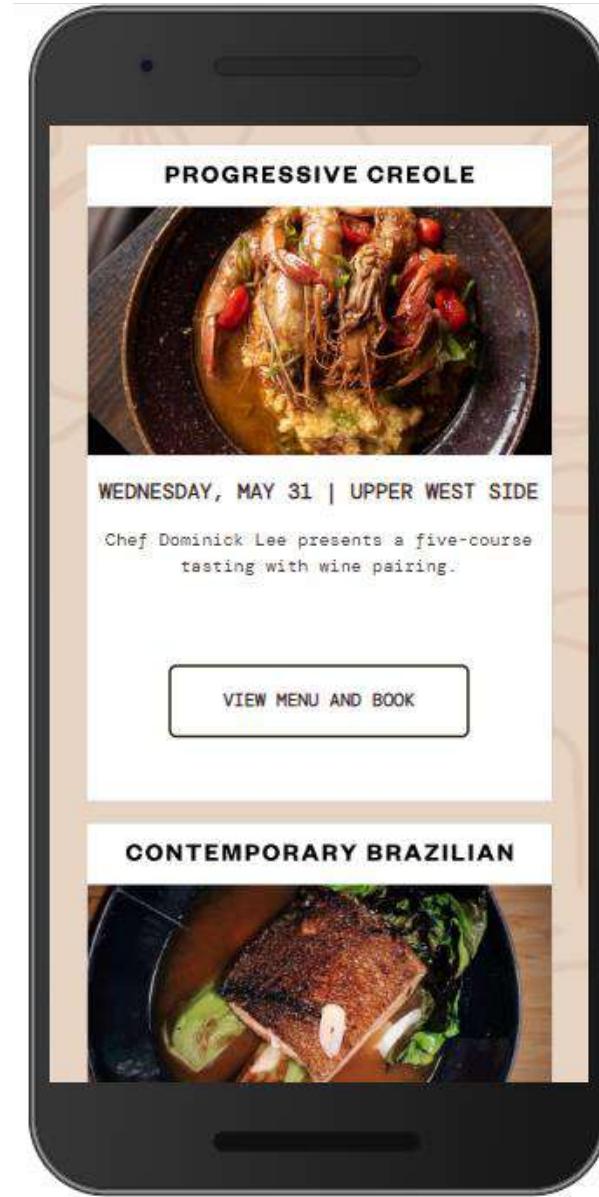
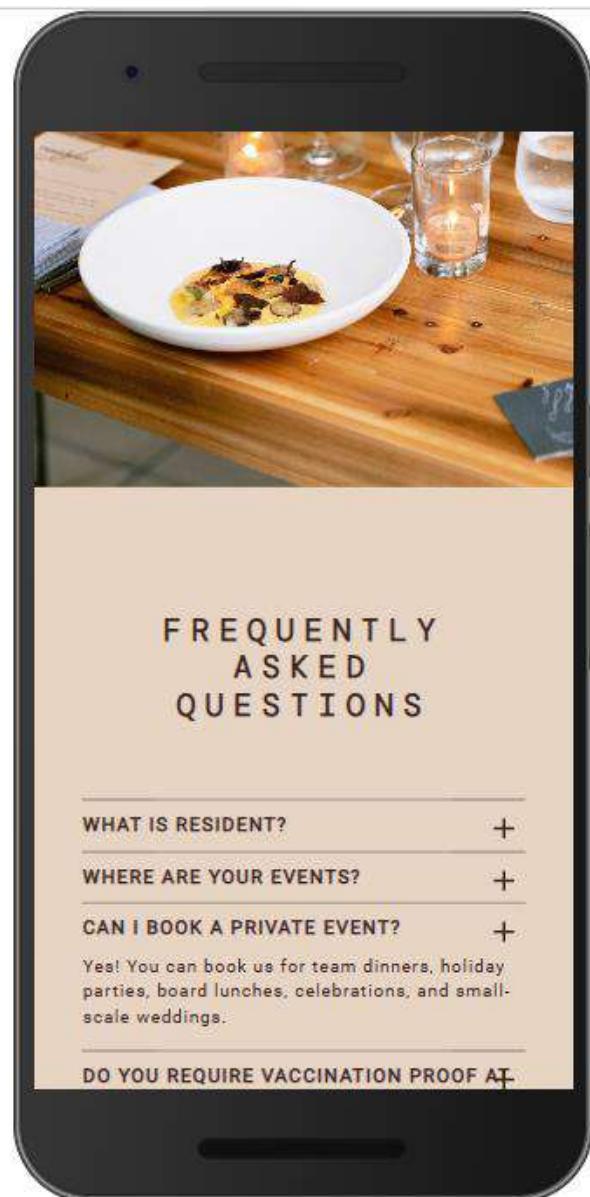
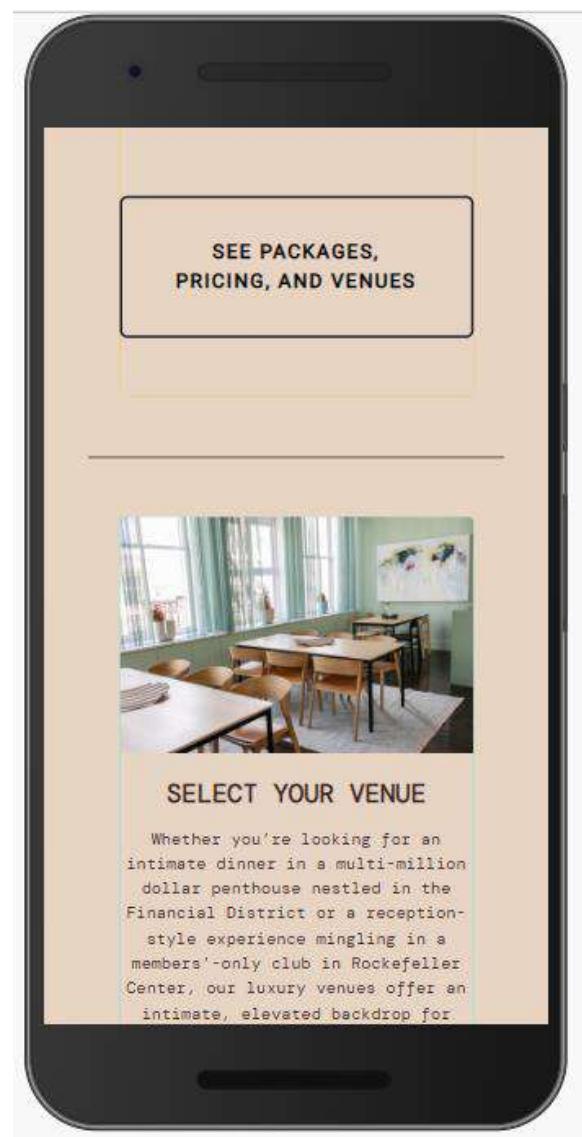
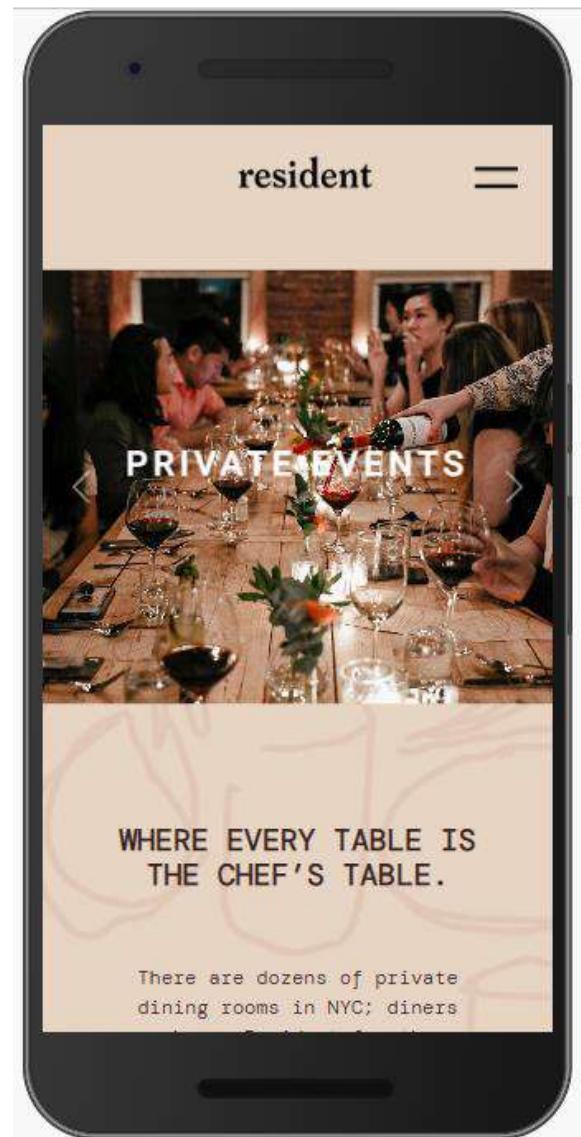
React, BootStrap, Typescript, Azure Web Service

Web Link

Github

This web is build for learning purpose. The web design is cloned from website of fine dinning restaurant located in New York (Resident).





The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- Explorer View:** Shows the project structure under "RESIDENT-MASTER". The "Joinus.tsx" file is currently selected.
- Code Editor:** Displays a TypeScript code snippet for a "Joinus" component. The component uses React Bootstrap components like Button, Form, and InputGroup. It includes a useEffect hook to scroll to the top of the page after submission.
- Terminal:** Shows the output of the build process, indicating a successful webpack compilation.
- DevTools:** An Edge DevTools window is open, showing the DOM structure of the "Joinus" component and its event listeners.
- Browser Preview:** A "resident" browser window displays a "NEWSLETTER SIGNUP" form with fields for First Name, Last Name, and Email Address.

The screenshot shows a development environment with a code editor and a browser preview side-by-side.

Code Editor (Left): A TypeScript file named `About.tsx` is open. The code defines a component with a carousel of images:

```
5
24 /}
25
26 div className="carousel slide about-carousel-img" id="about-card-img" data-t
27   <div className="carousel-inner">
28     <div className="carousel-item active">
29       |
32       |
35       |
38       |
41       |
44   </div>
45   <button className="carousel-control-prev" type="button" data-bs-target="#about-
46     <span className="carousel-control-prev-icon" aria-hidden="true"></span>
47     <span className="visually-hidden">Previous</span>
```

Terminal (Bottom Left):

```
To create a production build, use npm run build.
webpack compiled successfully
Files successfully emitted, waiting for typecheck results...
Issues checking in progress...
No issues found.
```

Browser Preview (Right): The Edge DevTools browser window shows a meal setup with a bowl of food, a bottle, and glasses. The page content includes:

- A heading: **BUILD YOUR EXPERIENCE**
- A paragraph: Our process is simple. Each Resident meal is bespoke in nature, giving you unlimited flexibility to design your perfect evening. From the number of courses, to specialty drinks, pre-dinner...

The browser interface shows the URL `http://localhost:3000/` and various developer tools like Elements, Styles, and Event Listeners.

The screenshot shows the Visual Studio Code interface with two files open in the main editor area. On the left is the `Joinus.tsx` file, which contains TypeScript code for a React component. On the right is the `index.css` file, which contains CSS styles for a card component. The status bar at the bottom indicates the file type as `TypeScript JSX`.

```
src > components > Joinus.tsx > [Joinus]
12 let joinOurTeamInfo = {
13   coverLetter: '',
14 }
15
16 const Joinus = (props: formSubmitData) => [
17   <div>
18     <Form>
19       <FormSection>
20         <FormSectionSection>
21           <FormSectionSectionSection>
22             <FormSectionSectionSectionSection>
23               <FormSectionSectionSectionSectionSection>
24             </FormSectionSectionSectionSectionSection>
25           </FormSectionSectionSectionSection>
26         </FormSectionSection>
27       </FormSection>
28     </div>
29   </Joinus>
30 
```

```
src > components > # Home.css > .background-container
105
106 div.card-home {
107   width: 90%;
108   max-width: max-content;
109
110   background-color: var(--card-bg-color);
111
112   border: 0;
113
114 }
115
116 .card-horizontal {
117   display: flex;
118   flex-direction: row-reverse;
119   /* flex: 1 1 auto; */
120   height: max-content;
121   width: 100%;
122
123   background-color: var(--card-bg-color);
124
125 }
126
127 .img-fluid#home-card-img {
128   width: 50%;
129   height: auto;
130   flex: 1 1 50%;
131   max-width: 550px;
132   background-color: var(--card-bg-color);
133
134 }
135
136 .card-body-home {
137   padding: 10px;
138 }
```

VS Code (TypeScript)

Web App For Support Test System (Show And Edit Test Parameter)

Software Tools And Programming Language

Frontend -> Angular, Typescript, Nodejs, BootStrap

Backend -> .Net Core, C#, MSSQL, MySQL, IIS, EF Core

[Web Link](#) [Github](#)

This web app connect with 2 database server (MSSQL and MySQL) that collect test result and test parameter from production test program in electronic manufacturing (GPS board and satellite associate product).

This web will be used by test engineer or technician to see detail of test station , gold board ,serial number ,product model, test equipment. Also to edit test parameter that will be used in test program.



Web GUI for Iridium Production Test

User : Kampanath Kappago User Level : Administrator Logout

ATE Profile

ConfigId	StationId	ConfigType	HardwareType	UserId	DateAdd	DateModify	DateRelease	ConfigName	ConfigRevNo	TestLimitNote	Action	
147	42	CAL	9522	3	2019-01-16T10:00:58.593	2019-01-16T10:00:58.593	2019-01-16T10:00:58.593	CalibrationLimits_9522_ATE11.txt	2019-Jan-16 10:00:47.512			
152	20	CAL	9522	1057	2019-01-18T16:11:26.75	2019-01-18T16:11:26.75	2019-01-18T16:11:26.75	CalibrationLimits_9522_ATE8V2.txt	2019-Jan-18 16:10:27.736			

tbATEConfigFile table from MSSQL

Show And Filter Table

Web GUI for Iridium Production Test

User : Kampanath Kappago User Level : Administrator Logout

Edit ATE Profile

Config Id	Station Id	Config Type
147	42	CAL
Hardware Type	User ID	
9522	3	
Config Rev No	Config Data	Is Active
2019-Jan-16 10:00:47.512	Begin<Step Name><Locals> Variable ValueGoldenValue_Pov	true
Test Limit Note	<input type="button" value="Upload New Config Data File"/>	
<input type="button" value="Cancel"/>		<input type="button" value="Submit"/>

Edit Test Parameter

Web GUI for Iridium Production Test

User : Kampanath Kappago User Level : Administrator Logout

MAIN NAVIGATION

- Home
- Stock
- Shop
- Non-ATE Profile
- ATE Profile
- Add New SN
- SN And IMEI Edit
- IMEI Import
- Equipment Edit MSSQL
- Gold Boards
- Fixture Config
- Model Edit
- FW
- Equipment Edit MySQL
- Station Calibration**
- Logout

Station Calibration

Station Name: ADE FTO Fixture Name: All

Start Date: 01/10/2018 Stop Date: 08/10/2020

Show all tbCalibration table from MSSQL

Count	Station Name	Fixture Name	socket No.	Cal Date Time	Software Version	User Name	Remark
1	ADE FTO	FTO	1	2018-01-17T10:33:13	1.0.6	panupt	
2	ADE FTO	FTO	1	2018-03-02T21:35:52	1.0.19	panupt	
3	ADE FTO	FTO	1	2018-03-03T09:42:22	1.0.19	panupt	
4	ADE FTO	FTO	1	2018-04-02T20:34:25	1.0.20	panupt	
5	ADE FTO	FTO	1	2018-04-11T08:25:41	1.0.21	panupt	
6	ADE FTO	FTO	1	2018-04-19T09:11:56	1.0.21	panupt	
7	ADE FTO	FTO	1	2018-04-26T15:26:31	1.0.21	panupt	
8	ADE FTO	FTO	1	2018-05-04T01:30:24	1.0.21	panupt	
9	ADE FTO	FTO	1	2018-05-11T15:26:29	1.0.21	panupt	
10	ADE FTO	FTO	1	2018-05-14T14:10:16	1.0.21	panupt	
11	ADE FTO	FTO	1	2018-05-22T07:28:49	1.0.21	panupt	
12							

Filter Start And Stop Date

The screenshot shows the Visual Studio Code interface with two tabs open:

- app.component.spec.ts**: A TypeScript unit test file. It imports TestBed and RouterTestingModule from '@angular/core/testing', and AppComponent from './app.component'. It uses describe and it blocks to test the creation of the app, its title, and rendering of the title.
- app-routing.module.ts**: A TypeScript file defining the app-routing module. It imports StockTbEquipmentEditComponent from './components'. It defines a routes array containing several path components, some with children, such as 'stock' which has paths like '' (Stock HomeComponent), 'Product' (Stock HomeComponent), 'create' (Stock Create Component), 'edit/:id' (Stock Edit Component), 'ird-create' (Stock Ird Create Component), 'ird-edit/:id' (Stock Ird Edit Component), 'UUTStatusType' (Stock UUT Status Type Component), 'tbProfile/:id' (Stock Tb Profile Component), 'tbProfileDetail/:id' (Stock Tb Profile Detail Component), 'mySQL' (Stock MySQL Component), 'mySQL-create' (Stock MySQL Create Component), 'mySQL-edit/:id' (Stock MySQL Edit Component), 'TbUutname' (Stock Tb Uutname Component), 'TbUutname-create/:id' (Stock Tb Uutname Create Component), 'TbUutname-edit/:id' (Stock Tb Uutname Edit Component), 'TbUutprofile/:id' (Stock Tb Uutprofile Component), 'TbUutprofile-create/:id' (Stock Tb Uutprofile Create Component), 'TbUutprofile-edit/:id' (Stock Tb Uutprofile Edit Component), 'tbProfile-edit/:id' (Stock Tb Profile Edit Component), 'tbProfile-create/:id' (Stock Tb Profile Create Component), 'tbProfileDetail-edit/:id' (Stock Tb Profile Detail Edit Component), and 'tbProfileDetail-create/:id' (Stock Tb Profile Detail Create Component).

The status bar at the bottom indicates the file is 52 lines long, uses UTF-8 encoding, is a TypeScript file, and shows the Go Live icon.

A screenshot of the Visual Studio Code interface. The title bar says "backend". The left sidebar shows a tree view of files under "BACKEND" and "Controllers". The main editor window displays C# code for a controller named "a9522b_ext_cntController". The code includes methods for handling GET and POST requests. A tooltip is visible over the word "string" in the SQL query string. The status bar at the bottom shows "pages: 4" and "UTF-8".

```
public class a9522b_ext_cntController : ControllerBase
{
    [HttpGet("{id}")]
    public IActionResult Geta9522b_ext_cntById(int id)
    {
        try
        {
            var result = ia9522BExtCntRepository.Geta9522b_ext_cntByID(id);
            if [result == null]
            {
                return NotFound(new { result = result, message = "failure" });
            }
            return Ok(new { result = result, message = "successfully" });
        }
        catch (Exception ex)
        {
            return StatusCode(500, new { result = ex, message = "failure" });
        }
    }

    [HttpPost]
    public IActionResult Adda9522b_ext_cnt([FromForm] a9522b_ext_cnt model)
    {
        try
        {
            var result = ia9522BExtCntRepository.Adda9522b_ext_cnt(model);
            return Ok(new { result = result, message = "successfully" });
        }
    }
}
```

VS Code (Back End) (C#)

A screenshot of the Visual Studio Code interface. The title bar says "config-test-parameter-master". The left sidebar shows a tree view of files under "CONFIG-TEST-PARAMETER" and "services". The main editor window displays C# code for a generic repository named "genericRepository". A tooltip is visible over the word "System.String" in the SQL query string. The status bar at the bottom shows "pages: 4" and "UTF-8".

```
public class genericRepository : IgenericRepository
{
    public IEnumerable<ateCalibrationATEName> GetTbAteCalByATENameColumn()
    {
        string query = @"SELECT DISTINCT tbStation.stationName
                        FROM tbCalibration LEFT OUTER JOIN
                        tbUUTName ON tbCalibration.UUTNameID = tb
                        tbFixture ON tbCalibration.fixtureID = tb
                        tbStation ON tbCalibration.stationID = tbStation.stationID LEFT OUTER JOIN
                        tbUser ON tbCalibration.userID = tbUser.userID";
    }

    public IEnumerable<ateCalibrationFixtureName> GetateCalFixtureNameByATEName(string ATENName)
    {
        string query = @"SELECT DISTINCT tbFixture.fixtureName
                        FROM tbCalibration LEFT OUTER JOIN
                        tbUUTName ON tbCalibration.UUTNameID = tbUUTName.UUTNameID LEFT OUTER JOIN
                        tbFixture ON tbCalibration.fixtureID = tbFixture.fixtureID LEFT OUTER JOIN
                        tbStation ON tbCalibration.stationID = tbStation.stationID LEFT OUTER JOIN
                        tbUser ON tbCalibration.userID = tbUser.userID
                        where tbStation.stationName LIKE '" + ATENName + "'";
    }
}
```

VS Code (Back End) (C#)

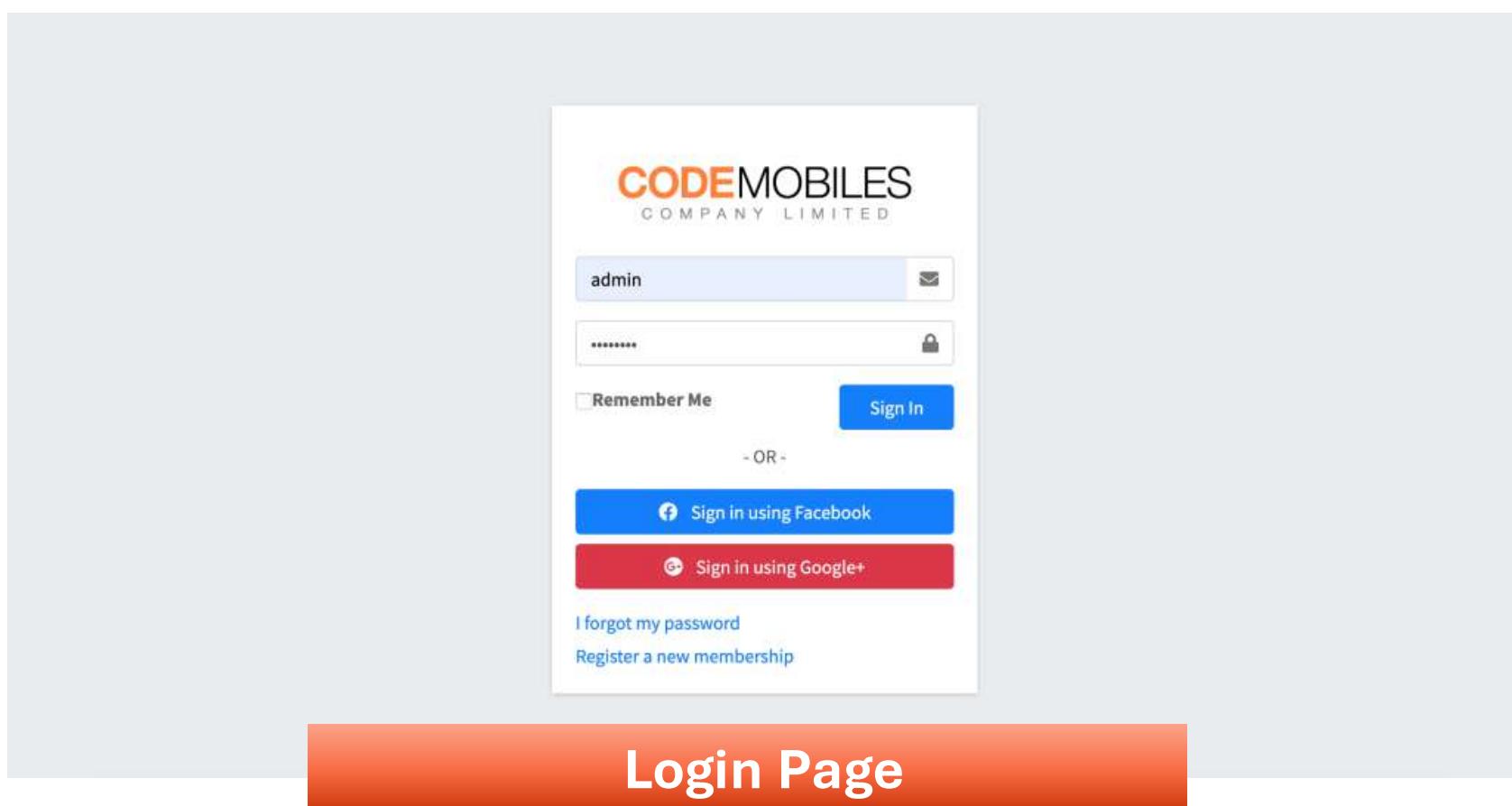
Web App For Online Shopping

Frontend -> Angular, Typescript, Nodejs, BootStrap

Backend -> .Net Core, C#, MSSQL, EF Core

[Web Link](#) [Github](#)

This website is build for learning purpose from Angular + ASP.net training course.



The screenshot displays the stock management page for the Arduino section of the website. The top navigation bar includes links for Home, Contact, Search, and a user icon. On the left, there's a sidebar with a main navigation menu and a sub-menu for "Arduino". The main content area shows three status indicators: "Stock 17", "Defect 0", and "Sold out 1". Below these is a search bar with the placeholder "Enter search keyword" and a green "Search" button. The main table lists five products with columns for ID, Name, Price, Stock, and Action (edit and delete icons). The products are:

ID	NAME	PRICE	STOCK	ACTION
17	Arduino Sensor Kit V5.0	฿150.00	13 PCS.	
16	4-Channel 5VDC Relay Module Relay Active High / LOW	฿185.00	1 PCS.	
15	4-Channel 5VDC Relay Module Relay Active High / LOW	฿185.00	1 PCS.	
14	4-Channel 5VDC Relay Module Relay Active High / LOW	฿185.00	1 PCS.	
13	Arduino MEGA 2560 R3 ไมโคร USB CH340 ชุนไพร แอดมาร์ท USB	฿200.00	100 PCS.	

A large orange banner at the bottom of the page reads "Stock Page".

ARDUINO .CODEMOBILES



MAIN NAVIGATION

- _STOCK
- หน้าร้าน
- ผู้ใช้งาน
- รายงาน
- รายการขาย
- กราฟ
- ออกจากระบบ

Edit Product

Name: Arduino Sensor Kit V5.0

Price: 150 Stock: 13

Cancel Submit

Upload image



Stock page (Edit product detail)

ARDUINO .CODEMOBILES



MAIN NAVIGATION

- _STOCK
- หน้าร้าน
- ผู้ใช้งาน
- รายงาน
- รายการขาย
- กราฟ
- ออกจากระบบ

Product	Price	Quantity
Arduino MEGA 2560 R3 ไข่ชีฟ USB CH340	฿200.00	- 1 +
Arduino Sensor Kit V5.0	฿1,350.00	- 9 +
4-Channel 5VDC Relay Module Relay Active High / LOW	฿185.00	- 1 +

Tax 7% ฿0.00

Total ฿1,735.00

Payment

Selling page



```
<div class="content-wrapper">
  <section class="content-header">
    <div class="container-fluid">
      <div class="row">
        <div class="col-12">
          <h1>Create Product</h1>
        </div>
      </div>
    </section>
    <section class="content">
      <div class="container-fluid">
        <div class="row">
          <div class="col-md-6">
            <div class="card card-primary">
              <form role="form">
                <div class="card-body">
                  <div class="row">
                    <div class="col-md-12">
                      <div class="form-group">
                        <label>Name</label>
                        <input [(ngModel)]="mProduct.name" type="text" class="form-control" name="Name" />
                      </div>
                    </div>
                  </div>
                  <div class="row">
                    <div class="col-md-6">
                      <div class="form-group">
                        <label>Price</label>
                        <input [(ngModel)]="mProduct.price" type="number" class="form-control" name="Price" />
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </section>

```

VS Code (HTML)

```
import { Component, OnInit } from '@angular/core';
import { Product } from 'src/app/models/product.model';
import { Location } from '@angular/common';
import { NetworkService } from 'src/app/services/network.service';

@Component({
  selector: 'app-stock-create',
  templateUrl: './stock-create.component.html',
  styleUrls: ['./stock-create.component.css']
})
export class StockCreateComponent implements OnInit {

  mProduct = new Product();
  imageSrc: string | ArrayBuffer = null;

  constructor(private localtion: Location, private networkService: NetworkService) {
    this.mProduct.name = '';
    this.mProduct.stock = 0;
    this.mProduct.price = 0;
  }

  ngOnInit() {
  }

  submit() {
    this.networkService.addProduct(this.mProduct).subscribe(
      data => {
        alert(data.message);
        this.localtion.back();
      },
    );
  }
}
```

VS Code (TypeScript)

The screenshot shows a SQL database editor window with the following details:

- Title Bar:** Shows the file name "database.sql" and a search bar.
- Toolbar:** Includes icons for file operations like Open, Save, Print, and a refresh symbol.
- Left Sidebar:** Displays a tree view of the database structure, including "C:\Users\cosin\Cosine_Data\JOB\Programming\Teacher Database" and "databasesql".
- Code Area:** Contains the following SQL script:

```
8 GO
9 CREATE TABLE [dbo].[Products](
10     [product_id] [int] IDENTITY(1,1) NOT NULL,
11     [Name] [nvarchar](max) NULL,
12     [Image] [nvarchar](max) NULL,
13     [Stock] [int] NOT NULL,
14     [Price] [int] NOT NULL,
15     [Created] [datetime2](7) NOT NULL,
16     CONSTRAINT [PK_Products] PRIMARY KEY CLUSTERED
17 (
18     [product_id] ASC
19 )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
20 ) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
21 GO
22 SET IDENTITY_INSERT [dbo].[Products] ON
23
24 INSERT [dbo].[Products] ([product_id], [Name], [Image], [Stock], [Price], [Created]) VALUES (1, N'NodeMCU Development Kit V2', '...', '...', '...', '...')  
...  
39 INSERT [dbo].[Products] ([product_id], [Name], [Image], [Stock], [Price], [Created]) VALUES (17, N'Arduino Sensor', '...', '...', '...', '...')
```
- Status Bar:** Shows "Restricted Mode", "0 A 0", "0", "UTF-8", "CRLF", "SQL", "MSSQL", and "Disconnected".

Database Script

Web App For Food Online

Software Tools And Programming Language

Frontend -> Angular, Nodejs, Typescript

Backend -> Express, Mongo

This web site is build for learning purpose

The screenshot shows the homepage of the Pizza Indy website. At the top, there is a banner with the text "Fresh Cooked Food At Your Door." Below the banner, there is a section titled "Customers Favourites Food" featuring four food items: "salad bowl", "Strawberry Cake", "Deluxe Burger", and "Paneer Grilled Sandwich". Each item has a small image and its price: 75 Baht, 120 Baht, 299 Baht, and 150 Baht respectively. The footer of the page includes social media links and the text "Pizza Indy GOOD FOOD, GOOD EAT".

The screenshot shows a table titled "Menu List" from the Pizza Indy website. The table has columns for "Menu Name", "Price", "img", and "Action". The rows show the following data:

Menu Name	Price	img	Action
salad bowl	75 Baht		Edit Delete
Strawberry Cake	120 Baht		Edit Delete
Deluxe Burger	299 Baht		Edit Delete
Paneer Grilled Sandwich	150 Baht		Edit Delete
Veggie Supreme Pizza	220 Baht		Edit Delete

The footer of the page includes social media links and the text "Pizza Indy GOOD FOOD, GOOD EAT".

Embedded System

Software Tools And Programming Language

MCS-51 Assembly Langauge

C++

VB6

Microcontroller

Embedded Linux

Example of My Project

1.Survey Robot (Senior Project)

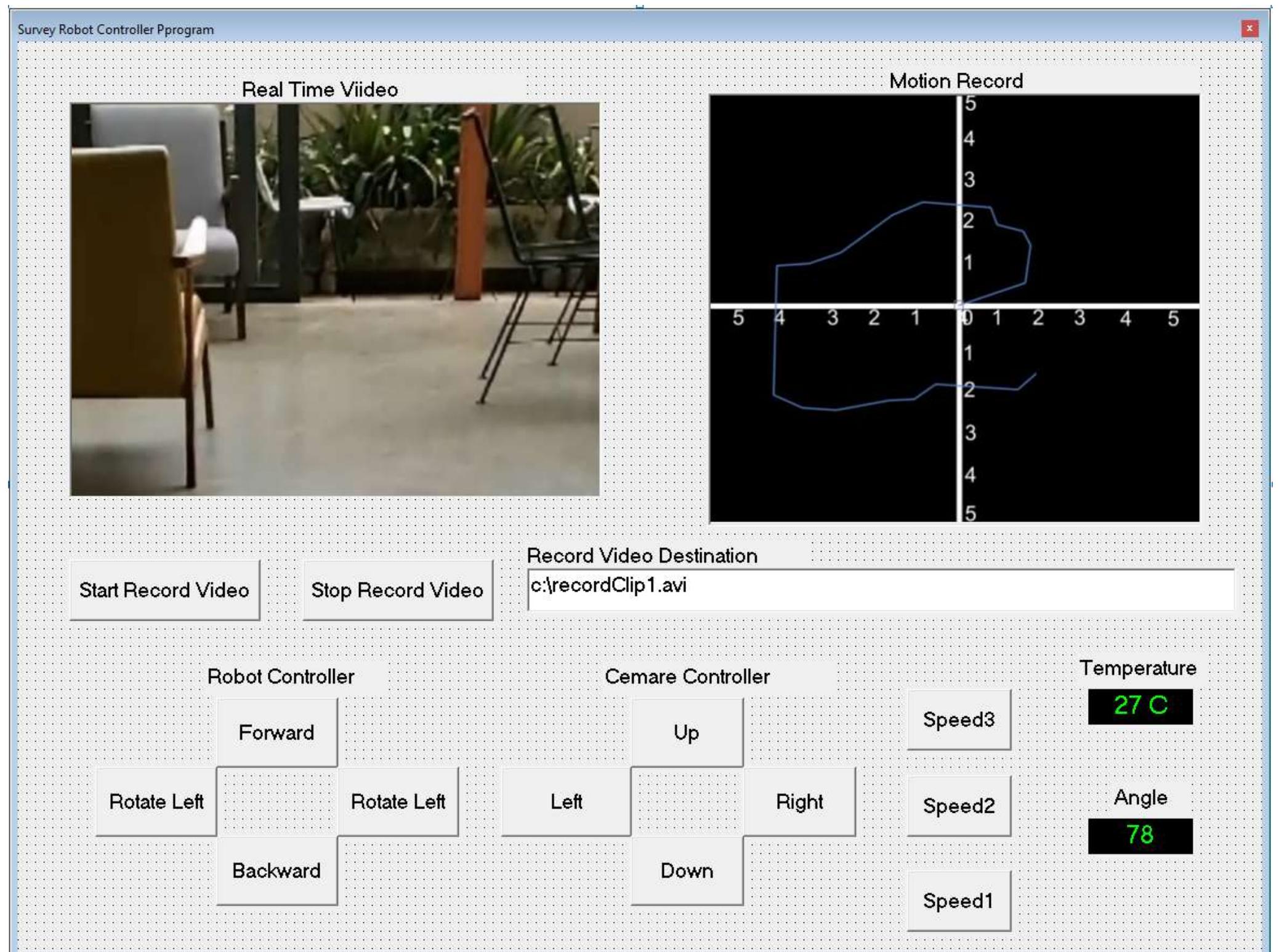
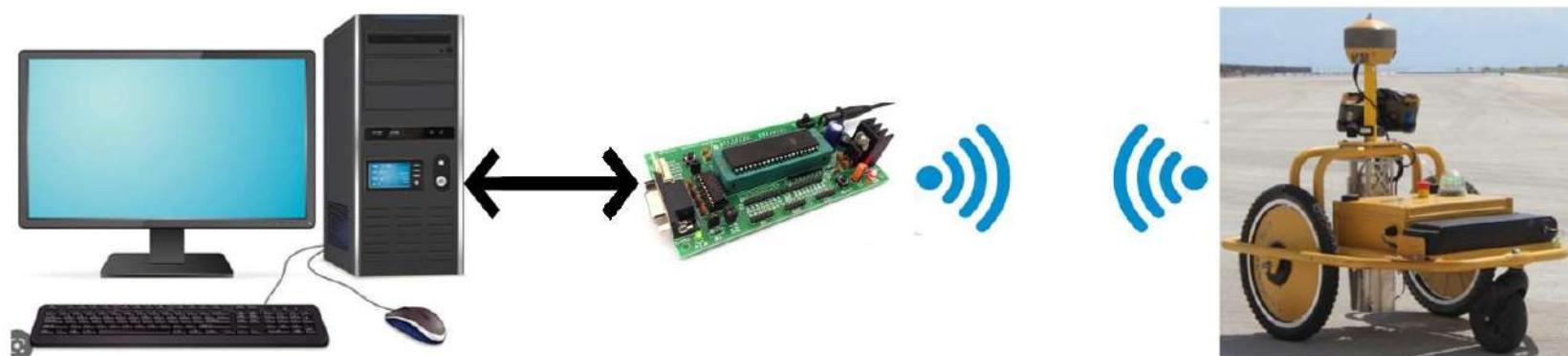
2.Simple Web Interfacing For Controlling Device Demo (Embedded Linux)

3. Basic Image Processing With Raspberry Pi And OpenCV (Object Detection)

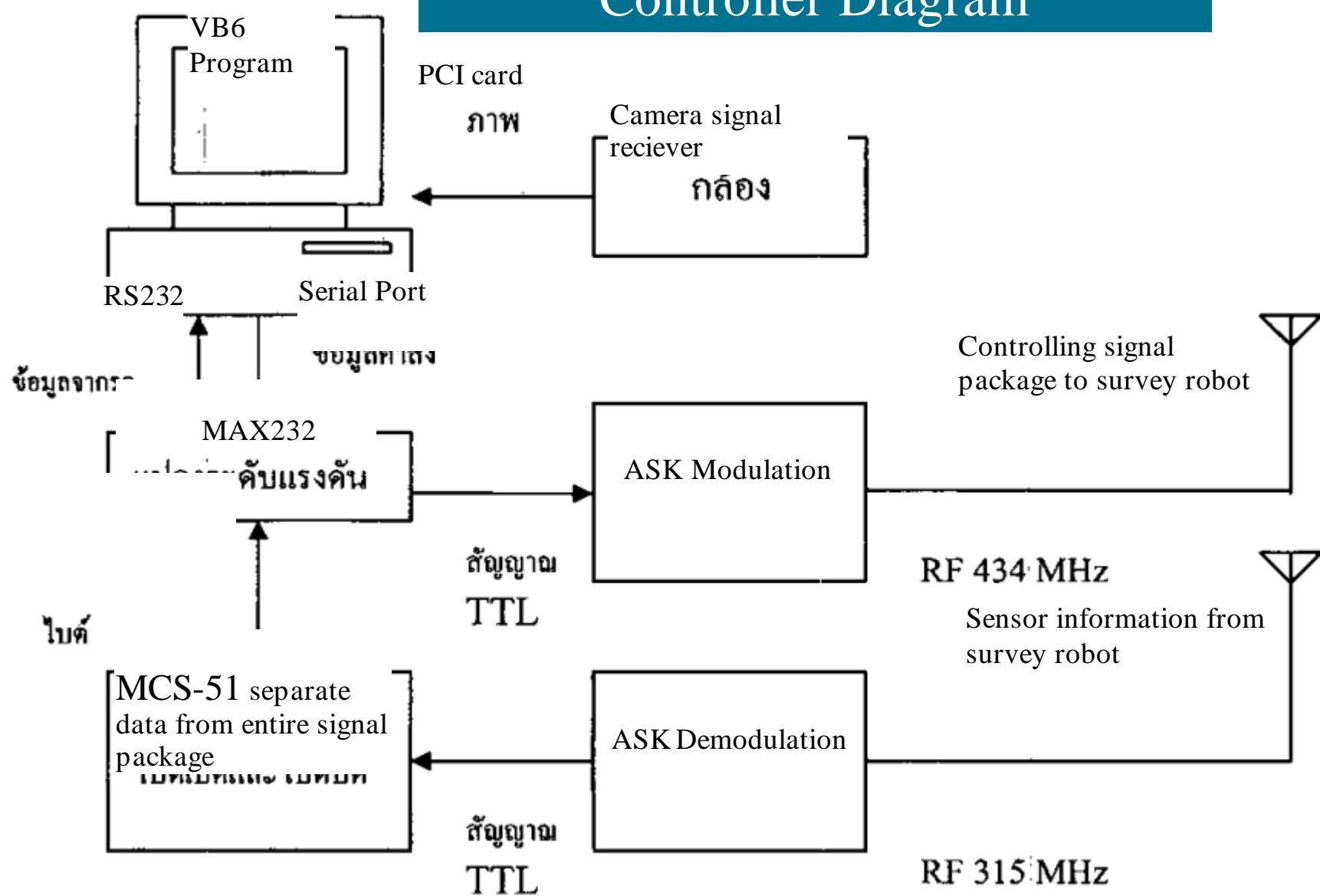
Survey Robot (Senior Project)

Software Tools and Programming Language : MSC-51 Assembly Langauge, VB6

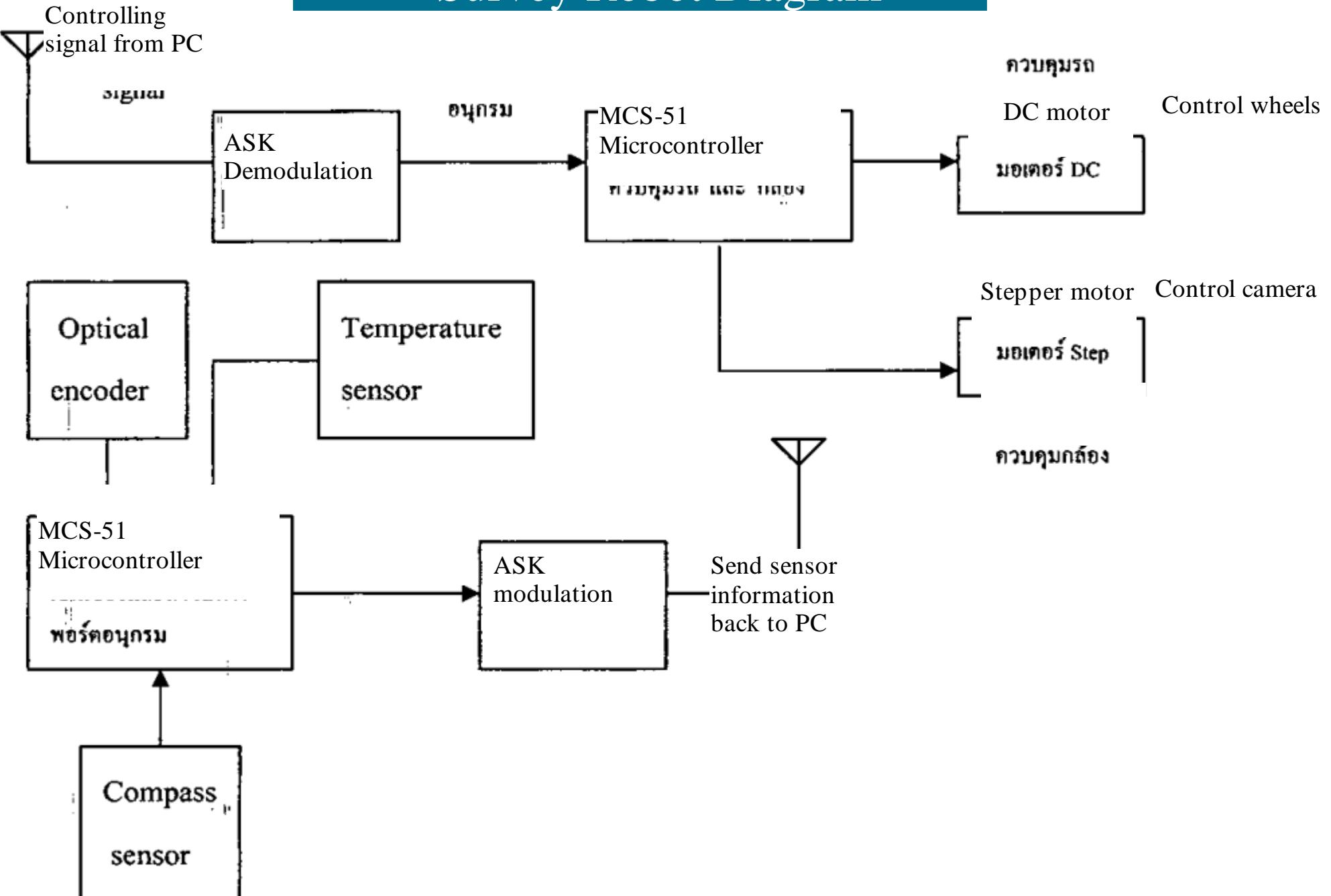
This survey robot can wirelessly control from computer using VB6 application. VB6 app can show real time video from camera installed at robot, show temperature from robot's temperature sensor, show robot's motion record by use data from robot wheel encoder and digital compass sensor. The camera's rotation also can control from VB6 application.



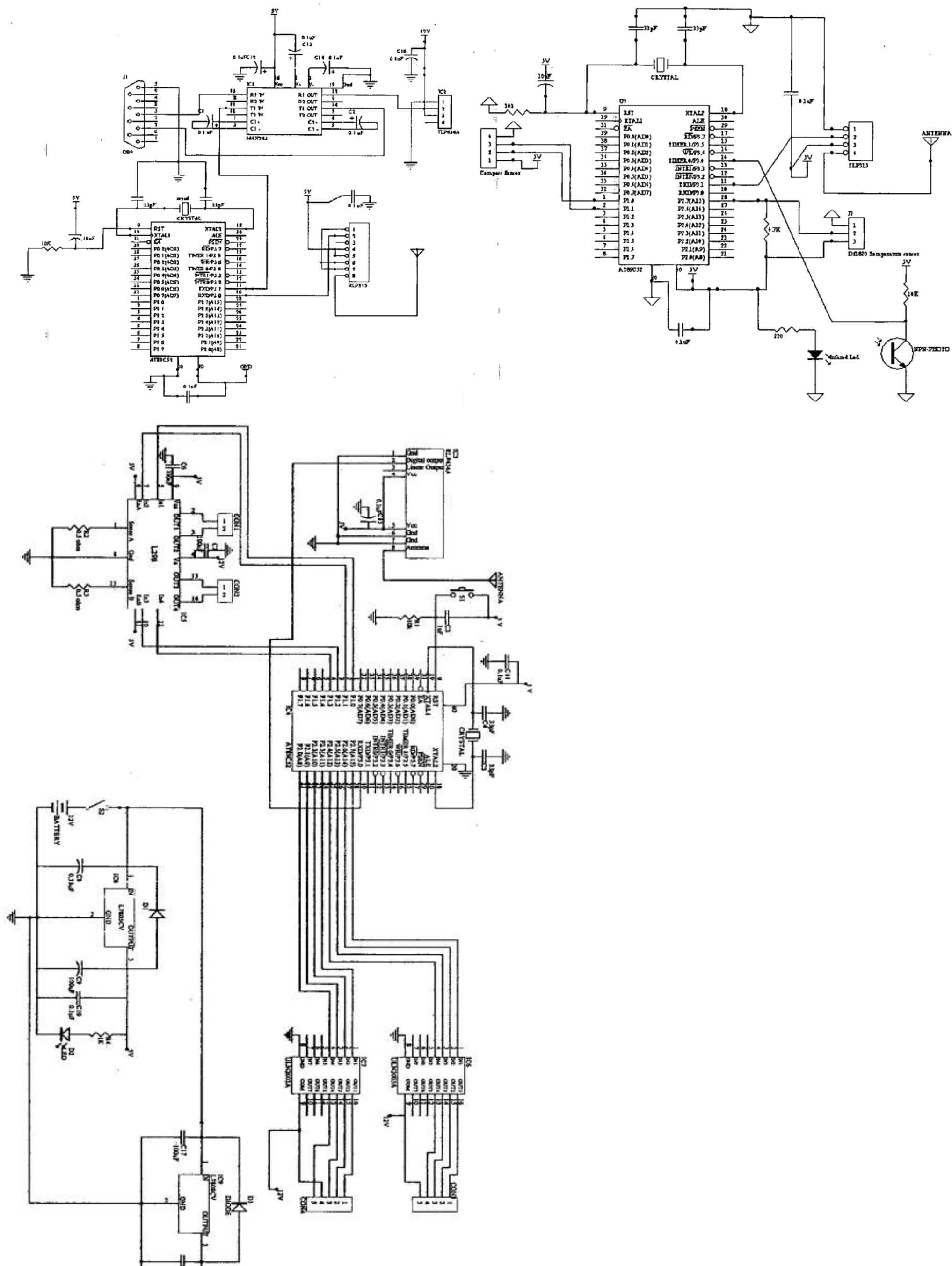
Controller Diagram



Survey Robot Diagram



Schematic



Example of assembly code

```
CAMERA_RIGHT: CJNE A,#09H,HOME  
DJNZ R1,CONTI4  
  
HOME: LJMP WAIT  
  
CONTI4: MOV A,R5  
CLR C  
RRC A  
JNC NEXT4  
MOV A,#00001000B  
  
NEXT4: MOV R5,A  
MOV P2,R5  
ACALL DELAY_100ms  
AJMP CAMERA_RIGHT  
  
DELAY_100ms: MOV R7,#26  
DELAY_100ms_1: MOV R6,#0E6H  
DELAY_100ms_2: NOP  
NOP
```

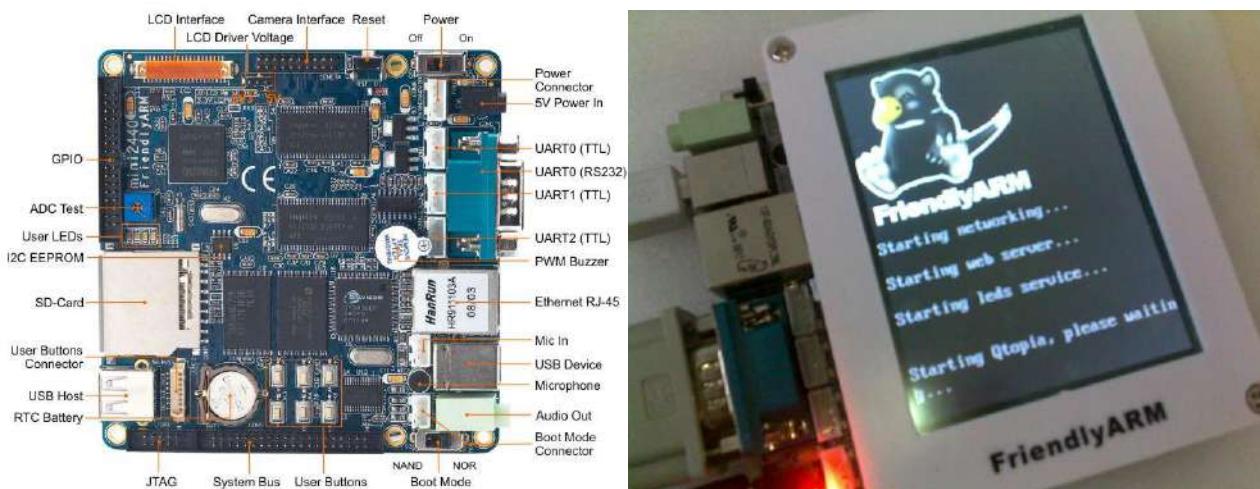
Example of VB6 code

```
Dim dataa(2)  
Dim dat(4)  
  
Private Sub Command10_Click()  
ezVidCap1.CaptureFile = txtAVI.Text  
ezVidCap1.CaptureVideo  
End Sub  
  
Private Sub Command10_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)  
Command10.BackColor = &HC0&  
End Sub  
  
Private Sub Command11_Click()  
ezVidCap1.CaptureEnd  
End Sub
```

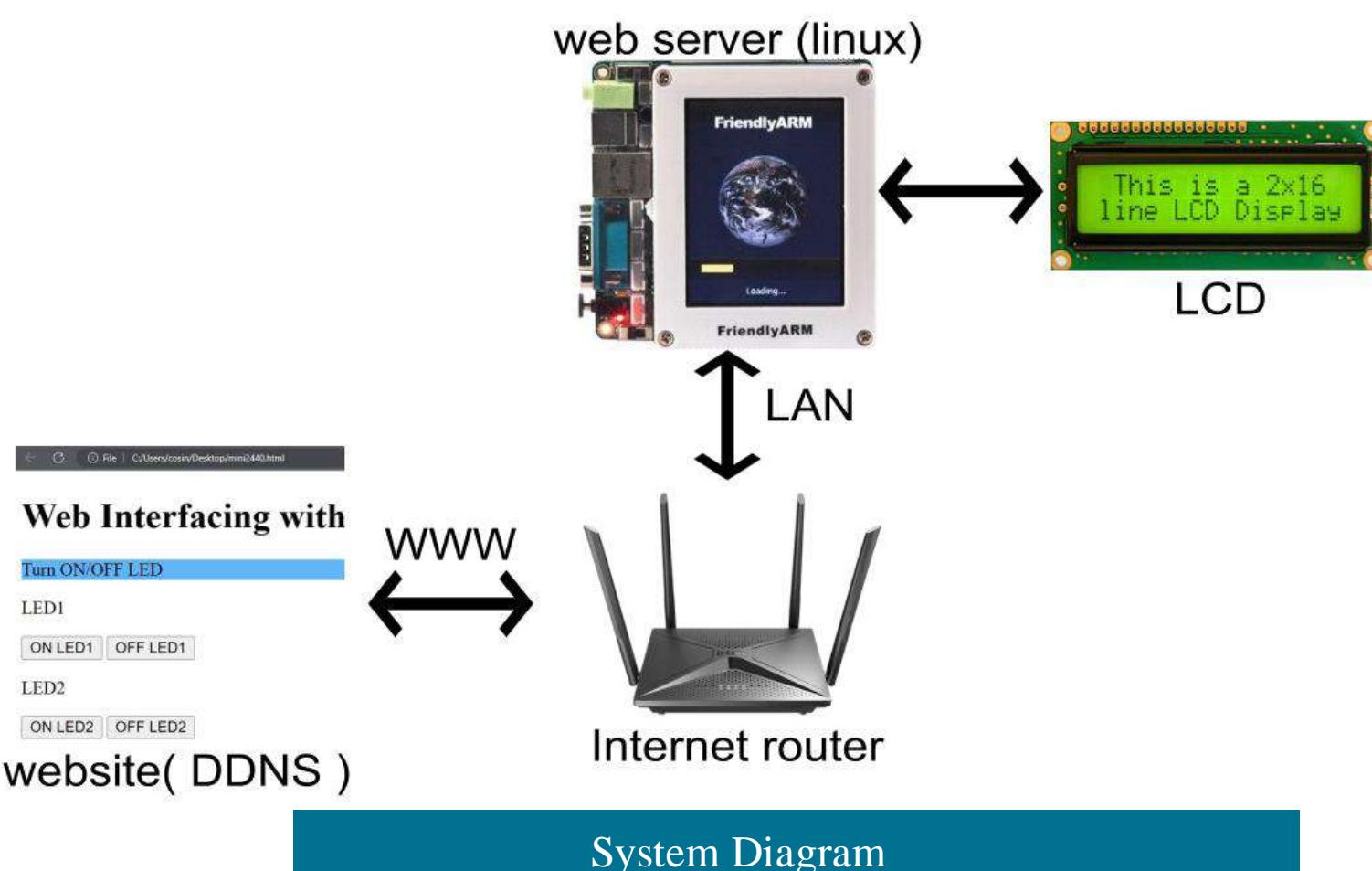
Simple Web Interfacing For Controlling Device Demo (Embedded Linux)

Software Tools and Programming Language : C++, HTML, Linux Qtopia

This simple website can turn ON/OFF LED on linux board, read button status (button is pressed or not), send text from text box to LCD that connect with linux. Web server will use CGI to run script that using linux device driver to control its peripheral device.



Mini2440 Board (ARM9 CPU, installed linux Qtopia)



Website

← ⌂ ⌂ File | C:/Users/cosin/Desktop/mini2440.html

Web Interfacing with embedded Hardware

Turn ON/OFF LED

LED1

LED2

LED3

Show Button Status

Button 1 : Pressed

Button 1 : Normal

Button 1 : Normal

Write text to LCD

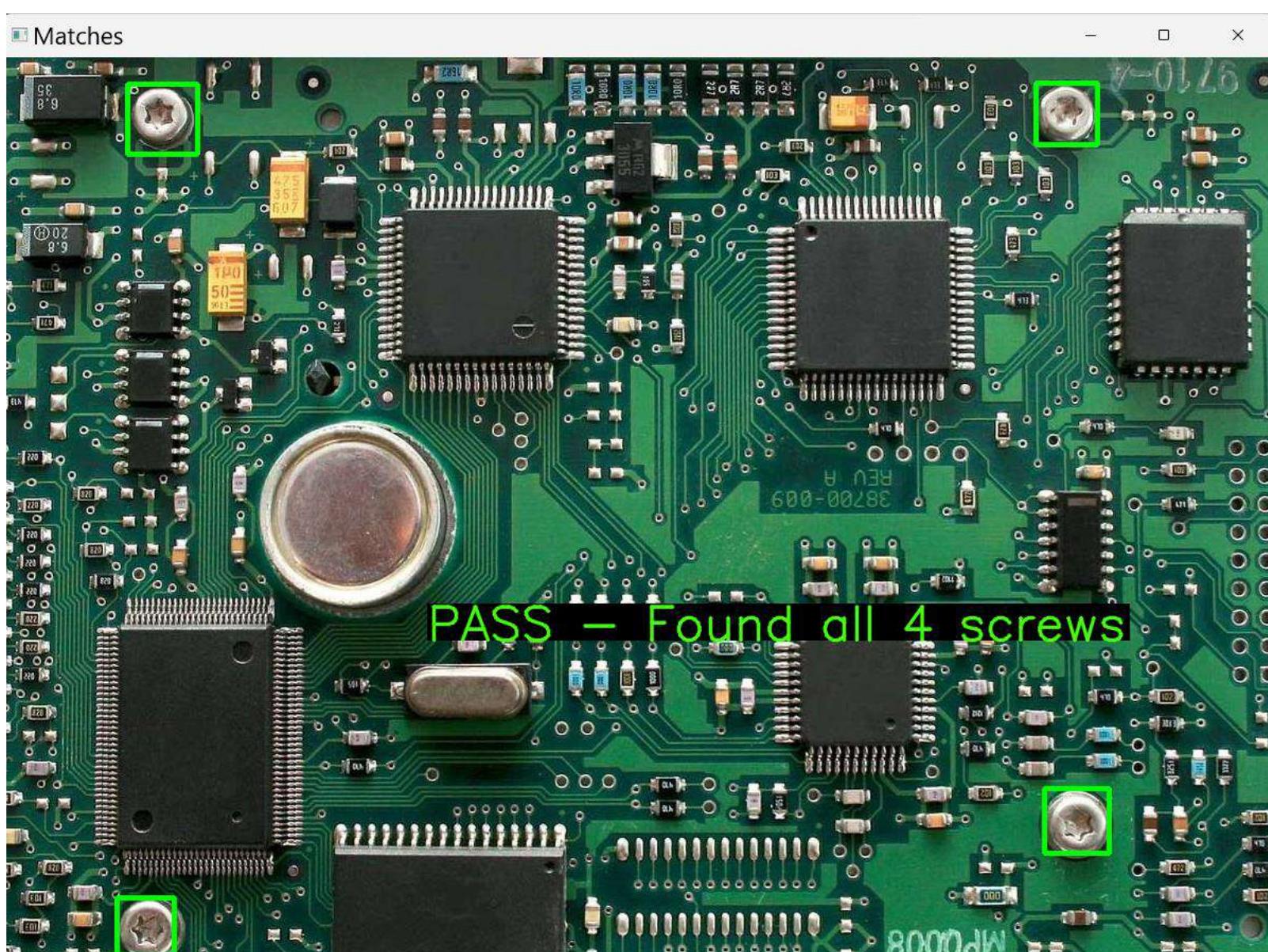
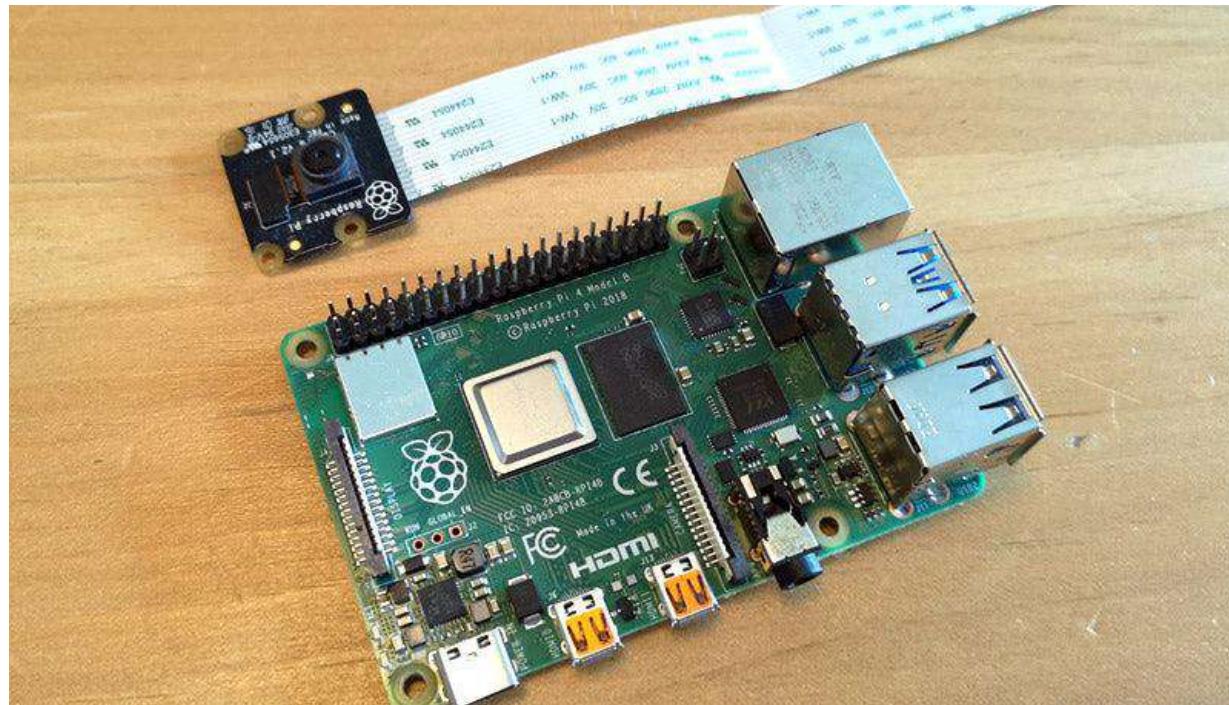
Example Of C++ Code

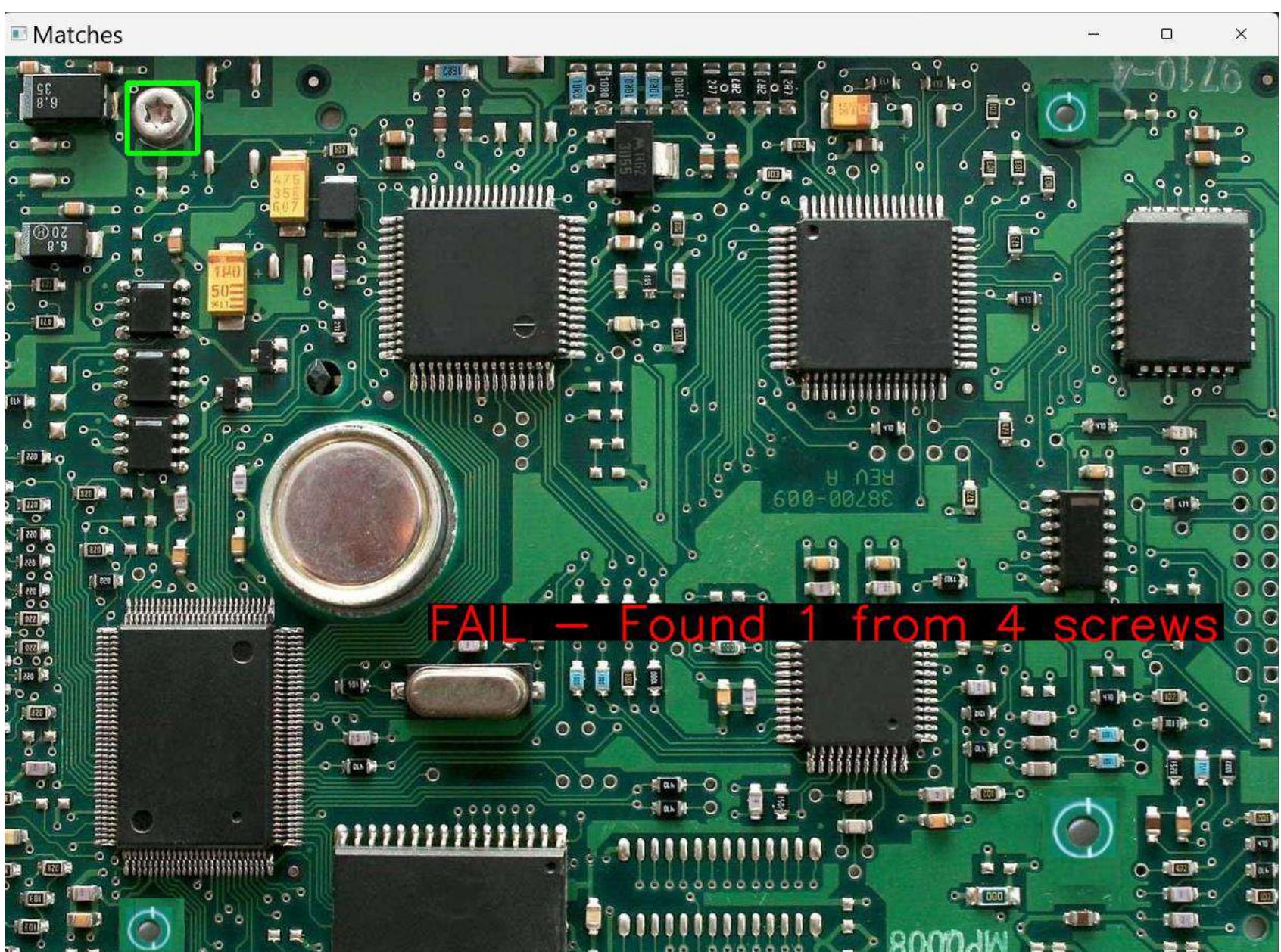
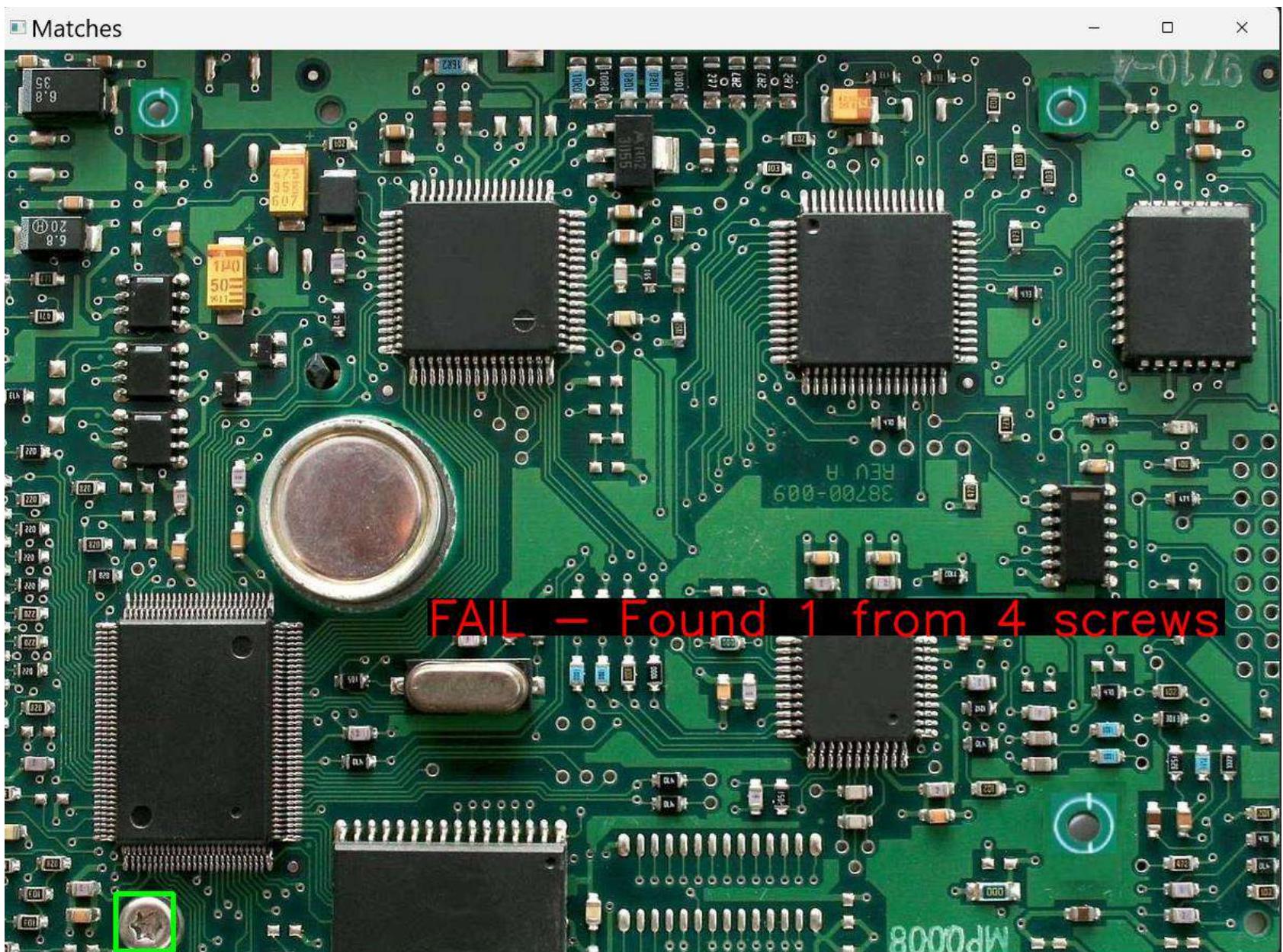
```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/ioctl.h>
5
6 int main(int argc, char **argv)
7 {
8     int on;
9     int led_no;
10    int fd;
11    if (argc != 3 || sscanf(argv[1], "%d", &led_no) != 1 || sscanf(argv[2], "%d", &on) != 1 ||
12        on < 0 || on > 1 || led_no < 0 || led_no > 3) {
13        fprintf(stderr, "Usage: leds led_no 0|1\n");
14        exit(1);
15    }
16    fd = open("/dev/leds0", 0);
17    if (fd < 0) {
18        fd = open("/dev/leds", 0);
19    }
20    if (fd < 0) {
21        perror("open device leds");
22        exit(1);
23    }
24    ioctl(fd, on, led_no);
25    close(fd);
26    return 0;
```

Basic Image Processing With Raspberry Pi And OpenCV (Object Detection)

Software Tools and Programming Language : Python, Ubuntu Linux, OpenCV

This software can detect missing screw on PCBA. The image can take from camera on Raspberry pi board. All 4 screws must presence on PCBA to PASS the test. With Cascade Classifier Training, you can training image manually with tool providing from OpenCV.





```

read2.pyx
36 fontScale = 1.5
37
38
39 if count > 3:
40     color = (16, 255, 72) #green B G R
41     text = "PASS - Found all 4 screws"
42 else:
43     color = (3, 3, 255) #red B G R
44     text = "FAIL - Found " + str(count) + " from 4 screws"
45
46 # Line thickness of 2 px
47 thickness = 2
48
49
50 rectangle_bgr = (0, 0, 0)
51
52 # get the width and height of the text box
53 (text_width, text_height) = cv.getTextSize(text, font, fontScale,
thickness)[0]
54 # set the text start position
55 text_offset_x = 400
56 text_offset_y = detection_image.shape[0] - 300
57 # make the coords of the box with a small padding of two pixels
58 box_coords = ((text_offset_x, text_offset_y), (text_offset_x + text_width + 2, text_offset_y - text_height - 2))
59
60 cv.rectangle(detection_image, box_coords[0], box_coords[1], rectangle_bgr, thickness)
61 cv.putText(detection_image, text, (text_offset_x, text_offset_y), font, fontScale, color, thickness, cv.LINE_AA, False)
62
63 cv.imshow('Matches', detection_image)
64
65

vision.pyx
74 points = []
75
76
77 # Loop over all the rectangles
78 for (x, y, w, h) in rectangles:
79     # Determine the center position
80     center_x = x + int(w/2)
81     center_y = y + int(h/2)
82     # Save the points
83     points.append((center_x, center_y))
84
85
86 # given a list of [x, y, w, h] rectangles and a canvas image to draw
87 # on, return an image with
88 # all of those rectangles drawn
89 def draw_rectangles(self, haystack_img, rectangles):
90     # these colors are actually BGR
91     line_color = (0, 255, 0)
92     line_type = cv.LINE_8
93
94     for (x, y, w, h) in rectangles:
95         # determine the box positions
96         top_left = (x, y)
97         bottom_right = (x + w, y + h)
98         # draw the box
99         cv.rectangle(haystack_img, top_left, bottom_right, line_color,
thickness=3, lineType=line_type)
100
101
102 def draw_rectangles_count(self, haystack_img, rectangles):
103     # these colors are actually BGR
104     count=0
105
106     for (x, y, w, h) in rectangles:
107         count=count+1
108
109

```

