GANPAT UNIVERSITY									
FACULTY OF ENGINEERING & TECHNOLOGY									
Programme	Master of Technology					Branch/Spec.	Electronics and Communication Engineering – Embedded Systems		
Semester	II					Version	1.0.0.0		
Effective from Academic Year		2026-27				Effective for the Batch admitted in		July 2026	
Course Code	3ES2PE02	Course Name				Programming for Embedded Systems			
Teaching Scheme						Examination Scheme (Marks)			
(Per week)	Lecture (DT)		Practical (Lab.)		Total		CE	SEE	Total
	L	TU	P	TW					
Credit	3	0	0	-	3	Theory	50	50	100
Hours	3	0	0	-	3	Practical	00	00	00

Pre-requisites

Basic knowledge of programming concepts such as variables, data types, operators, control statements, functions, arrays, and object-oriented programming (C++), along with Python fundamentals. Familiarity with basic computer architecture, memory concepts, bitwise operations, and problem-solving skills is recommended for developing embedded system applications.

Course Outcomes

On successful completion of the course, the students will be able to:

CO1	To introduce object-oriented programming concepts in C++ relevant to embedded systems.
CO2	To build proficiency in Python for embedded scripting, automation, and prototyping.
CO3	To enable students to interface hardware using both C++ and Python.
CO4	To compare performance and use-case trade-offs between C++ and Python in embedded applications.

Theory Syllabus

Unit	Content	Hrs.
1	**C++ for Embedded Systems:** Difference between C and C++ for embedded use; classes, objects, constructors/destructors; inheritance, polymorphism in resource-constrained systems; templates basics; embedded-specific concerns — avoiding dynamic memory, exception handling overhead, RAII; memory-mapped I/O using C++ abstractions.	8
2	**Advanced C++ Concepts for Embedded Applications:** Operator overloading; namespaces; static/const correctness; embedded C++ subset (EC++); real-time considerations in C++; interrupt service routines with C++ wrappers; introduction to embedded C++ frameworks (mbed, Arduino C++ style).	10
3	**Python Fundamentals for Embedded Applications & Hardware Interfacing:** Python basics recap — data structures, functions, modules; file handling; exception handling; Python for scripting and automation of embedded workflows; virtual environments and package management (pip). GPIO control using Python (RPi.GPIO / gpiozero style libraries); serial communication (pySerial); I2C/SPI interfacing using Python libraries; using Python on Raspberry Pi/microcontroller boards (Micro Python/Circuit Python basics).	10
4	**Data Handling & Automation with Python:** Working with sensor data logging; JSON/CSV handling; basic networking with Python (sockets, REST API calls) for IoT-embedded integration; automation scripts for testing	8

	embedded devices.	
5	**Comparative Study & Mini Project:** Performance comparison — C++ vs Python vs C in embedded contexts; choosing the right language for a task; case studies; mini project integrating C++/Python with an embedded board (sensor read + processing + display/log).	9
Practical Content		
Practical, assignments and tutorials are based on above syllabus.		
Text Books		
1	Michael Barr, Programming Embedded Systems in C and C++, O'Reilly — the standard foundational text for embedded C/C++ concepts, memory constraints, and hardware-level programming.	
Reference Books		
1	Bjarne Stroustrup, The C++ Programming Language, 4th Edition, Addison-Wesley — the authoritative reference for advanced C++ (OOP, templates, RAII) used in Units I–II of your syllabus.	
2	Simon Monk, Programming the Raspberry Pi: Getting Started with Python, McGraw-Hill — practical, hands-on coverage of Python for GPIO/hardware interfacing, directly matching Units III–IV.	
ICT/MOOCs Reference		
1	https://youtube.com/playlist?list=PLUQpHm_JtukII_8U9pucV61MX2YE2CSUt&si=oPAwHaOtV2BTmd1U	
2	https://youtube.com/playlist?list=PLUm-FI8ckafd1GP311wh9IyIRdPsm292n&si=drK-Cbbnj_haOnKH	

Mapping of CO with PO and PSO :															
Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	1	1	1	1	1	1	2	3	2	1
CO2	2	2	3	2	3	1	1	1	2	2	1	2	2	3	2
CO3	3	3	3	3	3	2	1	1	2	2	2	2	3	3	3
CO4	3	3	2	3	2	2	2	2	1	2	2	3	3	2	3