

NALLABOLU MANASA

Email: nallabolumanasa0@gmail.com | Phone Number: 9704878701

CAREER OBJECTIVE

To obtain a responsible position in a reputed organization where I can apply my technical knowledge, enhance my skills, and contribute effectively to organizational growth.

TECHNICAL SKILLS

- Programming Languages:
 - Shell scripting
 - Advanced C programming
 - OOP using C++
 - Data structures
- System programming:
 - Linux Kernel system calls
 - IPC mechanisms – Pipe, FIFO, Shared memory
 - Network Programming using TCP and UDP sockets
 - pThreads – Multi thread programming
- Embedded Linux:
 - U-boot, cross compiling, porting Linux kernel
 - Linux bring-up on Emertxe ARM board
- Embedded controllers:
 - Hands-on working with GPIOs, Analog I/Os, Memory usage, interfacing, character LCD
 - Peripherals usage - Timers, Counters and Interrupts
 - Communication protocols - UART, SPI, I2C, CAN etc

COURSE WORK

- Microprocessor
- Digital Electronics
- Digital Signal Processing

EDUCATION

- B.tech(EEE), JNTUACEA, 83%, 2021-2025
- Class – XII, Sai ram junior college, 96.4%, 2021
- Class – X, Sri Siva Chaitanya vidya mandir, 10GPA, 2019

PROJECTS

Project Number:1

Title	Image Steganography using LSB Encoding and Decoding
Project brief	The objective was to send a secret text file encoded inside an image of bmp file format. Encoded the length of the secret text and then encoded the data into the LSB of the image bytes. The decoding process involves decoding the length and then decoding the text bit by bit. The final output is the secret text after decoding.
Technologies used	Embedded C – File operations, Pointers, Bitwise operations, Functions, Makefiles, Command line arguments
Key challenges & Learnings	✓ Faced challenges in understanding the BMP image header and pixel structure and resolved them through literature study. ✓ Faced issues in bitwise data embedding and extraction and resolved them by careful manipulation of LSB operations to maintain data integrity. ✓ Faced memory and pointer-related errors during file I/O and resolved them through debugging and step-by-step validation.

Project number:2

Title	Lexical Analyser
Project brief	Developed a lexical analyzer to scan source code and break it into tokens such as keywords, identifiers, operators, and constants. The project helps in understanding the first phase of compiler design by analyzing and classifying program elements.
Technologies used	Advanced C – Loops, File handling, strings, conditional statements
Key challenges & Learnings	✓ Identifying and classifying different tokens such as keywords, identifiers, operators, constants, and delimiters from source code. ✓ Faced issues while Handling whitespace, comments, and special characters without affecting token recognition. ✓ Improved skills in string handling and file processing in C.

Project number:3

Title	Mp3 tag reader
Project brief	Created a program to read and display metadata from MP3 files, such as song title, artist, album, and year.
Technologies used	C Programming, File Handling, String Handling, Structures, Functions, Command-line I/O
Key challenges & Learnings	✓ Understanding the MP3 file structure and ID3 tag formats. ✓ Faced difficulties Correctly locating and extracting metadata such as title, artist, album, and year from MP3 files. ✓ Improved understanding of structures and string manipulation.

Project number:4

Title	Address Book
Project brief	Implemented an Address Book application for managing contact details including name, phone number, and email.
Technologies used	C Programming – File Handling, Arrays, Structures, Functions, String Handling, Command-line Input.
Key challenges & Learnings	✓ Managing contacts efficiently with add, search, update, and delete operations. ✓ Handling file operations to store and retrieve data correctly. ✓ Learned how to organize and manage data efficiently.

Project number:5

Title	Minishell
Project brief	Implemented a Minishell in C that mimics basic Unix shell behavior, supporting command parsing, execution of internal and external commands, process management, and signal handling.
Technologies used	C Programming, Linux, system calls, process management, signal handling.
Key challenges & Learnings	✓ Faced challenges in managing background processes and job control effectively. ✓ Faced challenges in ensuring proper process synchronization using <code>wait()</code> and <code>waitpid()</code>. ✓ Gained strong understanding of Linux shell internals and command execution flow.

Project number:6

Title	Trivial File Transfer Protocol (TFTP) Client–Server Implementation
Project brief	Implemented TFTP (Trivial File Transfer Protocol) in C to enable simple file upload and download between client and server using UDP, supporting basic operations like read/write requests and error handling.
Technologies used	C programming, Socket Programming (UDP), Linux system calls, Networking concepts,
Key challenges & Learnings	✓ Faced challenges in handling packet transmission using <code>sendto()</code> and <code>recvfrom()</code> functions, ✓ Implemented block-based file transfer with DATA and ACK packets. ✓ Learned how client-server communication works using UDP sockets.

Project number:7

Title	Arbitrary Precision Calculator
Project brief	Implemented an Arbitrary Precision Calculator in C to perform arithmetic operations on very large numbers beyond the limit of standard data types. The program uses data structures like linked lists to store and manipulate individual digits, enabling accurate addition, subtraction, multiplication, and division of large integers.
Technologies used	C programming, Data Structures (Linked List), Dynamic Memory Allocation, File Handling, Linux system calls
Key challenges & Learnings	✓ Faced challenges while implementing carry and borrow handling in arithmetic operations. ✓ Faced difficulties in managing linked list nodes during multiplication and division operations. ✓ Learned how to perform large number calculations using data structures.

Project number:8

Title	Inverted Search
Project brief	Implemented an Inverted Search system in C to efficiently store and retrieve words from multiple text files. The program creates a hash-based indexing structure that maps each word to the files in which it appears, along with occurrence count. It supports operations like creating a database, searching for a word, saving, and updating the database.
Technologies used	C programming, Data Structures (Hash Table, Linked List), File Handling, Dynamic Memory Allocation, Linux system calls
Key challenges & Learnings	✓ Faced challenges in implementing hashing and managing linked list nodes for storing words. ✓ Faced issues while updating and saving the database to files and resolved them through proper file handling.

- ✓ Learned how indexing improves search performance compared to scanning entire files.

Project number:	9
Title	CAN based automotive dashboard
Project brief	Developed a CAN-based automotive dashboard system to monitor and display vehicle parameters such as speed, temperature, fuel level, and RPM. The system uses CAN (Controller Area Network) protocol for communication between different electronic control units (ECUs). Sensor data is transmitted over the CAN bus and displayed on an LCD/SSD using a microcontroller, ensuring reliable and real-time data exchange
Technologies used	C programming, CAN Protocol, Microcontroller, MPLAB, Sensors, LCD/SSD Display.
Key challenges & Learnings	✓ Faced challenges in understanding and implementing CAN protocol communication between nodes. ✓ Faced issues in transmitting and receiving data frames accurately over the CAN bus. ✓ Learned how CAN protocol ensures reliable and efficient communication in automotive systems.

ACADEMIC PROJECTS

Title	Footstep Power Generation using Piezoelectric Sensors
Project brief	The system generates electricity by converting mechanical pressure from human footsteps into electrical energy using piezoelectric sensors. The generated power is stored in a battery or capacitor and can be used to charge small electronic devices or power low-energy applications like LED lights or displays
Technologies used	Arduino Uno, Piezoelectric sensors, Bridge rectifier, Rechargeable battery, Voltage regulator (LM7805)
Key challenges & Learnings	✓ Faced challenges in generating sufficient power output from piezoelectric sensors due to low voltage and current. ✓ Faced challenges in efficient energy storage using batteries/capacitors without significant losses. ✓ Learned how mechanical energy is converted into electrical energy using piezoelectric sensors.