

SMART BABY CRADLE USING IOT

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering with Internet of Things

By

KAMALIKA T (40732004)

SHASHANK BABU (40733026)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119**

APRIL 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KAMALIKA T (40732004)** and **SHASHANK BABU (40733026)** who carried out the project entitled "**SMART BABY CRADLE USING IOT**" under my supervision from November 2023 to April 2024.

Internal Guide

Dr. S. JAYANTHI, M.E., Ph.D.

HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Head of the Department
Dr. L. LAKSHMANAN, M.E., Ph.D.,
J. P. R. Ravi Varma
Chemical Engineering

Submitted for Viva voce Examination held on 15-04-2024

Internal Examiner

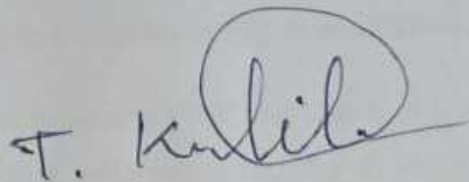
External Examiner

DECLARATION

I, Kamalika T (40732004), hereby declare that the Project Report entitled "Smart Baby Cradle using IoT" done by me under the guidance of Dr. S. Jayanthi, M.E., Ph.D. is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering with Internet of Things**.

DATE: 15/4/24

PLACE: Chennai

A handwritten signature in black ink, appearing to read 'T. Kamalika', with a large circular flourish above the name.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean, School of Computing, Dr. L. Lakshmanan M.E., Ph.D., Head of the Department of Computer Science and Engineering** and **Dr. S. Vigneshwari M.E., Ph.D., Head of the Department of Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Internal Guide **Dr. S. Jayanthi M.E., Ph.D.** for her valuable guidance, suggestions and constant encouragement which paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The standard automatic child cradle frameworks are often expensive and may not always provide the level of utility desired by parents. This project seeks to address the needs of mothers who find themselves overwhelmed and lacking the support of a cleaning professional or caregiver to assist with childcare. The proposed solution entails a smart cradle design with a side-to-side rocking action aimed at calming and soothing infants. Unlike traditional cradles, which typically require significant physical effort from parents to induce swinging motion, this smart cradle prototype aims to automate the rocking process, thereby easing the burden on caregivers. The versatility of this smart cradle makes it an ideal solution for busy parents who cannot devote their entire day to caring for their new-borns. Beyond home use, this technology can also benefit maternity hospitals by assisting staff responsible for infant care. By automating certain caregiving tasks and providing real-time monitoring capabilities, the smart cradle alleviates the workload of caregivers while ensuring the well-being of infants under their supervision. Moreover, the smart cradle will incorporate additional features to enhance its functionality. One such feature is an autonomously swaying mechanism driven by a geared motor system. This mechanism mimics the gentle rocking motion typically provided by caregivers, promoting infant relaxation and sleep. Additionally, the inclusion of a PCB for sensing moisture levels adds another layer of efficiency to the cradle. This feature enables the detection of wet conditions, such as diaper leaks, prompting timely intervention and ensuring the baby's comfort. In summary, this innovative smart cradle prototype offers a comprehensive solution to the challenges faced by parents and caregivers in providing optimal care for new-borns. Through the integration of advanced technology and thoughtful design features, it provides a convenient and reliable way to soothe, monitor, and attend to infants' needs, enhancing the caregiving experience for both parents and professionals alike.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	viii
	INTRODUCTION	1
1	1.1 Overview	1
	1.2 Objective of the Project	1
	1.3 Outline of the Project	3
	1.3.1 Embedded System	3
2	LITERATURE SURVEY	4
	2.1 Inferences from Literature Survey	5
	2.2 Open Problems in Existing System	6
3	REQUIREMENT ANALYSIS	8
	3.1 Feasibility Study/Risk Analysis	8
	3.2 Software Requirements Specification Document	8
	3.2.1 Embedded Software for the Cradle	8
	3.2.2 Cloud-Based Backend	9
	3.2.3 IoT Connectivity	9
	3.2.4 Security and Privacy Measures	9
	3.2.5 Remote Monitoring and Alerts	9
	3.2.6 Data Analytics and Machine Learning	10
	3.2.7 User Interface Design	10
4	DESCRIPTION OF PROPOSED SYSTEM	11
	4.1 Description of Hardware for Implementation and Testing Plan of the Proposed Model/System	12
	4.1.1 Arduino Uno	12

	4.1.2 Servo Motor	14
	4.1.3 Moisture Detection Sensor	16
	4.1.4 ESP Camera	18
	4.1.5 Buzzer	20
	4.2 Project Management Plan	21
5	IMPLEMENTATION DETAILS	22
	5.1 Overview of Implementation	22
	5.1.1 Sensors	22
	5.1.2 Actuators	22
	5.1.3 Control System	22
	5.1.4 Power Supply	22
	5.1.5 Materials and Construction	23
	5.1.6 Data Logging and Analysis	23
	5.1.7 Safety Regulations and Standards	23
	5.1.8 Testing and Quality Assurance	23
	5.1.9 User Experience Design	23
	5.1.10 Optional Features	23
	5.2 Testing	24
	5.2.1 Initial Inspection	24
	5.2.2 Safety Testing	24
	5.2.3 Electrical Safety	24
	5.2.4 Functional Testing	24
	5.2.5 Durability Testing	25
	5.2.6 Temperature and Humidity Testing	25
	5.2.7 Regulatory Compliance	25

	5.2.8 Quality Assurance	25
	5.2.9 Documentation	25
6	RESULTS AND DISCUSSION	26
	6.1 Arduino IDE	26
	6.2 Embedded C	27
7	CONCLUSION	29
	REFERENCES	30
	APPENDIX	32
	SOURCE CODE	32
	SCREENSHOTS	36
	RESEARCH PAPER	37
	CERTIFICATE	41

LIST OF FIGURES

FIGURE NO	FIGURE NAME	Page No.
1.1	Architecture of Embedded System	3
4.1	Arduino UNO board	12
4.2	Arduino UNO board with Digital Input/Output	13
4.3	Servo Motor	14
4.4	Servo Motor Mechanism	15
4.5	Moisture Detection Sensor	16
4.6	Moisture Detection Sensor Circuit Diagram	17
4.7	Working of Moisture Sensor	18
4.8	ESP32-CAM Module	19
4.10	Buzzer	20
5.1	Embedded C code	27

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

A traditional baby cradle serves as a tranquil space for infants to rest and drift off to sleep peacefully. However, conventional cradles lack the electronic capabilities necessary for automated rocking, requiring manual intervention from caregivers to soothe the baby. This reliance on manpower poses challenges for parents, as constant vigilance is needed to ensure the safety and comfort of their child. Recognizing these limitations, there is a pressing need for automatic cradles equipped with battery or power sources to provide enhanced convenience and functionality. The advent of automatic cradles revolutionizes the caregiving experience for parents by introducing automation into the soothing process. By incorporating battery or adapter-powered mechanisms, these modern cradles alleviate the need for manual rocking, offering a seamless and effortless solution for parents seeking to comfort their infants. This automation not only reduces the physical exertion required from caregivers but also ensures a consistent and gentle rocking motion, promoting uninterrupted sleep for babies.

1.2 OBJECTIVE OF THE PROJECT

- To design the development of an intelligent baby care, which has ability to monitor baby movement, bed-wet condition and body temperature.
- To make a baby cradle is safe and comfortable for baby with using PIR sensor to detect the movement of the baby body as well as bed-wet condition to keep away baby from hygienic environment.
- To make cradle innovation that is more flexible and less expensive to market.
- User friendly- simple and complete with instruction

As its name suggests, embedded means something that is attached to another thing. An embedded system can be thought of as a computer hardware system having software embedded in it. An embedded system can be an independent system or it can be a part of a large system. An embedded system is a microcontroller or microprocessor-based system which is designed to perform a specific task. For

example, a fire alarm is an embedded system; it will sense only smoke. An embedded system has three components:

- It has hardware.
- It has application software.

It has Real Time Operating system (RTOS) that supervises the application software and provide mechanism to let the processor run a process as per scheduling by following a plan to control the latencies. RTOS defines the way the system works. It sets the rules during the execution of application program. So, we can define an embedded system as a Microcontroller based, software driven, reliable, real-time control system. Characteristics of an Embedded System -

- Single-functioned – An embedded system usually performs a specialized operation and does the same repeatedly. For example: A pager always functions as a pager.
- Tightly constrained – All computing systems have constraints on design metrics,

But those on an embedded system can be especially tight. Design metrics is a measure of an implementation's features such as its cost, size, power, and performance. It must be of a size to fit on a single chip, must perform fast enough to process data in real time and consume minimum power to extend battery life. Reactive and Real time – Many embedded systems must continually react to changes in the system's environment and must compute certain results in real time without any delay.

Consider an example of a car cruise controller; it continually monitors and reacts to speed and brake sensors. It must compute acceleration or deaccelerations repeatedly within a limited time; a delayed computation can result in failure to control of the car.

Memory – It must have a memory, as its software usually embeds in ROM. It does not need any secondary memories in the computer.

Connected – It must have connected peripherals to connect input and output devices.

HW-SW systems – Software is used for more features and flexibility. Hardware is used for performance and security.

1.3 OUTLINE OF THE PROJECT

1.3.1 EMBEDDED SYSTEM

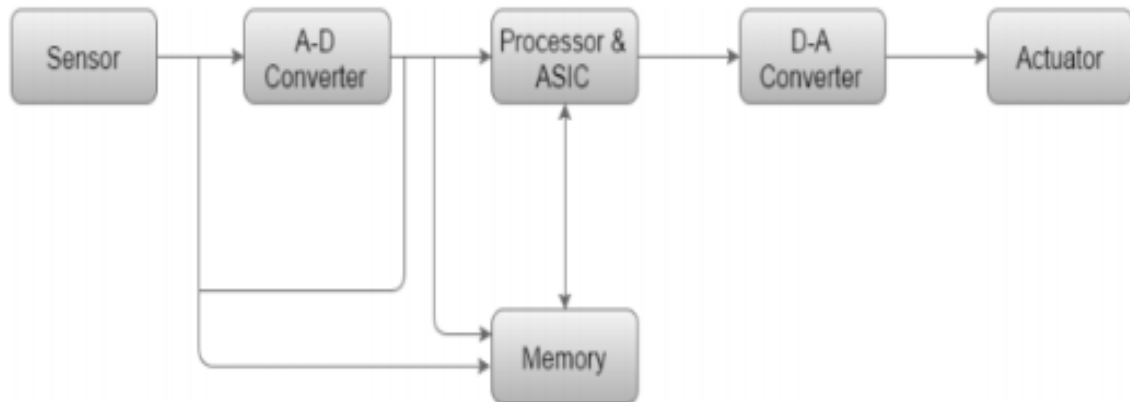


Fig. 1.1: Architecture of Embedded System

Sensor – It measures the physical quantity and converts it to an electrical signal which can be read by an observer or by any electronic instrument like an A2D converter. A sensor stores the measured quantity to the memory.

A-D Converter – An analog-to-digital converter converts the analog signal sent by the sensor into a digital signal.

Processor & ASICs – Processors process the data to measure the output and store it to the memory.

D-A Converter – A digital-to-analog converter converts the digital data fed by the processor to analog data.

Actuator – An actuator compares the output given by the D-A Converter to the actual (expected) output stored in it and stores the approved output.

CHAPTER 2

LITERATURE SURVEY

The automatic cradle system by Vijayamahantesh Hiremath, Dr. P. Vankataratnam [2017] : In this intelligent baby cradle system was developed. The cradle was capable of detection the movement of the baby and initiate cradle swing. in addition, within the event of Bed wet or physiological condition, the developed device is capable of causing SMS. The device may be accustomed minimize the work of the fogeys and nurses in home and hospitals severally

Smart Cradle by Natheera, S.Sundaravadivel, N.L.Visakan, S.Viswanathan and M.Vivekkumar [2018] : In this system mike module used to detect the cry of the baby and it connected with Arduino Uno. The baby cries then the mechanically wiegen swing are activated. The wet sensing element results a Boolean price either true or false indicating whether or not the bed within the cradle is wet or not. All are connect with Arduino Uno.

IOT Based Baby Monitoring System for Smart Cradle by Waheb A. Jabbar, Senior Member [2017] : The system consists of a baby cradle that mechanically swings employing a motor once the baby cries in keeping with the sound device signal. The oldsters will observe the traditional information recorded within the MQTT server cloud, like close temperature and remote switches, through the web victimization MQTT server, whereas the abnormal conditions are sent to the oldsters with triggering alarm to require acceptable actions. The oldsters also can monitor the baby's condition through Associate in The nursing external net camera and put on the lullaby toy settled on the baby's cradle remotely via the MQTT server to entertain the baby.

Smarter Cradle System for Enhanced Parenting by Sharmadha Senthil Nathan, Shivani Kanmani, Shruthi Kumar, Madheswari Kanmani [2018] : The system consists of automatic Rocking of the Baby Cradle, Detection of wet condition and detection of Temperature in Baby Cradle System. Cry analyzing system that detects

the baby cry voice and consequently the cradle swings until the baby stops crying. The speed of the cradle may be controlled as per the user want. Wet sensor and temperature sensor are put in a diaper and it connects with mobile through a microcontroller. It may be harmful to the baby.

Development of an Intelligent Cradle for Home and Hospital Use was published [2015] : In this research paper author states that system is designed monitor baby movement, bed-wet condition and body temperature. PIR sensor detects the movement of infant. After the location of enlivening of infant various occasions are terminated which incorporate sending notices to mother by SMS. And also observes the temperature of the infant to give therapeutic thoughtfulness regarding the child and cautions the folks if the body temperature of the child goes above to given threshold temperature. The cradle additionally incorporates a wet sensor which will caution the folks or the attendants for bunk wetting of the infant. It will help to keep the infant in a hygienic environment. System was giving three modules that was obviously good point. But there was absence of one important module of swinging cradle automatically.

2.1 INFERENCES FROM LITERATURE SURVEY

From Literature review 1

The concept of using Internet of Things (IoT) technologies in cradle systems has gained significant attention in recent years. Several studies and research papers have been published, exploring various aspects of smart cradles and their applications.

From Literature review 2

These literature survey proposed a smart cradle system that integrates IoT technologies for infant monitoring. The system utilized sensors to monitor the baby's vital signs, including temperature, heart rate, and respiration, and transmitted the data to a cloud-based server for analysis .

From Literature review 3

Many home-care systems are available but majority of this system are specially designed for the aged people and patients. These systems can monitor their health

status, automatically send out emergency signals, and have other functions. However, the caring methods for infants are not the same. Children and adults require different type of care because they are totally dependent for their normal functions on someone else.

From Literature survey 4

The existing cradles from literature survey it is understood that some cradles designed, but they usually have only one or two features. From that one or two features they itself were making trouble for taking care of baby or disturbing the baby from sleep.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

The concept of integrating smart technology into baby cradles represents a promising advancement in infant care, offering convenience and peace of mind for parents. By incorporating features such as smartphone control, video/audio streaming capabilities, and automatic comfort functions, these smart cradles aim to enhance the caregiving experience. However, despite their potential benefits, there are notable drawbacks and considerations that must be addressed to ensure the safety and effectiveness of such devices. One of the primary concerns regarding smart cradles is the potential risk posed by the presence of electronic components within reach of the baby. Babies are naturally curious and may attempt to interact with or even ingest these components, leading to safety hazards such as electrical shock or choking hazards. Additionally, there is the possibility of components becoming dislodged or broken during the baby's exploration, further increasing the risk of injury. Therefore, careful design and implementation of safety measures are essential to mitigate these risks and ensure the overall safety of the smart cradle. Moreover, existing baby cradles often lack comprehensive monitoring capabilities, limiting their functionality to serving as a basic sleeping space for infants. Smart cradles have the potential to address this gap by incorporating sensors to monitor various parameters of an infant's well-being, such as temperature, humidity, and movement. This data can be transmitted to the parent's smartphone, enabling them to track their baby's health and comfort levels in real time. However, the integration of these features must be carefully executed to ensure accuracy and reliability while minimizing the risk of false alarms or technical issues. In conclusion, while smart

cradles offer promising benefits in terms of convenience and monitoring capabilities, it is crucial to address safety concerns and prioritize the well-being of the baby. By implementing robust safety measures, conducting thorough testing, and ensuring reliable functionality, smart cradles have the potential to revolutionize infant care and provide peace of mind to parents. However, ongoing research and development efforts are needed to overcome existing challenges and fully realize the potential of smart cradles in enhancing the caregiving experience. Furthermore, parents' apprehensions regarding the safety of smart cradles may outweigh their interest in the additional features offered by such products. While features like smartphone control and real-time monitoring capabilities provide convenience and reassurance to parents, safety concerns may deter them from fully embracing these technologies. Thus, manufacturers must prioritize safety in the design and development of smart cradles, implementing robust safety features and conducting thorough testing to alleviate parents' worries and build trust in the product.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY/RISK ANALYSIS

The feasibility of developing a smart baby cradle using IoT (Internet of Things) technology depends on several key factors. First, market demand and competition need to be carefully assessed to determine if there is a significant demand for such a product and whether it can stand out in a competitive marketplace. Technical feasibility is another critical aspect, involving the availability of IoT components and the ability to design a safe and functional cradle. Ensuring compliance with regulatory and safety standards for baby products is paramount. Cost analysis is essential, considering both initial development costs and ongoing operational expenses, and it must be balanced against revenue projections. Moreover, risk assessment is necessary to identify and mitigate potential challenges, ranging from technical issues to data security concerns. The project's operational feasibility, including supply chain management and scalability, should also be evaluated. Lastly, ethical and environmental considerations, user acceptance, development timelines, and a clear marketing and distribution strategy are essential components of the feasibility assessment. Ultimately, a thorough examination of these factors will help determine whether a smart baby cradle using IoT is a viable and worthwhile endeavour.

3.2 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

3.2.1 Embedded Software for the Cradle

Developing firmware to control the hardware components of a smart cradle, including motors and sensors, is essential for ensuring its functionality and safety. This firmware serves as the bridge between the physical components of the cradle and the software interface, allowing for seamless communication and control. In addition to basic functionality, such as controlling motor speed and monitoring sensor data, the firmware plays a crucial role in real-time monitoring of the cradle's status to ensure the safety and comfort of the baby.

3.2.2 Cloud-Based Backend

Implementing cloud servers for data storage and processing is crucial for enabling remote access, scalability, and advanced analytics capabilities in the smart cradle system. By leveraging cloud infrastructure, such as Amazon Web Services (AWS) or Microsoft Azure, the smart cradle can securely store sensor data, user preferences, and other relevant information in a centralized and easily accessible location. This cloud-based approach also facilitates seamless integration with the accompanying mobile app, enabling users to access their data and control the cradle from anywhere with an internet connection.

3.2.3 IoT Connectivity

The smart cradle system can leverage communication protocols such as MQTT (Message Queuing Telemetry Transport), CoAP (Constrained Application Protocol), or HTTP (Hypertext Transfer Protocol) for efficient and reliable data transmission between devices and cloud services. These protocols enable real-time communication, asynchronous messaging, and support for various data formats, ensuring compatibility and interoperability within the system.

3.2.4 Security and Privacy Measures:

By implementing end-to-end encryption and secure login mechanisms, such as multi-factor authentication, the smart cradle system can enhance security and protect user privacy, ensuring that sensitive data remains confidential and secure from unauthorized access or interception. These security measures are essential for building trust and confidence among users and stakeholders in the smart cradle system, ultimately enhancing the overall security posture and reliability of the system.

3.2.5 Remote Monitoring and Alerts:

By incorporating alerts, notifications, and live video/audio streaming functionalities into the smart cradle system, parents can stay informed and connected with their baby's activities in real-time, enhancing their peace of mind and confidence in the system. These features provide valuable insights and reassurance, enabling parents

to respond promptly to any changes or unusual conditions detected by sensors and ensuring continuous monitoring of their baby's safety and comfort, even when they are not physically present.

3.2.6 Data Analytics and Machine Learning:

Integrating advanced algorithms and predictive analytics into the smart cradle system enhances its capabilities to analyse baby's sleep patterns and anticipate their needs, providing valuable insights and recommendations to parents for optimizing their baby's sleep and overall well-being. These features leverage data collected from sensors and user interactions to generate personalized recommendations and dynamically adjust the cradle's settings to promote better sleep quality and comfort for the baby.

3.2.7 User Interface Design:

Creating a visually appealing and intuitive UI for the mobile app enhances usability, engagement, and satisfaction with the smart cradle system. Controls for adjusting settings and accessing information should be prominently displayed and clearly labelled. Users should be able to easily adjust cradle settings, such as rocking speed or sleep schedules, and access real-time information, such as sensor data or alerts. Providing feedback, such as confirmation messages or progress indicators, helps users understand the outcome of their interactions and reduces uncertainty. By prioritizing clear navigation, interactive elements, and customization options, the app provides a seamless and enjoyable user experience for parents or caregivers, enabling them to monitor and control the smart cradle with ease and confidence.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

The proposed idea of a smart cradle represents a significant innovation in infant care, offering a cost-effective and user-friendly solution that has the potential to improve the quality of care for infants. By incorporating advanced monitoring capabilities and communication features, this system provides parents with assurance that their infants are well taken care of, even when they are not physically present. The smart cradle system is designed to be easy to use and accessible to a wide range of users, regardless of their technical expertise or familiarity with technology. The smart baby cradle has noise detection which will sense the noise when the baby is crying or making loud noise. A servo motor will rotate (swing) the cradle up to an angle that is safe when infant is inside the cradle. A GSM module to communicate with a remote device that is operated by parents. And an Arduino microcontroller to control and monitor the activities of the cradle. Also whenever the wet is detected it gives an buzzer alert and also sends an SMS alert to the parents. The intuitive user interface and clear controls make it simple for parents to navigate and interact with the system, ensuring a seamless and stress-free experience. The smart cradle represents a significant advancement in infant care technology, offering parents peace of mind and confidence in knowing that their baby is being well cared for. By providing continuous monitoring, real-time communication, and ease of use, this system has the potential to revolutionize the way parents care for their infants and improve the overall quality of infant care.

4.1 DESCRIPTION OF HARDWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

4.1.1 Arduino Uno

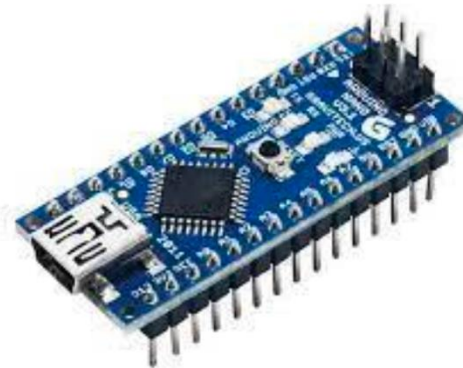


Fig. 4.1: Arduino Uno Board

The Arduino Uno microcontroller board, based on the ATmega328 chip, is a versatile and user-friendly platform that serves as the central control unit for a wide range of projects, including humanoid robots. With its 14 digital input/output pins, 6 analog inputs, and PWM outputs, the Arduino Uno offers ample connectivity options for interfacing with sensors, actuators, and other peripheral devices. One of the key advantages of the Arduino Uno is its ease of use and accessibility. Designed with beginners and hobbyists in mind, the Arduino development environment provides a simple and intuitive interface for writing, uploading, and debugging code. This makes it an ideal choice for robotics projects, where precise control and real-time responsiveness are essential. The Arduino Uno's built-in USB connection allows for easy communication with a computer, enabling rapid prototyping and development. Additionally, the presence of a power jack and compatibility with AC-to-DC adapters or batteries makes it easy to power the Arduino Uno and connected peripherals in various environments. Furthermore, the Arduino Uno's open-source nature and extensive community support make it a popular choice among robotics enthusiasts and educators alike. A vast array of libraries, tutorials, and example projects are available online, providing valuable resources for learning and experimentation. There are many versions of Arduino boards introduced in the market like Arduino Uno, Arduino Due, Arduino Leonardo, Arduino Mega, however, most common

versions are Arduino Uno and Arduino Mega. If you are planning to create a project relating to digital electronics, embedded system, robotics, or IoT, then using Arduino Uno would be the best, easy and most economical option.

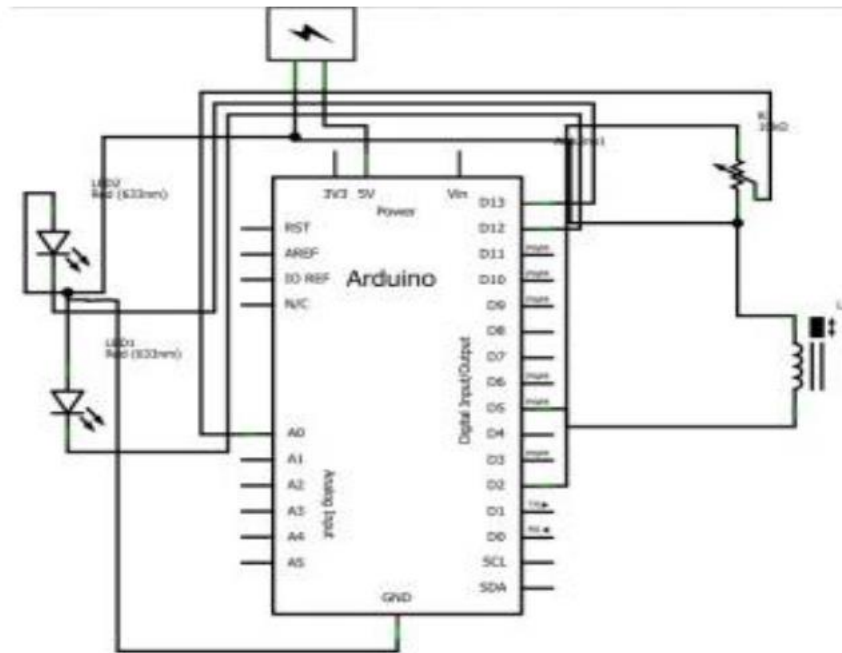


Fig. 4.2: Arduino Uno with Digital input/Output

There are various types of Arduino boards in which many of them were third-party compatible versions. The most official versions available are the Arduino Uno R3 and the Arduino Nano V3. Both of these run a 16MHz Atmel ATmega328P 8-bit microcontroller with 32KB of flash RAM 14 digital I/O and six analogue I/O and the 32KB will not sound like as if running Windows. Arduino projects can be stand-alone or they can communicate with software on running on a computer. For e.g. Flash, Processing, Max/MSP). The board is clocked by a 16 MHz ceramic resonator and has a USB connection for power and communication. You can easily add micro SD/SD card storage for bigger tasks. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware.

4.1.2 Servo Motor



Fig. 4.3: Servo Motor

In a smart baby cradle, the inclusion of a servo motor serves a pivotal role in creating a soothing and comforting environment for infants. The servo motor is responsible for generating gentle rocking motions, mimicking the calming sensation of being cradled in a caregiver's arms. This controlled movement not only aids in lulling the baby to sleep but also promotes relaxation and comfort, fostering a conducive atmosphere for restful slumber. By integrating IoT connectivity, parents gain the flexibility to remotely control the cradle's rocking motion, adjusting speed and patterns to suit their baby's preferences. Furthermore, safety features such as obstruction detection and overload protection ensure the baby's well-being during operation. With its low noise and energy-efficient operation, the servo motor contributes to a serene sleeping environment while minimizing disruptions. the integration of a servo motor within a smart baby cradle represents a significant leap forward in infant care technology. By combining the soothing power of gentle rocking motions with the convenience and versatility of IoT connectivity, this innovative solution provides parents with a reliable and effective tool for promoting their baby's sleep and overall well-being. With its focus on safety, convenience, and comfort, the servo motor-equipped smart baby cradle exemplifies the transformative potential of technology in enhancing the caregiving experience for both parents and infants alike.

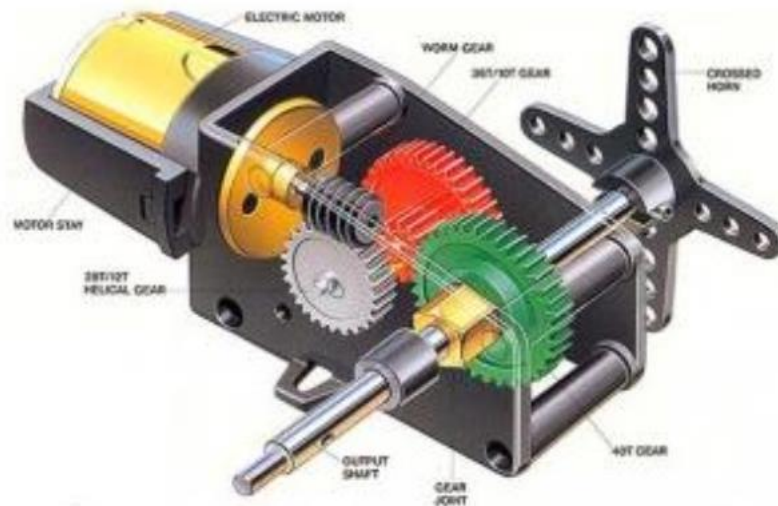


Fig. 4.4: Servo Motor Mechanism

The very simplest servomotors use position-only sensing via a potentiometer and bang-bang control of their motor; the motor always rotates at full speed (or is stopped). This type of servomotor is not widely used in industrial motion control, but it forms the basis of the simple and cheap servos used for radio-controlled models. More sophisticated servomotors use optical rotary encoders to measure the speed of the output shaft and a variable-speed drive to control the motor speed. Both of these enhancements, usually in combination with a PID control algorithm, allow the servomotor to be brought to its commanded position more quickly and more precisely, with less overshooting. The servo motor is actually an assembly of four things: a normal DC motor, a gear reduction unit, a position-sensing device and a control circuit. The DC motor is connected with a gear mechanism which provides feedback to a position sensor which is mostly a potentiometer. From the gear box, the output of the motor is delivered via servo spline to the servo arm. For standard servo motors, the gear is normally made up of plastic whereas for high power servos, the gear is made up of metal. A servo motor consists of three wires- a black wire connected to ground, a white/yellow wire connected to control unit and a red wire connected to power supply. The function of the servo motor is to receive a control signal that represents a desired output position of the servo shaft and apply power to its DC motor until its shaft turns to that position.

4.1.3 Moisture detection sensor

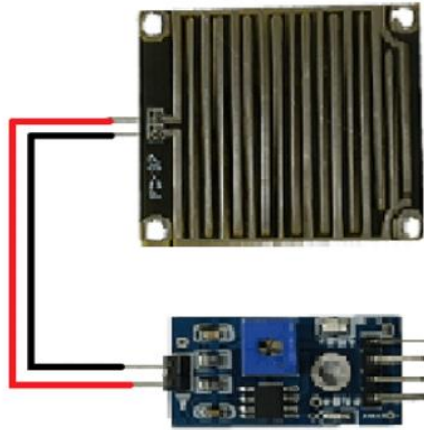


Fig. 4.5: Moisture detection sensor

A moisture detection sensor, also known as a humidity sensor or raindrop sensor, is a device designed to measure the moisture content in the surrounding environment. These sensors are commonly used in various applications, including agriculture, industrial processes, HVAC systems, and consumer electronics. The level of moisture in the air is often expressed as relative humidity (RH), which is the ratio of the amount of moisture present in the air to the maximum amount the air could hold at that temperature. The Moisture Sensor contains two open plates that act as probes for the sensor which together work as a variable resistor. The sensor will detect wetness under the baby sleeping over the cradle. Sensor will monitor the damp mattress of the baby. The output of the soil moisture sensor fluctuates continuously in the ADC value range of 0 to 1023. This phrase might refer to the percentage moisture value. The control module of the raindrop sensor has 4 outputs. VCC is connected to a 5V supply. The GND pin of the module is connected to the ground. The D0 pin is connected to the digital pin of the microcontroller for digital output or the analog pin can be used. To use the analog output, the A0 pin can be connected to the ADC pin of a microcontroller. In the case of Arduino, it has 6 ADC pins, so we can use any of the 6 pins directly without using an ADC converter. The sensor module consists of a potentiometer, LN393 comparator, LEDs, capacitors and resistors. The pinout image above shows the components of the control module. The rain board module consists of copper tracks, which act as a variable resistor. Its resistance varies with respect to the wetness on the rain board.

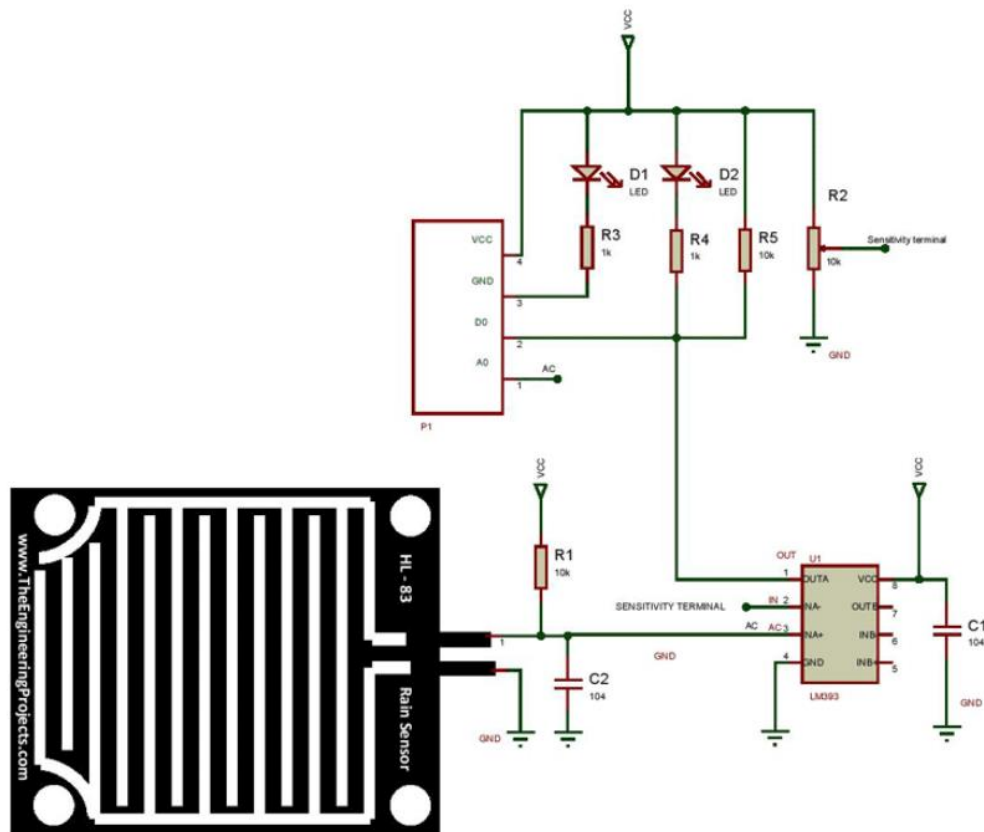


Fig. 4.6: Moisture Detection Sensor Module Circuit Diagram

As shown in the above figure, the R1 resistor and the rain board module will act as a voltage divider. Capacitors C1 and C2 are used as a biasing element. The input for the non-inverting terminal is taken from the connection point of the R1, and rain board module. Another point is taken from this connection and connected to the AO terminal of the control module.

The input to the inverting terminal of the LM393 is taken from the potentiometer (R2). The R2 resistor acts as a voltage divider, and by varying R2 we can vary the input voltage to the inverting terminal, which in turn affects the sensitivity of the control module. The connections are shown in the above fig. The resistors R3 and R4 will act as current limiting resistors, while resistor R5 will act as a pull-up resistor to keep the bus in a high state when not in use.

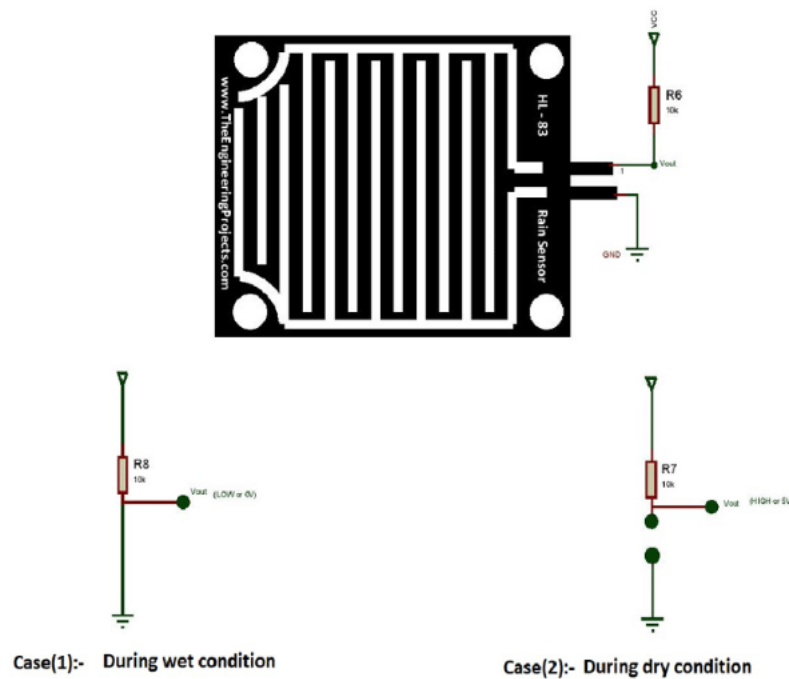


Fig. 4.7: Working of Moisture Sensor

When the A0 pin is connected to the microcontroller, an additional analog to digital converter (ADC) circuit is used. In the case of Arduino, it consists of 6 ADC pins, which can be directly used for calculation purposes.

Applications of Rain sensor:

- Automatic windshield wipers
- Smart Agriculture
- Home-Automation

4.1.4 ESP32 Camera

The ESP32-CAM is a small size, low power consumption camera module based on ESP32. It comes with an OV2640 camera and provides onboard TF card slot. The ESP32-CAM can be widely used in intelligent IoT applications such as wireless video monitoring, WiFi image upload, QR identification, and so on. When working with the ESP32-CAM, keep in mind that handling the camera module and processing images or video may require a good understanding of programming and microcontroller concepts. Additionally, power considerations, especially when using the camera, are important to ensure stable operation.



Fig. 4.8: ESP32-CAM Module

The microcontroller boasts dual-core processing power and a rich set of built-in peripherals, making it well-suited for a diverse array of IoT (Internet of Things) projects. With integrated Wi-Fi and Bluetooth connectivity, the ESP32-CAM empowers developers to seamlessly integrate wireless communication into their applications, facilitating tasks such as image transmission, remote monitoring, and surveillance. This connectivity also enables the board to serve as a hub for smart home systems, where visual data can be leveraged to enhance automation and user interaction.

Furthermore, the onboard flash memory, typically around 4MB in capacity, offers ample storage space for captured images, configuration data, or other application-specific information. This local storage capability enables the ESP32-CAM to operate autonomously, even in scenarios where continuous internet connectivity may not be available. Moreover, the board's GPIO pins provide flexible interfacing options, allowing developers to connect a variety of sensors, actuators, and peripherals to expand the scope and functionality of their projects. typically featuring an OV2640 sensor capable of capturing images up to 2 megapixels (1600x1200 pixels), this board extends the ESP32's utility beyond its traditional applications.

4.1.5 Buzzer



Fig. 4.9: Buzzer

A buzzer or beeper is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

Early devices were based on an electromechanical system identical to an electric bell without the metal gong. Similarly, a relay may be connected to interrupt its own actuating current, causing the contacts to buzz. Often these units were anchored to a wall or ceiling to use it as a sounding board. The word "buzzer" comes from the rasping noise that electromechanical buzzers made. A joy buzzer is an example of a purely mechanical buzzer and they require drivers. Other examples of them are doorbells. A piezoelectric element may be driven by an oscillating electronic circuit or other audio signal source, driven with a piezoelectric audio amplifier. Sounds commonly used to indicate that a button has been pressed are a click, a ring or a beep. A piezoelectric buzzer/beeper also depends on acoustic cavity resonance or Helmholtz resonance to produce an audible beep

4.2 PROJECT MANAGEMENT PLAN

During the initiation phase, the project team will define the project's scope, objectives, and overall vision. This includes identifying key stakeholders, such as parents, designers, engineers, and potential end-users. Additionally, a project charter will be developed to outline the project's goals and initial requirements, along with high-level timelines and budget estimates.

In the planning phase, a comprehensive project plan will be created, detailing all tasks, dependencies, and resource allocation. This plan will outline the technical specifications for the Smart Baby Cradle, including IoT sensors and connectivity requirements. It will also cover design considerations, hardware and software development, regulatory compliance, and potential risks and mitigation strategies. Moreover, a detailed schedule and budget will be established.

During the execution phase, the actual development of the Smart Baby Cradle will take place. This will include the creation of the hardware components, IoT connectivity, and associated software applications. Regular team meetings and progress tracking will be implemented to ensure the project stays on schedule and within budget. Quality control measures will be in place to guarantee the product meets the required standards.

Finally, in the closure phase, the Smart Baby Cradle project will be completed, and all deliverables will be handed over to the stakeholders. This includes product testing, documentation, user manuals, and any required training for end-users. A post-project review will be conducted to assess the project's success, and lessons learned will be documented for future reference.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 OVERVIEW OF IMPLEMENTATION

5.1.1 Sensors

Motion Sensors detect the movement and position of the baby in the cradle and Temperature and Humidity Sensors can monitor the environmental conditions to ensure the baby's comfort. Additionally, the Weight Sensors measure the baby's weight for safety and monitoring purposes. Sound Sensors detect the baby's cries or sounds for alerts and monitoring.

5.1.2 Actuators

Motorized Rocking Mechanism provides gentle rocking motion to soothe the baby. Temperature Control System adjusts heating or cooling elements to maintain a comfortable temperature. LED Lights or Night Light Illuminates the cradle area softly for night-time visibility. Audio System Plays lullabies or white noise to help the baby sleep.

5.1.3 Control System

Microcontroller or Single Board Computer (SBC) Controls the operation of sensors and actuators based on input from sensors and user commands. User Interface Includes buttons, touch screens, or mobile apps for caregivers to interact with the cradle, adjust settings, and receive alerts. Safety Features Implement safety mechanisms such as automatic shut-off in case of malfunction or danger. Networking Connectivity Allows remote monitoring and control via Wi-Fi or Bluetooth, enabling caregivers to check on the baby's status from their smartphones.

5.1.4 Power Supply

Battery Backup ensures uninterrupted operation even during power outages. Low-Power Design optimizes power consumption to prolong battery life.

5.1.5 Materials and Construction

Sturdy Construction: Use durable and non-toxic materials for the cradle structure. **Breathable Fabric** ensure good airflow to prevent overheating and suffocation risks. **Washable Components** design components such as mattress covers to be easily removable and washable for hygiene.

5.1.6 Data Logging and Analysis

Data Storage record sensor data for monitoring the baby's sleep patterns, temperature preferences, etc. and **Analytics** analyze the data to provide insights to caregivers and possibly detect patterns or abnormalities.

5.1.7 Safety Regulations and Standards

Compliance: Ensure that the smart cradle meets safety standards and regulations for infant products in your region.

5.1.8 Testing and Quality Assurance

Prototyping build prototypes for testing and iteration before finalizing the design. **Quality Control** conduct thorough testing for reliability, safety, and performance.

5.1.9 User Experience (UX) Design

Intuitive Interface Design the user interface to be user-friendly, especially for sleep-deprived caregivers. **Feedback Mechanisms** provide clear feedback through indicators or notifications to reassure caregivers.

5.1.10 Optional Features

Camera System Enables video monitoring of the baby. **Breathing Monitoring** utilizes sensors to monitor the baby's breathing patterns. **Integration with Smart Home Systems** allows integration with other smart home devices and automation systems.

5.2 TESTING

Testing a smart baby cradle involves several steps to ensure its safety, functionality, and reliability. Here's a general outline of the testing process:

5.2.1 Initial Inspection

Perform a visual inspection of the cradle to check for any manufacturing defects, loose parts, or physical damage.

5.2.2 Safety Testing

Ensure that the cradle meets safety standards for baby products. This includes testing for sharp edges, secure fastenings, non-toxic materials, and appropriate weight limits.

5.2.3 Electrical Safety

If the cradle has electronic components or is powered, conduct electrical safety tests to verify insulation, grounding, and protection against electrical hazards such as short circuits or overheating.

5.2.4 Functional Testing

Test all the features of the smart cradle to ensure they work as intended. This may include:

- **Motion:** Test the rocking or swinging motion of the cradle to ensure it's smooth and gentle.
- **Sound:** If the cradle includes sound features such as music or white noise, test the audio quality and volume levels.
- **Lights:** Test any lighting features for brightness and functionality.
- **Connectivity:** If the cradle connects to a mobile app or other devices, test the connectivity and communication to ensure it functions correctly.

5.2.5 Durability Testing

Perform stress tests to simulate extended use and ensure the cradle can withstand normal wear and tear without failure. This might include repeated rocking cycles or weight testing.

5.2.6 Temperature and Humidity Testing

Evaluate how the cradle performs under different environmental conditions, ensuring it remains safe and functional within specified temperature and humidity ranges.

5.2.7 Regulatory Compliance

Ensure the cradle complies with relevant regulations and standards for baby products in the target market. This may include safety standards, electromagnetic compatibility (EMC), and product labelling requirements.

5.2.8 Quality Assurance

Conduct quality assurance checks throughout the testing process to verify that the cradle meets the manufacturer's quality standards and specifications.

5.2.9 Documentation

Document the testing process, including test results, observations, and any issues encountered. This documentation is essential for quality control, troubleshooting, and regulatory compliance purposes.

CHAPTER 6

RESULTS AND DISCUSSION

6.1 ARDUINO IDE

Programming the Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials. The ATmega328 on the Arduino Uno comes pre burned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files). You can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU boot loader). See this user-contributed tutorial for more information. Automatic (Software) Reset Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer.

One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 Nano farad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the boot loader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half second or so, the boot loader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

6.2 EMBEDDED C:

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software. Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile phone, washing machine, digital camera, etc. These all device working is based on microcontroller that are programmed by embedded C.

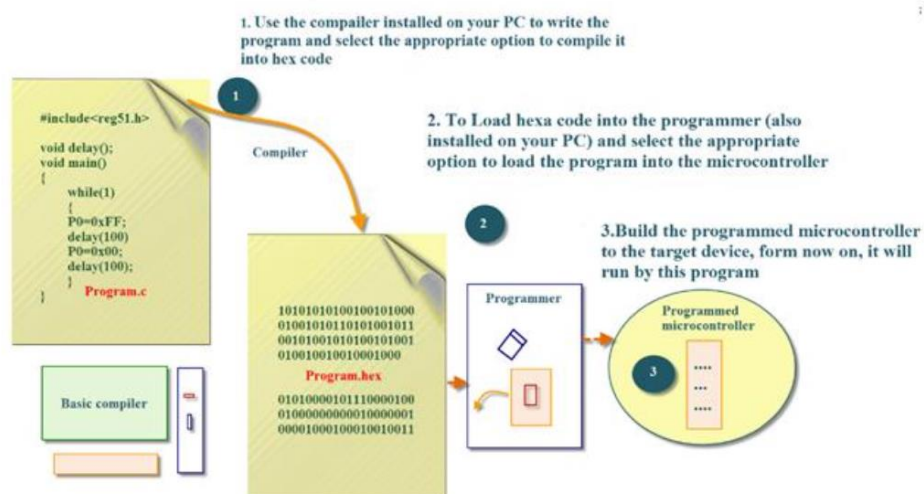


Fig. 5.1: Embedded C code

Embedded software needs to include all needed device drivers at manufacturing time, and the device drivers are written for the specific hardware. The software is highly dependent on the CPU and specific chips chosen. Most embedded software engineers have at least a passing knowledge of reading schematics, and reading data sheets for components to determine usage of registers and communication system. Conversion between decimal, hexadecimal and binary is useful as well as using bit manipulation. Web applications are rarely used, although XML files and other output may be passed to a computer for display. File systems with folders are typically absent as are SQL databases. Software development requires use of a cross compiler, which runs on a computer but produces executable code for the target device. Debugging requires use of an in-circuit emulator, JTAG or SWD. Software developers often have access to the complete kernel (OS) source code. Size of the storage memory and RAM can vary significantly. Some systems run in 16 KB of Flash and 4 KB of RAM with a CPU operating at 8 MHz, other systems can rival contemporary computers. These space requirements lead to more work being done in C or embedded C++, instead of C++. Interpreted languages like BASIC (while e.g. Parallax Propeller can use compiled BASIC) and Java (Java ME Embedded 8.3 is available for e.g. ARM Cortex-M4, Cortex- M7 microcontrollers and older ARM11 used in Raspberry Pi and Intel Galileo Gen. 2 are not commonly used; while an implementation of the interpreted Python 3 language – Micro Python – is however available expressly for microcontroller use, e.g. 32-bit ARM-based (such as BBC micro: bit) and 16-bit PIC microcontrollers. Communications between processors and between one processor and other components are essential. Besides direct memory addressing, common protocols include I²C, SPI, serial ports, and USB. Communications protocols designed for use in embedded systems are available as closed source from companies including InterNiche Technologies and CMX Systems. Open-source protocols stem from uIP, lwip, and others are used

CHAPTER 7

CONCLUSION

In an era where the demands of modern life often leave caregivers grappling with the delicate balance between professional obligations and nurturing childcare, the smart baby cradle System emerges as a beacon of innovation and compassion. With a profound understanding of the challenges faced by working mothers and nurses worldwide, this meticulously crafted solution offers not just practicality, but a genuine sense of support and relief.

The inherent portability of the smart baby cradle System further enhances its practicality, allowing caregivers the flexibility to effortlessly relocate the system between rooms as needed. Whether transitioning between workspaces or accommodating changing routines, this adaptable feature ensures continuity of care and comfort for both mother and child.

Beyond its functional excellence, the smart baby cradle System boasts a distinct advantage in its affordability, with a low initial cost and minimal operating expenses making it an accessible option for caregivers of all backgrounds. Its customizable options not only cater to individual preferences but also open the door to potential enhancements and increased efficiency, ensuring that the system evolves alongside the needs of its users. Because GSM technology is employed, the user may communicate over a greater distance. It is a convenient device for remotely monitoring the baby's health care.

REFERENCES

"Arduino," [Online] Available: "<https://www.arduino.cc>", Sep 2017.

Aquib Nawaz, "Development of an Intelligent Cradle for Home and Hospital Use," National Inst. of Technology, 2015.

B. Song, H. Choi, and H. S. Lee, "Surveillance tracking system using passive infrared motion sensors in wireless sensor network," in Information Networking, 2008. ICOIN 2008. International Conference on, 2008, pp. 1-5.

Cynthia L. Altenhofen, "BABY MONITOR SYTEM," Patent no .6,043,747, March 28, 2000.

Dr. Andrew Rawicz, Fanchao Yu Liu, Xiago Lu, Kiru Sri, "Functional Specification for Smart Baby Cradle," Simon Fraser University, 24th Jan 2016.

Dr. Andrew Rawicz, Fanchao Yu Liu, Xiago Lu, Kiru Sri, "Proposal for Smart Baby Cradle," Simon Fraser University, 24th Jan 2016.

Fatih Elmas, Abdurrahman Yilmaz, Muhammed Garip, "Rocking motion for the baby sleeping on Mothers Lap: Modelling and Prototyping Automatic Swing Cradle Design," Mechtronics Eng. Dept. Yildiz Tech. University.

G. Wong, "Automatic baby crib rocker," ed: Google Patents, 1976.

I. Khattab, A. Menon, and A. Sakr, "Effect of Mode of Incorporation of Disintegrants on the Characteristics of Fluid-bed Wet-granulated Tablets," Journal of pharmacy and pharmacology, vol. 45, pp. 687-691, 1993.

J.-H. Choi and V. Loftness, "Investigation of human body skin temperatures as a bio-signal to indicate overall thermal sensations," Building and Environment, vol. 58, pp. 258-269, 2012.

K. N. Ha, K. C. Lee, and S. Lee, "Development of PIR sensor based indoor location detection system for smart home," in SICE-ICASE, 2006. International Joint Conference, 2006, pp. 2162-2167.

M. Blea and M. Harper, "Automatically rocking baby cradle," ed: Google Patents, 1973.

M. Margolis, Make an Arduino-controlled robot: "O'Reilly Media, Inc.", 2012.

M. Moghavvemi and L. C. Seng, "Pyroelectric infrared sensor for intruder detection," in TENCON 2004. 2004 IEEE Region 10 Conference, 2004, pp. 656-659.

M. S. Zaghloul, "GSM-GPRS Arduino Shield (GS-001) with SIM 900 chip module in wireless data transmission system for data acquisition and control of power induction furnace," International Journal of Scientific & Engineering Research, vol. 5, 2014.

P. A. Marques, "ambient temperature and nest temperature variation in enclosed nests (spanish sparrow) and open-cup nests (iberian azure-winged magpie)," international studies on sparrows, p. 14.

P. Jamieson, "Arduino for teaching embedded systems. are computer scientists and engineering educators missing the boat?," Proc. FECS, pp. 289- 294, 2010.

Rajat Arora, Heli Shah, Rohan Arora, "Smart Cradle Gear to Ensure Safety of baby in Cradle," Int. J. of Informative and Futuristic Research, Mar 2017.

Ronen Luzon, "INFANT MONITORING SYSTEM," Patent no US 2002/0057202A1, May 16, 2002.

Savita P. Patil, Manisha R. Mehtre, "Intelligent Baby Monitoring System," VIT Pune.

Y. George, "Baby cradle rocked by electricity," ed: Google Patents, 1949.

Y. Wong, "Automatic baby crib rocker," ed: Google Patents, 1976.

APPENDIX

A. SOURCE CODE

```
#include <Servo.h>
int servoPin = 3;
Servo myservo;
#include <SoftwareSerial.h>
SoftwareSerial mySerial(4, 5);
int wet_sensor = A0;
int Sound_sensor = 9;
int pos = 0;
int buzzer = 7;
int led = 6;
void setup() {
// put your setup code here, to run once:
Serial.begin(9600);
mySerial.begin(9600);
pinMode(wet_sensor, INPUT);
pinMode(Sound_sensor, INPUT);
pinMode(buzzer, OUTPUT);
pinMode(led, OUTPUT);
myservo.attach(servoPin);
}
void loop() {
// put your main code here, to run repeatedly:
int a = analogRead(wet_sensor);
Serial.print("WET RANGE = ");
Serial.println(a);
if (a < 500) {
tone(buzzer, 500, 2000);
mySerial.println("AT+CMGF=1");
delay(1000); // Delay of 1000 milli seconds or 1 second
mySerial.println("AT+CMGS=\"+919092098217\"\\r");
```

```

number47
delay(1000);
mySerial.println((char)26);// ASCII code of CTRL+Z
delay(1000);
}
int b = digitalRead(Sound_sensor);
Serial.print("SOUND = ");
Serial.println(b);
if (b ==0) {
for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
// in steps of 1 degree
myservo.write(pos); // tell servo to go to position in variable 'pos'
delay(30); // waits 15ms for the servo to reach the position
}
for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees
myservo.write(pos); // tell servo to go to position in variable 'pos'
delay(30); // waits 15ms for the servo to reach the position
}
}

```

```

#include "appGlobals.h"
void setup() {
logSetup();
// prep SD card storage
if (startStorage()) {
// Load saved user configuration
if (loadConfig()) {
// initialise camera
if (psramFound()) {
LOG_INF("PSRAM size: %s", fmtSize(esp_spiram_get_size()));
if (esp_spiram_get_size() > 3 * ONEMEG) prepCam();

```

```

else snprintf(startupFailure, SF_LEN, STARTUP_FAIL "Insufficient PSRAM for
app");
}
else snprintf(startupFailure, SF_LEN, STARTUP_FAIL "Need PSRAM to be
enabled");
}
}
#ifdef DEV_ONLY
devSetup();
#endif
// connect wifi or start config AP if router details not available
startWifi();
startWebServer();
if (strlen(startupFailure)) LOG_ERR("%s", startupFailure);
else {
// start rest of services
startSustainTasks();
#ifdef INCLUDE_SMTP
prepSMTP();
#endif
#ifdef INCLUDE_FTP_HFS
prepUpload();
#endif
prepPeripherals();
#ifdef INCLUDE_MIC
prepMic();
#endif
#ifdef INCLUDE_TELEM
prepTelemetry();
#endif
#ifdef INCLUDE_TGRAM

```

```

prepTelegram();
#endif
prepRecording();
checkMemory();
}
}
void loop() {
// confirm not blocked in setup
LOG_INF("===== Total tasks: %u =====\n",
uxTaskGetNumberOfTasks() - 1);
delay(1000);
vTaskDelete(NULL); // free 8k ram
}

```

B. SCREENSHOT

