Modular Self-Reconfigurable Robot for Implementation of Biomimetic Structures

THESIS

Submitted in partial fulfillment

of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

S SANKHAR REDDY CH.

Under the supervision of

PROF. ANUPAMA K. R

Under the co-supervision of

DR. ANITA AGRAWAL

BIRLA INSTITUTE OF SCIENCE & TECHNOLOGY, PILANI

2018

BIRLA INSTITUTE OF SCIENCE & TECHNOLOGY, PILANI

CERTIFICATE

This is to certify that the thesis entitled **"Modular Self-Reconfigurable Robot for Implementation of Biomimetic Structures"** submitted by **S SANKHAR REDDY CH.** ID No **2012PHXF0001G** for award of Ph.D Degree of the Institute embodies original work done by him under my supervision.

Signature of the Supervisor K.R Anupama

Name in capital block letters : **PROF. ANUPAMA K. R**
Designation : **PROFESSOR**
Department of Electrical and Electronics Engineering

Date : 30/4/2018

Signature of the Co-Supervisor

Name in capital block letters : **DR. ANITA AGRAWAL**
Designation : **ASSISTANT PROFESSOR**
Department of Electrical and Electronics Engineering

Date : 30/4/18

ABSTRACT

Application specific designs, lack of reconfiguration capabilities, and minimal scope for self-healing are major drawbacks of traditional robots. The research in the field of modular robotics provides a solution to real-world applications while addressing the demerits of conventional robots. Modular robotics employs homogeneous robotic units with minimal degrees of freedom equipped with capabilities such as self-assembly, re-configuration, and locomotion. The modular robotic designs are an optimal integration of robot mechanics, embedded systems, communication and custom-made sensor-actuator mechanisms. An efficient design of a modular robot aids in forming numerous structures, increasing lifetime of the robot and the lifetime of coordinated structure formed using such modular robots. Apart from the robotic design, optimizations in modular robotics can be extended to design of mechanisms for docking and reconfiguration, locomotion strategies, power consumption, degrees of freedom and communication.

The primary objective of the domain of modular robotics is to utilize numerous homogeneous modular robotic units for implementation of sophisticated robotic applications. Modular robotic units are individually incapable of overcoming the tasks like hole crossing, navigation in uneven terrains etc. due to their small form-factor and limited capabilities, but a co-ordinated robotic system formed using such modular robotic units overcoming the obstacles is a possibility and such designs are thoroughly researched in the domain of modular robots. The major challenges in the domain of modular robotics can be observed in the development of robotic designs that are capable of providing different degrees of freedom to a co-ordinated robotic system along with possibilities of forming numerous co-ordinated structures. Such challenges are addressed by embedding an intelligent combination and placement of sensors and actuators for providing few degrees of freedom on individual modular robot and providing numerous docking faces/interfaces for autonomous/manual assembly. Another major challenge observed in long-term operational capabilities of modular robotics is energy conservation. Since modular robots are small in form-factor and are not connected to external energy sources, optimizations are required to minimize the energy consumed for sensing, processing and communication along with the robotic design to increase the life-time of individual robotic units and co-ordinated robotic systems.

A modular robotic design named HexaMob capable of forming chain and biomimetic structures was designed for supporting applications requiring autonomous distributed sensing in uneven terrains and hazardous locations. The HexaMob design is envisaged to employ the image processing capabilities and odometric sensor data for achieving autonomous nature in docking, reconfiguration and locomotion. Numerous docking interfaces present on the faces of HexaMob facilitate formation of different structures and zero-energy latching mechanism employed for docking increases the operational lifetime of the coordinated structures. An implicit back-drive preventing worm gear based mechanism employed at two actuators is a novel and unique feature inculcated into robot for implementing locomotion. The worm gear mechanism provides increased torques that will aid in applications with heavy loads and also a precise control over angular velocity and displacement. The four degrees of freedom (one pitch, one yaw and 2 degrees from mobility) on each HexaMob robotic module provides autonomous nature, and coupled with advanced sensor interfaces such as image sensors, the overall capabilities of the HexaMob robotic module are further enhanced.

An embedded platform referred as FlexEye was prototyped with an objective of integrating an efficient electronic control unit into the HexaMob robotic module. FlexEye is capable of fusing odometry and image processing data for providing autonomous nature to HexaMob along with wireless communication capabilities. The FlexEye prototype was compared with numerous wireless platforms developed in the domain of wireless sensor networks for better interpretation on energy conservation capabilities. FlexEye platform can support large-scale applications such as distributed sensing and also provides necessary scope for future expansion. An analysis on power consumption, the details of image acquisition and processing using FlexEye are analyzed as a part of research.

A wireless communication platform referred as Quanta was developed to support communication and monitoring of activities inside each modular robot along with events in the external environment. The Quanta platform with its hardware and communication mechanisms is capable of scheduling concurrent events with latencies in milliseconds. The strategic packet structure and light-weight middleware in Quanta aids in rapid scheduling of events in modular robotics such as monitoring, and also events such as locomotion in a coordinated robotic system. The details on software such as state machines for enabling remote monitoring and control are described.

The simulations performed for identification of an optimal transceiver for communication on parameters such as power consumption and packet delivery rate of radio modules are presented.

Declaration

I, S SANKHAR REDDY CH., hereby declare that this thesis entitled "Modular Self-Reconfigurable Robot for Implementation of Biomimetic Structures" submitted by me under the guidance and supervision of Prof. Anupama K. R and co-supervision of Dr. Anita Agrawal is a bonafide research work. I also declare that it has not been submitted previously in part or in full to this University or any other University or Institution for award of any degree.

Ch. Sasanka

Signature of the student

Name of the student : S SANKHAR REDDY CH.

Reg. No : 2012PHXF0001G

Date : 30/4/2018

Acknowledgments

I would like to express my sincere gratitude to my supervisor and mentor Prof. Anupama K. R. for the continuous support in conducting this research, for her patience and motivation. She had whole heartedly helped me in this endeavor at all stages of this work. Her guidance helped me in all the time of research and writing of this thesis. From bottom of my heart, I extend my sincere thanks to her for all her efforts and help at various levels without which this work not have been completed. My sincere gratitude extends to my co-supervisor Dr. Anita Agrawal for her patience and timely support in various aspects of the research and in numerous challenges faced during this work. Her guidance directed me into right approach towards research.

I am very grateful to members of my Doctoral Advisory Committee for thesis mentor ship and guidance, Dr. Ramesha C. K. and Dr. A. Amalin prince for sparing their valuable time in reviewing my thesis and giving constructive suggestions to improve it. Their valuable suggestions have helped in greatly enhancing the quality of the thesis.

I would extend my sincere thanks to Prof. M.K. Deshmukh, Professor and Head, Dept.of E.E.E for his constant support and encouragement at various levels.

I thank Prof. Sauvik Bhattacharya, Vice-Chancellor, BITS Pilani, Prof. G. Raghurama, Director, BITS Pilani-K. K. Birla Goa Campus, Prof. Sasikumar Punnekkat, former Director, BITS Pilani - K. K. Birla Goa Campus, Prof. K. E. Raman, former Director, BITS Pilani - K. K. Birla Goa Campus, Prof. S. K. Verma, Dean, ARD, BITS Pilani - Pilani Campus, Prof. Prasanta Kumar Das, Associate Dean, ARD, Prof. Sunil Bhand, Dean, SRCD, Prof. D. M. Kulkarni, Dean, Administration, BITS, Pilani - K. K. Birla Goa Campus, and Prof. S. D. Manjare for giving me an opportunity to carry out research studies at the institute and also by providing necessary infrastructure and facilities to carry out my work.

I would extend my sincere thanks to Prof. A. P. Koley, associate dean, Instruction division, BITS Pilani K K Birla goa campus for his moral support at various levels. I also extend my sincere thanks to Dr. Gautam Bacher, DRC Convener, for support and motivation. I also thank Dr. Debasis Patnaik, Dr. Pravin Mane, Dr. Nitin Sharma, Dr. Narayan Manjarekar, Dr. Sheron Figardo, Mr. Meghanand Bhamare, Dr. Sarang Dhongdi, Ms. Meetha Shenoy, Dr. Shibu Clement, Mr. Vijay Suryawanshi, and Dr. Gaurav Singh for their constant support, motivation and suggestions.

I acknowledge with due gratitude to Mr. V. V. Aditya and Mr. Patlolla Sharath for their constant support. I would extend my sincere thanks to Mr. Vijaykumar V. Patil, Mr. Anil Kumar and Mr. Anoop, lab technicians for their constant help during my research work carried out in the lab.

I would like to express my sincere gratitude to my loving parents Mr. Bhaskar reddy and Mrs. Nagamalleswari, for always believing in me, for their continuous support, encouragement, motivation and prayers.

I thank my wife Mrs. Divya for taking responsibilities of family on my behalf and giving encouragement in carrying out work.

I am thankful to all my friends and relatives who directly or indirectly helped me in completing my thesis.

Above all, my gratitude towards the almighty that provided me with blessings and mercy in all the possible forms.

S SANKHAR REDDY CH.

Contents

List of Tables

List of Figures

Abbreviations

ACM	**A**ctive **C**ord **M**echanism
ARQ	**A**utomatic **R**epeat **R**equest
CAN	**C**ontrol **A**rea **N**etwork
CEBOT	**Ce**ll-structured Ro**bot**
CIF	**C**ommon **I**ntermediate **F**ormat
CKbot	**C**onnector **K**inetic ro**bot**
COTS	**C**ommerical **O**ff-**T**he-**S**helf
CMOS	**C**omplementary **M**etal **O**xide **S**emiconductor
CPLD	**C**omplex**P**rogrammable **L**ogic **D**evice
DCMI	**D**igital **C**a**m**era **M**edia **I**nterface
DMA	**D**irect **M**emory **A**ccess
DMIPS	**D**hrystone **M**illion **I**nstructions **P**er **S**econd
DOF	**D**egrees **O**f **F**reedom
ESB	**E**nhanced **S**hock **B**urst
FIFO	**F**irst **I**n **F**irst **O**ut
FPGA	**F**ield **P**rogrammable **G**ate **A**rray
FPU	**F**loating **P**oint **U**nit
FSK	**F**requency **S**hift **K**eying
GFSK	**G**aussian **F**requency **S**hift **K**eying
GOPS	**G**iga **O**perations **P**er **S**econd
GPIO	**G**eneral **P**urpose **I**nput **O**utput
I^2C	**I**nter **I**ntegrated **C**ircuit
ICC	**I**nstantaneous **C**enter of **C**urvature

IMU	**I**nertial **M**easurement **U**nit
ISM	**I**ndustrial **S**cientific **M**edical radio
JPEG	**J**oint **P**hotographic **E**xpert **G**roup
LED	**L**ight **E**mitting **D**iode
MEMS	**M**icro **E**lectro-**M**echanical **S**ystems
ModRED	**Mod**ular **R**obot for **E**xploration and **D**iscovery
MSK	**M**inimum-**S**hift **K**eying
MSRR	**M**odular **S**elf **R**e-configurable **R**obot
MWSNs	**M**obile **W**ireless **S**ensor **N**etworks
NMI	**N**on **M**askable **I**nterrupts
OOK	**O**n/**O**ff **K**ey
OS	**O**perating **S**ystem
PetRo	**Pet Ro**bot
PLL	**P**hase **L**ocked **L**oop
QVGA	**Q**uarter **V**ideo **G**raphics **A**rray
RGB	**R**ed **G**reen **B**lue
RISC	**R**educed **I**nstruction-**S**et **C**omputing
ROM	**R**ead **O**nly **M**emory
RTC	**R**eal **T**ime **C**lock
RTOS	**R**eal **T**ime **O**perating **S**ystems
SCCB	**S**erial **C**amera **C**ontrol **B**us
SDIO	**S**ECURE **D**igital **I**nput **O**utput
SDRAM	**S**ynchronous **D**ynamic **R**andom **A**ccess **M**emory
SIMD	**S**ingle **I**nstruction **M**ultiple **D**ata
SPI	**S**erial **P**eripheral **I**nterface
SRAM	**S**tatic **R**andom **A**ccess **M**emory
SXGA	**S**uper **E**xtended **G**raphics **A**rray
TCP/IP	**T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol
UART	**U**niveral **A**synchronous **R**eceiver/**T**ransmitter
USB OTG	**U**niversal **S**erial **B**us **O**n-**T**he-**G**o
VGA	**V**ideo **G**raphics **A**rray

VSMs	**V**isual **S**ensor **M**otes
WSNs	**W**ireless **S**ensor **N**etworks
YaMoR	**Y**et **a**nother **Mo**dular **R**obot

Modular robot

Biomimetics

Chain structures

Back-driving restriction

Self-reconfigurability

Autonomous docking

Image processing

Energy optimization

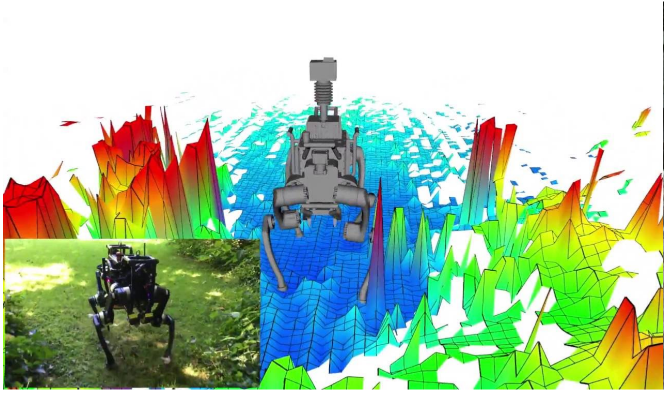
Distributed sensing

Chapter 1

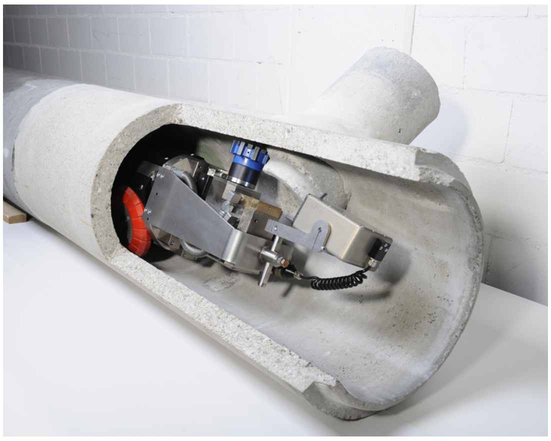
Introduction

Robotics over the decades has found a place in numerous fields such as manufacturing and automation, medicine, and space exploration. The technological improvements in processors, advanced sensor and actuation mechanisms in smaller form factors and better analysis tools enhanced the rate of adaptation of robotics further. Many large-scale applications such as terrain mapping, disaster management, and pipeline monitoring are conventionally executed manually and the efficiency of such operations is often realized to be very less due to human participation. Robots are being employed in these large scale applications as shown in figure 1.1 and it can be inferred from the figure that the presence of robotic units aids in rapid completion of the tasks considering the scale of these applications in spatial perspective. The applications often pose challenges in terms of cost and time due to inaccessibility to the regions, a risk to human life and hard pressing deadlines. Employment of biomimetic robotic structures for above mentioned large-scale applications is already under consideration and the research in robotics is further being directed into inculcating characteristics such as self-reconfiguring and self-healing as primary traits into robotic structures.

The major drawbacks of the conventional robotics such as application specific designs, complex debugging issues, high maintenance costs, limited scope for generalization etc. can be overcome by utilization of modular reconfigurable robotic units aiding the distributed techniques for sensing, actuation and adaptive mechanisms in robotic platforms. The advantages of such strategies can be visualized from the perspectives of ease of replacement of robotic parts, reduction in



a) b)



c)

Figure 1.1. Biomimetic Robots - Real world applications - a) Terrain mapping [1] b) Disaster management [2] c) Pipeline monitoring [3]

production complexity of individual robotic units, wide range of applications, reusability of numerous software modules and transfer of processing load to centralized servers etc. Research in modular robotics domain integrates these advantages and can be a promising solution to demerits of conventional robots.

The notion of distributed sensing and control using homogeneous and simplistic units of hardware equipped with similar software is a well-researched concept in the domains of wireless sensor networks (**WSNs**), mobile wireless sensor networks **(MWSNs)** and swarm robotics. Though the applications of these domains vary significantly, few common objectives such as energy optimizations in hardware, form-factor reduction, reconfiguration techniques, and software optimization can be observed to be common among them. The techniques and strategies employed in the WSN, MWSNs and swarm robotic domains can be implemented in the research of modular robotics due to the similarities present in their domain characteristics.

The domain of WSNs primary aims at distributed sensing of parameters in a environment while maintaining the pervasiveness of embedded devices. The non-intrusive requirements of WSNs impose several constraints on the hardware such as lack of immediate access to hardware for maintenance, unavailability of external power sources, and lack of connectivity due to hazardous and unpredictable nature of environment. Such restrictions have inspired researchers to investigate the possibilities of optimizations in hardware and software in order to increase the lifetime of WSNs. Figure 1.2 provides an overview of the hardware characteristics and software features of modules implemented for WSNs. Since mobility is rarely an integral part of the nodes in WSNs, the nodes employ virtual clustering and virtual self-reconfiguration of a network for achieving better connectivity and throughputs. The primary merit of WSNs hardware is observed in simplistic units achieving the automation of large scale tasks while preserving the longevity of entire network for years.

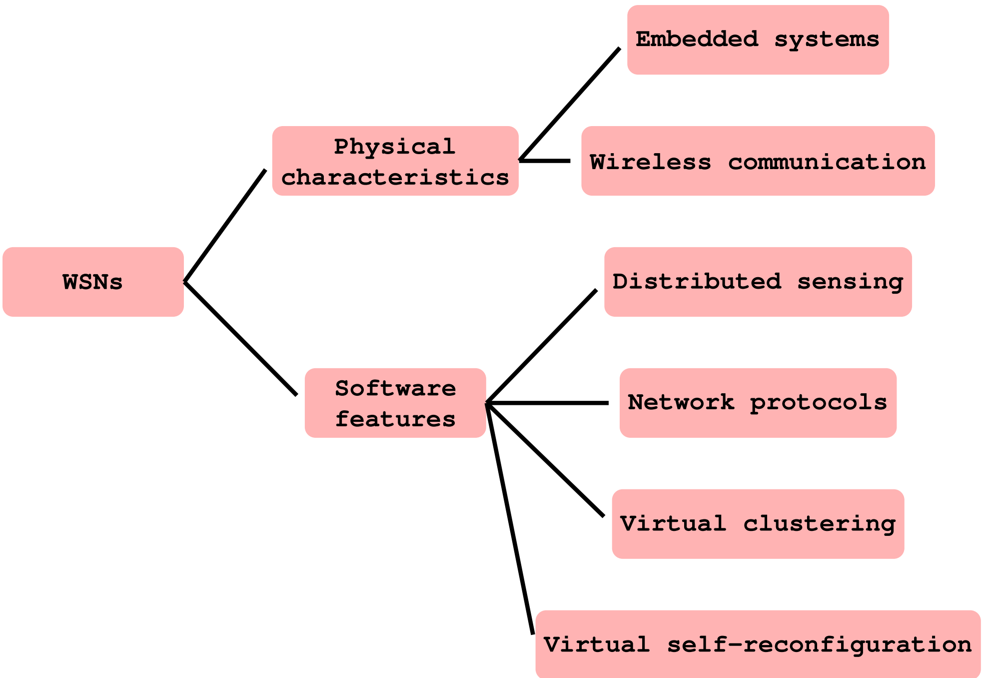


Figure 1.2. Wireless sensor networks - Physical characteristics and software features

MWSNs integrate mobility into WSNs by the addition of mobile nodes and units. Though mobile units are an integral part of the network, wheeled robots equipped with WSN motes are deployed into the network only to identify the merits and demerits of protocols in the presence of mobility. The priority towards the design of wheeled robots while optimizing power consumption is very minimal. Another major feature of MWSNs is that the network is capable of physical reconfiguration due to presence of mobile nodes and hence making self-healing further better. The MWSNs nodes are equipped with various algorithms for supporting localization, navigation, network reconfiguration for self-healing via node mobility etc. Figure 1.3 provides an overview of the physical characteristics and software features of modules implemented for MWSNs.

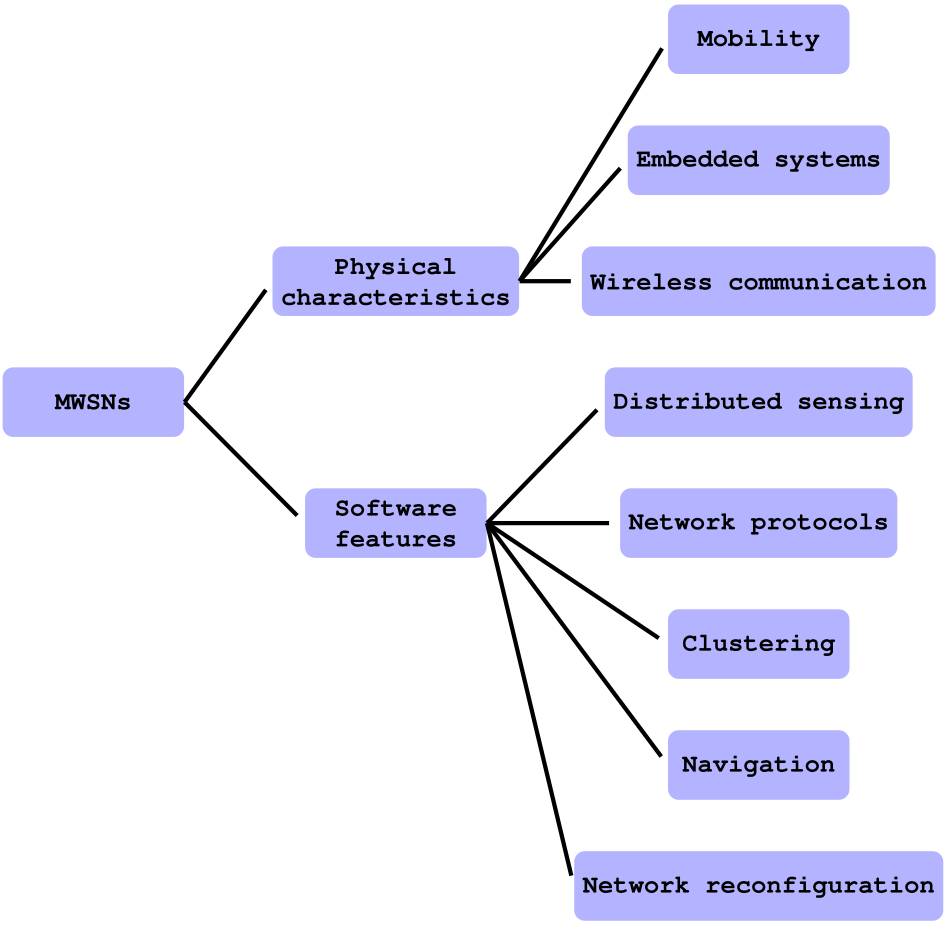


Figure 1.3. Mobile wireless sensor networks - Physical characteristics and software features

Swarm robotics domain deals primarily with the coordination dynamics of multi-robot testbeds. Some of the major research interests are in the development of algorithms for multi-robot tasks such as aggregation and dispersion, task scheduling, centralized and decentralized processing etc. Though wireless communication plays a vital role in implementing the test scenarios, the communication mechanisms often deviate from the utilization of standard wireless interfaces used in WSNs and MWSNs by employment of light (Infrared, Laser, color LEDs), acoustics etc. The robots designed for swarm robotics test scenarios are often very lightly capable embedded platforms and limited to exhibiting simple movements such as sliding using vibration motors upon processing of co-ordination algorithms. Figure 1.4 provides an overview on physical characteristics and software features of the robotic units in the domain of swarm robotics.

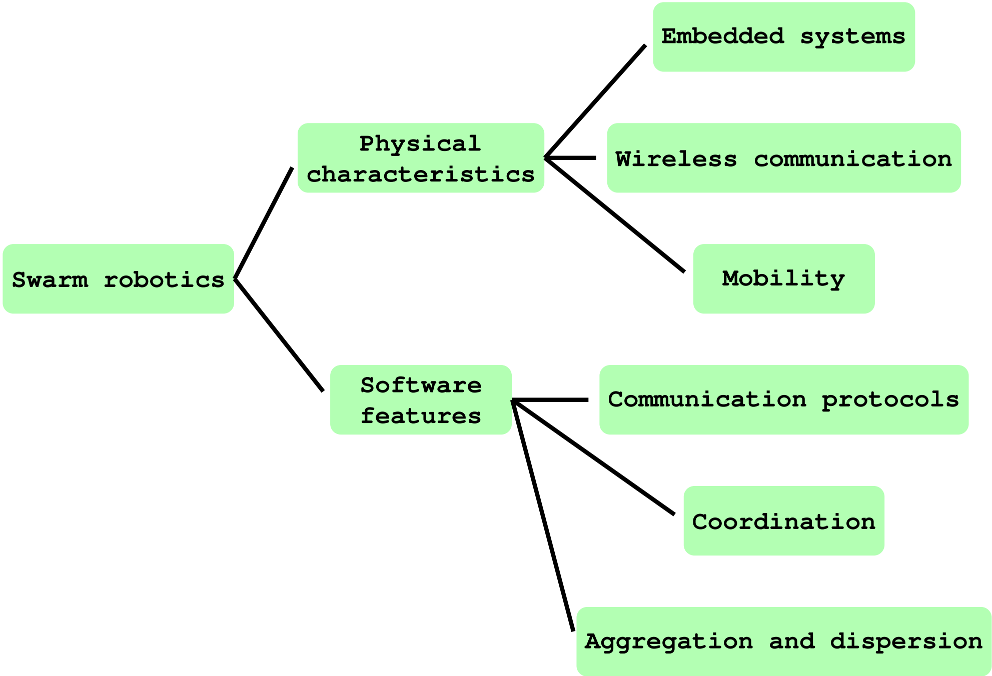


Figure 1.4. Swarm robotics - Physical characteristics and software features

Though the majority of the research in these domains choose to prioritize between few objectives such as communication optimization or mobility optimization etc., the real-world applications implemented using distributed homogeneous systems often have to meet optimization demands at every level of hardware and software for the efficient functioning of a prototype in unpredictable ambient conditions. The parameters such as power consumption, processing capabilities, data storage, form factor, mobility, and self-reconfiguration are vital for the autonomous operation of the systems and hence the corresponding directions are to be explored even in the case of modular robotics.

Modular robotics is one of the domains in the area of robotics which explicitly deals with the homogeneous robotic systems capable of self-reconfiguration designed for providing locomotion features similar to humanoid and biomimetic robots. The research in modular robotics is still in nascent stages, rigorous analysis and developments in infrastructure are still to be done for real-world scenarios. Many planes necessary for an efficient operation of modular robots such as communication protocols, aggregation and dispersion, navigation, optimization of power consumption etc. are not dealt in this domain. A major emphasis in modular robotics research domain so far has been observed in design and development of individual robotic modules capable of forming numerous structures. The mechanisms for docking and undocking of modules supporting reconfiguration while providing locomotion are also prototyped rigorously as a part of modular robotics research. Figure 1.5 provides an overview on physical characteristics and software features of the robotic units in the domain of modular robotics.

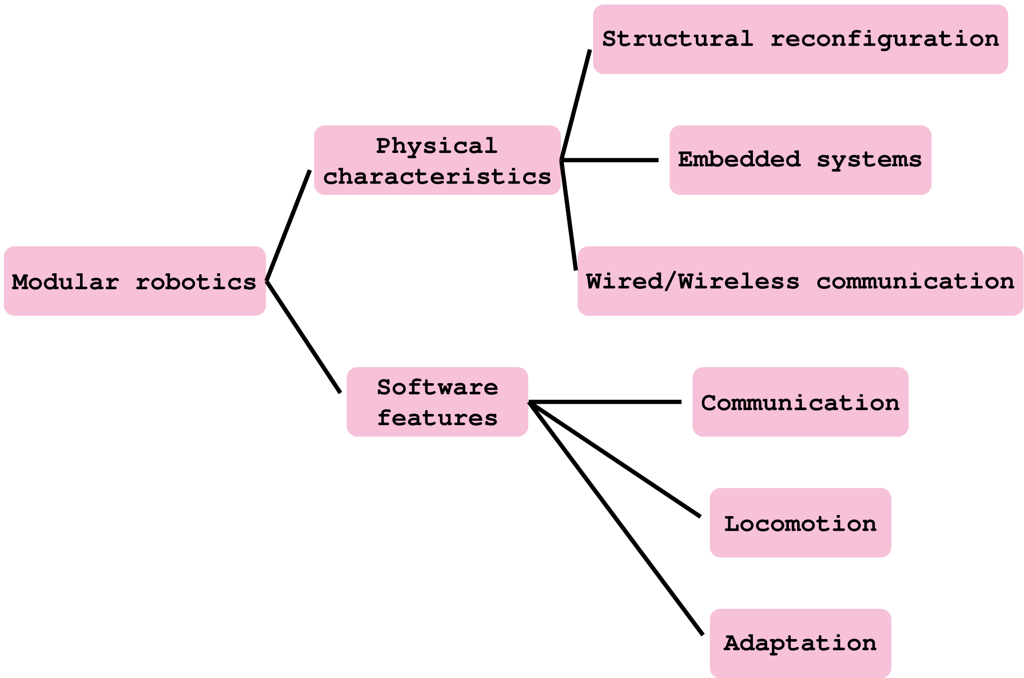


Figure 1.5. Modular robotics - Physical characteristics and software features

The major advantage of modular robotics is in its capabilities for extending the limited scope of distributed sensing and actuation by providing autonomous nature to robots. The strategies advocated in WSNs, MWSNs and Swarms can be re-utilized in modular robotics domain for enhancing autonomous nature of robots in real-world applications. The affordable and efficient physical platforms like modular robotic units can be utilized for implementation of algorithms and protocols developed in WSNs, MWSNs and swarms due to the similarity in physical characteristics as shown in figure 1.2, 1.3, 1.4 and 1.5. Figure 1.6 provides a broad overview of real-world applications that can be implemented using the research from domains discussed so far. The applications mentioned in the figure 1.6 were already researched in individual domains with limited features and restricted hardware capabilities.

The Swarm-Bot robotic project[4] attempted to integrate research work across multiple domains by the employment of adaptable hardware structures and algorithms and succeeded partially in

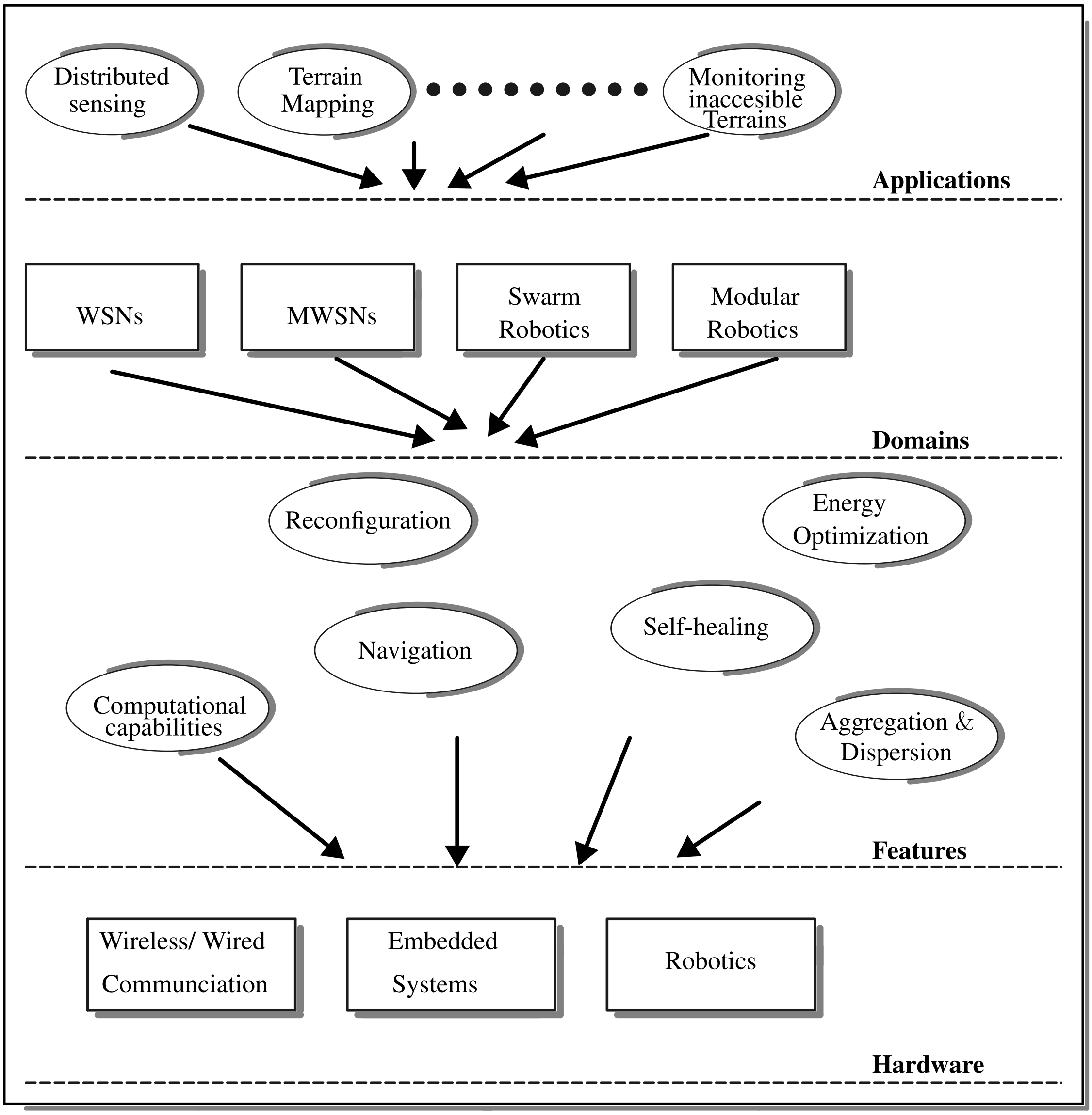


Figure 1.6. Distributed applications - Layered approach in implementation

the same as shown in figure 1.7. The notion of utilization of biomimetic structures capable of crawling, walking etc. can overcome challenges posed by unpredictable environment by adaption of physical structures dynamically. Further integration of numerous algorithms developed in the WSNs, MWSNs and swarms can prove to be an advantage for complete automation of the application.

Figure 1.7. Swarm-bot modular chain structure[4]

1.1 Gaps in research

Though the field of modular robotics was rigorously researched for the past five decades, the research can still be perceived to be in nascent stages due to its limited developments. The major priority of the researchers in modular robotics domain so far is placed into the development of new modular robotic units and resources were invested into the design of hardware structures, sensor-actuator mechanisms, docking strategies and locomotions. In spite of such efforts, the real-world applications utilizing modular robots almost negligible. The major constraints observed in numerous modular robotic designs are

- Inability of robotic modules to form multiple structures.
- Lack of optimizations in robotic modules for longer operational times.
- Semi-autonomous nature of robotic designs.

Embedded system is one of the least researched components in modular robotics. Though 8-bit/16-bit microcontroller based embedded systems provided satisfactory solutions for docking the robots for forming complex structures and reconfiguration, such systems failed to provide complete autonomous capabilities to the individual robots. Autonomous capabilities can be incorporated into modular robots for zero human intervention by the employment of sensor fusion techniques utilizing numerous transducers like vision sensors, inertial measurement units, magnetometers, rotary encoders, etc. It is also necessary to incorporate necessary power optimizations in the hardware so that the longevity of a modular robot as well as the coordinated robotic structure can be increased.

Emphasis on research for communication strategies especially employing wireless communication techniques for rapid and efficient locomotion in centralized/de-centralized mechanisms is also minimal in the domain of modular robotics. An efficient and light-weight communication strategies that are independent of embedded hardware and operated by employing the on-board features to the optimal level will be of significant aid in efficient implementation of locomotion in robotic structures. Incorporation of sleep and wake strategies in transceivers can improve the optimizations in power consumption further.

1.2 Motivation

Many applications such as disaster management in earthquakes, floods and hurricanes, underground exploration, and autonomous terrestrial exploration pose huge challenges to human life due to the risks present from the external environment. Due to the restricted availability of trained personnel and less time for disaster management, robots can be provided as supplementary tools to monitor pipelines, tunnels, caves etc. Since an application specific design is an inefficient solution to such problems where the terrains and obstacles vary significantly, a self-reconfigurable biomimetic structure made out of simple homogeneous robotic units can be an effective solution. The domain of modular robotics rigorously explores the development of self-reconfigurable robotic units that can be employed in wide range of applications.

Major research in the field of modular robotics is limited to the hardware models tested in the laboratories for sophisticated functionalities. Few robotic designs developed for formation lattice structures and chain structures so far were developed for demonstration purposes and such designs seldom found their place in real-world applications. Enhancement of inherent capabilities of individual robotic modules for enabling automation along with powerless and energy efficient latching and docking mechanisms can improve the performance of coordinated robotic structures in real-world scenarios.

The motivation of research is to improvise various aspects of modular robotics such as degrees of freedom, power conservation, communication mechanisms, structural possibilities and autonomous capabilities so that a suitable modular robot capable of forming biomimetic structures and a supplement communication mechanisms can be provided for real-world applications.

1.3 Objectives

The major objectives of research work presented in this thesis are,

- To design a modular robot with minimal degrees of freedom so that it can be employed in formation of biomimetic structures with reconfiguration capabilities for facilitating adaptation in dynamic environments.
- To develop an efficient embedded platform for handling control and communication events in the modular robotic system. Large storage feature for distributed sensing, image processing capabilities and the control over power consumption are considered as primary requirements so that the platform can aid in increasing the operational times of the robot and the biomimetic structures.
- To develop a communication and concurrent control mechanisms for remote control and monitoring of robots. The mechanisms that are to be designed are meant to facilitate centralized and decentralized control enabling peer to peer communication in robots without additional overhead on network.
- To implement image processing algorithms for facilitating autonomous docking in modular robots as well as providing future scope for better navigation capabilities in the real-world applications.

1.4 Organization of thesis

Chapter 2 of the thesis describes various hardware models of modular robots developed so far as part of literature survey. In this chapter, more emphasis is given in describing the physical features of the modular robots such as external structures, docking mechanisms, various degrees of freedom, types of interfaces and flexibility of the robotic modules for reconfiguration. The tables summarizing comparison of various physical features of modular robots is provided at the end of the sections for better understanding. Emphasis on embedded electronic hardware and communication is minimal in chapter 2 due to fewer details provided by the designers regarding the same.

Chapter 3 summarizes the novel design of a modular robot - HexaMob designed for implementing chain and biomimetic structures. The autonomous capabilities of the HexaMob robotic module equipped with a mobility unit and four degrees of freedom are detailed. The important features of HexaMob robotic module such as capabilities in forming and maintaining biomimetic structures, optimizations for reducing energy consumption in robots are provided in the chapter. A detailed comparison provided at the end of the chapter in terms of numerous features provided by various modular robots developed so far summarizes the advantages of HexaMob robotic module over other modular robotic designs.

Chapter 4 of the thesis explains about a visual sensor mote - FlexEye prototyped to operate as a control unit for the HexaMob robot. The chapter lists out various camera platforms optimized in hardware and software for their application in WSNs and makes a detailed comparison in power consumption of various motes with FlexEye mote. A summary on numerous features for interfacing sensor and actuator mechanisms provided by the on-board chip employed in FlexEye mote along with rapid memory transfer capabilities for image acquisition, processing and storage are listed in the chapter. The details on the acquisition of high-resolution images in spite of having low on-chip memory capabilities without using external RAM for better processing are provided. Latency details on the execution of various image processing algorithms are also provided for better estimation of the computational capabilities of the FlexEye platform.

Chapter 5 provides details of a communication platform - Quanta, designed for supporting centralized and de-centralized communication strategies for implementation of locomotion and other initialization/operational activities in modular robots. The simulations performed for optimal choice of a transceiver using real-time latencies are presented. The state machines implemented for communication and concurrent control on FlexEye platform are explained along with packet structure. Major characteristics of the Quanta platform are summarized at the end of the chapter for providing an overview on the performance of the platform.

Chapter 6 concludes the thesis by summarizing the major outcomes from the research and also provides the details on future scope of the research explained in the thesis.

Chapter 2

Literature Survey

2.1 Introduction

Modular Robotics provides a unique advantage over traditional robotic technologies in terms of reconfigurability, re-usability, and ease in manufacturing. Traditional robots such as robotic arms, hexapods etc. provides solutions that are particular to each real-world application and the generated prototypes cannot be used in other applications. Most traditional robotic solutions are operated in a controlled environment and any changes in environments render the traditional solutions less useful due to the absence of flexibility. The repair and maintenance of such conventional designs generally require separate trained personnel for each model and hence increasing the expenditure of industries. The next phase of robotic designs is developed in the perspective of the assembly of modular units for increasing ease of repairing, replacing, control etc. The researchers in later phases of development introduced the concept of automation, self-healing, re-configuration etc. creating modular self re-configurable robots(**MSRR**). Many applications such as management of large facilities[5], space exploration[6], surveillance in military zones, disaster management, prosthetics for physically disabled etc. often require adaptable and self-healing abilities and MSRR is often considered as a viable solution for the same. The major difference of MSRR designs over modular robotic designs can be visualized as the abilities of designs to attach/detach in/from a formation as per the requirement of application with minimal human intervention.

The growing demand for reusable, space constrained and multi-purpose solutions for real-world applications is a great motivator for research in the field of MSRR. The researchers in the domain of MSRR provided numerous solutions to real-world applications via various prototype designs, communication algorithms, co-ordination and dispersion techniques using selected test scenarios. The development of novel prototypes for MSRR is an analytical process that often has deep roots in intuition and derives better fruits from the experience on basic locomotions and laws of physics. A detailed survey on wide range of modular designs of outer structures, physical interfaces between modules, communication protocols, sensor technologies for docking and alignment, co-ordinate movement algorithms, environment characteristics etc. provide better insight towards the development of novel solutions with efficient utilization of latest technologies. The scope of literature survey is limited to summarizing the hardware architectures along with sensor and interfacing technologies of various MSRR.

2.2 Modular Robots - Hardware Architectures

The hardware architectures of MSRRs are evolving along with technologies and so does the paradigm used for categorizing the robots. The first prototype developed in MSRR research is the cell-structured bot(CEBOT) consisting of heterogeneous separate units capable of binding together and since then the research was directed to the development of systems capable of forming different structures mimicking biological organisms. Yim et al. [6, 7] suggested two possible classifications of modular robotic systems - the classification based on structures formed by MSRRs and classification based on reconfiguration strategies. Gilpin et al. [8] added few more sub-classifications under structures category by including research from micro electro-mechanical systems(**MEMS**). Moubarak et al. [9] categorized MSRR systems based on the locomotion of the individual modules and coordinated structures along with form factors. The classifications proposed so far are done as per the state of art research in recent MSRR technologies, prototypes etc. available till the date of publication. Few MSRR characteristics are observed to be falling in the middle of earlier classifications and the identification of a category and subcategory for MSRRs is becoming increasing difficult due to the sophisticated designs and features of robots.

The classification of MSRR based on various categories and subcategories such as physical characteristics, abilities etc. is provided in figure 2.1. The widely accepted classification is in the perspective of possible structural formations when independent MSRR are brought together and five subcategories are recognized under structures as per the current MSRR research. They are - *Lattice*, *Chain*, *Hybrid*, *Truss*, and *Free-form* structures.

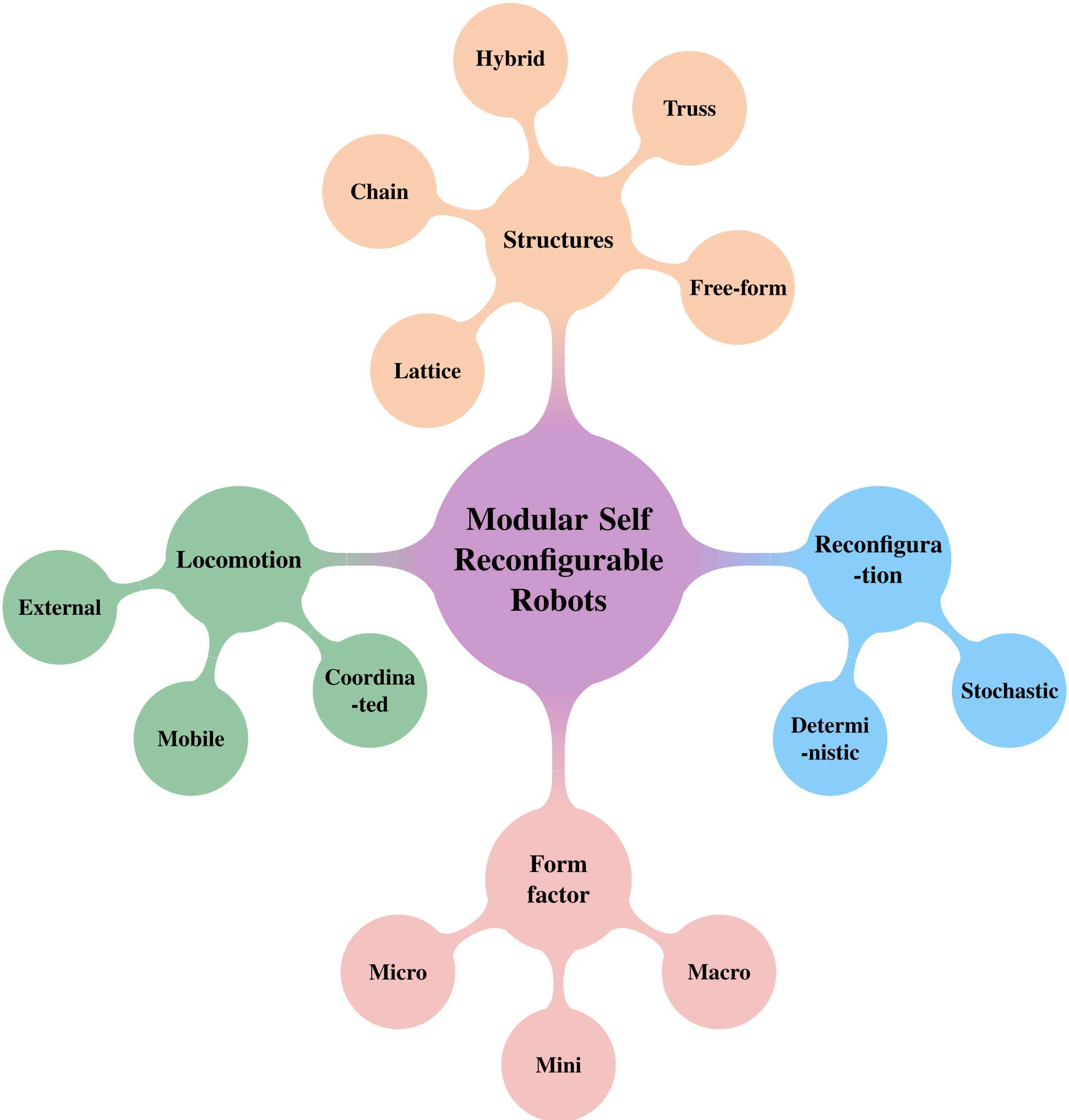


Figure 2.1. Classification of MSRR designs based on hardware characteristics

The MSRR designed for lattice structures are inspired from atomic structures like cubic centered lattice, tetrahedron etc. and are equipped with actuators to form similar structures. The individual robotic units occupy discrete positions in space and lack capabilities to reach random positions/ orientations if necessary due to the limitation in actuator assemblies. The lattice architectures provide easy control mechanisms and usually do not require closed-loop control due to their predefined actuator positions in 2D and 3D space. The robotic units under chain category are serially connected robotic units and are capable of forming complex structures like snakes, centipedes etc. The actuators of these robots are assembled to provide end effector random positions in space. The control of chained systems is more complex and often require feedback to confirm the position of modules in space for reconfiguring structures. The majority of lattice and chain systems are designed without wheels on individual units and hence mobility is realized only during coordination of robots. These MSRR designs can be placed in the coordinated sub-category under locomotion category shown in figure 2.1. Few MSRR designs equipped with wheels are capable of forming lattice or chain structures depending on the design and hence can be placed in the mobile sub-category under locomotion. The hybrid designs provide more advantages compared to lattice and chain robotic structures due to their capabilities in easy adaptation to surroundings by forming both lattice, chained structures and a mixture of both. The MSRR with truss based designs support formation of random structures due to the employment of telescopic links and heterogeneous units for forming structures but require complex algorithms for handling assembly and formation of structures. The free-form category MSRR is the most flexible category in the perspective of attaching and detaching from the system. They can form arbitrary structures and normally maintain weak bonds with neighbors. Chain and hybrid differ from the free-form structures in terms of rigidity in bonding.

The recent contributions to research in MSRR employ disturbances and vibrations in the environment for assembly of robots and hence creating two sub-categories based on Reconfiguration - *Deterministic* and *Stochastic*. Deterministic reconfiguration type of MSRR has precise control over the structures, assembly, and reconfiguration either by employing closed-loop control or advanced actuator assemblies. The stochastic type of MSRR usually don't have control over the assembly of units but retains the ability for disassembly. Hence the reconfiguration after completing a particular structure assembly and the time required for the same is dependent largely on environmental factors.

Many researchers have developed designs in micro to macro form-factors for addressing various test scenarios in MSRRs. The form-factor scaling is performed at a trade-off with capabilities and also increased dependency on events happening in the surrounding environment. Henceforth in this chapter, the MSRR robots occupying the volume equal to and more than a cube of 5 cm side are referred as macro-structures, models occupying less than the volume of macro designs but visible to the naked eye are referred as mini structures and designs not easily visible to the naked eye are referred as micro-structures. The widely accepted classification of MSRRs - classification based on structures is adopted for broadly summarizing the research so far in the thesis. The other classifications are provided implicitly while providing the details of locomotion, dimension and mobility.

2.2.1 Lattice structured systems

Metamorphic robotic system [10–12] is the first lattice structure category robotic design capable of changing structures in the 2D environment. The designers have explored the possibility of hexagonal and square lattice structures using metamorphic robotics systems. A hexagonal skeleton was developed for mimicking the robot outer structure with 6 servo motors at each corner and male and female connectors on alternate sides for docking as shown in figure 2.2a. After successful docking between the cells, each cell can revolve around the periphery of a neighboring cell by gradually changing their structure. The square structured prototypes for lattice structures employs sliding mechanism using gender-based connectivity for movement along the lattice structures.

Murata et al. [13] developed a 2D lattice category MSRR called Fracta. The individual robot in the fracta consists of a top and a bottom module with permanent magnets and middle modules equipped with electromagnets. The assembly is shown in figure 2.2b. The docking process begins with the insertion of the middle layer into the empty space between the top and bottom layer of neighboring modules by activating electromagnets. The operating principle was tested using modules equipped with castors on frictional less surface.

Molecule is a 3D structure supporting design developed by Rus D. [14] and each unit consists of two atoms with a right angle rigid bond binding them. The connectors equipped with electromagnets are present on side faces of each atom. The bonded two atom system is referred to as ”Molecule” and each atom has two degrees of freedom(**DOF**) with one provided by the motor at a connector on the face and another due to the motor at the bond as shown in figure 2.2c. The Molecule as a whole provides 4 DOF and can be used for creating arbitrary structures like walls.

Kurokawa et al. [15, 16] prototyped a 3-D unit in cubical structure with connectors on all faces. Each connector can rotate independently along their axis providing the 3-D unit 6 DOF as shown in figure 2.2d. The connectors on all faces are connected to a single 7W motor using worm gear mechanism controlled by an independent solenoid driven switching technique. The arms are connected using a connection cuff capable of moving back and forth along the axis of the arm. The connection hand, mounted on the cuff closes at one extreme of sliding displacement and opens at the other.

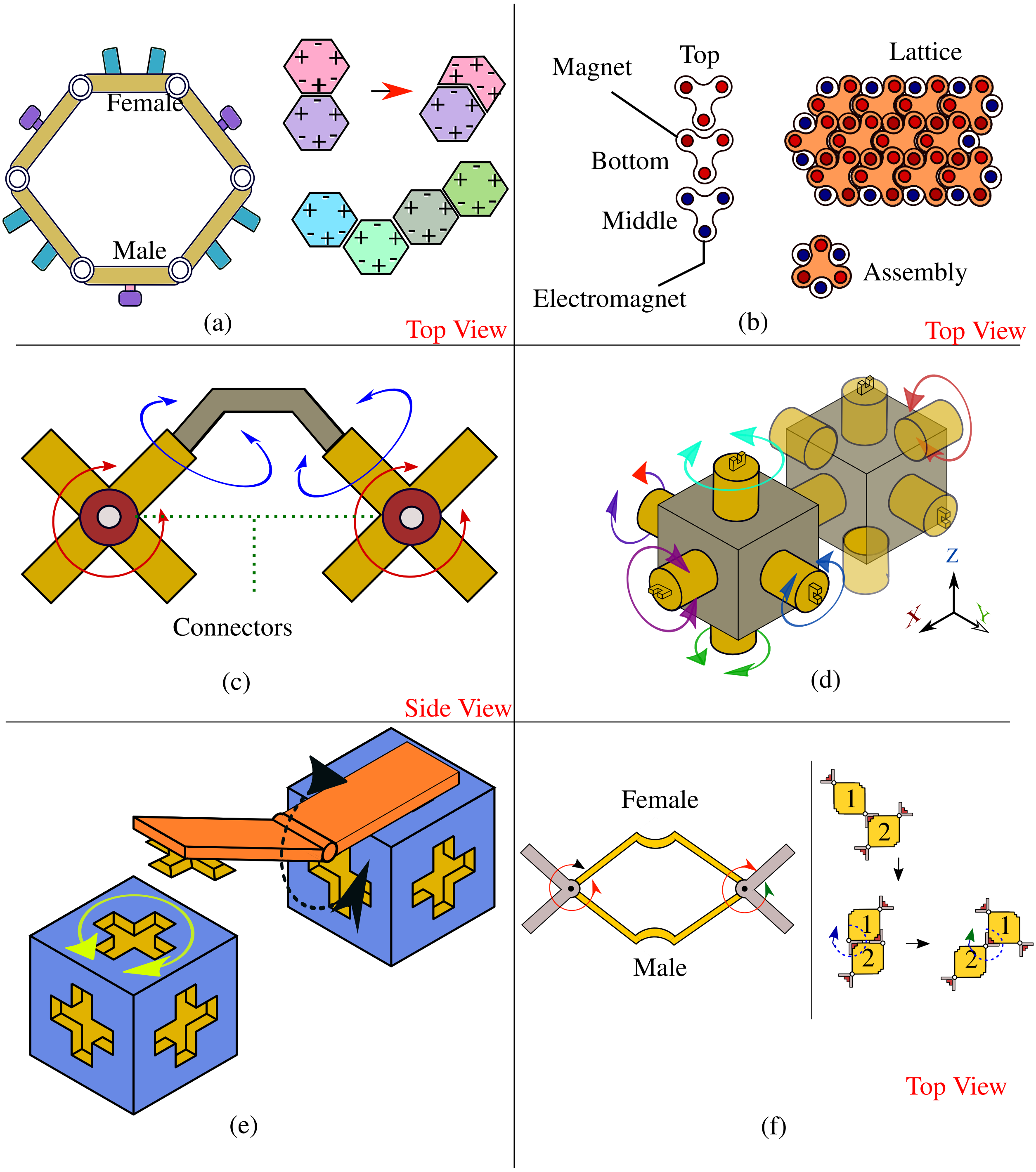


Figure 2.2. Lattice MSRR hardware models a) Metamorphic [10–12] b) Fracta[13] c) Molecule[14] d) 3-D Unit [15, 16] e) I-Cubes [17, 18]f) Micro unit 1 [19, 20]

I-Cubes proposed by Ünsal et al. in [17, 18] is another cubical structure robotic design with two units - Cubes and Links. The faces of cubes have female connectors to mount the links using a lock and key mechanism. A cube at a given time can have zero to six links connected to its faces. The links are independently controlled multi-joint units that are shared and transferred between cubes. The horizontal beam of the link constitutes a joint at the center of two horizontal beams and can be rotated as shown in figure 2.2e. The cubes can rotate with respect to the link after successful latching and hence providing locomotion to the cubes present in the system.

A mini form factor design referred to as Micro-unit has been developed by Yoshida et al. [19, 20]. Micro-unit was prototyped in two different models and each module in the system has square skeleton structure with two static female connecting parts at two ends of a diagonal and rotating male connecting parts at the end of other diagonal as shown in figure 2.2f. The first prototype designed can form structures in 2D with docking controlled by torsion springs made from shape memory alloys(**SMA**). The design employs torsion springs and stoppers coupled with SMA for generating a rotation mechanism. The research team has also attempted further miniaturization of modules by removing the control unit present in earlier prototype and designed the second model providing capabilities for forming structures in 3D.

The Vertical robot published in [21] is a cubical structure of 90mm side independent units. Each cube is equipped with two hands each placed on parallel side faces similar to human hands and rest of the faces are equipped with magnetic sheets. The cells are capable of extending and rotation only along the axis normal to surface they are mounted on. The design facilitates movement of robots only along vertical plane and hence stacking is the only method supported for navigation. The hands of two robots can be docked for lifting and the docking technique is facilitated by a genderless lock and key passive connector. The extension of hands is controlled using sliding mechanism.

Crystalline[22] is a cuboid structured robot with expansion and retraction capabilities on the side faces. The expansion and retraction of faces are performed on all sides simultaneously using rack and pinion mechanism. The active connection mechanism is present on the two neighboring side faces and passive connector mechanism is present on others. Since the system is not designed for docking on top and bottom faces, the crystalline MSRR structures are limited to 2D scenarios. The Telecubes module developed by John et al.[23] is an improvisation to crystalline design with support for 3D structures. The six faces of each module can expand and contract in the direction normal to the face similar to crystalline. Unlike crystalline, telecubes can move in the vertical axis and hence has capabilities of forming 3D structures. Each face on the telecubes module is divided into four quadrants with magnet pole pieces in odd and magnetic metal in even quadrants with chamfered borders for passive docking. The modules couple when they are close to each other since the connection plates on them are mirror images and the SMA springs present in the system pull magnetic pole pieces inside for undocking. The cubic structured module - EM-cubes published in [24] also employed magnets on four faces for docking and locomotion. The permanent magnets are installed provides firm bonding and electromagnets facilitates locomotion. The electromagnets are activated alternatively to create attractive and repulsive forces simultaneously generating couple force at two ends of the cube for locomotion.

M-blocks developed by John et al. [25, 26] are cubical MSRR prototyped in two versions - M-Blocks and 3D M-Blocks. The M-blocks and 3D M-blocks are equipped with an inertial actuator at the center of the body for applying controlled torque at the center of mass of the module and hence rotating the M-block MSRR in clockwise and anticlockwise directions. The M-block cells have capabilities of individual movement for docking. The faces and edges of both models are embedded with permanent magnets as shown in figure 2.3. The rapidly accelerating and decelerating internal rotation mechanism sources the locomotion and edge magnets control locomotion of modules around other robots using pivot action. The face magnets support alignment between the modules after locomotion. The M-blocks provide actuation in a single direction and 3D M-blocks can actuate in six directions by changing inertial actuator orientation to any of three orthogonal axes for 3D movements.

Figure 2.3. M-Block MSRR modules [25, 26]

A mini form factor MSRR - MICHE[27] is designed for forming lattice structures in 3D with the aid of environment. Three faces of cubic structured MICHE are equipped with switchable magnets and rest of the faces are covered with steel plates. The magnets are placed away from the geometric center of plates for avoiding repulsion forces between magnets of two robots during docking. The magnet switching is controlled by internal microcontrollers communicating via infrared(**IR**) transceivers and hence providing capabilities for retaining structures to MICHE MSRR. The MICHE MSRR falls under the stochastic category for its dependence on the environment for aggregation and locomotion.

Pebbles[28] is another stochastic category cubic structure designed to form lattice structures in 2D. The four side faces on the robot can act as a connection plates due to their internal contact with four custom-designed electro-permanent magnets as shown in figure 2.4.

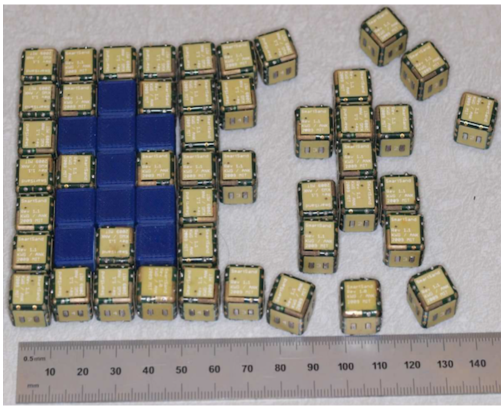


Figure 2.4. Pebbles MSRR modules [28]

White et al. [29, 30] proposed stochastic robotic modules prototyped in two models with one supporting only 2D structures and other supporting 3D structures. The 2D structure modules are designed in both triangular and square base structures. The sides of a module's base are equipped with electromagnet for coupling. The docking and undocking is controlled by actuation and de-actuation of electromagnets enabled via H-Bridges. The stochastic 3D version modules are cubic structures of 10 cm side with permanent magnets placed radially from center and electromagnets at the center of each face. The latching/unlatching is controlled by polarity of the electromagnet. Programmable parts[31] MSRR is another stochastic category robot with a triangular chassis equipped with latching mechanism on all sides. Each side is equipped with a fixed magnet and a rotating magnet controlled by DC motors placed adjacent to each other. During the latching process, fixed magnets of a module faces the rotating magnets of other robotic modules. Hence a module can perform undocking by retracting the rotating magnets in itself and the neighboring modules. The IR sensors inserted into sides handles communication between the modules. The XBOT MSRR[32][33] consists of ’X’ shaped cuboid modules capable of forming 2D structures stochastically. Each leg in ’X’ shape is equipped with a pair of compliant arms with magnets at their tips as shown in figure 2.5. The arms bond different modules together and the coupling/decoupling process is controlled by push-pull process regulated by SMA wires wounded around the frame and arms.

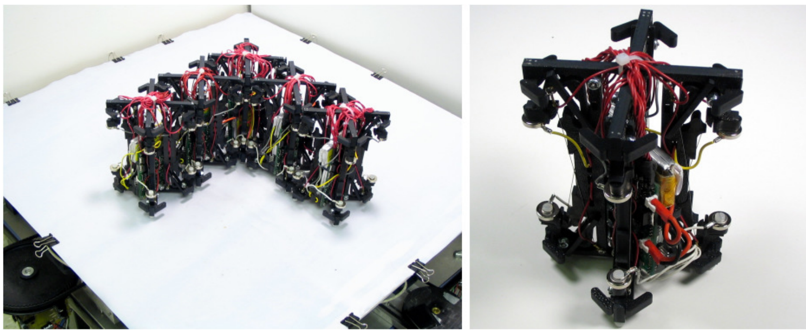


Figure 2.5. X-Bot MSRR modules [32][33]

The ATRON module proposed in [34, 35] is a lattice structured design along with minimal flexibility for forming chain structures in 3D as shown in figure 2.6. The modules are composed of two hemispheres mounted on each other on flat side and each hemisphere is capable of rotating 180^{o} independently. The two hooks(active male) and two passive female connectors placed equidistantly around periphery of each hemisphere in alternate positions facilitate docking. The hooks are driven by worm gears and female connectors are two rigid bars firmly connected to chassis of the module. The rotation of a hemisphere with respect to other provides locomotion in the structures. The tetrapod structured Pet Robot(**PetRo**) MSRR developed by salem et al.[36] is a self-mobile lattice category design proposed for forming 3D structures. The central hub and four legs together form tetrapod structures. Each free end of the legs are connected to a wheel providing one DOF along the leg axis and another DOF is added at the central hub perpendicular to the leg axis with a rotation of $\pm$ 45^{o}. The wheels are also proposed to play role of connection plate between various PetRo modules forming complex structures similar to pets. The IR sensors present on the connector faces aids in alignment for docking. The grooved pins and chamfered holes on the connection surfaces comes opposite to each other for alignment and with the support from magnets the docking process is completed successfully.

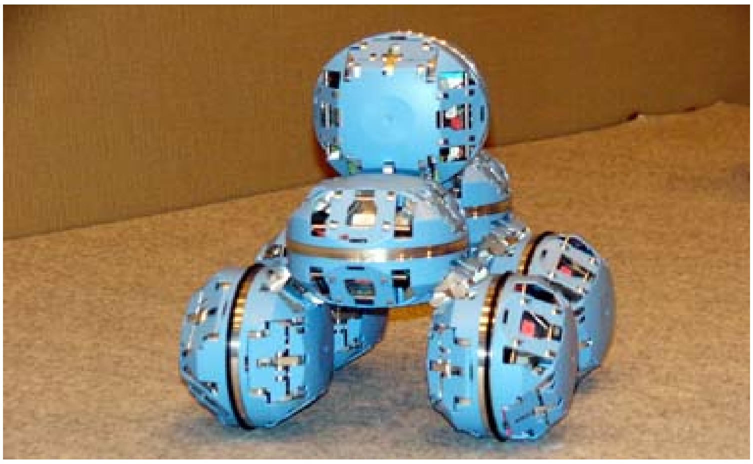


Figure 2.6. ATRON MSRR modules [34, 35]

Table 2.1 provides a broad comparison of various lattice MSRR designs described in the previous sections. The comparison is listed as per the categories mentioned in figure 2.1. Since the shape generally defines the robustness of structures while the number of actuators along with type of actuator defines the parameters such as form-factor, power consumption etc., the details of actuators and structures are also listed. The structures of connection faces and number of connection faces on each MSRR module aids in identifying the probable structures possible when visualized in association with the shape of the robotic module. Since the connection faces are implemented using a wide range of technologies, various jargons are adopted for categorizing them. The number of connection faces column in table 2.1 lists details of a single robotic module in an MSRR design and is separated into two sub-columns - active and passive types for providing better visualization while interpreting locomotion capabilities. The paradigm adopted for connection interfaces can be listed as Male, Female, Active and Passive interfaces. The active connection interfaces are generally constructed using mechanical/electrical actuation mechanisms for docking and the same are absent in passive connection interfaces. The passive connection interfaces still contribute to docking due to the presence of passive materials such as permanent magnets, sockets for screws, velcros etc. The active and passive terminology is widely applied for genderless docking mechanisms and gender-based docking designs differentiate between interfaces using male and female connection faces. The entries in connection faces column in table 2.1 are listed as Male(**M**), Female(**F**) and Dual role (**DL.** - active and passive interfaces present on same face). In the case of presence of heterogeneous modules in MSRR designs, the listed number is total count of the active and passive interfaces present on heterogeneous units.

Table 2.1. Comparison of Lattice structured MSRR designs.

Category				**Robot**	**Shape**	**DOF**	**Coupling**		**No. of Connection faces faces**		**Ref.**
Structures	***Reconfiguration***	***Form factor***	***Locomotion***				***Interface***	***Actuator***	***Active/Male***	***Passive/Female***	
				Micro-unit I	Square	2D	Latch	SMA	2F	2M	[19]
		Mini	**Co-ord.**	Micro-unit II	Cubical	3D	Latch	SMA	3F	3M	[20]
				Metamorphic I	Hexagon	2D	Lock	DC Motor	3M	3F	[10–12]
				Metamorphic II	Square	2D	Lock	DC Motor	2M	2F	[11]
				Fracta	Triangular	2D	Elect. Mag.	Current	3M	3F	[13]
				Vertical	Cubical	2D	Key and lock	—	0	4	[21]
				3-D unit	Cubical	3D	Hooks	DC Motor	6	0	[15, 16]
				Molecule	Cuboid	3D	Elect. Mag.	Current	0	6	[14, 37]
				Crystalline	Cuboid	2D	Key and lock	DC Motor	2M	2F	[22]
				I-cubes	Cubical	3D	Key and lock	Servo motor	2M	6F	[17, 18]
Lattice	**Detrmn.**	**Macro**	**Co-ord.**	Telecubes	Cubical	3D	Swtc. Mgnts.	SMA	6 Dl. faces		[23]
				ATRON	Octagonal	3D	Hooks	DC Motor	4M	4F	[34, 35]
				EM-cube	Cubical	2D	Elect. Mag.	Current	2	2	[24]
				M-blocks 2D	Cubical	2D	Perm. Mag.	—	0	6	[25]
				M-blocks 3D	Cubical	3D	Perm. Mag.	—	0	6	[26]
			Mobile	Petro	Tetrahydral	3D	Latch	Manual	0	4	[36]
				Pebbles	Cubical	2D	El-Pm. Mag.	Current	4	0	[28]
		Mini	**Extern.**	Miche	Cubical	3D	Rot. Pr. Mg.	Motor	3	3	[27]
	Stocst.			Stochastic 2D	Cubical	2D	Elect. Mag.	Current	4	0	[29]
				Stochastic 3D	Cubical	3D	Elect. Mag.	Current	6	0	[30]
		Macro	**Extern.**	Prog. Parts	Triangular	2D	Perm. Mag.	DC Motor	3	0	[31]
				X-bot	Cubical	2D	Perm. Mag.	SMA	4	0	[32][33]

2.2.2 Chain structured systems

The CEBOT [38, 39] MSRR belongs to the mobile category comprising of heterogeneous modules and has two hardware prototypes referred as Series I and Series II. The design facilitates 3D structure formation and comprises of three types of cells -

(a) Wheel mobile cell

(b) Rotation joint cell

(c) Bending joint cell

The cells are fitted with castors at bottom for frictional less movement and are equipped with male and female connectors for docking. The wheel mobile cell shown in figure 2.7 having mobile capabilities initiates docking with the necessary cells. The cells are equipped with SMA couplers for active latching of the male connector during docking and position sensors mounted on the cells provide feedback on the docking process. The cells in series-I prototypes require precise control and alignment for docking. The cells in series II prototypes are replaced with tapered female socket with worm gear for the active latch mechanism instead of SMA while maintaining the same docking process. The physical characteristics of cells in CEBOT are listed in Table 2.2.

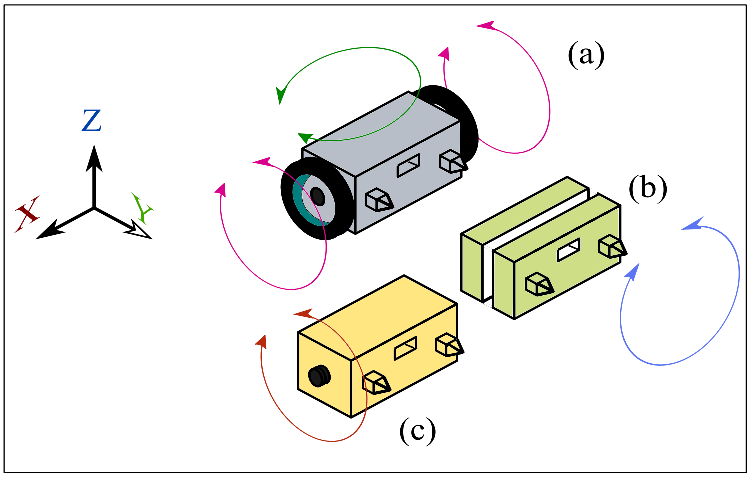


Figure 2.7. Structure of cells in CEBOT MSRR [38, 39] - a) Wheel mobile cell b) Rotation joint cell c) Bending joint cell

Endo et al. [40–42] developed active cord mechanism(**ACM**) MSRR for mimicking snake alike chain structures in 2D. The ACM MSRR have three different versions - ACM, ACM-R2 and

Table 2.2. CEBOT - Cell physical characteristics

	Series I	**Series II**
Dimensions (mm)	190 * 90 * 50	176 * 126 * 90
Weight (Kg)	1.2 (Mobile cell) 1.2 (Target cell)	2.7 (Mobile cell) 1.0 (Target cell)
Connectivity surface	Flat	Tapered
Coupler actuator	SMA	DC Motor

ACM-R3. Each unit in ACM is a wheeled square chassis robot without any actuator present for controlling individual mobility. A servo motor is equipped with every unit to rotate the robot at the joint axis. The ACM MSRR is a combination of such individual homogeneous units assembled manually. The ACM-R2 is an improvement to ACM MSRR and has capabilities of forming 3D structures. The ACM-R2 MSRR is equipped with pitch and yaw motors in the joint unit between units for proving 2 DOF. The ACM-R3 is designed using custom frame body and wheels for providing robust support in the formation of 3D structures and also facilitating manual assembly of robots with $\pm 90^{o}$ offsets with respect to each other.

Brown at al.[43] prototyped a two-sided tracked vehicle called Millibot capable of forming 2D structures for applications like movement in uneven terrains, stair climbing etc. The millibot MSRR is approximately an elliptical structure robot capable of self-docking using male and female connectors via latching mechanism actuated by SMA and is shown in figure 2.8a. The male connectors are installed in the front on a lifter capable of lifting objects vertically with the help of harmonic drives. Amoeba-I is another tracked MSRR with self-mobility proposed by Liu et al. [44, 45] for forming 3D structures. Each unit is a tracked elliptical structure capable of moving itself and is equipped with pitch joint on one side and yaw joint on the other. The robots when manually connected using physical links provides various DOF as shown in figure 2.8b. The amoeba-I MSRR locomotion combinations are numerous depending on the orientation of link between the modules as well as actuation of corresponding joints. Dazhai et al. [46, 47] developed an improvised version of millibot - JL 1, and JL 2 in terms of DOF by providing yaw and pitch control mechanism to each bot and also gear based docking mechanism at the cost of the weight of the robot. The major difference between JL-1 and JL-2 is that earlier employed latching mechanism for docking and later employed gripper for docking. The gripper on JL-2 can also be utilized as manipulator arm for holding objects in the environment.

Lyder et al.[48] developed the Thor MSRR made up of modular blocks. The blocks are analytically developed motors, gears, right angle joints, gears and wheels that can be utilized for forming various single robotic structures similar to lego structures. The blocks can be assembled in various configurations due to symmetry in block designs and Thor is a robot build with a gripper using such blocks. Thor robot is equipped with wheels for mobility, gripper to dock with neighboring modules etc. and hence making it a MSRR.

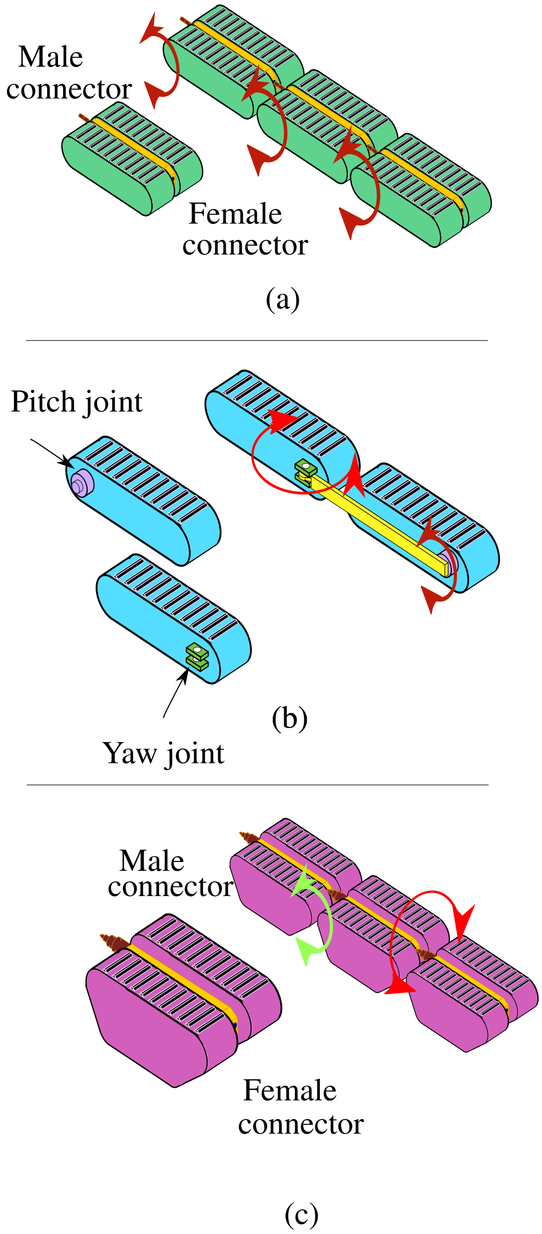


Figure 2.8. Chain structured mobile MSRR hardware models and structures a) Millibot [43] b) Amoeba [44, 45] c) JL-1 [46, 47]

Yim [49] designed Polypod MSRR that falls under the chain structure category and with capabilities of forming 3D structures. Polypod consists of two types of modules: segments and nodes. The nodes are rigid modules in cubical structure with a single connector on each face providing six connectors from batteries. The segments are formed using 10-bar linkages providing two degrees of freedom to the system and are capable of expanding or contracting in length as well as inclining towards left and right. The segments and nodes together facilitate the formation of complex structures in 3D as shown in figure 2.9. The Polypod is actuated using a small DC motor and position sensors are used for measuring angles of the linkages. The control architecture is implemented in three levels with the highest level deciding the behavioral modes, the middle level executing the behavioral mode and the lower level translating the commands to actuator joint space. The connection plates between the modules also facilitate the electrical connectivity for power and communications. In spite of the absence of wheels, the system is capable of movements like a snake, caterpillar, rolling track turning, the moonwalk dance etc.

Figure 2.9. Polypod MSRR modules[49]

Castano et al.[50] designed CONRO MSSR to form structures like snakes or hexapods in 3D. Each module in CONRO consists of three segments

(a) Passive connector

(b) Body

(c) Active connector

Two servo motors with rotation axis in orthogonal orientation are attached to the body as represented in figure 2.10. The pitch motor is connected between the active connector and body. The yaw motor is connected to the body and the passive connector. The docking mechanism and communication is handled using the feedback from IR transceivers present on the faces of active and passive connectors. The SMA equipped locking system present in passive connector latches the modules together after successful docking. A hormone-based centralized and decentralized control for coordinate movements in modular robots was researched on CONRO robots in [51, 52]. Further research on docking and alignment issues in the CONRO robot modules are addressed in detail in [53].

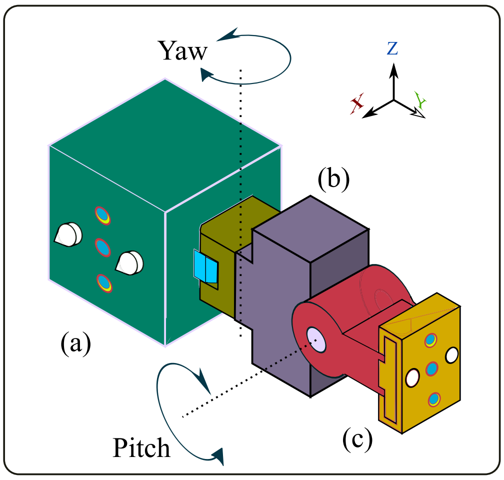


Figure 2.10. CONRO MSRR robotic module architecture [51, 52]

A Modular Robot for Exploration and Discovery(**ModRED**) was proposed by Dasgupta et al.[54] is similar to CONRO MSRR with modification in DOF. The ModRED robot consists of 3 cuboid blocks with 2 pitch motors - one at first block and other at last block. A prismatic motor is placed along with pitch motor in the last block for elongation of bond between the center and last block in the horizontal plane. A roll motor is placed at the center block for rotating the front block with respect to center block. The first and last blocks are equipped with brackets as connectors with grooves and pins in structure of square along with a solenoid controlled mechanism for latching.

Polybot[55, 56] MSRR is a chain structure inspired robotic design capable of forming 3D structures. The polybot is a cubic structure prototyped in three major versions - G1, G2 and G3. The G1 version of Polybot is a quick prototype with connection plates on front and back faces of 5 cm cube. The connection plates orientation with respect to each other can be changed with DC motor mounted outside the cube whose axis of rotation is normal to the side faces. The G1 prototype has no mechanism for latching and unlatching and hence docking is done manually. Since the connection plates are equipped with grooved pins and holes symmetrically, it is possible to dock two polybot G1 modules back to back even with an offset of 90^o. The Polybot G2 is similar to G1 and additionally equipped with electromechanical latches and SMA controlled by software. The docking mechanism is guided by IR transceivers mounted on face plate and the robot is shown in figure 2.11. The Polybot G3 are miniaturized modules with dimensions around 50 * 50 * 50 mm^3. The externally visible DC motor in G1 and G2 version is made internal by changing the mechanism to dc pancake motor with harmonic gear along with active braking feature.

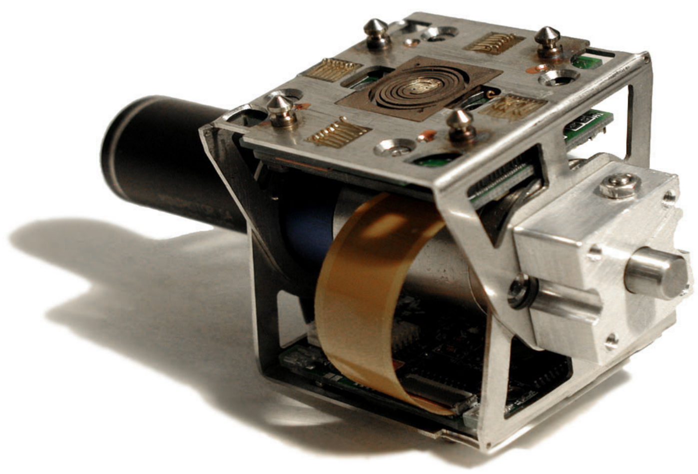


Figure 2.11. Polybot G2 MSRR modules[56]

The Transmote[57] module design is similar to Polybot with major difference in latching mechanism and number of connection surfaces. The front side face of the transmote is equipped with a conical structure used for docking with female socket present at the back of the robot. The Transmote facilitates twist and lock mechanism controlled by a servo motor for docking between robots. Transmote MSRR has a connection provision on one side face along with front and back faces providing more stability to 3D structures. The GZ-I MSRR robotic module proposed in [58] is similar to the Transmote with three connector faces and slightly different physical construction. The GZ-I modules were not equipped with docking sensors, actuators etc. and hence are assembled manually. The yet another modular robot(**YaMor**)[59] robot is a semi-cylindrical box structured robot capable of forming 2D chain structures. A triple beam in shape '⊔' is connected to the side faces of a semi-cylindrical box at the free ends of beams. Each robot module has one DOF and the system does not support autonomous docking. The velcros placed on the beams, side faces and back of the robots are used for docking with neighboring modules manually. The YaMor robot is a complete integrated solution with wireless communication capabilities and field programmable gate arrays(**FPGA**) for reconfigurable computation purposes. Table 2.3 provides comparison on various chain modular robots developed as part of research.

2.2.3 Hybrid structured systems

Mondada et al.[4, 62–68] developed a completely integrated autonomous robot called S-BOT capable of forming lattice structures in 2D and chain structures in 3D and hence a hybrid category robot. The robot is a cylindrical structured track robot designed for research in swarm robotics. The robots are capable of localization and navigation in uneven terrains. The robots employ gripper mechanism for docking with a ring covering the periphery of the robot. Since the ring is present around the periphery, the docking can be done almost from every direction. The optical sensors present in gripper modules form a closed control loop for providing feedback on docking process. The S-Bot employs same features of modular robots such as modularity, reconfiguration etc.

The M^3 MSRR proposed by Kutzer et al.[69, 70] is capable of forming a 3D chain and lattice structures along with mobility features and is developed in two versions - M^3 and M^3 express. The models are 'L' shaped robot with two wheels on parallel sides of the long beam and one omnidirectional wheel on outside face of short beam parallel to surface and perpendicular to the common rotational axis of other two wheels as shown in figure 2.12. The wheels play a dual role - enabling mobility and connection plates for docking. The M^3 module is equipped with two hooks on wheels separated by 180^o. The units are latched together when wheels of two modules come face to face with an offset of 180^o or 360^o. The custom designed slip rings aids robots with docking as well as mobility using same wheels. In the M^3 express module each wheel is equipped with two magnets at the ends of the diameter, a yoke and four locking pins. The yokes are connected to servo motors in a sliding mechanism for activating a slip disk with metallic screws. The disk is

Table 2.3. Comparison of Chain structured MSRR designs.

Category				Robot	Shape	DOF	Coupling		No. of Connection faces faces		Ref.
Structures	*Reconfiguration*	*Form factor*	*Locomotion*				*Interface*	*Actuator*	*Active/Male*	*Passive/Female*	
				CEBOT I	Cuboid	3D	Latch	SMA	1M	1F	[39]
				CEBOT II	Cuboid	3D	Latch	DC Motor	1M	1F	[38]
				ACM-R1	Rectangle	2D	Latch	Manual	0	2	[40]
				ACM-R2	Rectangle	3D	Latch	Manual	0	2	[41]
				ACM-R3	Rectangle	3D	Latch	Manual	0	2	[42]
				Millibot	Elliptical	2D	Latch	SMA	1M	1F	[43]
			Mobile	UNI-Rover	Cylind. arm	3D	Latch	Servo Motor	1	1	[60]
				Amoeba - I	Cuboid	3D	Latch	Manual	0	2	[44][45]
				JL-1	Trapezoid	3D	Latch	DC Motor	1M	1F	[46]
				JL-2	Trapezoid	3D	Gripper	DC Motor	1M	1F	[47]
				Thor	Cuboid. arm	3D	Gripper	DC Motor	1	0	[48]
Chain	**Detrmn.**	**Macro**		Steering.	Cuboid	2D	Latch	—	1M	1F	[61]
				Polypod	Cubical	3D	Latch	Manual	0	6	[49]
				CONRO	Cuboid	3D	Latch	SMA	1M	3F	[50][53][52][51]
				Polybot I	Cubical	3D	Latch	Manual	1M	1F	[55]
				Polybot II	Cubical	3D	Latch	SMA	1M	1F	[56]
				Polybot III	Cubical	3D	Latch	SMA	1M	1F	[56]
				Yamor	Semi-Cylnd.	3D	Velcros	Manual	0	4	[59]
			Co-ord.	GZ-1	Cubic	3D	Latch	Manual	0	4	[58]
				Transmote	Cubic	3D	Key and lock	Manual	1M	2F	[57]
				ModRED	Cuboid	3D	Latch	Solenoid	2 Dl. faces		[54]

normally separated due to internal springs and the actuation of servo motor mounts the slip disk into wheels bringing metallic screws onto the face of wheels at the ends of other diameter for docking.

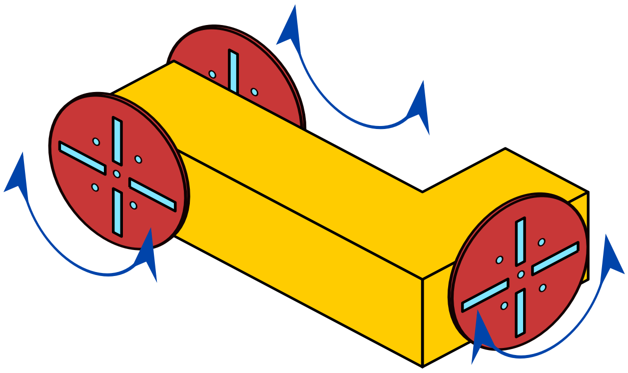


Figure 2.12. M^3 express MSRR robotic module architecture[69, 70]

iMobot[71, 72] is another mobile hybrid MSRR prototyped by Harry et al. The iMobot MSRR is a cuboid structured formed from the assembly of two semi-cylindrical modules as shown in figure 2.13. The side faces of iMobot are equipped with chamfered flat sheets capable of rotating continuously and hence providing mobile abilities to the robot. The semi-cylindrical modules are capable of rotating 180^o along their axis independently. The four rotation mechanisms together aid iMobot to mimic movements such as crawling, rolling, standing etc along with lattice and chain structures. The iMobot modules can be assembled manually all the sides and hence forming various complex structures required for numerous real-time applications.

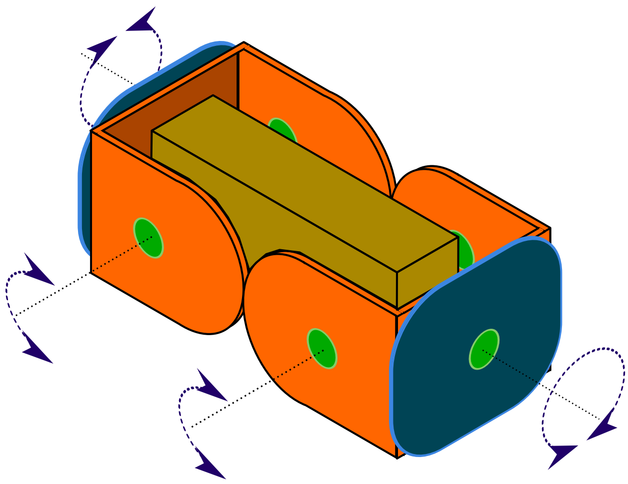


Figure 2.13. iMobot MSRR module architecture[71, 72]

The SMORES MSRR design proposed by Yim et al.[73] is similar to iMobot consisting of a single semi-cylindrical cubic structure on which three of four side faces of the cube are equipped with circular discs. Two circular discs on parallel faces that play a dual role of movement and docking and a third disc is used for rolling neighboring modules after docking. Another internal motor provides pitch movement abilities to the system by lifting wheel orthogonal to the common axis of rotation of the two side wheels in parallel. The locomotion is designed using orthogonally placed gears. Each face is equipped with four magnets with same polarity magnets occupying alternate positions and hence at a time eight magnets participate in a docking when the connection plates face each other with an offset of 90^o or 270^o. The docking keys selector present internally can extend through the center of all faces creating a necessary gap for undocking.

Trimobot[74] is a fully integrated mobile category hexagonal MSRR capable of forming lattice structures in 2D and chain structures in 3D. The robot is equipped internally with three omnidirectional wheels on the alternate sides of hexagonal structures for movement in the 2D plane. The sides of trimobot are fixed with 5 passive connection faces and an active connection face on the outside. A pitch joint is embedded with active connector face on one side of the hexagonal structure to facilitate lifting of modules in the vertical plane and hence forming chain structures in 3D. The active connector face is also equipped with a camera for docking purposes. The docking is enabled using four hooks present on active connector face and is controlled using rotation mechanism. The hooks are activated during docking when the passive and active connector faces of various modules face each other.

M-Tran is a hybrid configuration modular robot capable of forming 3D structures in both the lattice and chain configurations and has three versions - M-Tran I [75, 76], M-Tran II[77, 78] and M-Tran III[79]. M-Tran robotics system consists of active and passive modules in the semi-cylindrical structures and a link is permanently fixed in the active unit as shown in figure 2.14. The active, passive modules and links are equipped with four permanent magnets in a square structure on outside faces providing three connection surfaces on each module and two connection surfaces on the link. The passive units can be coupled at the back of active units in two different angular orientations - 0^o, 360^o and 90^o, 270^o due to the alignment of magnets. The connection surfaces are also designed to aid electrical connectivity between the modules. The servo motors present in the active unit enables the rotation of the link and the connection is established between units after a link present on active units enters the passive unit. The latching process is controlled by SMA coils by extending or retracting the magnets in the passive units docked with magnets in the link. The M-Tran II latches/unlatches link with the passive part at 89% more efficiency when compared to M-Tran I with a trade-off observed in time. The M-Tran III is an improvised design when compared to previous versions. The latching/unlatching between the link and passive part is replaced with hooks controlled by motor and hence providing a more stable connection.

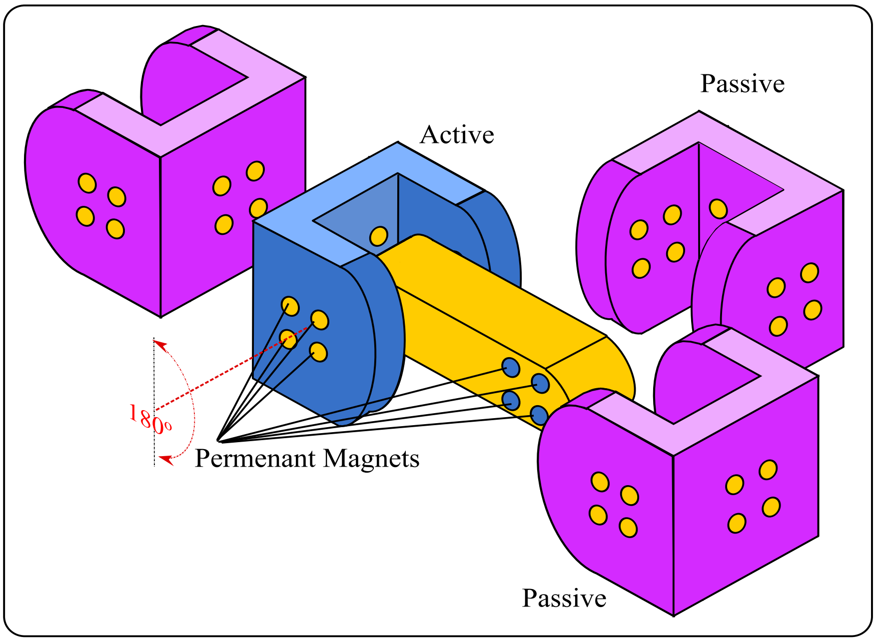


Figure 2.14. M-Tran MSRR robotic module architecture[75, 76]

The Superbot module proposed in [80][81] is formed by a permanent bonding between two semi-cylindrical cells using a link similar to iMobot MSRR. The cells are capable of rotating by 180^o along their individual axis and also can also roll with respect to bond binding them. The superbot MSRR has connectors on all faces, making 6 connectors in total available on each superbot module. The rotating bond and the two cells together provides 3 DOF for each superbot module - 180° yaw, 180° pitch and 270° roll. The superbot is capable of forming both lattice and chained structures and hence making it a hybrid category robot. The connector kinetic robot(**CKbot**) MSRR design proposed by Yim et al.[82] is similar to SMORES MSRR with the reduction in self-mobility and rolling capabilities in individual units. The CKbot MSRR has auto docking/un-docking features enabled by magnetic faces and also via screws if a manual assembly is necessary. The CKbot MSRR is designed to test the self-healing capabilities of the robotic system with the aid of vision after sudden events such as explosions etc.

Zykov et al.[83] developed Molecubes - a cubic structure based hybrid category MSRR. The cube is an assembly of two parts made by splitting the cubic structure of 10 cm along the plane normal to a longest diagonal as shown in figure 2.15. One half of the cube can be rotated with respect to other in multiples of 120^o with the help of internal servo motor coupled with the worm gear. The system is capable of forming both chained and lattice structures. The permanent pole magnets present around the center of faces facilitates coupling and the polarity of electromagnets at center can be utilized for severing or strengthening the bonds.

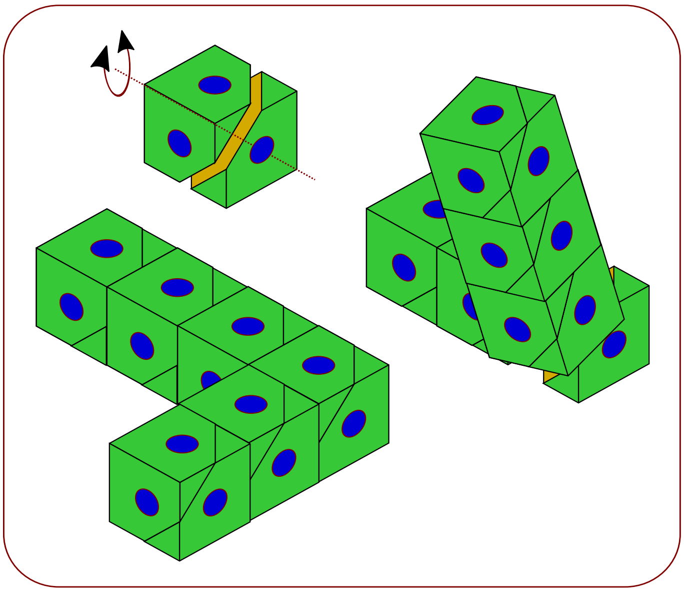


Figure 2.15. Molecubes MSRR robotic module architecture[83]

The UBot[84–86] MSRR system consists of cubic structured cells capable of rotating in discrete steps along the longest diagonal similar to Molecubes. The internal faces are chamfered for facilitating rotation. The Ubot robotic cells are categorized into active and passive modules with active modules providing four active connection interfaces and passive modules providing four passive connection interfaces. The active and passive modules have the same outer structures and rotation mechanisms. The hooks present on active connection interfaces enables firm docking with passive connectors. The active and passive modules are latched using a hook and sliding mechanism guided by position sensors for forming lattice and chain structures in 3D making UBot a hybrid category robot.

Roombots[87, 88] MSRR is another hybrid architecture designed to form a chained and lattice structure in 3D. Each roombot robot has two cells of spherical structure bonded together and each cell is a combination of two half-spheres mounted on each other along the faces as shown in figure 2.16. The locomotion is facilitated by three gear motors - one at the bond between cells and one is present in each cell for rotating the other half spheres. Each roombot robot can be equipped with ten active connections on a half sphere to a single active connection and 8 passive connections.

The connection mechanism between various Roombots is implemented with mechanical latches for holding the neighboring modules at the holes present on the surface.

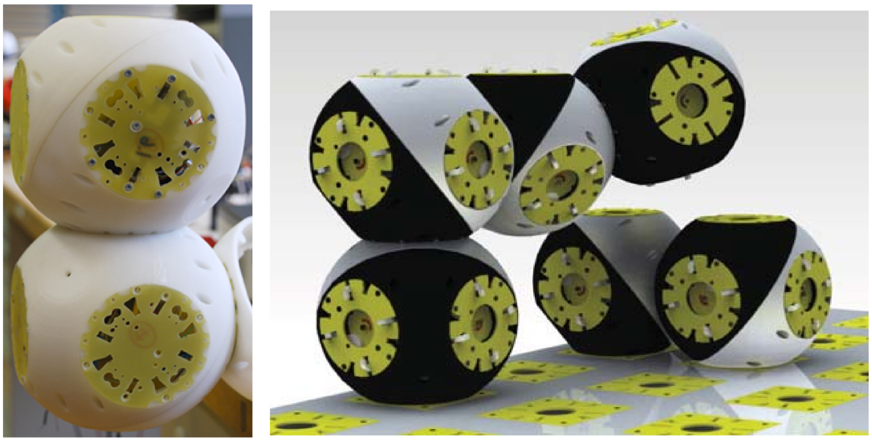


Figure 2.16. Roombot MSRR modules[87, 88]

Soldercubes developed by Jonas et al.[89, 90] is a hybrid category MSRR with the shape similar to a cell in the dual-cell structure of Roombots. The six genderless connector faces of each cell facilitate docking between modules and coordinates movements. The connector faces are custom made symmetrically designed PCB boards with soldering contacts. The contacts on the connector faces can be melt upon transmission of current at low temperatures and hence making a bond between modules for forming structures along with mechanical and electrical connections. The bond can be broken using the same mechanism of melting the contacts. The soldercubes module has an embedded mechanism for rotation of single connector face providing single DOF to the module but facilitating various DOF after docking with similar modules as shown in figure 2.17.

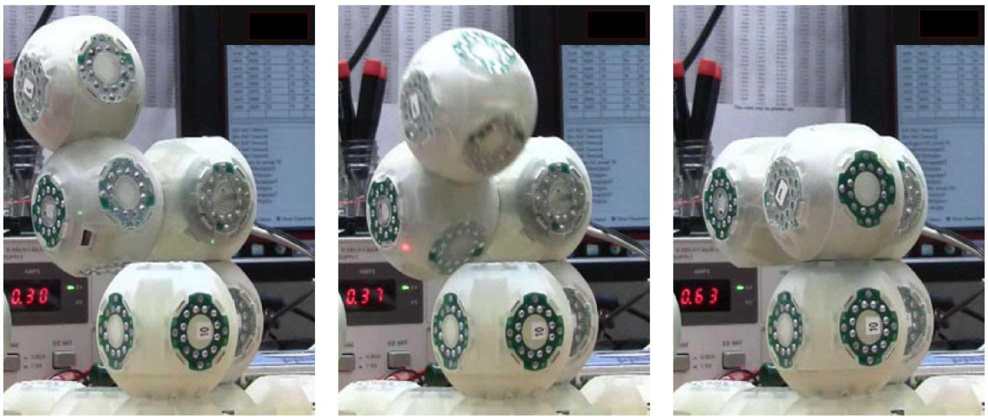


Figure 2.17. Soldercube MSRR modules[89, 90]

Table 2.4 provides comparison on various hybrid modular robots developed as part of research.

2.2.4 Truss structured systems

Hamlin et al.[91, 92] prototyped a Truss based MSRR - Tetrobot for forming random structures using heterogeneous units - Links and Joints. The links in the tetrobot robot are cylindrical rods of fixed length and reconfiguration is supported only at the joints. A three-axis concentric multi-link spherical joint capable of expansion and contraction in 3D is designed to hold three links together. The assembly between joints and links along with reconfiguration is performed by controlling joints using motors. Ramchurn et al. proposed a conceptual truss design MSRR - ORTHO-BOT[93] with telescopic links having split toroidal at two ends and with one toroid connected to link via revolute joint. The split-toroid joint aids in inter-connectivity between modules providing 2 DOF rotation. The locomotion of co-ordinated system is simulated for structures such as hexapod.

Odin[94] MSRR consists of heterogeneous units - cubic closed packed joints and telescopic links along with capabilities to form structures in 3D as shown in figure 2.18. The CCP has twelve female connector sockets each with internal female PCB connector. The telescopic links are extendable cylindrical structures with flexible connectors on both ends equipped with male PCB connectors. The modules are not capable of autonomous docking and are fitted manually. The joints act as power sharing and communication interfaces between the controllers present in links. The Morpho truss system developed by Yu et al.[95] consists of active links, passive links, and joints. The active links can expand and contract due to internal actuation of motors and the passive links expand and contract due to external forces. The links are joined together manually using a cubic structured interfacing unit with a connector on each face. A surface membrane is covered over a 3D-skeleton structure formed using links and joints for realizing structures like conveyor belts with adapting topologies. Hjelle et al.[96] developed Hinge MSRR for reconfiguring truss structures. The design of truss system used as a testbed is similar to Odin MSRR. The joints have 18 female connectors and the struts are passive cylinders fastened by threaded inserts. Instead of providing locomotion in struts or joints, the hinge robot maneuvers from one strut to another till it reaches the destination and rotates the struts with help of servos by firmly holding them and hence reconfiguring the structure.

A concept of shape-shifting materials was introduced by Amend et al.[97] for programmable structures. The system consists of links and nodes like general truss systems. The links are beams

Figure 2.18. Odin MSRR modules[94]

of granular material instead of static metal structures. The nodes are connectors between the beams supporting the transfer of granular materials from one beam to another. The nodes are capable of jamming the movement of materials and hence modifying the stiffness of beams for changing structures. Galloway et al.[98] developed a reconfigurable truss system called factory floor to demonstrate the idea of an auto-assembly of truss structured systems. The CKBots equipped with a manipulator is used for assembly of custom structures by placing various elements together. The joints in factory floor MSRR are cubical structures with passive connectors on each face and the struts are hollow cuboid rods with grippers at both ends for docking. The pressing action performed by manipulator at the center of strut creates a couple force internally leading to the opening of grippers.

A comparison on truss category modular robots is listed in table 2.4.

2.2.5 Free-form structured systems

Tokashiki et al.[99] prototyped a MSRR capable of forming free-form structures in 2D. The cylindrical structured MSRR(referred as Transform. henceforth) is equipped with gear on the top and bottom of the cylinder that are actuated by motors as shown in figure 2.19. The robots are also equipped with 6 pole magnets around the periphery for providing bonding between the robots by attraction. The robots can move around when the gears of neighboring modules are locked with each other with magnets maintaining the structural integrity of the system.

Goldstein et al. developed a cylindrical structured MSRR named Claytronics[100–102] of diameter 44 mm for demonstrating the structure mimicking in 2D. The periphery of cylindrical structure is equipped with 24 spherical electromagnets in two rings present one below the other. The robots by themselves are immobile and require support of neighboring robots for forming structures as well as locomotion(on frictionless surfaces). The modules have point contacts due to the shape of electromagnets and hence can implement various structures at much faster pace compared to other latched and rotating structures as shown in figure 2.20.

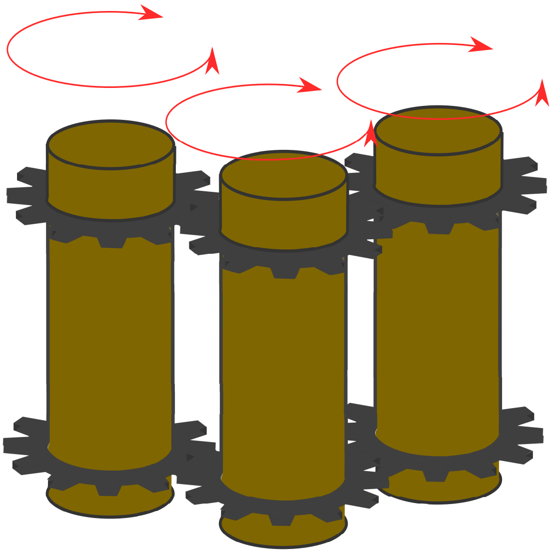


Figure 2.19. Transform. MSRR robotic module architecture

Figure 2.20. Claytronics MSRR modules[100–102]

Slime[103, 104] is another cylindrical design capable of forming a free-form structure similar to claytronics MSRR. The slime MSRR is equipped with 6 solenoids each controlling a 60^o section of 360^o periphery. Each cylinder section is equipped with a velcro to make contact with the neighboring robots. The spring action regulated by pneumatic air cylinders can extend and retract the cylinder sections for making and breaking the bond between robots. An extra solenoid placed downwards controls the position of a friction plate with respect to ground for increasing/decreasing friction during attachment/detachment process. The mini form-factor MSRR - Catoms[105] is another cylindrical structure utilizing electrostatic forces for locomotion. The Catoms MSRR consists of a cylindrical wafer of 1mm diameter and electrode strips placed vertically around the periphery of the cylinder. The electrodes are sourced such that every alternate electrode holds charges of opposite polarities. The stability of structures is maintained by static fields and locomotion mechanism is controlled by changing the polarities of electrodes on modules.

A micro form-factor scratch drive MSRR - MEMS was developed by Donald et al.[106, 107] for forming free-form structures. The module consists of an arm and a scratch drive forming an 'L' shaped structure, whose structures are controlled by the voltages applied to the module. The long beam acts as a scratch drive for turning and the short beam in the structure is used for movements. The pulsating voltages applied to the system from the bottom surface creates various structures in arm and scratch drive with different frictional effects contributing to the movement. The authors have explored various control algorithms and movement strategies for aligning the robots in a structure required using pulsating voltages. A comparison on free-form category modular robots is listed in table 2.4.

The design optimization strategies such as FEM modeling and multi-body dynamics are rarely performed on the designs in modular robotic research. The actuators available in the market doesn't provide large torques in small form-factor at minimal power consumption and the limitations in the actuator technologies restrict the number of modules that can participate in a coordinated structure. On the other hand, the strength of the surface materials of modular robots is strong enough to manage majority of loads due to the wide range of alloys available.

The power optimization strategies are yet to cross a threshold since more emphasis is placed on the identification of suitable sensor-actuator mechanisms in small form-factor. The power conservation in modular robotics can be extended from optimizing sensor-actuators interfaces and microcontroller regulation to optimization of locomotion and retention of structures.

Table 2.4. Comparison of Hybrid, Truss and free form MSRR designs.

Category				Robot	Shape	DOF	Coupling		No. of Connection faces faces		Ref.
Structures	*Reconfiguration*	*Form factor*	*Locomotion*				*Interface*	*Actuator*	*Active/Male*	*Passive/Female*	
				S-BOT	Cylindrical	3D	Gripper	Motor	1	1	[4][62][66][63][64]
				M^3	L shaped	3D	Hooks	DC Motor	3 DL. conn.		[69]
				M^3express	L shaped	3D	Latch	SMA, Servo	3 DL. conn.		[70]
			Mobile	iMobot	Cuboidal	3D	Latch	Manual	0	6	[71]
				SMORES	Cubical	3D	Perm. Mag	DC Motor	3	1	[73]
				Trimobot	Hexagonal	3D	Hooks	DC Motors	1M	6F	[74]
Hybrid	**Determn.**	**Macro**		M-Tran I	Semi-Cylnd.	3D	Perm. Mag	SMA	6	2	[76, 108]
				M-Tran II	Semi-Cylnd.	3D	Perm. Mag	SMA	6	2	[77, 78]
				M-Tran III	Semi-Cylnd.	3D	Hooks	DC Motor	6	2	[79]
				Superbot	Cuboid	3D	Latch	Manual	0	6	[80, 81]
				Molecubes	Cubical	3D	Elect. Mag.	Current	6	0	[83]
			Co-ord.	CKBot	Cubical	3D	Perm. Magnets	Manual	0	4	[82]
				UBot	Cubical	3D	Hooks	DC Motor	2	2	[84–86]
				Roombots	Cuboidal	3D	Latch	Manual	0-10	0-10	[87]
				Neurobot	Cubical	3D	Latch	DC Motor	1M	1F	[9]
				Soldercubes	Cubical	3D	Binder Mat.	Current	6	0	[89, 90]
				Tetrobot	Cylindrical	3D	Spherical Jnt.	Manual	2M	3F	[91, 92]
				ORTHO-BOT	Linact	3D	Split toroid	Manual	0	2	[93]
				Odin	Cylindrical	3D	CCP jnt.	Manual	2M	12F	[94]
Truss	**Determn.**	**Macro**	**Co-ord.**	Morpho	Cubical	3D	Cubic Jnt.	Manual	1	6	[95]
				Shape shift.	Amorphous	3D	3 way pipe	Manual	2M	3F	[97]
				Hinge	Cylindrical	3D	18 axis node	Manual	2M	18F	[96]
				Factory Flr.	Cylindrical	3D	Grippers	Couples	6M	2F	[98]
		Micro	**Co-ord.**	MEMS	L shaped	2D	Alignment	Voltage	—	—	[106, 107]
		Mini	**Co-ord.**	Claytronics	Cylindrical	2D	Electr. Mag.	Current	24	0	[100–102]
Free Fr.	**Determn.**		**Co-ord.**	Catom	Cylindrical	2D	Electrodes	Voltage	8	0	[105]
			Co-ord.	Slimebot	Cylindrical	2D	Velcros	Pnu. air Cyld.	0	6	[103, 104]
		Macro	**Co-ord.**	Transform.	Cylindrical	2D	Perm. Mag.	—	6	2	[99]

Communication plays a vital role in generating locomotions by employing peer to peer communication or centralized communication strategies. The communication techniques are least researched of all the optimizations possible due to availability of numerous industrial standards for immediate use and also primarily due to nascent stages of development of modular robots.

2.3 Summary

The MSRR modules summarized so far are designed in various shapes such as squares, triangle etc. for 2D scenarios and cube, cuboid, cylinder etc. for 3D scenarios so that the modules can have maximum contact surfaces for docking with neighbors while providing stability for a coordinated structure as they adopt in the environment. Research in MSRR has also been extended to the development of robotic development environments, Communication protocols(Wired and Wireless), Middleware development[109][110], Human-machine interface improvement etc. which can be generally coupled with MSRR robotic modules providing a complete platform for rapid research in MSRR.

In this chapter, a summary of various modular self-reconfigurable robotic structures is provided in terms of form-factor, mobility, structural capabilities and reconfiguration strategies. Research in MSRR as can be visualized as a deeply creative process employing various technologies from sensor-actuator mechanisms requiring in-depth understanding about the merits/demerits of various locomotion and sensor and actuation technologies. The research involves intensive prototyping and many MSRR models developed in past research with limited autonomous capabilities can be researched again due to the availability of miniaturized sensor and actuator assemblies.

Chapter 3

HexaMob - A Modular Robotic Design For Implementing Biomimetic Structures

3.1 Introduction

Though robots are extensively utilized in various fields of automation, few applications such as disaster management and navigation in uneven terrains require robots that are capable of adopting according to the environmental constraints. The biomimetic robotic designs such as ANYmal shown in figure 1.1a prove to be more effective for such applications instead of wheeled robots. The biomimetic robotic structures are designed by identifying the necessary links and joints in biological organisms from the analysis of their locomotion as shown in figure 3.1.

The locomotion capabilities of such replicated models reaches close to biological organisms when the joints used in replication provide enough degrees of freedom and necessary number of links to support the structures. The robotic design perspectives evolved from utilization of uniquely designed wheeled robots for every real-world scenario to utilization of biomimetic robots for a application. Though biomimetic robotic structures added new capabilities to robotic designs, reconfigurability remained as a major setback in utilization of the robots in the applications where

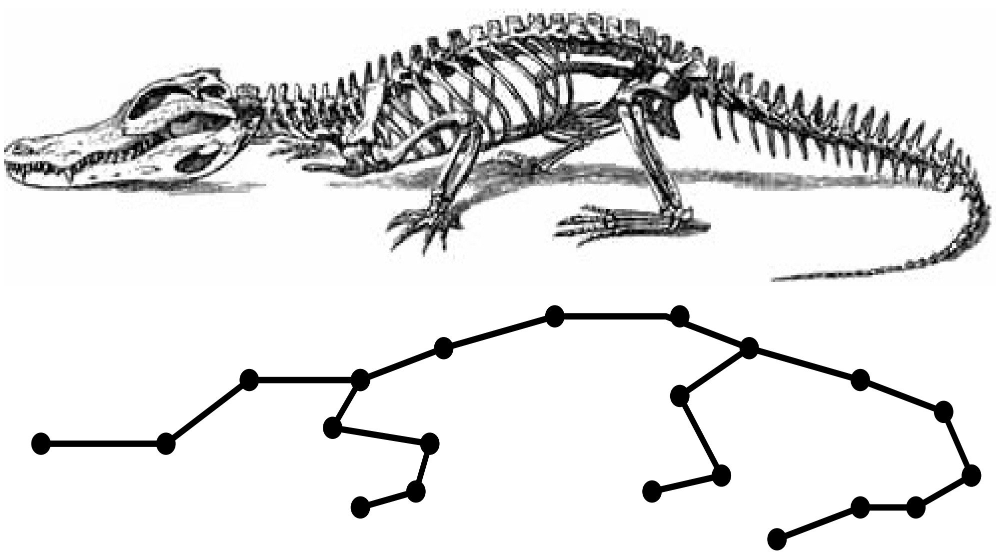


Figure 3.1. Link and Joint structure of vertebrates

constraints are not defined clearly and real-time demands are present. Modular robotics can aid in improving the reconfigurability of the robot by utilization of intelligent combination of link and joints that are reconfigurable as per the requirements.

A novel modular robotic design named HexaMob is proposed in this chapter that is capable of forming biomimetic structures. This chapter is organized as follows. The details on motivation to the current work along with previous research work in modular robotics domain that is closely related to HexaMob robotic module is provided in detail for better understanding. The design characteristics of HexaMob robotic module is explained in the next section. Further sections provide details on various optimizations and sensor mechanisms considered during the design of the HexaMob robotic module. A comparison on characteristics of robotic modules including HexaMob is provided at the end of the chapter with respect to the related work provided.

3.2 Related work

The research in modular robotics is conventionally categorized using various parameters/features facilitated by individual robotic units such as structural formation capabilities, locomotion and form-factor. Table 3.1 provides a list of categories proposed so far based on the research in hardware modeling in the domain of modular robotics. The chain and hybrid category modular robots are gaining prominence in relative to other categories due to their suitability in real-world applications.

Table 3.1. Classification of modular self-reconfigurable robots

Parameter	Type
Structures	Lattice Chain Hybrid Truss Free-form
Locomotion	Mobile Coordinated External
Form-factor	Micro Mini Macro
Reconfiguration	Stochastic Deterministic

Many robotic modules are developed with capabilities for forming chain structures. Modular robotic designs such as ACM [40–42], Millibot [43], Uni-rover [60], Sambot [111], Scout[112, 113] and Trimobot [74] are capable of forming chain structures along with mobility support. Robotic designs such as Polypod [49], CONRO [53], Polybot [56], Transmote [57], ModReD [54, 114], and CKbot [82] are also capable of forming chain structures designed without self-mobility feature in independent robotic modules. Hybrid category robotic units such as M^3 robot [69], M^3 Express [70], iMobot [72], SMORES [73], M-TRAN I-III [76, 77, 79], UBot [85], Soldercubes [90], HyMod [115] and CoSMO [116] are capable of forming both lattice and chain structures with few designs equipped with capabilities for self-mobility.

The majority of modular robotic designs explored the concept of developing an immobile robot [81][90][117] which is coupled with homogeneous units for forming the coordinated structures. The major drawback of such systems is the requirement of manual intervention in most of the implementation scenarios. Few modular units are assembled manually into structures to initiate locomotion and reconfiguration due to lack of aggregation and dispersion abilities in individual robots. A relative comparison on various reconfiguration features and structures of modular robots provided in [9] and an extensive survey including software components published in [118] provide deep insight into research in the domain of modular robotics.

Every design in modular robotics domain is unique in the perspective of characteristics and features of the design. Numerous modular robotic designs are prototyped in research for real-world applications and it was realized that the chain and hybrid modular robotic designs are suitable for real-world applications due to their abilities in forming biomimetic structures like gaits, centipedes etc. The overall capabilities of a modular robotic depends on performance of autonomous features, sensor actuator interfaces, chassis structures, locomotion mechanisms etc. Though there are no standardized metrics to quantify the capabilities of modular robots, the performances of designs and prototypes can be made in terms of capabilities to form different structures and reconfigurability.

- **Capability to form numerous structures** - The ability to form numerous structures can be directly related to the number of docking faces. The chassis structure and number of docking faces together contribute significantly to the stability of such structures in both static as well as moving robotic structures.

- **Reconfigurability** - The ability to reconfigure in modular robots can be visualized in three different perspectives as mentioned below -

  1. The sensor capabilities to recognize neighboring robotic modules for reconfiguration
  2. The docking capabilities to interface with another robot for formation of required co-ordinated robotic structure.
  3. Docking faces made available for forming different structures by manually, semi-autonomous and autonomous methods.

  Similar to the hybrid design(lattice and chain) approach followed by the researchers in robotic designs, the reconfiguration features in modular robots are also handled in a hybrid manner, making them difficult for assessment. Certain robotic designs emphasized only on providing various degrees of freedom for locomotion and enhanced the possibilities of forming numerous co-ordinated structures while ignoring the capabilities such as autonomous recognition of neighboring robots and electronically guided docking mechanisms. Though such designs appear to lack reconfiguration capabilities, numerous co-ordinated structures formed using docking faces demonstrates the possibility of utilization of a modular robotic design for multiple purposes in a given application. Such designs can be categorized as reconfigurable robotic designs. Certain designs are equipped with autonomous docking features using latches, hooks, soldering pads, etc. on their docking faces for restructuring. Such designs are commonly referred as self-reconfigurable robotic designs in the field of modular robotics in spite of their deficiency in recognizing the neighboring robotic modules for autonomous docking when they are placed well apart. Very few robotic designs are integrated with advanced sensors like image sensors for neighbor detection and docking, such designs can be categorized as self-reconfiguring as well as considered to be close to practical solutions instead of laboratory prototypes.

Few robotic designs are summarized below to provide understanding on their biomimetic as well as self-reconfiguration capabilities in more detail.

M^3 modular robot is an L shaped robot equipped with three wheels (one omni-directional and two regular wheels) that are capable of supporting mobility as well as formation of structures. Each wheel as shown in figure 3.2a supports mobility and also consists of sockets for docking with neighboring robotic modules. The wheels also aid in rotation and lifting of robots after successful docking. M^3[69] robotic module was prototyped in 2010 and it employs hooks for the sake of docking. M^3Express[70] prototyped in 2012 was morphologically similar to M^3 robotic module as shown in figure 3.3 and it employs internal magnets coupled with slip rings that control the position of magnets for docking and undocking.

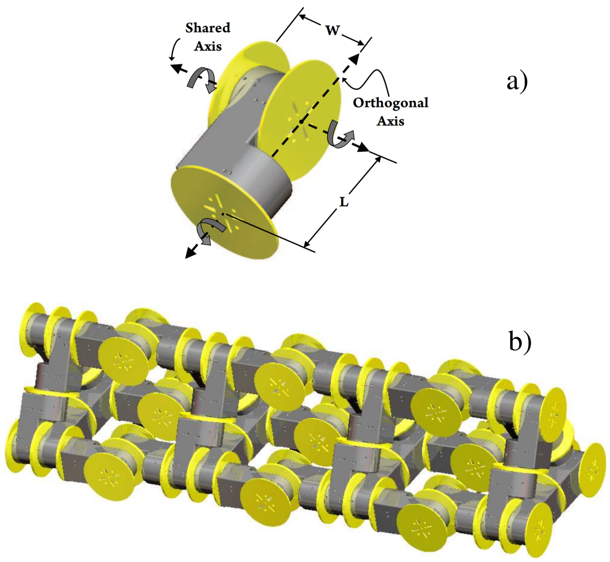


Figure 3.2. M^3 robot **a)** M^3 Design **b)** M^3 structures

The structures possible(shown in figure 3.2b) with M^3 and M^3Express robotic modules are very few due to limited possibilities in docking and absence of self-locking mechanism for retaining the structures. M^3 robotic doesn't facilitate formation of symmetric chain structures that are necessary to form spine while mimicking a organism. Another major disadvantage can be observed in the design in which there is no torque improvement for handling loads due to the coincidence of rotational axis of the wheels with rotational axis of the servo motors. Additional set of servo motors are also interfaced for implementing the docking mechanism and hence leading to increased form factor and power consumption.

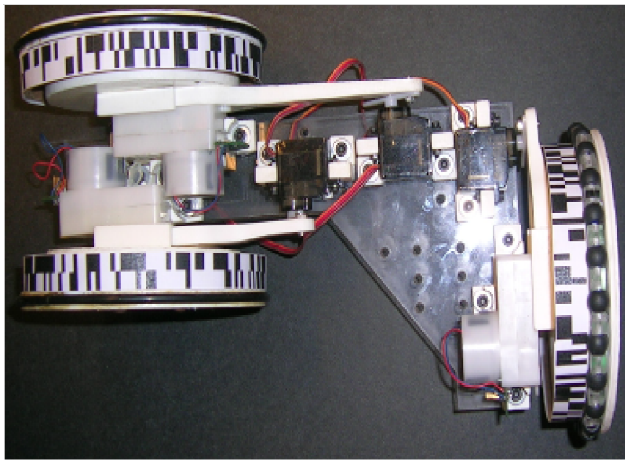
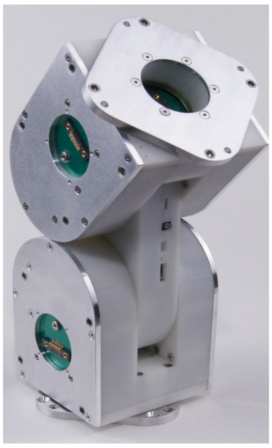


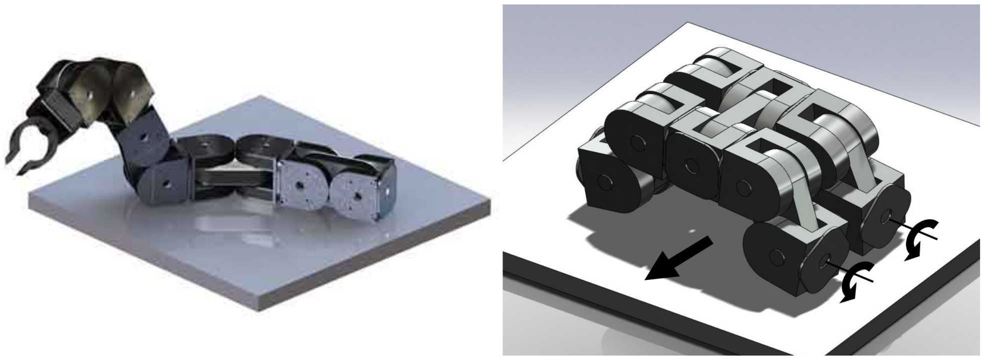
Figure 3.3. M^3 Express modular robot

iMobot[71] robotic module is an assembly of two semi-cylindrical structures as shown in figure 3.4a comprising of six faces for docking of which two faces can function as wheels for providing mobility to the robot. Numerous structures are possible with iMobot design due to it's four degrees of freedom and multiple faces for manual latching. Authors demonstrated the platform capabilities in locomotion such as crawling, folding, standing, rolling etc. The absence of self-locking can be visualized as a characteristic that leads to continuous power consumption in the design. The disadvantages of the iMobot design are similar to M^3Express robot along with zero autonomous features for docking and navigation.

CoSMO [116] robotic module is similar to SMORES in exterior design. The CoSMO robotic module is a cubic shaped structure with mechanical connectors mounted on four vertical walls of the cube and docking is guided by IR sensors. Three degrees of freedom is available with a CoSMO robotic unit of which two degrees are from mobility and one from pitch movement of a vertical face. The unique feature of CoSMO robot is its mobility which is enabled by



a)



b)

Figure 3.4. iMobot modular robot a) iMobot Design b) iMobot sample structures

two screwdrive type wheels providing omnidirectional movement and heterogeneous designs for achieving complex tasks of autonomous nature.

SMORES [73] hybrid robotic module consists of a three-wheeled cuboid structure similar to M^3 robot with four degrees of freedom as shown in figure 3.5. The front wheel apart from rotating can be lifted for generating pitch movement. Docking is facilitated and maintained by the magnets mounted on wheels and undocking is facilitated by rotation of wheel on neighboring module controlled(lock/unlock) by an internal shaft(shown in figure 3.6b and 3.6c) bringing magnets of same polarity face to face. A well designed gear train is mounted internally to regulate the torque and velocity ratios so that the rotational velocity of the side wheels is regulated and front face has double torque for upward/downward tilt when motors are controlled in synchronization.

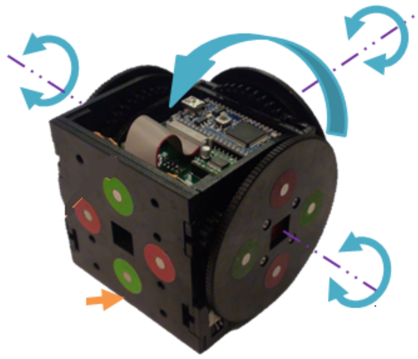
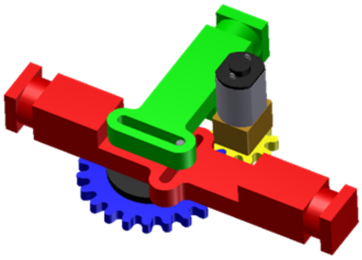
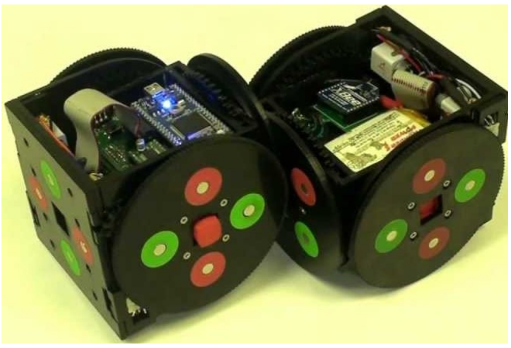


Figure 3.5. SMORES modular robot



a)



b)

Figure 3.6. SMORES modular robot a) Undocking mechanism c) SMORES docking

The SMORES-EP [119] robotic module is an enhancement to SMORES robot employing electro-permanent magnets on faces in place of permanent magnets. Though back-driving restriction is absent in the designs, the inertia of its gear train and motors provides non-zero resistance for external torques observed while forming structures. The possibilities of forming various structures using SMORES are numerous.

Trimobot[74] robotic module is unique due to its autonomous capabilities as compared to the other modular designs presented in this thesis so far. The robotic module as shown in figure 3.7a is equipped with five inactive faces and one active face for docking. The active face consists of rotating hooks for docking/undocking, can rotate to provide pitch movement and also equipped with camera module (not shown in figure) for object recognition and docking.

The docking mechanism on the active face consists of four spur gears activated by a motor connected to one of them and is shown in figure 3.7b. The hooks can be locked and unlocked as per the requirements when the robotic modules approach each other for docking. Trimobot proposes numerous improvements in terms of autonomous capabilities of robots due to employment of vision sensors as well as better navigation features due to omni-directional wheels present in robotic module. The sample structures proposed by authors using Trimobot robot modules are shown in figure 3.7c.

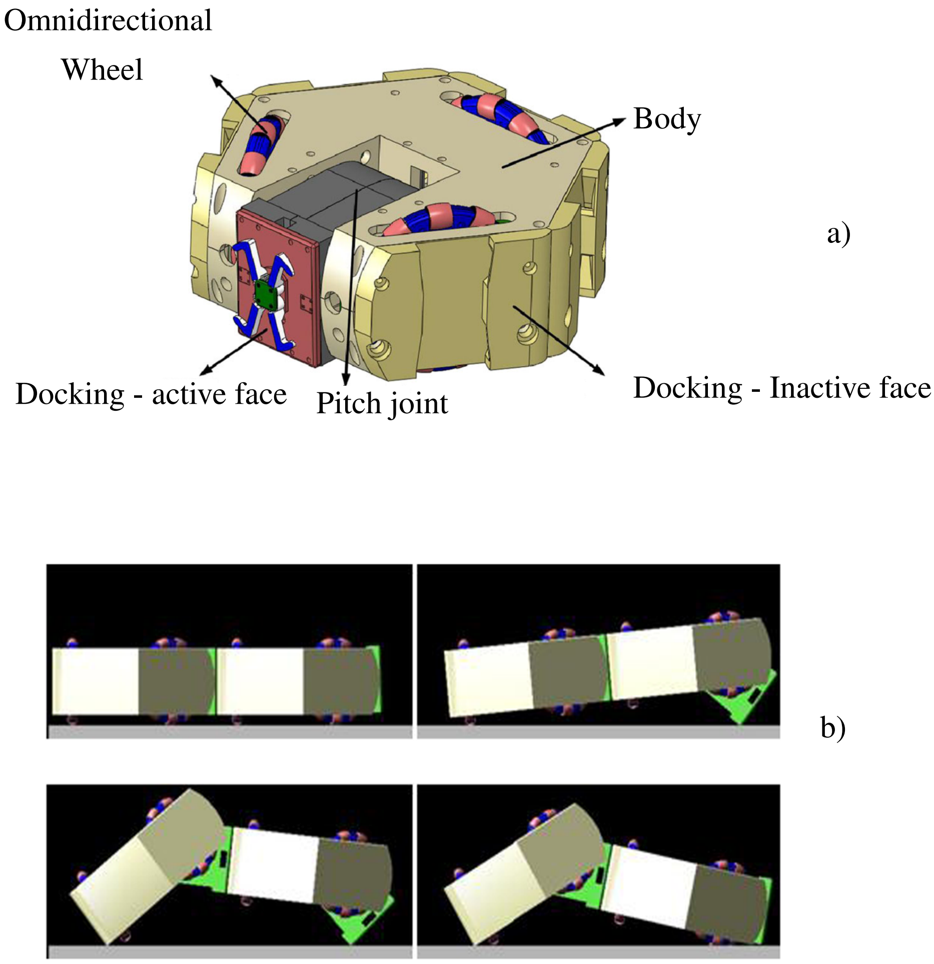


Figure 3.7. Trimobot modular robot - a) Design b) Docking mechanism c) Structures

The modular robotic designs provided in chapter 2 and summarized above lack back driving restriction mechanism in them. Though lattice designs doesn't face continuous power dissipation issues, such issues are obvious in chain and hybrid systems where energy needs to be supplied continuously for maintaining structures. The back driving restriction mechanism can also act as a fail-safe state apart from reducing the power consumption. The major drawbacks that can be inferred from the summary provided above is that - The M^3 robotic module due to its unsymmetrical design cannot be used for forming chain structures. The implementation of most of the vertebrate organismic structures is impossible due to this drawback as most of the mimicking is done by branching various arms from chain structure forming spinal cord. iMobot robot can form symmetric chain structures and hence can be used for mimicking. But the design is not inculcated with any docking mechanisms or any advanced sensor mechanisms for recognizing robots in the neighborhood. SMORES is an advanced modular design developed while accounting for many locomotion patterns. The setback in case of SMORES robot is in it's lack of back driving mechanism and mechanism for recognizing neighbors. it can also be observed that certain locomotion patterns will have to go through unnecessary movements for taking end effector to a position. Similar drawbacks can be observed in CoSMO robotic module. Trimobot is a totally self-configurable design due to its vision sensor and docking mechanisms but with a major setback in forming structures due to its chassis structure and DOF.

The modular robotic designs described in chapter 2 also suffer from similar drawbacks mentioned for the robots referred above. The HexaMob robotic module detailed in this chapter is a modular design(robot with chain formation capabilities equipped with mobility) with four degrees of freedom (2 degrees for mobility and 2 degrees for structural reconfiguration). The design addresses major requirements of modular robotics such as self-reconfiguration, homogeneity and more possibilities for structures as well as stability during locomotion. The following sections explain in detail about various design considerations and choices made while modeling the HexaMob robotic module.

3.3 HexaMob - Design

The motivation for initialization of modular robot is acquired from the S-bot project[4]. The S-bot is a fully integrated system capable of assembly, disassembly and autonomous navigation through guidance from various sensor network protocols. it is capable of overcoming obstacles such as steps and can navigate through certain troughs in rocks as shown in figure 3.8.

Figure 3.8. Applications of S-bot

S-bot is one of the few practical solutions developed for real-world application that can form lattice structures in 2D and 3D chain structures. In spite of being a completely integrated system, the robotic structures formed using S-bot still suffers from the drawbacks such as adaptability to different structures that are necessary for majority of real-world applications. The pitfalls in the S-bot can be identified as chassis structure and degrees of freedom. The gripper module present in the S-bot can be used for linking with the neighboring robots and can form a chain. Apart from gripping,the gripper can also link the robots and hence forming structures in 3D. It can be identified from the figure 3.8 that single DOF is present on each robotic module and because of the single degree of freedom the end effector can only have limited freedom in 3D space.

The external body parts of biological organisms are often non-spherical in structure and require different DOF at subsequent joint instead of greater DOF at a single joint. Experiments were performed on few prototypes after understanding biological organisms and demerits of S-bot for identification of issues in prototyping. The first robotic structure prototype is of square chassis that is 3D printed using ABS material with single passive claw for linking between the robots as shown in the figure 3.9. The side faces of the claw are connected to two servo motors each with torque capabilities of 3.2 Kg/cm operating in synchronization for providing double torques during lifting. Since the docking process is passive and and not designed to interface with specific sockets, the docking can be done from 3 sides of the chassis.

The linking between the numerous modules of first prototype is done manually and numerous experiments were conducted to identify synchronization between motors, power consumption and stability issues. The structures shown in figure 3.9 are generated using sequential activation of various servo motors using pulse width modulation. Major drawbacks identified in the design

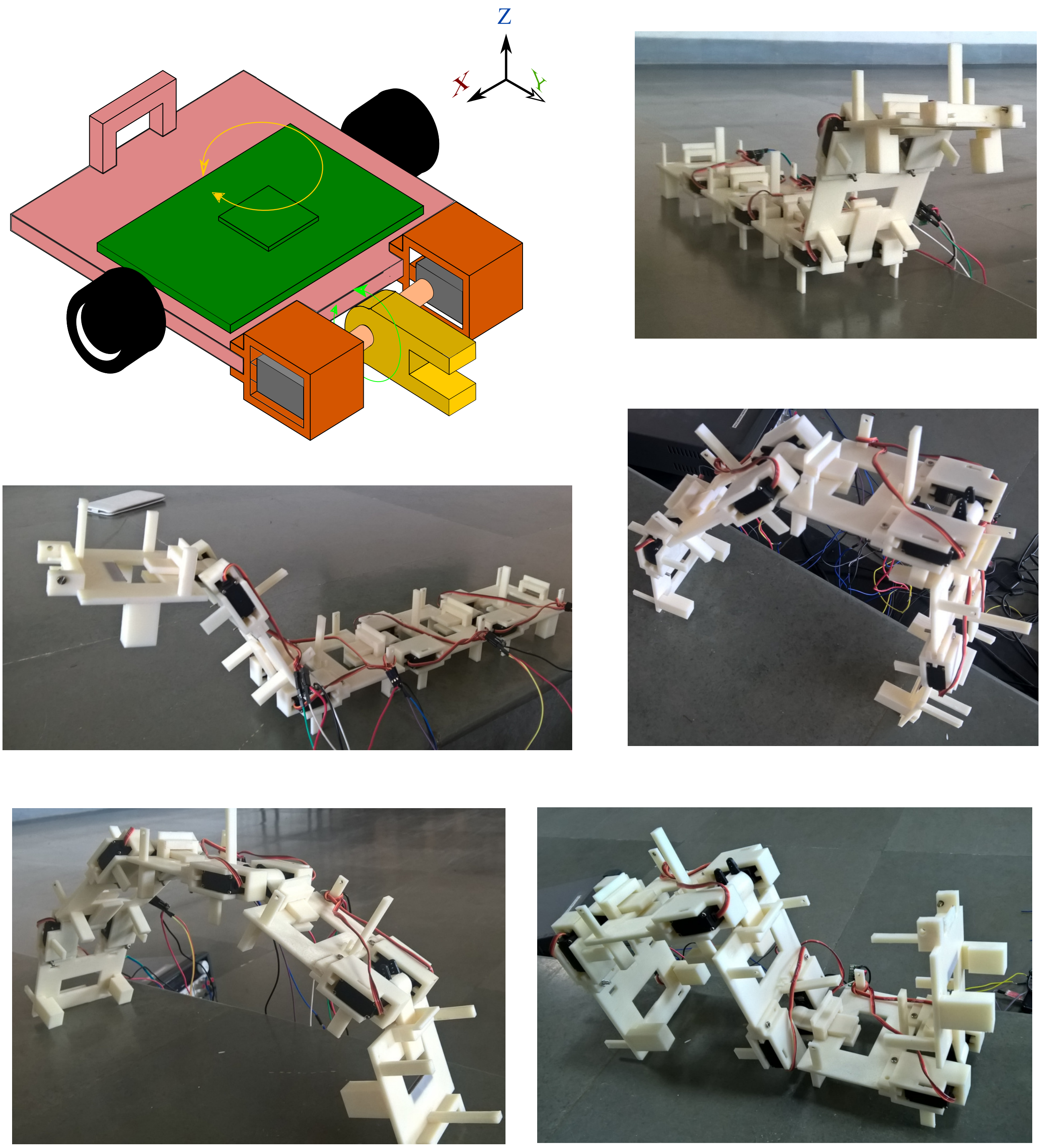


Figure 3.9. Prototype 1 - SQ-Bot

of first prototype apart from a large form-factor are - synchronization and stability issues. The servo motors fail to reach perfect synchronization due which they content for maintaining different position of the claw with respect to their rotational axis. The current consumption spiked in such situations from 50mA to 400mA and eventually leading to burning of servo motors.

Another demerit in design of the initial prototype is observed in docking technique. The docking is observed to be unstable as more and more loads are as there are no Velcros in place and also due to the tolerances left for loose fit between the inside faces of the claw. It can be visualized that the initial robot prototype can approximately meet the performance of Trimobot[74] at best with few performance issues after inculcating an embedded system with image processing capabilities and vision sensor.

The demerits of prototype 1 are corrected in prototype 2 using double claw mechanism instead of a single claw. The double claw mechanism is activated using two servos, each controlling a claw as shown in figure 3.10. Though the spikes in power consumption are reduced due to double claw mechanism, it also added few demerits to the robot. The form-factor of the prototype has increased due to placement of servos between the claws and the claws were observed to be diverging occasionally during operation outwards due to the absence of a common shaft and manufacturing defects in servo motors. This characteristic lead to instability in docking.

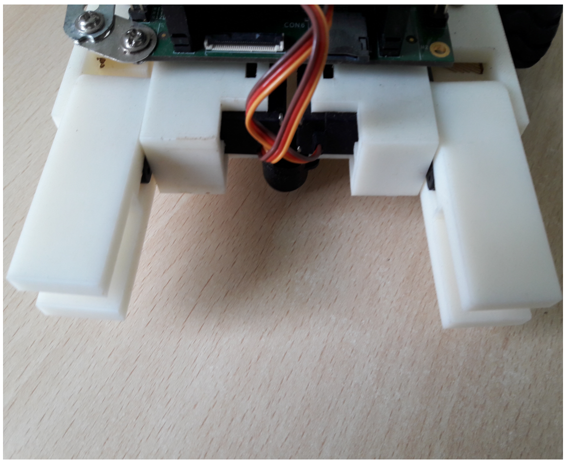


Figure 3.10. Prototype 2 - double claw

The third prototype - HexaMob is designed eliminate the drawbacks of the previous designs while adding an extra degree of freedom so that the robotic structure can form chain structure and can also be used as limbs of the biological structure for navigation. The major requirements excepted out of the HexaMob robotic modules are

- To facilitate DOF in orthogonal axes so that the gait models can be mimicked using the homogeneous modules.
- To provide torque improvements at all degrees of freedom

- To provide necessary sensor-actuator interfaces for reconfiguration.

The major challenges identified in implementing the requirements is to facilitate the torque improvements at the DOF that are often disregarded in modular robotic designs while also providing the back driving restriction for retaining the robotic structure in a fail-safe state. The identification of appropriate sensor actuator mechanism is also vital for the system to be functionally robust as per the application demands.

The HexaMob robotic module is designed to support the formation of chain and biomimetic structures in 2D and 3D. The hybrid category design of HexaMob along with its mobility support makes it a viable testbed for research in modular robotics. The design of HexaMob robotic module is shown in figure 3.11. HexaMob robotic module is an assembly of three separate sections - Front chassis, Back chassis, and Mobility unit. The front chassis is equipped with two claws rotated by a common shaft and rotation of the shaft is sourced by a worm-gear present in front chassis. The claws are capable of positioning robotic modules mounted on them from vertically upwards(90^o) to vertically downwards(-90^o). The front chassis also consists of docking sockets on the side faces for forming different structures. The back chassis consists of three docking faces including a face at the back along with sockets on side faces. A second worm gear mechanism is present at the center of the assembly of the front and the back chassis facilitating rotation around a vertical axis via a hinge mechanism. The DC motor and worm necessary for controlling the worm-gear present at the center of assembly are mounted in back chassis.

The design of the front/back chassis is made to resemble a hexagon structure both in the top and side views so that stresses on claws and body can be minimized in few chain structures. The robotic modules mount on chassis of each other at extreme angles of rotation and hence reducing the stresses on claws/hinges as proposed in [120]. The hinge mechanism coupled with worm-gear system present at the center of assembly provides precise control in navigation, docking and degrees of freedom for HexaMob during formation of various structures. The orientations possible with a single HexaMob robotic module upon activation of two worm gears (W.G 1 at the front and W.G 2 at the center) are shown in figure 3.12.

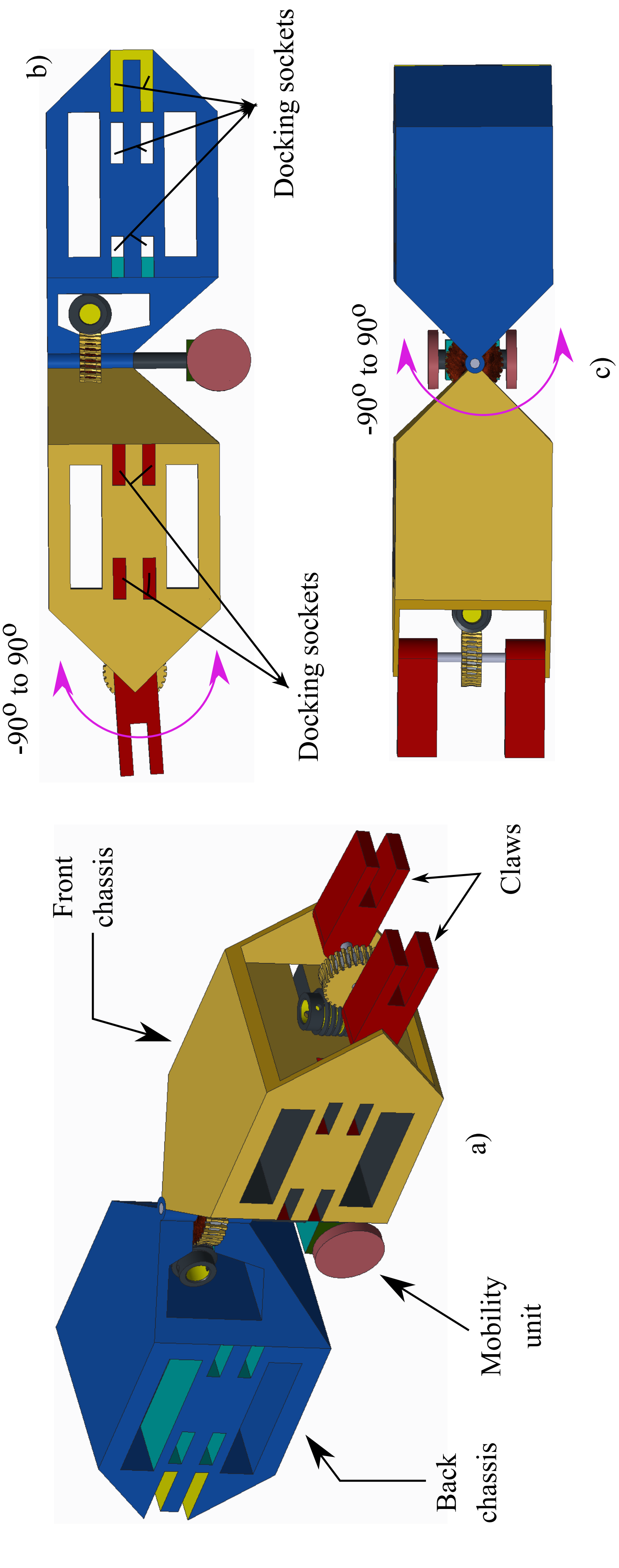


Figure 3.11. HexaMob robotic module. a) 3D view b) Top view c) Side view

3.4 HexaMob - Docking and Structures

The major merits of the HexaMob robotic module stem from the utilization of standard components in designing the robotic modules. HexaMob provides flexible possibilities for micro-sizing/ macro-sizing the design in relative to numerous modular robots due to the employment of claws and vision for docking and locomotion. The locomotion in/using HexaMob robotic module can be implemented by coordinated operation of a Twin-claw male interface actuated using worm-gear, a Hinge coupled worm-gear, Mobile unit, and the Vision system. The design choices for various mechanisms along with the possibilities of power and communication sharing are explained in further sections. HexaMob robotic module are implemented by

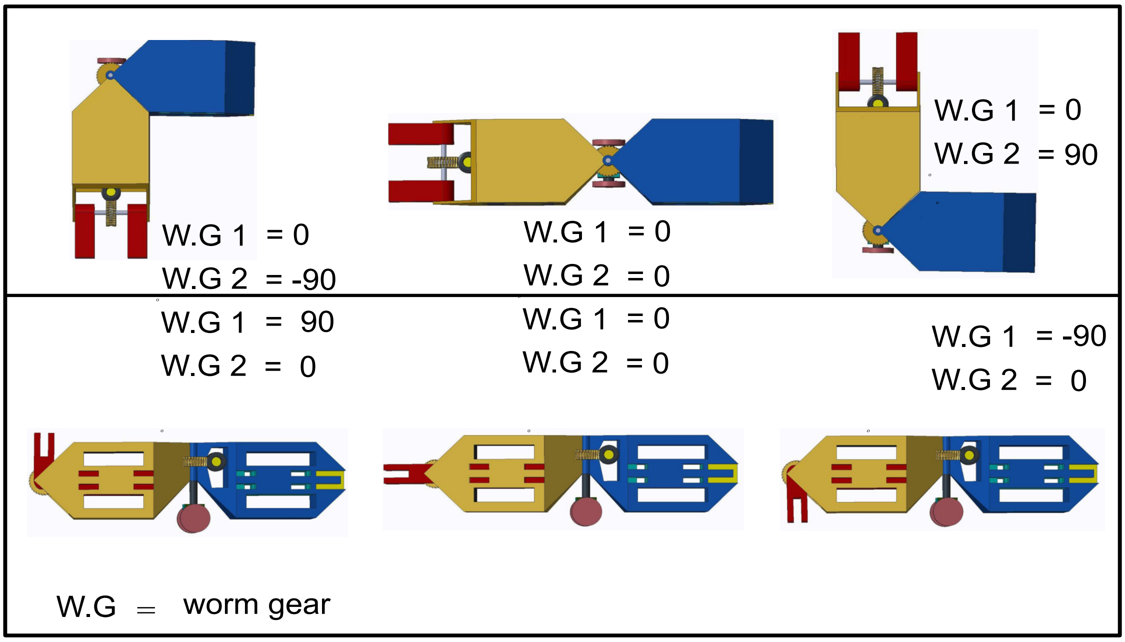


Figure 3.12. HexaMob - possible orientations from locomotion

coordinated operation of a Twin-claw male interface sourced using worm-gear, a Hinge coupled worm-gear, Mobile unit, and Vision system. The design choices for various mechanisms along with the possibilities of power and communication sharing are detailed below.

3.4.1 Twin-claw Mechanism

HexaMob robotic modules are designed to employ energy-less docking mechanism for forming structures. Numerous male interfaces for docking proposed in [120] are tested out using 3D printed models for identification of demerits and possible faults. The single claw mechanism in figure 3.13a is found to be unstable during locomotion due to wobbling and the torque generation using two servo motors operating in synchronization and sharing a common axis of rotation proved to be an unsuitable mechanism due to high power consumption issues. It has been observed that a slight mismatch in the assembly of the servo motor system (due to manufacturing defects of motors or assembly materials) can push current consumption limits of both servo motor to the maximum in spite of the absence of a load. The double claw mechanism shown in figure 3.13b is tested in the process of rectifying the power consumption issues and it has been found that such mechanism leads to structural faults and higher form-factor. In order to address issues such as continuous power consumption, latching without active parts while providing stability, and more accurate control of speed and rotation, the enveloped worm gear is chosen as a actuator mechanism due to its implicit locking of back-driving and torque improvement capabilities. The DC motor mounted in front chassis controls the position of claws using rotary encoders capable of measuring angles to the precision of 0.6^o.

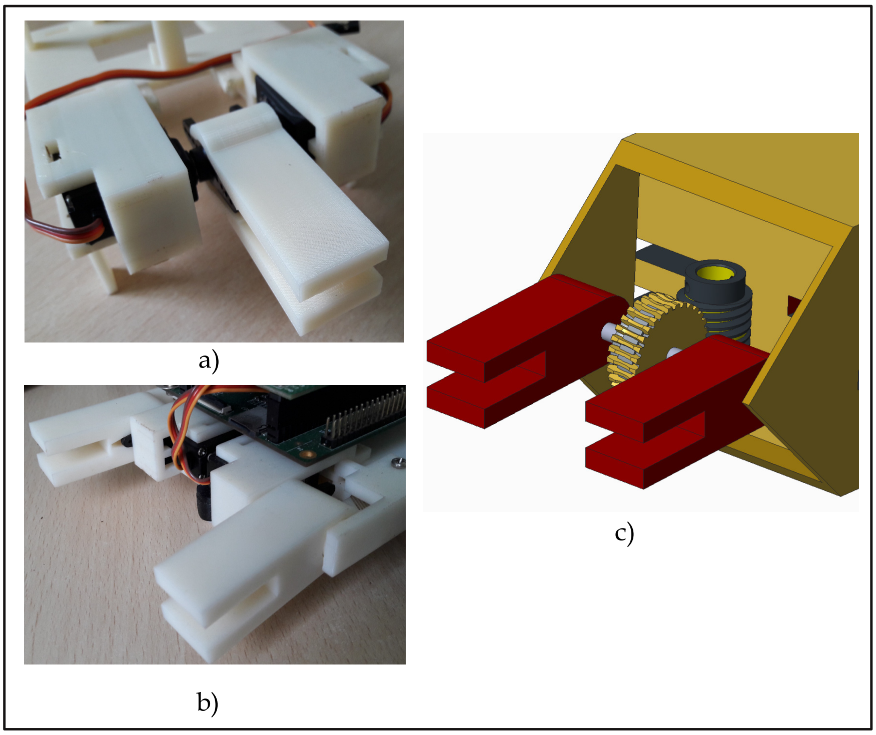


Figure 3.13. Docking Prototypes. a) Single claw - double servo b) Double claw - double servo c) Double claw - Worm gear

The docking between various HexaMob robotic modules is facilitated by claws (male) and five female faces (each equipped with 4 sockets on each face) as shown in figure 3.11. The latching between male and female parts is firmly maintained by teeth present on the internal faces of the claws and on internal faces of the sockets as shown in figure 3.14. The docking process is initiated by aligning the front faces of the claw parallel to female faces of the neighboring module in front of sockets. Due to implicit gaps provided in design for sliding, the claws slide into the sockets of a neighboring module with zero force. After sliding, adjusting the angle of orientation of the claw using its worm gear leads to locking of internal teeth present in claw and female faces.

The teeth together with thin films of Velcros (not shown in figure 3.14) placed on claw and sockets can provide a firm binding during docking and locomotion for lighter loads. Since the twin claw mechanism is sourced using an enveloping worm gear as shown in figure 3.13c and docking can be implemented without using energy, the HexaMob robotic module with its back-driving restriction capabilities from worm-gear maintains the structures with zero energy consumption.

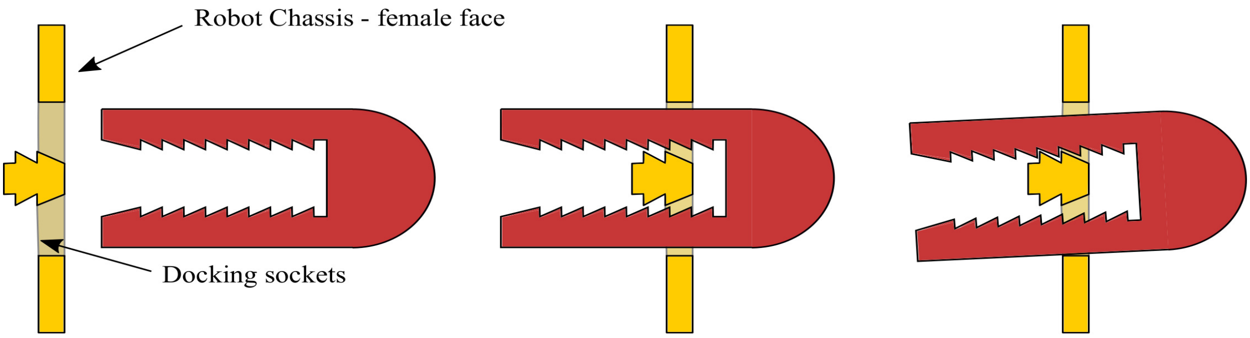
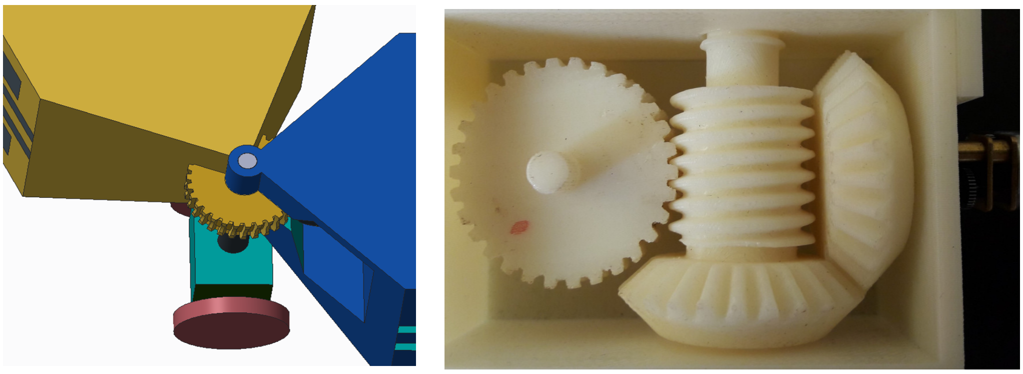


Figure 3.14. HexaMob - Latching mechanism between male and female interfaces

3.4.2 Hinge coupled Worm gear

The first degree of freedom in HexaMob design is facilitated by twin claw mechanism located at the front of the robotic modules. The second degree of freedom is enabled by the barrel hinge mechanism located at the center of HexaMob. The hinge mechanism is rotated by a worm gear whose driving motor is mounted in the back chassis. Since direct interfacing of a DC motor to the worm extends it to outwards (refer to figure 3.11b and 3.11c) of back chassis and obstructs the rotation, a bevel gear train is coupled with the worm gear for accommodating the motor in the back chassis. The gear train connecting the worm gear and bevel gears is shown in figure 3.15.

The gear in the worm-gear mechanism is an integral part of the front chassis design and worm controlling the rotation of gear along with bevel gears are placed inside the back chassis of



a) b)

Figure 3.15. HexaMob - worm and bevel gear assembly a) Worm gear location in HexaMob robot b) 3D-prototype for testing of back-drive restriction

HexaMob. The mobility unit increases the inertia on the back chassis of HexaMob since it is fixed to it and hence the front chassis will rotate relative to the back chassis upon activation of bevel and worm gears providing better control in navigation/docking. The back-driving restriction mechanism was tested successfully and hence continuous power consumption can be reduced to zero while maintaining the structures along with improved control over angular velocity of rotation.

3.4.3 HexaMob - Mobility

Mobility is a critical feature in limiting the human intervention or maximizing the automation in robotics. Numerous wheeled steering mechanisms are considered while designing the HexaMob robotic module. The steering mechanisms such as Skid steering, Tricycle drive, Synchronous drive, Omnidirectional drive and Articulated drive are found to be conflicting with form factor constraint of the homogeneous modular design. Miniaturized robotic models of HexaMob can have two actuators in the mobility unit due to small form factor requirement for non-obstructive rotation at the center. An option of distributing the mobility actuators to front chassis and back chassis is also not viable as it adds unnecessary complexity to the steering kinematics. Differential wheel drive mechanism is chosen for the implementation due to its advantages such as easy reverse steering, and simple turning mechanism during navigation. Since the axis of rotation of worm gear mechanism at the center and axis of rotation of the mobility unit (axis when two wheels rotate in opposite direction with equal velocity) are coincident, the steering kinematics and docking process are further simplified. The ease of control using differential drive mechanism can be observed in steering kinematics involving very few parameters as explained using figure 3.16. The rotation of a differential drive robot can be varied by controlling the linear velocity of the two wheels shown in figure 3.16. The point at which the robot rotates is called instantaneous center of curvature (**ICC**) present at a distance of R from the mid-point between two wheels. The relationship between linear velocities of the wheels and the angular velocity of the robot with respective to ICC are provided in the equations below.

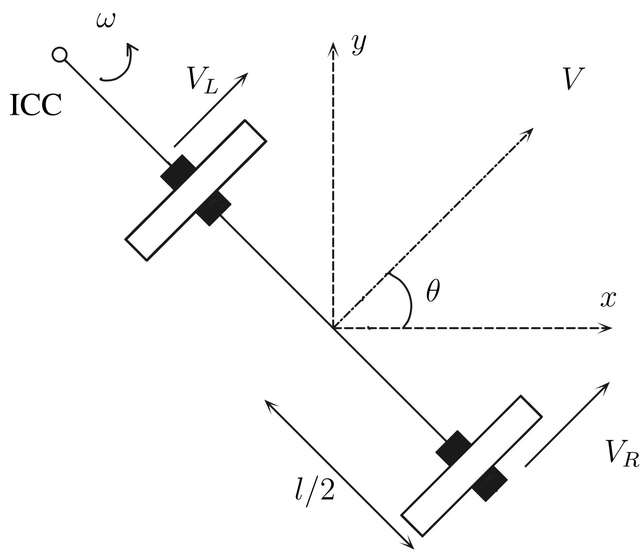


Figure 3.16. HexaMob mobility - Differential drive

$$\omega(R + l/2) = V_R \tag{3.1}$$

$$\omega(R - l/2) = V_L \tag{3.2}$$

where, ω is the angular velocity of rotation around ICC, l is the distance between the two wheels, V_L is velocity of left wheel and V_R is velocity of right wheel. The relationships can be further written as

$$R = \frac{l}{2}.\frac{V_L + V_R}{V_R - V_L} \tag{3.3}$$

$$\omega = \frac{V_R - V_L}{l} \tag{3.4}$$

It is possible to estimate the rate of change of position by counting the number of rotations made by the vehicle's drive wheels and prior knowledge of wheel's diameter. For a robot at a position

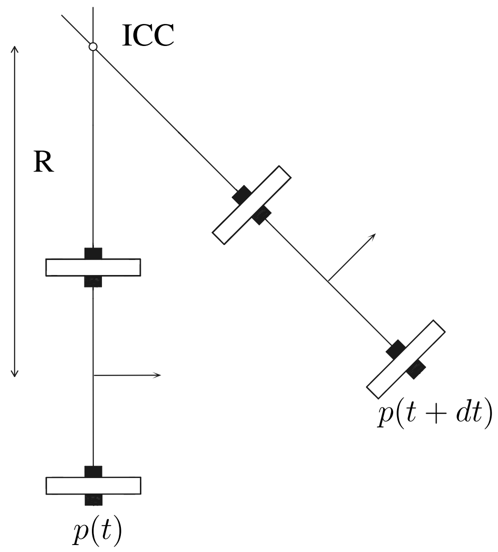


Figure 3.17. HexaMob mobility - path prediction using forward kinematics

(x, y) and facing a line making θ with the x-axis as shown in figure 3.16, it can be easily deduced for a situation that

$$(ICC_x, ICC_y) = (x - Rsin\theta, y + Rcos\theta) \tag{3.5}$$

At time $t + \delta t$ the pose of the robot as shown in figure 3.17 is given by

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} cos(\omega.\delta t) & -sin(\omega.\delta t) & 0 \\ sin(\omega.\delta t) & cos(\omega.\delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega.\delta t \end{bmatrix} \tag{3.6}$$

The major advantage of differential drive in HexaMob is co-incidence of it's axis of rotation with a DOF in the HexaMob robot. It provides an added advantage in docking as well as navigation for precise control of front chassis of the HexaMob. The mobility unit is fixed to the back chassis adding extra inertia to it, and castor wheels can be placed at suitable locations for providing stability to the HexaMob module. The motors can also be placed in back chassis such that the motor shafts can be extended into the mobility unit's chassis in parallel from which the wheels can be connected to rotational shafts using bevel gears while redesigning HexaMob for heavier loads and larger form factor.

3.4.4 Vision system

The Vision system is a guidance mechanism for docking and navigation of HexaMob robotic modules. Unlike conventional modular designs, HexaMob design aims to inculcate a capable embedded electronic platform into the mechanical structure similar to the trimobot robotic module for assisting in docking and navigation using vision sensor. The concept of docking when the robotic modules are in proximity to each other was explored rigorously by the employment of IR sensors [102][56][24] and magnets [13][26] in the domain of modular robotics and success of such docking process is also limited to few orientations of robotic modules with respect to each other. Vision sensors aids in overcoming major limitations in autonomous docking during reconfiguration in both 2D and 3D scenarios and also automating the self-aggregation and self-dispersion processes. The major requirements of the modular robotics electronic platforms such as low power consumption, various low power operational modes and reactiveness along with image processing capabilities are fulfilled by FlexEye platform[121] described in chapter 4 and the same platform is utilized(without transceiver) for identifying the error in alignment process during docking. The vision sensor recognizes an area of interest and calculates the alignment error between male and female interfaces with respect to its camera axis as shown in figure 3.18.

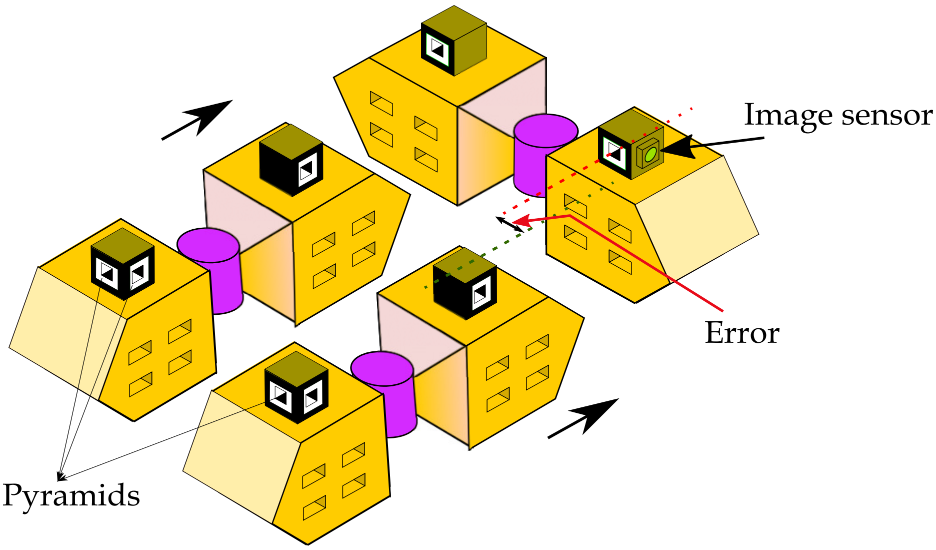


Figure 3.18. HexaMob robotic module - Alignment using vision sensors

Energy-less guidance mechanisms proposed using vision sensors are limited to 2D scenarios in which color processing or pattern recognitions are employed and such guidance mechanisms are ineffective in case of reconfiguration of modular robotics where the alignment errors are observed in all the three planes. A 3D object utilized for alignment reference whose image reflects unique errors for various alignments in 2D/3D plane is a apt substitute to conventional image processing mechanisms. A mini-sized pyramid mounted on top of HexaMob robotic module(shown in figure 3.18) with faces painted in different colors as shown in figure 3.19 can generate unique errors as per alignment and angle of approach.

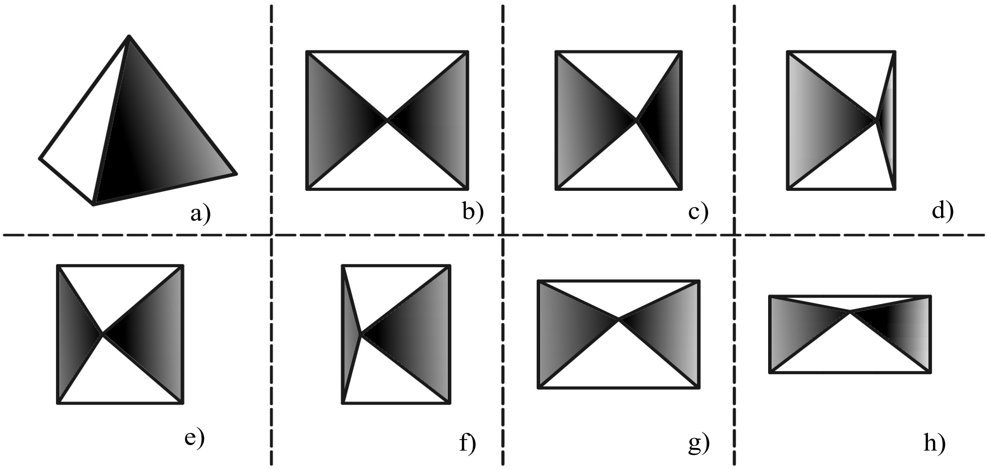


Figure 3.19. HexaMob robotic module - Alignment error detection using pyramid structure a) Pyramid structure with faces colored differently b) Appearance of pyramid on acquired image in case of zero error in alignment c) Appearance of pyramid on acquired image in case of error in alignment is towards left and zero error from top and bottom d) Appearance of pyramid on acquired image in case of error in alignment is large on left and zero error from top and bottom e) Appearance of pyramid on acquired image in case of error in alignment is towards right and zero error from top and bottom f) Appearance of pyramid on acquired image in case of error in alignment large on right and zero error from top and bottom g) Appearance of pyramid on acquired image in case of error in alignment is zero on left and right but non-zero error from bottom h) Appearance of pyramid on acquired image in case of error in alignment is zero on left and right but large non-zero error from bottom

Algorithm 1 implemented for recognition of error in alignment during docking process is a lightweight algorithm that recognizes a particular area of interest from a given image after converting it from a grayscale image to a binary image. The binary image is analyzed to recognize the presence of an area of interest - a rectangle with dark borders and relatively bright internal area. The center of the rectangle and its location with respect to camera axis provides the error in alignment. A solid of pyramid structure with 1 cm side and 1 cm height, colored black on alternate faces is glued with it's base parallel to the rectangular frame with dark borders as shown in figure 3.20 for testing the algorithm. The location of the apex of a pyramid in the image and area of bright and dark triangles provide information regarding the error in alignment with respect to the center of a image during docking when the male interface approach from various angles. The major advantage of using pyramid structure is its structural symmetry that also aids in docking when the robotic modules are floating in the air during reconfiguration.

Algorithm 1 Error detection and alignment

procedure ALIGNMENT
    Cnstrint ← *original length/original width*
    $m \leftarrow no.of\ rows$
    $n \leftarrow no.of\ columns$
    $image_Y[m][n] \leftarrow image[m * n]$
    $image_{bin}[m][n] \leftarrow$ binary($image_Y$,*thrshd*)
    $horztl_{edges}[n] \leftarrow$ scan($image_{bin}[m][n]$,*horzntl*)
    $vertl_{edges}[m] \leftarrow$ scan($image_{bin}[m][n]$,*vertl*)
    $locations[\,][\,] \leftarrow$ identify($horztl_{edges}, vertl_{edges}$)
    $Count_{Loc} \leftarrow$count($locations$)
loop1:
    if $Count_{Loc} > 0$ **then**
        $Count_{Loc} \leftarrow Count_{Loc} - 1$
        $Areas[\,][\,] \leftarrow$Traverse($locations, Cnstrint$)
        goto *loop1*.
        close;
    $Count_{Areas} \leftarrow$ count($Areas$)
    if $Count_{Areas} > 1$ **then**
        $i \leftarrow Count_{Areas}$
loop2:
        if $i > 0$ **then**
            $i \leftarrow i - 1$
            $image_{crop}[p][q] \leftarrow$image[$Areas[\,][\,], i$]
            $image_{cn}[p][q] \leftarrow$ binary($image_{crop}$,*thrshd*)
            $Areas_2[\,][\,] \leftarrow$ Traverse($image_{cn}, Cnstrint$)
            goto *loop2*.
        close;
    $Error \leftarrow$ Analyze($Areas_2$)

The algorithm searches for the rectangles in images and after successful recognition of rectangle(s), it searches for triangles and calculates their areas. The recognition starts with the identification of the rectangular frame to which pyramid is attached. After successful identification of frame from possible locations on the HexaMob, the pyramid projection is searched and analyzed for orientation and precise error. The identification of rectangular frame is possible in almost all cases except at the extreme conditions where the camera plane and the pyramid plane are orthogonal to each other or near to the orthogonal angles. The precise angle at which the recognition fails at extreme angles is capped only by the quality of the vision sensor and the resolution of a image acquired for processing. Since the angle between the pyramid plane and camera plane while approaching for docking cannot be 90^o or even close to angles such as 80^o, the recognition is almost always successful.

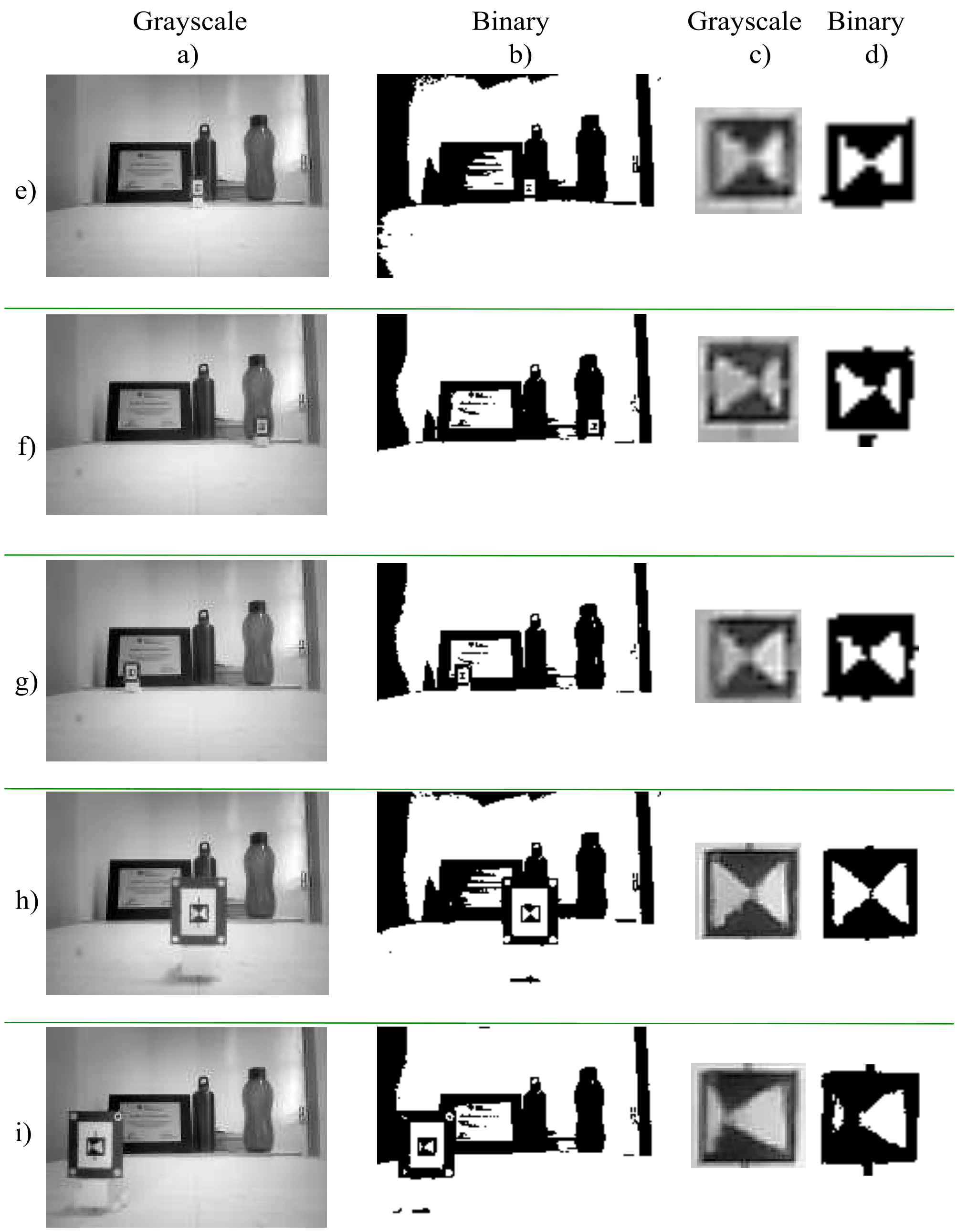


Figure 3.20. **a)** Acquired images **b)** Binary images from image processing **c)** Acquired high resolution images of pyramid for further analysis **d)** High resolution binary images of pyramid from image processing **e)** Scenario with target object aligned with vertical axis of vision sensor and placed at 65 cm distance from camera module **f)** Scenario with target object to the right of vertical axis of vision sensor and on horizontal line normal to vertical axis drawn at a distance of 65 cm from sensor **g)** Scenario with target object to the left of vertical axis of vision sensor and on horizontal line normal to vertical axis drawn at a distance of 65cm from sensor **h)** Scenario with target object at center of camera axis and on a perpendicular line drawn at 20 cm from vision sensor **i)** Scenario with target object to the left of vertical axis of vision sensor and on horizontal line normal to vertical axis drawn at a distance of 20 cm from sensor

The method of calculating the area of triangles in different scenarios shown in figure 3.20 provides better results as the robotic modules move closer to each other and hence assisting in precise docking. The recognition was successfully tested in from the distance of 65 cm to 5 cm from the vision sensor. The rotation of front claws and center hinge controlled by worm gears provides precise control in both horizontal and vertical planes and hence the HexaMob docking process on five female sides can be completed in both 2D and 3D scenarios with the help of five pyramids each mounted towards the sides as shown in figure 3.18.

3.4.5 Power and Communication sharing

The majority of lattice structured and few chain structures modular robotic designs prototyped so far implemented power and communication channel sharing. The feature of sharing is implemented by designing docking interfaces embedded with metallic contacts aiding in connecting the batteries on multiple robotic modules using a common two-wire bus. Similarly, another two-wire bus was also made available at the same interface for sharing a communication bus for enabling locomotion and reconfiguration. The HexaMob robotic module establishes two contacts at each claw and hence there are four contacts in total with each neighboring robot after docking. The power and communication bus sharing can be implemented(if necessary) in HexaMob robotic units by placing thin uninsulated copper lines from batteries and microcontroller communication peripherals stretched over the length of internal faces of claws as shown in figure 3.21.

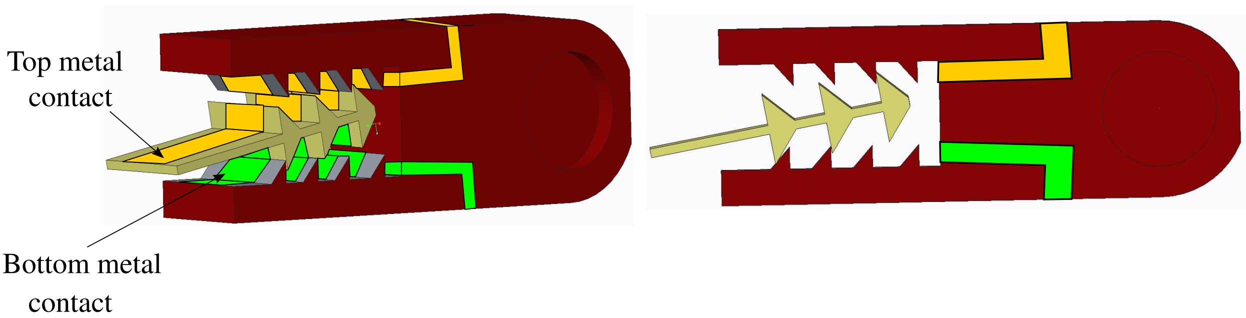
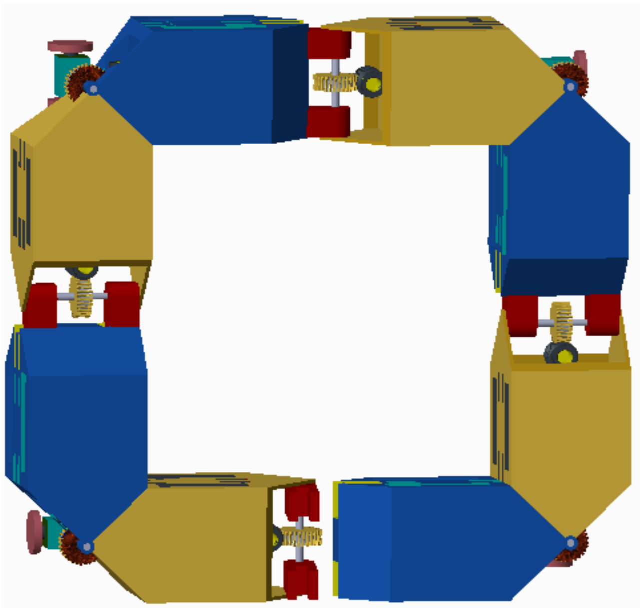
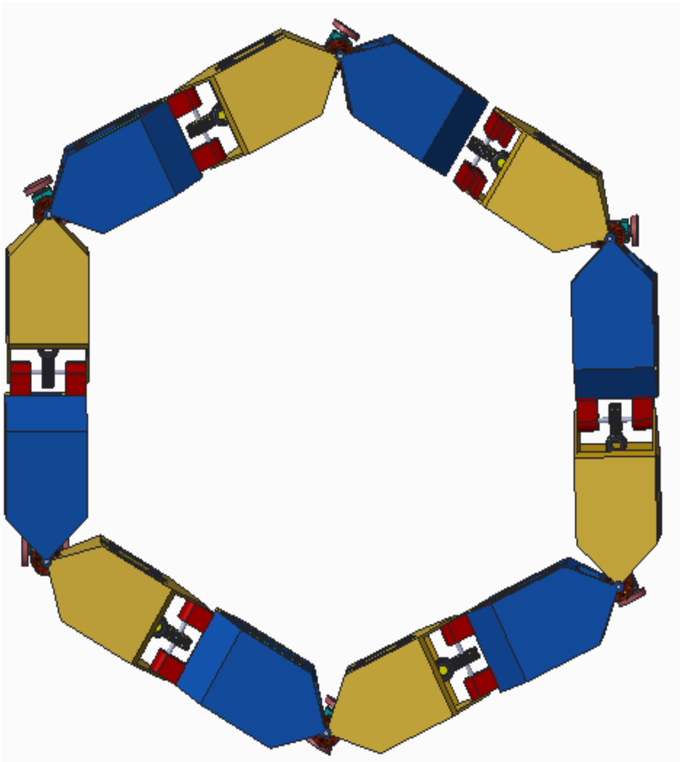


Figure 3.21. HexaMob robotic module - Power and communication sharing

Numerous features embedded into the design of HexaMob robotic module aids in formation of multiple structures autonomously. The HexaMob design also made it possible to mimic biological organisms such as centipede and vertebrates with relative ease in docking and locomotion.

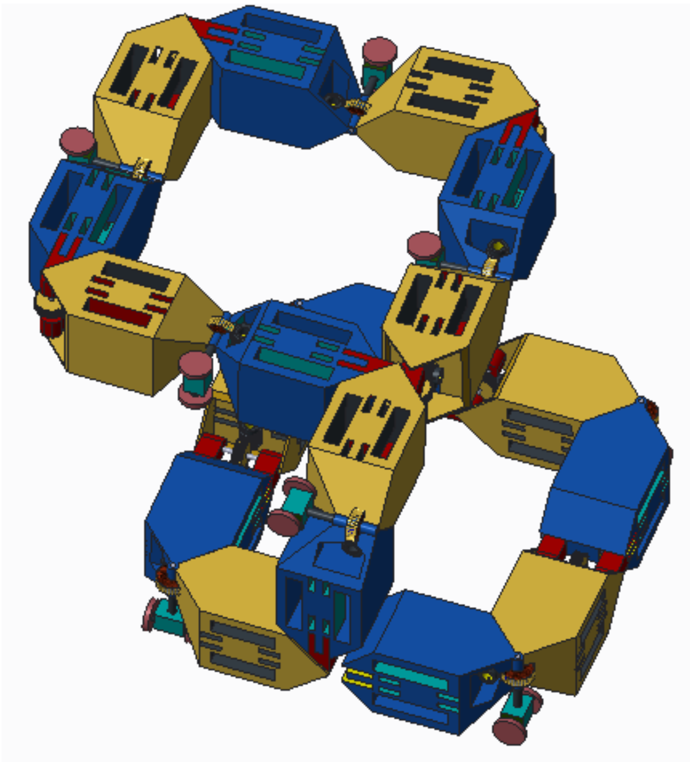


a)

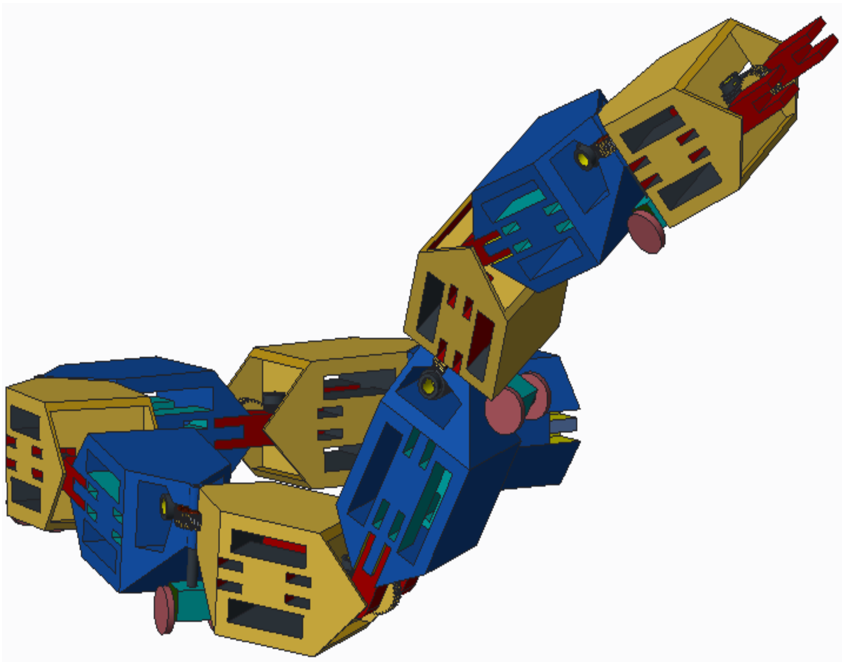


b)

Figure 3.22. HexaMob structures a) Square Lattice(Top view) b) Hexagon Lattice(Top view)

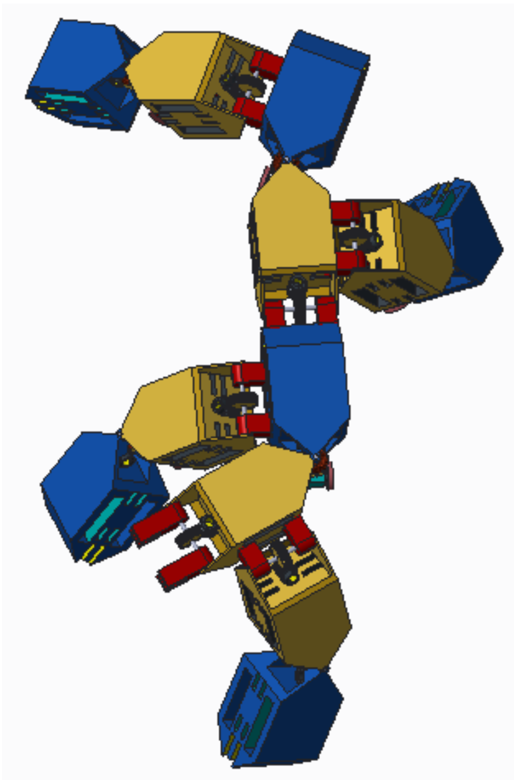


a)

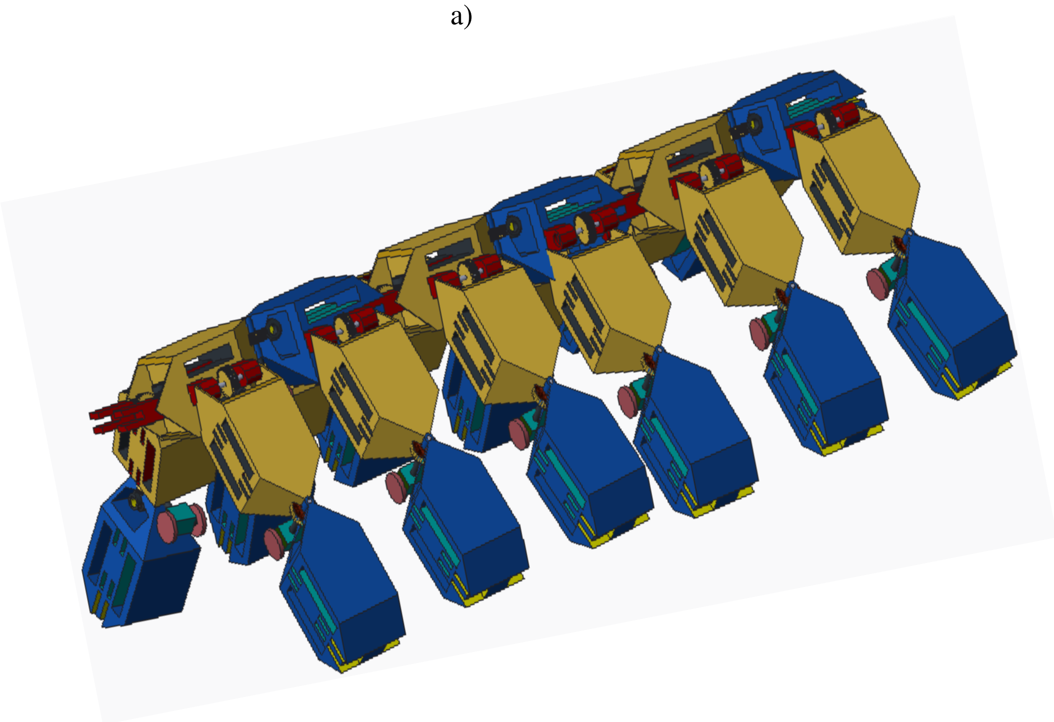


b)

Figure 3.23. HexaMob structures a) 3D stacked Square Lattice b) Chain structure



a)



b)

Figure 3.24. HexaMob structures a) Vertebrate structure b) Centipede

3.5 Path to implementation

The HexaMob robotic unit is designed with enough spacing for integrating the embedded control unit into the system for production of a complete robotic module. The top and bottom faces of the HexaMob can be replaced with PCB boards with necessary electronics for facilitation of communication and navigation capabilities. The space between sockets and the top/bottom faces can be utilized for mounting Li-Po batteries, other sensors during macro-sizing etc. Major design time is expended on analysis and modeling the worm gears as per the application requirements. Since worm gears suffer from friction present between materials participating in sliding motion, the choice of materials plays a vital role in efficiency and performance of system and hence the performance of HexaMob robotic module. The self-locking capabilities of worm gear for parameters shown in figure 3.25 can be observed at condition[122]

$$f_{stat} > cos\phi_n tan\lambda \tag{3.7}$$

where,
f_{stat} represents static frictional co-efficient
$\boldsymbol{\lambda}$ represents lead angle
$\boldsymbol{\phi_n}$ represents normal pressure angle(not shown in figure 3.25).

Various design parameters of worm gear mechanism can be calculated using the following equations

$$pitch\, diameter\, d_G = \frac{N_G.p_t}{\pi} \tag{3.8}$$

where,
$\boldsymbol{N_G}$ represents number of teeth in gear
$\boldsymbol{p_t}$ represents transverse circular pitch.
In general, the pitch diameter of the worm 'd_w' will be selected to fall into the range for optimum horse power capacity of the gearset.

$$\frac{C^{0.875}}{3.0} \leq d_w \leq \frac{C^{0.875}}{1.7} \tag{3.9}$$

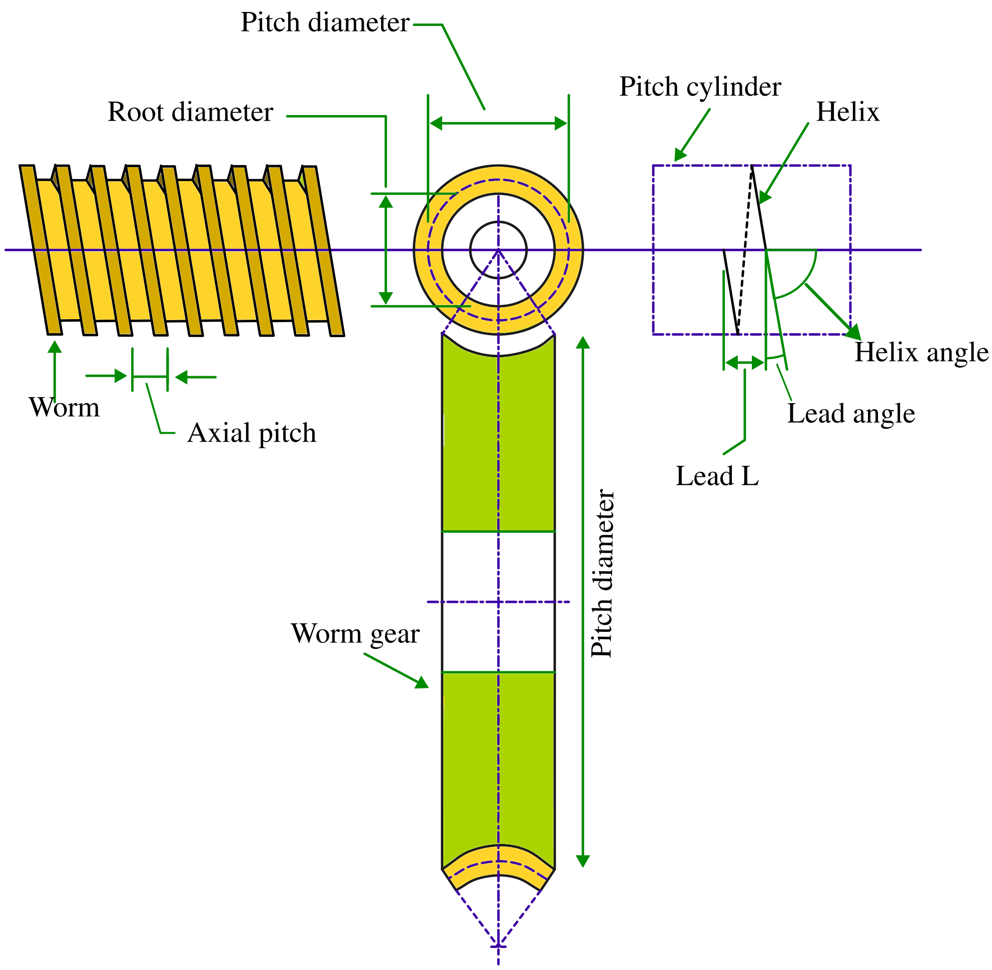


Figure 3.25. Worm gear design parameters

where,

C represents center distance

The lead 'L' and the lead angle 'λ' of the worm have following relations:

$$L = p_x.N_W \tag{3.10}$$

$$tan\lambda = \frac{L}{\pi.d_w} \tag{3.11}$$

where,

$\boldsymbol{p_x}$ represents axial pitch.

$\boldsymbol{N_W}$ represents number of teeth in worm

The efficiency of worm gear mechanism can be calculated using

$$\eta = \frac{cos\phi_n - f.tan\lambda}{cos\phi_n + f.cot\lambda} \tag{3.12}$$

The output torque 'T_{out}' can be calculated in absence of friction as

$$T_{out} = \frac{N_G}{N_W} * T_{in} \tag{3.13}$$

It can be observed from the relations above that the frictional co-efficient plays a vital role in efficiency along with enabling the self-locking mechanism. The kinetic friction of various materials at numerous sliding angles are available in [123]. American Gear Manufactures Standard association also made available optimal tables for numerous lead angles and normal pressure angles. It is possible to determine the efficiency and output torques for a given worm-gear specifications and suitable material as per the design requirements. The HexaMob robotic module apart from worm-gear mechanism(s) uses standard components for implementing the structures and hence provides relative ease in prototyping and research.

3.6 Results and discussions

A comparison on a number of modular robotic designs equipped with the self-mobility feature is provided in table 3.2 and table 3.3. The unsymmetrical design of M^3 robotic module poses some constraints in forming structures in spite of having a precise docking mechanism and better navigation system. iMobot robot provides better mobility and numerous possibilities for the formation of structures but exhibits less autonomous nature due to the absence of docking and undocking mechanisms. SMORES is a sophisticated modular design equipped with docking and undocking capabilities and lack autonomous features due to lack of sensors for navigation and docking. Trimobot is unique in the perspective of enhancement of navigation and docking capabilities including omnidirectional mobility but has limitations in different structures possible with the robotic module.

The HexaMob robotic design is unique in terms of its back-driving restriction capabilities with zero energy consumption as well as no extra actuators. The design of HexaMob is made with the

Table 3.2. Comparison of hardware features in modular robotic designs

Robot	Shape	Docking		Connection		Mobility	Ref.
		Interface	Actuator	Active	Inactive		
M^3	'L' shaped	Hooks	DC motor	3	0	Omni	[69]
M^3express	'L' shaped	Latch	SMA, Servo	3	0	Omni	[70]
iMobot	Cuboid	Latch	Manual	0	6	Differential	[71]
SMORES	Cube	Magnets	DC motor	3	1	Differential	[73]
Trimobot	Hexagonal	Hooks	DC motor	1 Male	5 Female	Omni	[74]
CoSMO	Cube	Lock	DC motor	4	0	Omni	[116]
Scout	Cube	Lock	DC motor	4	0	Tracks	[112, 113]
Sambot	Cube	Hooks	DC motor	1 Male	5 female	Differential	[111]
HexaMob	Hexagonal	Claw	Worm gear	1 Male	5 Female	Differential	

Table 3.3. Comparison of reconfiguration capabilities in modular robotic designs

Robot	Degrees of freedom	Bck.-drvg. rstrctn. capabilities	Pwr. shrng capabilities	Reconfigure Atnms. features	Ref.
M^3	3D	✘	✘	⊙ † ‡⋆	[69]
M^3express	3D	✘	✘	⊙ † ‡⋆	[70]
iMobot	3D	✘	✘	⊙ ‡ ⋆	[71]
SMORES	3D	✘	✘	⊙ † ‡⋆	[73]
Trimobot	3D	✘	✘	† † ‡	[74]
CoSMO	3D	✘	✓	⊙ † ‡⋆	[116]
Scout	3D	✘	✓	⊙ ⊙ † † ‡⋆	[112, 113]
Sambot	3D	✘	✓	⊙ † ‡	[111]
HexaMob	3D	✓	✓	⊙ ⊙ † † ‡	

⊙ - Biomimetic capabilities, † - Autonomous features, ‡ - Chain structures, ⋆ - Lattice structures

perspective of minimizing the stresses on various internal components during configuration and locomotion while meeting all constraints of modular robots. Numerous interfaces on HexaMob increase the capabilities of the design in forming various structures and the back-driving restriction feature in locomotion improves the longevity of system by reducing the power consumption. The back-driving restriction mechanism of the worm gears aids in reducing the continuous power consumption of DC/servo motors that are commonly employed as actuators for locomotion and reconfiguration in modular robots. Though frictional losses are more in worm gear mechanisms, such losses can be reduced by employment of appropriate lubricants. The energy consumption because of each electric actuator is given as,

$$E = \int_0^t V * I * dt \tag{3.14}$$

where V= voltage at which the DC/servo motor is operating, I = current through the DC/servo motor and t is the time period of operation.

It can be observed from equation 3.14 that as time progresses, the effort in maintaining a joint in position in a modular robot with no back-drive prevention mechanism leads to an increase in the energy consumption. The total losses due to continuous power consumption in each modular robot keeps accumulating and can be realized using following relationship.

$$Energy\, losses = \sum_{x=1}^{n} E(x) \tag{3.15}$$

where n = maximum number of actuators participating in maintaining the structures. The sophisticated and complex locomotions involves numerous such modular robots to participate in coordination and hence the cumulative losses only due to actuation of motor/servos can be observed as

$$Total\, energy\, losses = \sum_{y=1}^{m} \sum_{x=1}^{n} E(y, x) \tag{3.16}$$

where m = maximum number of modular robots participating in maintaining the co-ordinated structure.

It can also be identified that irrespective of whether a coordinated structure is mobile or immobile, the losses keep accumulating in case of lack of implicit back-drive prevention mechanism at the joints providing various DOF and such losses are substantial when compared with frictional losses observed in the worm-gear mechanisms. Along with the advantages of back-drive prevention, the worm gear systems also provide torque improvement with minimal space occupancy as per the relationship below.

$$T_{out} = \frac{N_G}{N_W} * T_{in} \tag{3.17}$$

Where,T_{out} = output torque, N_G = Number of teeth present on gear, N_W = Number of teeth present on worm and T_{in} = input torque.

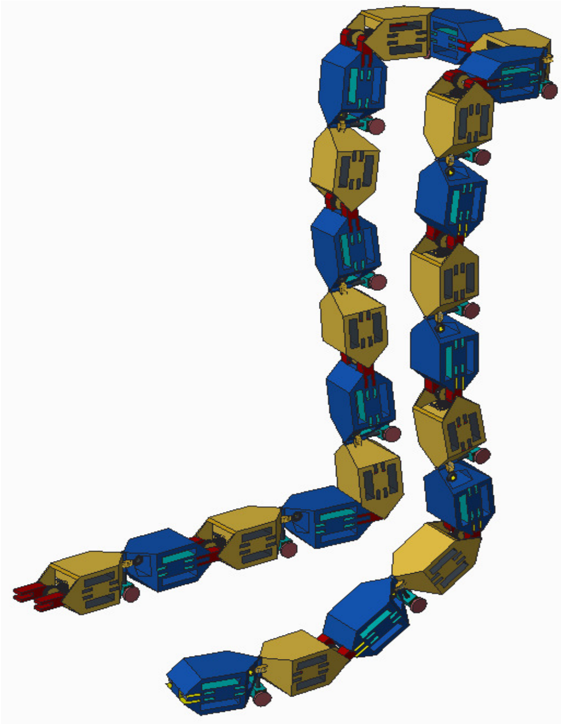


Figure 3.26. Hexamob coordinated gait structure

The torque gain is achieved in minimal form factor due to the use of worm gear which is advantageous while scaling down the design for producing smaller robotic structures. The implicit torque improvement due to the wormgears can be inferred as an additional advantage because of which implementation of gait structure shown in figure 3.26 is also feasible and hence HexaMob facilitates formation of biomimetic structures better as listed in table 3.3. The actuators proposed to be employed in the design for locomotion are micro-metal gear motors with extended shaft for mounting magnetic rotary encoders to monitor parameters such as angular velocity and acceleration. The 16-bit timers present inside the microcontroller of FlexEye module provides numerous features that aids in monitoring and controlling the actuators in real-time.

The HexaMob robotic module can be easily micro-scaled and macro-scaled due to its design simplicity and modularity. Due to the independence of docking mechanism using vision sensors with other internal elements of design, the vision based alignment error detection mechanism for docking can be re-utilized without major modifications in the scaled robotic designs.

3.7 Summary

HexaMob modular robotic design can be inferred as an energy efficient modular architecture with greater improvement in reconfiguration and autonomous capabilities due to the following features.

- An implicit back-driving prevention mechanism with no extra actuators proposed in HexaMob design leads to improvement in the lifetime of the robot. Energy conservation in the co-ordinated structure also improves by several fold during locomotion because of the back-driving prevention and simultaneous torque improvement. The absence of extra actuators for facilitating the back-driving prevention is an added advantage and it provides relative ease in up-scaling and down-scaling of robot.

- Visual sensors are employed for docking between the HexaMob robotic modules. The advanced features such as autonomous navigation can be added to the robotic structures using vision sensors and vision based algorithms are reusable in scaled robotic designs without major modifications. Each HexaMob robot can function independently in decision-making as they are designed to be equipped with a mobility unit, a vision sensor and a capable electronic processing unit.

- Numerous coordinated structures can be formed using HexaMob due to improved torques at worm gears in small form factor. The torque improvement provides increased capability in carrying the loads in locomotion and hence improving the suitability of HexaMob in real-world applications.

Chapter 4

FlexEye - A Flexible Camera Platform for Low-Power Distributed Sensing

4.1 Introduction

The crucial traits of MSRR such as re-configuration and coordinated locomotions are achieved by continuous coordination of sensor-actuator assemblies with electronic processing platforms and constant communication between robotic modules. Digital control platforms embedded with advanced sensors like vision and ultrasonic transducers can aid in navigation as well as self-reconfiguration capabilties of a robotic system. The microcontrollers present on the electronic platforms when coupled with suitable algorithms attain higher efficiency in performance in the perspective of reactiveness, speed, and power consumption. Precise control of various processes in reconfiguration and communication in robotic applications is feasible due to the latest advancements in electronic platforms and a through analysis on electronic platform is necessary before finalizing the same for a robotic application.

The objectives of the research described in this chapter are

- To prototype an energy efficient embedded platform for modular robots for
  - Rapid image acquisition and processing for enabling self-reconfiguration of robots.
  - Providing high-precision internal peripherals for interfacing sensors and actuators.
  - Facilitating rapid wireless communication for facilitating centralized and decentralized control and monitoring.
  - Facilitating large storage capabilities suitable for distributed sensing applications.

The microcontroller boards embedded in MSRR platforms that have been described in chapter 2 of the thesis are custom-developed boards primarily designed as per the constraints such as form factor of design, the number of sensors and actuators and algorithm complexities. The communication capabilities of majority of the MSRR designs are limited to wired communication with neighboring modules in a coordinated structure. It can be realized that autonomous capabilities of individual robots are restricted due to lack of wireless communication capabilities in robots for aggregation. The development of a completely integrated MSRRs is a multidisciplinary research work and can adopt crucial traits of WSNs, MWSNs such as power and communication optimizations in both centralized and decentralized scenarios for improvement in efficiency. The WSNs and MWSNs provide optimized hardware solutions for wireless communication so that various centralized/decentralized scenarios can be realized at a minimal trade-off in power consumption. A summary of various static WSNs and MWNs motes are provided in table 4.1 and table 4.2 respectively for facilitating better comparison on hardware capabilities of the wireless platforms.

The current chapter in this thesis elaborates on a hardware prototype and its performance in power consumption and image processing. As a prototype, a camera platform suitable for WSNs, MWSNs and swarms is developed using an optimum choice of the commercial off-the-shelf(**COTS**) components and compared with the existing computationally intensive hardware architectures. The prototype is henceforth referred as FlexEye. The details on various visual camera motes**(VSMs)** developed in the field of WSNs are explained in brief in section 4.2. The section 4.3 provides details of the FlexEye hardware platform implemented using COTS for proof of concept. The section 4.4 describes the software architecture implemented in the FlexEye platform. An analysis on the performance in power consumption and image processing capabilities of the FlexEye platform is presented in section 4.5 and the conclusions are provided in section 4.6.

Table 4.1. Comparison of WSNs motes

Mote Features							
Mote Name	**Processor**	**8/16/32 bit**	**Freq. MHz**	**RAM KB**	**ROM KB**	**Radio**	**Ref.**
MICA 2	ATmega 128L	8 bit	7.37	4	128	CC1000	[124]
MICA 2 DOT	ATmega 128L	8 bit	4	4	128	CC1000	[125]
MICA Z	ATmega 128L	8 bit	7.37	4	128	CC2420	[125]
IRIS	ATMega1281	8 bit	7.37	8	128	AT86RF230	[126]
TELOSB	TI MSP430	16 bit	8	48	10	CC2420	[127]
IMOTE2	PXA271 XScale	32 bit	416	256	32000	CC2420	[128]
CRICKET	ATmega 128L	8 bit	7.37	4	128	CC1000	[129]
LOTUS	ARM7 Cortex M3	32 bit	100	64	512	RF231 Atmel	[130]
STARGATE	Intel PXA255	32 bit	400	64000	32000	No radio	[131]
T MOTE	TI MSP430	16 bit	8	48	10	CC2420	[132]
Oracle SPOT	ARM 926ej-S	32 bit	400	1000	8000	CC2420	[133]

Table 4.2. Comparison of MWSNs testbeds

Testbed Features						
Testbed	**Core Processor**	**Freq. (MHz)**	**RAM**	**ROM**	**Radio**	**Ref.**
Indoor Mapping Robotic Testbed (IMR)	Intel ATOM	1660	2 GB	Expandable	802.11g router	[134]
MINT-M	Uses Roomba robot	NA	NA	NA	IEEE 802.11 a/b/g	[135]
Mobile Emulab	Stargate Board	400	64 MB	32MB	MICA 2 mote radio Wi-Fi radio	[136]
Kansei	Stargate Board	400	64 MB	32 MB	Tmote Sky radio Wi-Fi radio	[137]
CoNet	Intel Core 2 Duo	2260	8 GB	120 GB	Wireless a/b/g/n Bridge MICA 2 nodes	[138]
ISROBOTNET	Intel Core 2 Duo	2260	8 GB	120 GB	Wi-Fi radio	[139]
iRobotSense	IRIS motes	7.37	8 KB	128 KB	AT86RF230 Transceiver	[140]

4.2 Related work

VSMs are custom designed hardware platforms with on-board CMOS sensors supporting image processing. Apart from the image processing capabilities, the hardware platforms are designed to satisfy the requirements like minimal power consumption, form factor reduction, large memory, more GPIOs for future proofing etc. A custom-made embedded platform can be compared with VSMs in terms of performance and efficiencies as they are optimized in numerous parameters and capable for real-world applications. The details on few VSM platforms are listed below.

4.2.1 Cyclops

The Cyclops[141] mote shown in figure 4.1 is a multi-processor platform built using custom design. The cyclops mote consists of an ATMEL ATmega128L processor - an 8-bit RISC core processor operating at 7.3728MHz and 3.3V. The ATMEL ATmega128L processor has 128KB of flash program memory and 4KB SRAM. The second processor is a low power Xilinx XC2C256 CoolRunner complex programmable logic device **(CPLD)** operating at 1.8V. The CPLD is connected to a 16MHz clock and it provides a 4MHz clock to the camera. The ATmega processor handles RF communication and the second processor performs image processing, hence facilitating isolation and concurrency in the image sensing and networking computations.

The SRAM of 64KB operating from 2.3V to 3.6V power supply for image buffering and 512KB of CMOS flash programmable and erasable read only memory for permanent storage are interfaced to the system. Both the external SRAM and the flash memory are kept in standby modes when they are not in use. The camera used in cyclops (CIF resolution CMOS camera module - ADCM-1700) supports three image formats of 8-bit monochrome, 24-bit RGB color, and 16-bit YCbCr color and is programmable through a synchronous serial I^2C port. The power consumption of the cyclops mote has a peak value of 110.1mW and minimum of 0.8mW. Though the author clearly describes the acquisition of images at low frame rates on cyclops platform, quantitative information is not provided in this regard for making a comparison.

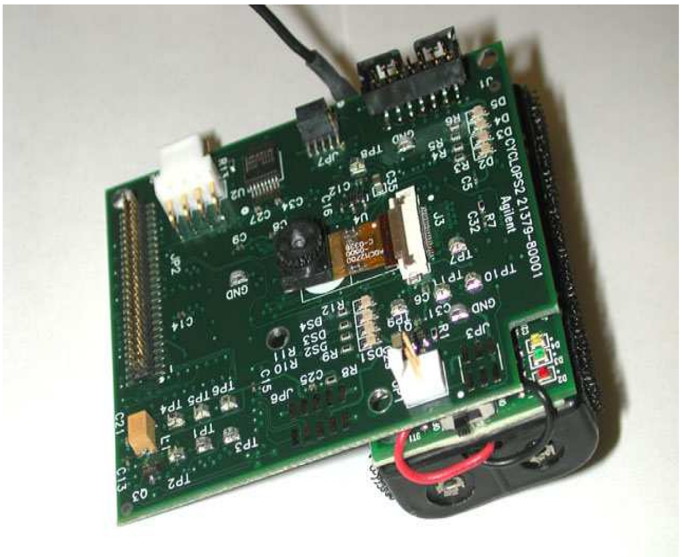


Figure 4.1. Cyclops VSM platform [141]

4.2.2 Mesheye

The Mesheye[142] mote shown in figure 4.2 is equipped with Atmel AT91SAM7S family of microcontrollers - an ARM7TDMI ARM Thumb processor with 32-bit RISC architecture with maximum clock frequency of 55 MHz, internal SRAM of 64 KB and internal flash memory of 256 KB. The built-in power management controller can power down peripherals by disabling their clock source and hence providing low power modes. An internal programmable PLL provides an option of choosing various clock frequencies. The mote is capable of interfacing eight 30 x 30 pixel imagers(ADNS-3060, 6-bit grayscale) and one VGA(640 x 480 pixels) camera module(ADCM-2700). The data from the VGA camera is read through general-purpose I/O pins for the purpose of avoiding additional on-board components like CPLDs, FPGAs etc. resulting in a frame rate of 3 frames per second(**fps**).

Though the authors describe that Mesheye supports eight pixel imagers, the data from them is acquired via a SPI interface. Therefore, acquisition procedure from the eight pixel imagers is done sequentially. The strategy followed in acquiring the VGA color image is not described by the author which is of critical importance since the total size of the image is exceeding the on-board RAM boundaries. The system switches between various cameras present on board under control of software and thus reducing the power consumption. The peak power consumption of the mesheye mote is 175.9mW.

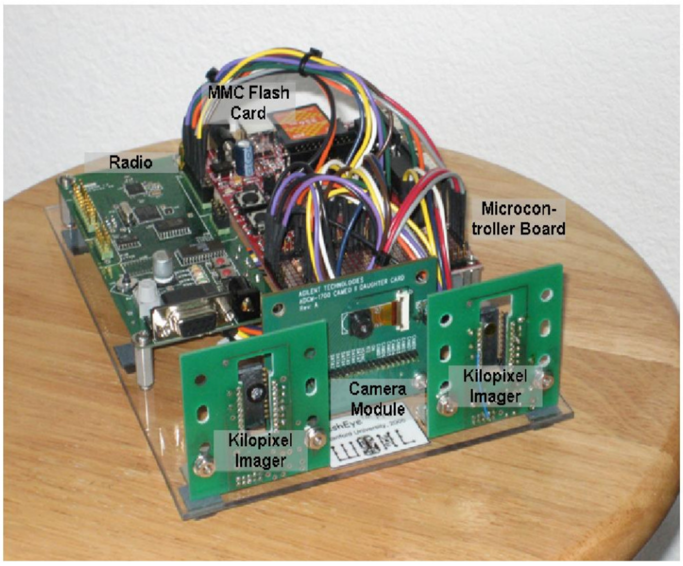


Figure 4.2. Mesheye VSM platform[142]

4.2.3 FireFly Mosaic

The Firefly Mosaic Mote[143] consists of three main components; the CMUcam3 vision processing board, the FireFly sensor node and the PC interface board as shown in figure 4.3. The CMUcam3 board consists of an OmniVision OV6620 camera, Averlogic AL440b FIFO chip acting as a frame buffer, and a 32-bit LPC2106 ARM7TDMI microcontroller running at 60MHz with 64KB on-chip RAM and 128KB on-chip FLASH memory. The image acquisition and processing are completely implemented using the dedicated LPC2106 microcontroller. The Atmel Atmega128L, a 8-bit processor with 8KB RAM and 128KB flash memory is coupled with a Chipcon CC2420 - an IEEE 802.15.4 radio to handle the RF communication.

The authors employed JPEG compression on images instead of taking raw images. Moreover, details regarding frame rate of the images captured by the CMUcam3 hardware module are not made available. The resolution of the captured image is 352 x 288 pixels and the peak power consumption of the Firefly Mosaic mote is 572.3mW when all the external peripherals employed are enabled in the mote.

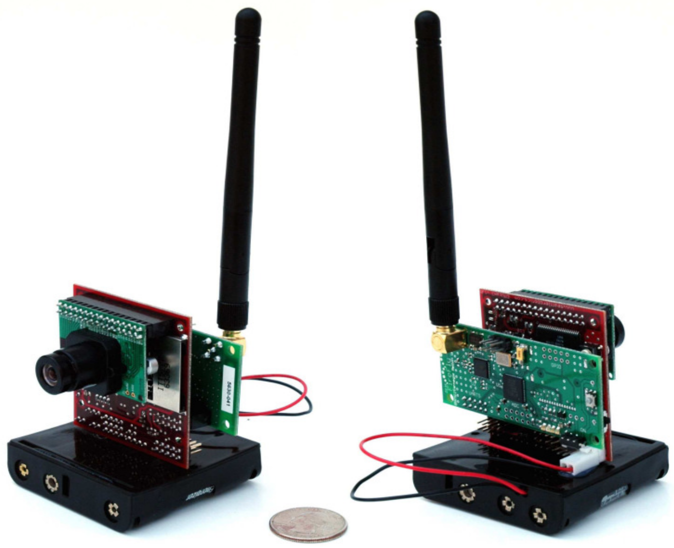


Figure 4.3. Firefly Mosaic VSM platform[143]

4.2.4 MicrelEye

The Micreleye[144] shown in figure 4.4 is a fully integrated hybrid platform combining a FPGA and a microcontroller on the same chip. The chip contains a 40K gate FPGA and an AVR 8-bit RISC microcontroller with on-board SRAM 36 KB, of which maximum of 16 KB is for data memory and 20 KB for program storage. The microcontroller is clocked at 14.74 MHz. An external memory of 1MB with an automatic power-down feature was interfaced to the board for storing the frames acquired and for processing images.

The camera module is OV7640 from Omnivision operating at 2.5V. The acquired image resolution is 320 x 240 (QVGA) resolution at 30 fps frame rate. The higher frame rate in MicrelEye mote is possible because of the FPGA interface implemented between the camera module and AVR microcontroller for discarding the U and V bytes in the data of the YUV 4:2:2 frames before the data is received by the microcontroller for processing. Therefore, the mote supports grayscale image acquisition at high frame rates. The peak power consumption of the MicrelEye mote is 500mW.

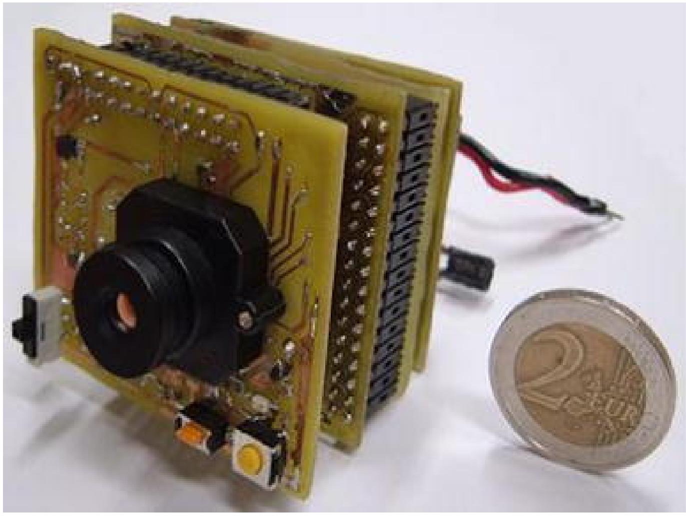


Figure 4.4. Micreleye VSM platform[144]

4.2.5 Citric

The Citric mote[145] shown in figure 4.5 is a custom-made high-end platform consisting of the PXA270 fixed-point processor with a maximum speed of 624 MHz, 256 KB of internal SRAM, and a wireless MMX co-processor to accelerate multimedia operations. The processor is scalable in voltage and frequency for low power operations. The PXA270 also provides the Intel Quick Capture Interface for high-speed transfer of data from the camera instead of using custom hardware. The citric mote is interfaced to a T-mote consisting of Texas MSP430 (8MHz) microcontroller having 10KB RAM and 48K flash for handling RF communications.

The camera used in citric mote is OmniVision OV9655, a low voltage SXGA (1.3 megapixel) CMOS image sensor that offers the full functionality of a camera and image processor on a single chip. The PXA270 is interfaced to 64 MB of SDRAM and 16MB of Intel NOR FLASH providing ample space to buffer images during acquisition and processing along with algorithm testing. The citric mote is capable of acquiring the images of the resolutions 1280 x 1024 and 640 x 480 at the frame rates 15fps and 30fps respectively. The peak power consumption of Citric mote is measured as 927mW.

Figure 4.5. Citric VSM platform[145]

4.2.6 WiCa.

The Wireless Camera mote[146] shown in figure 4.6 is a unique custom design because of its implementation using Xetal-II processor–a 107 GOPS, 600 mW massively parallel processor explicitly designed for video analysis. The intensive data processing is handled by a linear processing array consisting of 320 processing elements and a 10 Mbit on-chip frame memory. The device provides interfaces to three independent video channels of 10-bit resolution with the transfer rate of 80 Mpixels/s.

Though the capabilities of the device are very high, the processor programming is done using extended C language. Since it is a single instruction multiple data**(SIMD)** processor, it is more applicable to the data-intensive applications instead of hybrid applications where varying demands for data and control processing exists. The processor has peak power consumption of 600 mW. The

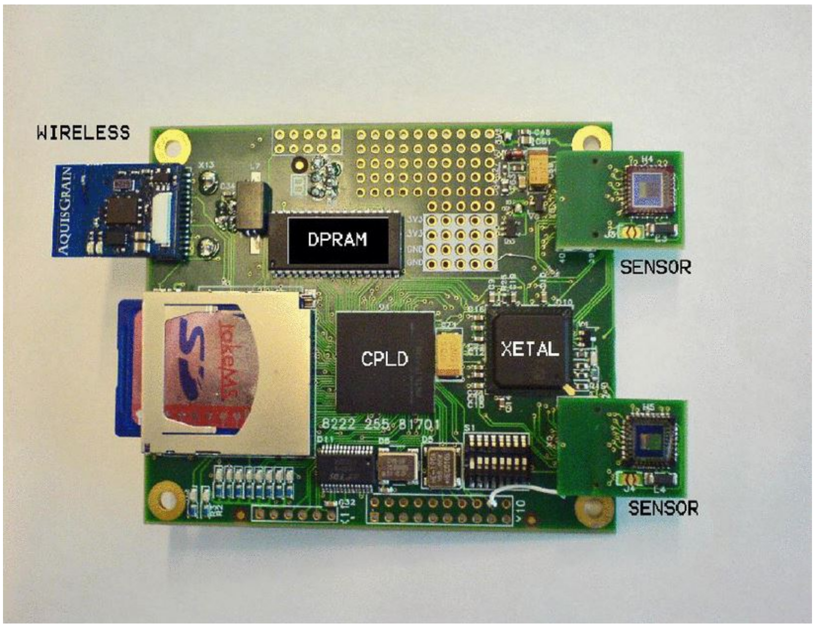


Figure 4.6. Wica. VSM platform[146]

VSM platforms listed above were mostly built using used custom-designed embedded platforms for handling various demands of visual sensor networks like surveillance, target tracking etc. The demerits of the platforms are listed in 4.3.

All the platforms described above require external memory interfacing that leads to larger form-factor of the embedded platform along with development time and performance. The Cyclops, Mesheye and Firefly mosaic interfaced external hardware for handling high data rates from camera modules which can be considered as another reason for larger form-factor. Another major drawback of mentioned platforms other than Citric and Wica. are that they are meant for soft-real time application in which timely constraints can be ignored. Few platforms like Citric, Firefly, Cyclops and Mesheye are coupled with another COTS wireless sensor mote for explicit handling of communication and hence they are multi-processor platforms with large form-factor.

The citric platform is more suitable for super scalar applications like running operating systems and the Wica. platform is designed for explicitly for image and video processing. Hence citric and wica. are capable beyond the requirements and also with unnecessary shortcomings like memory deficiencies and less ports for sensor interfacing etc.

Table 4.3. Demerits and applications of VSM platforms

Platform	Drawbacks	Applications
Cyclops	Onboard ROM and RAM deficiency External IC for memory and image acquisition	soft real-time like WSNs
Mesheye	Onboard ROM and RAM deficiency External IC for memory, lower FPS large form-factor	soft real-time like WSNs
Firefly mosaic	Onboard ROM and RAM deficiency External IC for image acquisition	soft real-time like WSNs
Micreleye	Onboard ROM and RAM deficiency	soft real-time like WSNs
Citric	Onboard ROM and RAM deficiency	Hard real-time like multimedia, avionics
Wica	Onboard ROM and RAM deficiency	Hard real-time like multimedia, graphics

Applications such as modular robots require better performance for tracking real-time data from the rotary encoders, inertial measurement units, vision sensors along with meeting communication demands. All the platforms described above make the system unnecessary complex when the mobility/navigation interfaces are added to the embedded platform. The optimal requirements that are to be interfaced/present to/on the embedded platform in modular robotics domain are

- Sufficient on-board RAM and ROM.
- Necessary on-board interfaces for transducers such as vision sensors, inertial measurement units for providing self-reconfiguration capabilities .
- Sufficient capabilities for on-board processing like image acquisition and processing.
- Single processor platform facilitating communication, navigation and other necessary tasks due to form-factor constraints.

The FlexEye VSM - a flexible camera mote is prototyped using COTS components at very low cost while meeting the above mentioned requirements while also featuring low power consumption, small form-factor and support for future expansion to facilitate research

4.3 FlexEye Camera Mote

The primary operational features expected from an embedded system of a modular robot is to control actuators for locomotion and track the movement in real-time for closed-loop control. The 8-bit microcontrollers available in the market like 8051ED2 boards or 16-bit architectures from ATMEL and Texas Instruments will also suffice the task. Self-reconfigurability of a modular robot is achieved when the robot in the neighborhood is detected successfully and docking with it after recognition. Such capabilities can be inculcated in the modular robot by using suitable microcontroller platform equipped with processing capabilities for data-intensive tasks like image processing. The characteristics of various embedded platforms along with their suitable applications are shown in figure 4.7

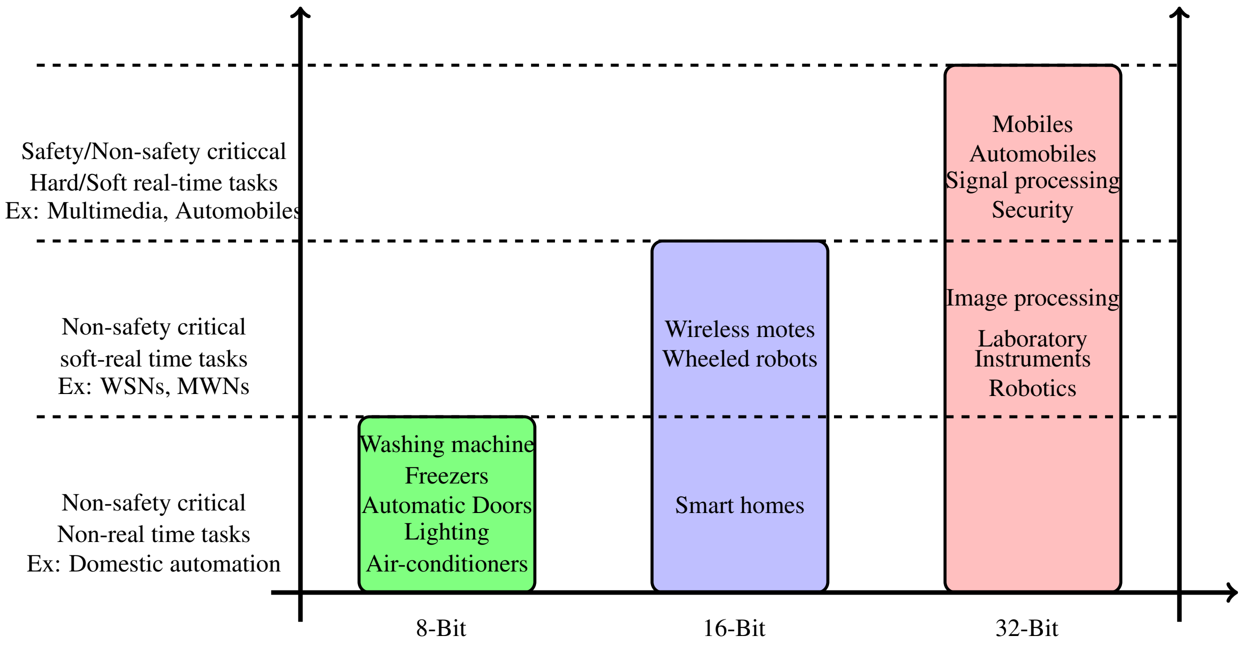


Figure 4.7. Characteristics and applications of Embedded platforms

The modular robots like M^3, iMobot and SMORES can be implemented using 8-bit and 16-bit platforms because of absence of vision sensors and requirements for heavy computations. The CoSMO and Trimobot robotic module require high-end (32-bit) microcontroller platforms since they are equipped with vision sensors for object recognition and navigation. Significant improvement in performance of the 32-bit embedded platforms in relation to 8-bit/16-bit platforms is observed because of increase in the bit size, improved clock frequencies, manufacturing technologies and micro-architectures.

Advanced RISC Machine(**ARM**) architectures from ARM limited is proven and standardized 32-bit platforms available in the market today. A wide-range of architecture profiles are available from ARM limited starting from high-end applications using Cortex-A series and to energy efficient applications using Cortex-M series. The Cortex-M series devices are designed explicitly for operating system independent and low energy applications. Initial possibilities of an embedded platform are explored with established camera platform - Fire Bird V[147] designed by Nex Robotic private limited. The platform is embedded with LPC2148 microcontroller designed using ARM7TDMI architecture similar to Mesheye platform. The Microcontroller suffers from the same drawbacks of Mesheye VSM such as larger form factor, lack of sufficient on-board memory capabilities and interfaces for rapid image acquisition and processing.

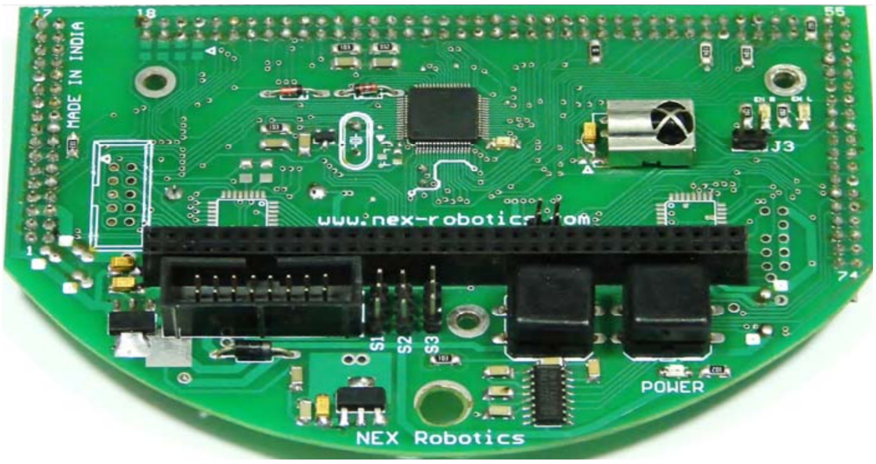


Figure 4.8. LPC2148 - Nex robotics platform

Recent STM32 series of microcontroller from STMicroelectronics provided solution to majority of the requirements of the modular robotics like inbuilt interfaces for camera modules, sufficient on-chip memory etc. while not compromising on energy conservation. The Flexeye mote prototyped is developed using STM32 Microcontroller boards and details on the same are provided in further sections.

The FlexEye mote consists of five independent hardware modules interfaced together.

- STM32F4 discovery board
- STM32F4 baseboard
- OV9655 camera

- CC2500 transceiver
- MicroSD card

The STM32F4 discovery board[148] from STMicroelectronics is the central core of FlexEye platform that implements image acquisition and wireless communication. The discovery board is mounted on the STM32F4 baseboard for providing external connectivity via Ethernet, RS-232, camera and MicroSD card interfaces. The hardware specifications and features of various modules employed in FlexEye are provided in the sub-sections below.

4.3.1 STM32F4 Discovery Board

The STM32F4 Discovery Board consists of STM32F407VGT6[149] microcontroller - an ARM Cortex M4 architecture (32-bit RISC architecture) based microcontroller providing digital signal processing extensions, single precision FPU, optional memory protection unit etc. The 1.25 DMIPS/MHz core together with interrupt features like single non-maskable interrupts(NMI) and 240 physical interrupts enables interfacing to external devices and concurrent process handling. The STM32F407VGT6 microcontroller has inbuilt features such as frequency scalability, peripherals like SPI, I^2C, UART, CAN etc. for sensor interfacing, SDIO for memory card access, Digital camera interface(DCMI) for interfacing camera modules, flexible memory controllers for external memories etc. The STM32F407VGT6 microcontroller operates on 1.8 to 3.6V supply with three low power modes of operation - Sleep mode, Standby mode and Stop mode.

4.3.2 OMNIVISION OV9655 Camera

The OV9655 CameraChip[150] is a 1.3 MegaPixel CMOS image sensor which offers the functionality of a camera and an image processor. The OV9655 provides full-frame, sub-sampled, scaled or windowed images in a wide range of formats such as SXGA (1280 x 1024) to 40 x 30 using 8-bit/10-bit parallel data bus. The OV9655's on-chip image processing features like exposure control, gamma, white balance, color saturation, white pixel canceling etc aids in reducing the power consumption during image processing. The camera module can be configured for various resolutions, image formats, preprocessing functions using the serial camera control bus(**SCCB**) interface available on the module.

4.3.3 CC2500 - RF transceiver

The CC2500[151] is a low-cost and low power 2.4 GHz wireless transceiver intended for the 2400-2483.5 MHz ISM band. The RF transceiver is integrated with a configurable baseband modem supporting various modulation formats like OOK, 2-FSK, GFSK, and MSK and has a configurable data rate up to 500 KBaud. The CC2500 module can be deployed in a platform for research purposes since it supports configuration capabilities for packet handling, data buffering, burst transmissions, clear channel assessment, link quality indication etc. A CC2500 module with a SMA antenna is interfaced with the module and the current consumption at 3V is observed as 48mA for transmission distance of 62 meters and 4.6mA while receiving data.

The FlexEye VSM is shown in figure 4.9. The platform is a prototype and the placement of various modules can be changed as per the requirements of deployment in an application

Figure 4.9. FlexEye mote prototyped using COTS components

4.4 Image acquisition - Software

The suitability of the FlexEye platform for soft real image processing applications was verified using image acquisition and processing tests for three different image resolutions – QQVGA(160 x 120 pixels), QVGA (320 x 240 pixels) and VGA (640 x 480 pixels). The STM32F407VGT6 microcontroller has 1MB on-chip flash for program memory and limited SRAM of 196KB. Though the SRAM is limited in terms of storing images greater than QQVGA resolution, by parallel utilization of the resources on-board(SRAM, DMA, and MicroSD card), the ARM Cortex M4 can be used for acquiring images of higher resolutions that cross RAM boundaries and still facilitate processing on high-resolution images. The memory constraints were handled by employing pipelining strategies in acquiring and storing the image data using SDIO, DMA and DCMI peripherals present in the system. The DCMI peripheral also supports dynamic image cropping, hence saving memory space utilized during image acquisition. The DMA on the microcontroller is employed to handling the process of acquiring data from DCMI buffers and storing the same in SRAM with minimum intervention of the processor core. The hardware architecture employed for implementing the high-speed image acquisitions on the platform developed is shown in figure 4.10. The software libraries developed for implementing the pipelined acquisition are operated in four layer cycle during the image acquisition process. The four layers are

1. Configuring peripherals and devices.
2. Fetching image data and storing in SRAM.
3. Storing fetched image data in MicroSD card.
4. Retrieving image data stored in MicroSD card.

The layer 2 and layer 3 ensures the utilization of hardware resources in the pipeline for accelerating the process of acquiring and saving the images by employing DMA, buffers and SDIO peripherals. The implementation of layer 3 is necessary for scenarios, where the size of images being acquired is greater than SRAM boundaries and layer 4, is used when processing has to be done on the acquired images and also for transmitting image data to PC for debugging purposes.

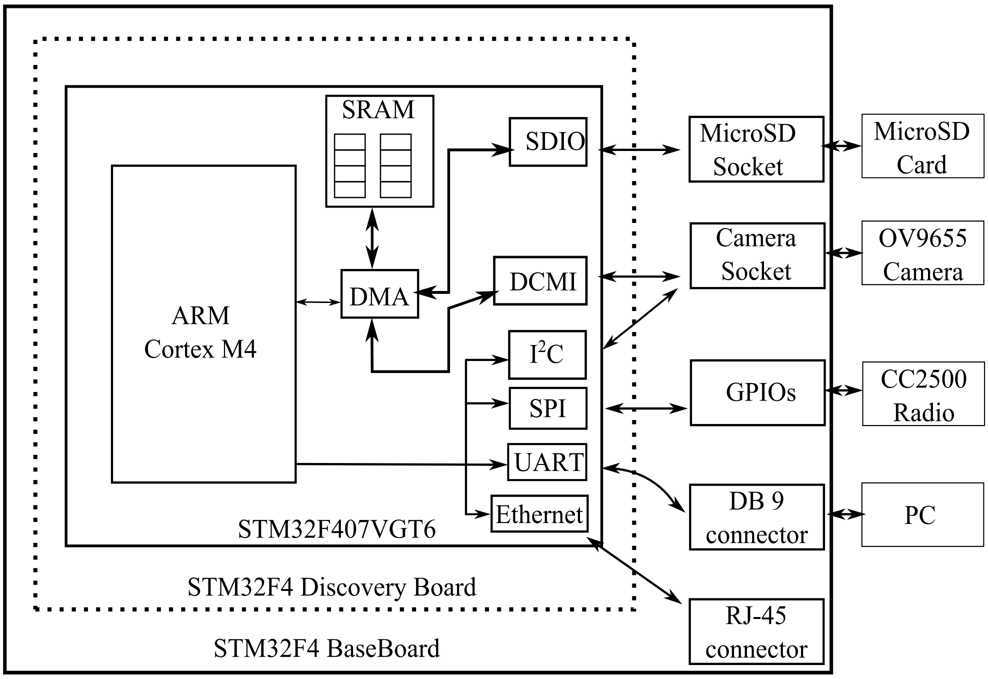


Figure 4.10. FlexEye VSM - image acquisition hardware architecture

4.4.1 Layer 1 - Configuring peripherals and Devices

The layer 1 primarily handles the configuration of the OV9655 module via I^2C peripheral, initialization of DCMI and DMA for proper synchronization among each other and initialization of MicroSD card using SDIO peripheral. It is a one-time execution and needs to be executed again only when a different resolution image from the current resolution has to be acquired. The OV9655 module is configured in layer 1 for required resolution, brightness, internal clock speeds, testing and debugging etc. The DCMI peripheral is capable of receiving high-speed data flows from external 8-bit CMOS camera modules. The data are packed into a 32-bit data register by DCMI for transferring through a stream of DMA channel. The two DMA controllers present on STM32F407VGT6 microcontroller employ dual advanced high-performance master bus architecture with independent FIFOs and streams for fast transfer of the data between memories and memory to/from peripherals.

4.4.2 Layer 2 - Fetching the Image data and storing in SRAM

The image data is transferred continuously at high data rates on 8-bit parallel data bus from camera and the same data can be stored in the MicroSD card operating via 4-bit parallel data bus. The write speeds to the MicroSD card are lower in relation to acquisition rates due to different data bus lengths and operating frequencies. The feasible way of acquiring high resolution data without significant compromise in data rates is to use two circular buffers in SRAM (referred as Buffer 1 and Buffer 2 henceforth in the thesis) after synchronizing the operating frequency of the camera module with write speeds of MicroSD card. The image data will be directed to Buffer 1 using DMA and after filling the Buffer 1, the DMA will be re-directed data to Buffer 2 for writing new image data. The data from a filled buffer can be transferred to MicroSD card while other buffer is being filled with recent data. Since writing and reading of the data is done continuously in a loop till the complete image frame is acquired, the buffers in operation are also referred as circular buffers. The data sent per second will quadruple when image resolution changes from QQVGA to QVGA and QVGA to VGA resolutions. Since the data transfer rate to the MicroSD card is approximately constant, these change in data rates because of changing resolutions forces the acquiring and storing images processes out of synchronization. Hence for proper synchronization, the rate of data transfer from camera module is synchronized according to the resolution of image being transmitted by configuring the internal clock frequency of OV9655 module. The hardware peripherals used in STM32F4 Discovery board and the block diagram for implementing layer 2 and the rest of software cycle is shown in figure 4.11.

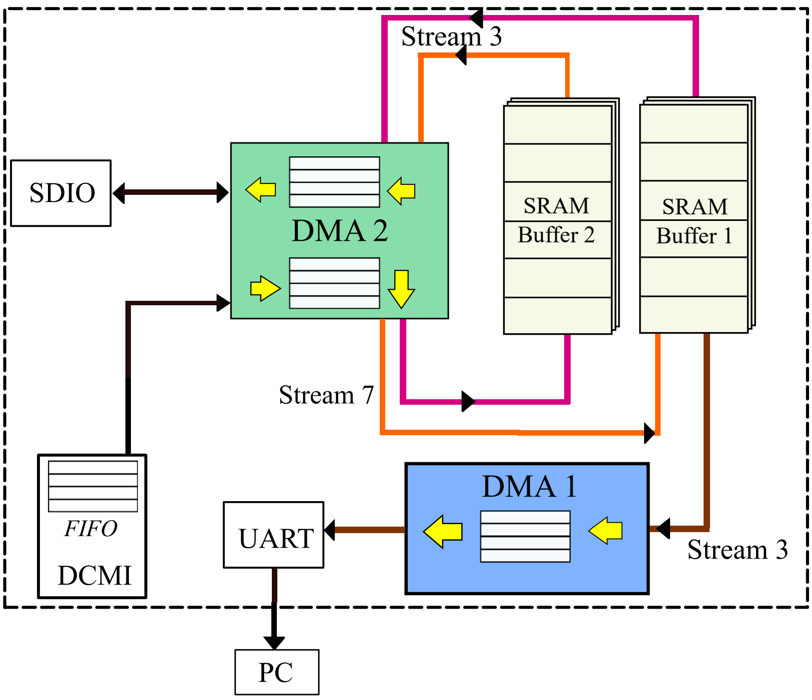


Figure 4.11. FlexEye - active hardware peripherals for image acquisition

4.4.3 Layer 3 - Storing the image data from SRAM

After proper synchronization between the data rates of acquisition from OV9655 and writing in MicroSD card, the data can be transferred to MicroSD card after complete utilization of a buffer in SRAM. The data transfer to the MicroSD card is performed by the same DMA that performs the acquisition operation from the OV9655 module by concurrently filling the second buffer. The storage of the image in MicroSD card is performed directly instead of using a file system like file allocation table, net technology file system etc. for reducing the possible latencies in performance assessment. The sequence of various events executed during concurrent operation of the layer 2 and layer 3 is shown in figure 4.12. As shown in figure 4.12, the DMA first fills Buffer 1 with data(till t_1) from the OV9655 and DMA switches to Buffer 2(from t_1 till t_2). While the data is being loaded to Buffer 2, the unread data from Buffer 1 is shifted to MicroSD card. The data transfer rates from camera are configured so that there is a small time gap *P* left after every write operation to MicroSD card. The time gap *P* is left to accommodate the minute variations of write speeds in MicroSD card.

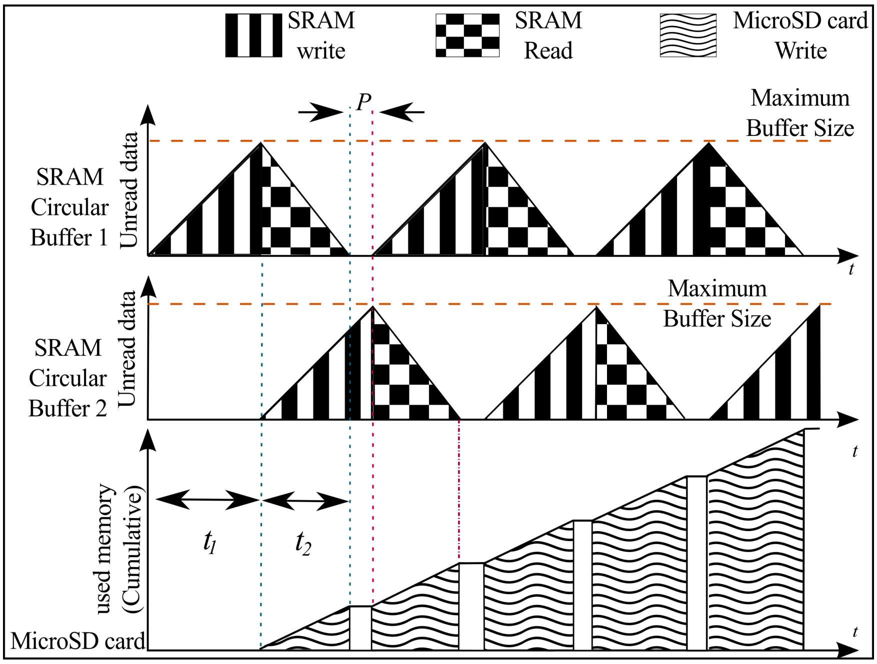


Figure 4.12. FlexEye - Event sequences in Layer 2 and Layer 3

4.4.4 Layer 4 - Retrieve the image data from MicroSD card

The complete image or a portion of an acquired image stored can be retrieved from the MicroSD card and stored in SRAM for further image processing. The acquiring functionality is also implemented using DMA.

4.5 Results and discussions

The performance of FlexEye VSM platform in analyzed in terms of image acquisition rates, power consumption and latencies in completing image processing functions.

4.5.1 Analysis on Power Consumption

Power consumption of a electronic platform plays vital role in determining the longevity of the robotic systems. Modular robotics employs locomotion by distributed control and such control requires minimal processing capabilities at each node except at the top node in the decision hierarchy. The ability to control the performance of a microcontroller in terms of speed of operation in such scenarios can be an added advantage in energy optimizations.

4.5.1.1 Power consumption - Chip level

The STM32407VGT6 microcontroller employed in the FlexEye provides frequency scalability features using which the performance of the chip can be controlled as per the requirement. A part from frequency scaling, the lower power modes such as sleep mode reduces the power consumption of microcontroller further and hence making power consumption of robotic modules not participating in locomotion almost negligible. Figure 4.13 and 4.14 provides details on power consumption of STM32F407VGT6 at different frequencies in run and sleep modes respectively. It can be observed from figure 4.13 and 4.14 that power consumption can be reduced to minimal levels for frequencies below 20MHz which are typical operating frequencies of WSN motes listed in table 4.1.

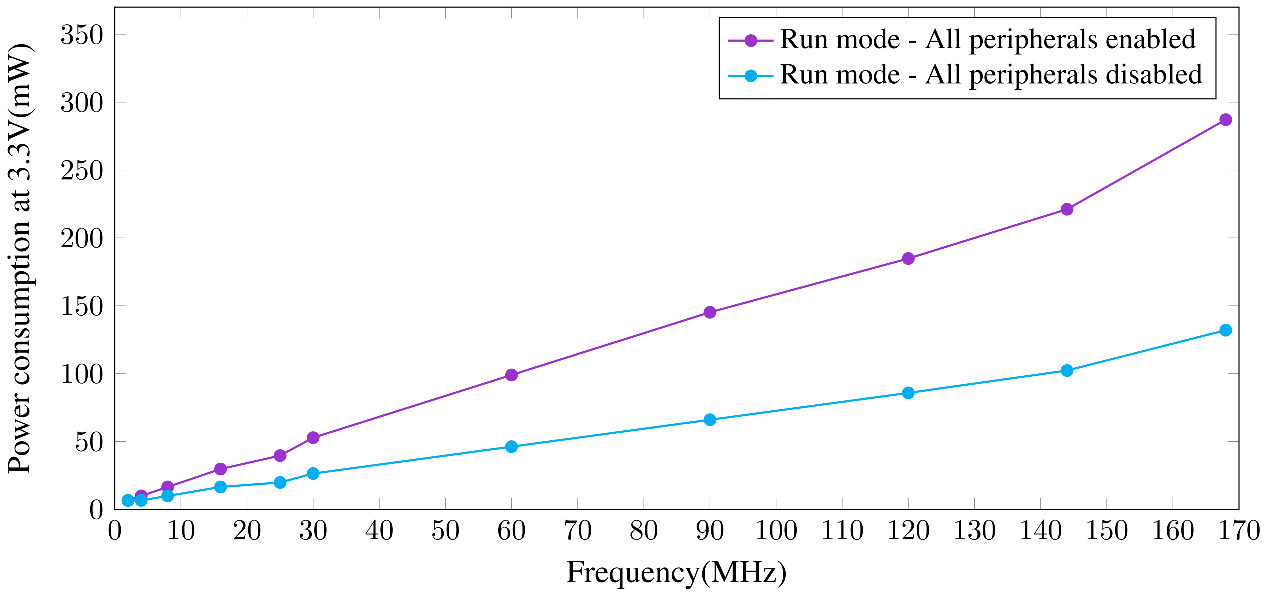


Figure 4.13. STM32F407VGT6 Power consumption - Run mode

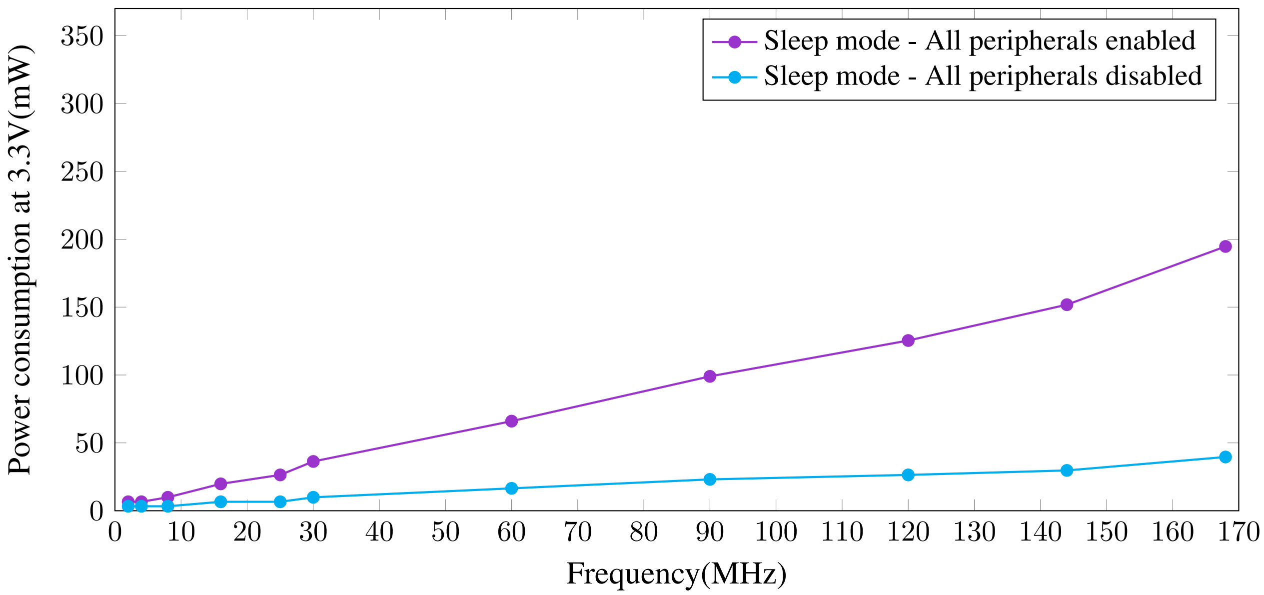


Figure 4.14. STM32F407VGT6 Power consumption - Sleep mode

4.5.1.2 Power consumption - Board level

The on-board components of baseboard in FlexEye platform are connected to power supply pins of discovery board for powering and the input current of baseboard was measured from power supplies(3V and 5V from discovery board). The input currents of baseboard for external interfaces such as RJ-45, RS-232, on-board crystal oscillator for camera etc., in different scenarios is summarized in table 4.4. It can be observed from table 4.4 that baseboard input current is consuming 41.8 mA for maintaining the active state of the hardware drivers (RJ-45 and RS232) that are not utilized during image acquisition. The operating current of the MicroSD card and the camera together isolated from the unused interfaces on the baseboard is observed from table 4.4 as 19.5 mA.

Table 4.4. Current consumption of Baseboard on FlexEye VSM

Camera	MicroSD card	3V Supply	5V Supply	Tot. Current
X	X	41.8mA	8μA	≈41.8mA
X	√	42.4mA	8μA	≈42.4mA
√	X	60.5mA	8μA	≈60.5mA
√	√	61.3mA	8μA	≈61.3mA

It has been observed that the idle state input current of the microcontroller at 168 MHz with default peripheral settings after the reset state is 48.5 mA and it increases to 53.6 mA after initializing the peripherals used for acquiring the images. The increase in current input can be attributed to enabling of GPIOs, DMA, DCMI, I^2C and SDIO peripherals. The current intake during the image acquisition process was observed as 67.2 mA. Hence, the total input current for the acquisition of images including baseboard consumption is 86.7 mA approximately and power consumption at 3.3V supply for the image processing can be calculated as 286.11 mW. Since a high transmission range CC2500 module with a whip antenna is interfaced, the input current of the CC2500 module during continuous transmission was measured as 48 mA for a range of 62 meters. Therefore, the power consumed by radio module during wireless transmission is 144 mW. The peak current consumption of the platform is 214.3 mA after including current input to microcontroller, interfaces on baseboard, camera and radio modules - baseboard total current of 61.3 mA, 93 mA during peak processing of microcontroller at 168 MHz operating frequency from figure 4.13 and 60 mA current in transceiver for transmission range of 120 meters. Hence the peak power consumption is 707.19 mW with the device operating at 168 MHz on full load.

Table 4.5 provides comparison between the VSM platforms discussed in section 4.2 and FlexEye platform. The Cyclops and Mesheye motes are power efficient platforms achieved at a cost of reduced capability for data intensive applications. The processing power of Firefly Mosaic mote is less than FlexEye and has relatively high power consumption for available specifications. The Citric platform is a mote with very powerful processor but provides less support for future expansion. Though the MicrelEye mote appears to be a close competitor, FlexEye platform supports color images while Micreleye supports only gray scale images. The Wica mote provides

Table 4.5. Comparison of VSM motes

Platform	**Processor**	**Frequency (MHz)**	**External RAM/ Processor/ onboard Buffer**	**Expansion support**	**Peak Power (mW)**
Cyclops	ATmega128L(8-bit)	7.37	Xilinx CPLD	√	110.1
Mesheye	AT91SAM7S(32-bit)	55	–	√	175.9
Firefly	LPC 2106(32-bit)	60	AL440b FIFO	√	572.3
MicrelEye	AVR(8-bit)	14.74	FPGA	X	500
Citric	Intel PXA270(32-bit)	624	Texas MSP430	X	927
Wica	XETAL-II	84	–	√	600
FlexEye	STM32F407(32-bit)	168	–	√	707.19

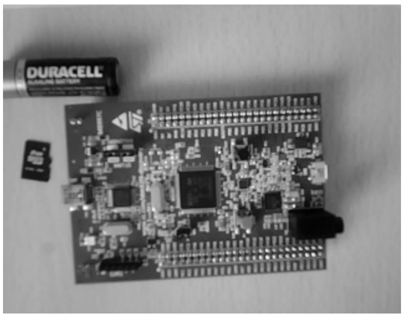
extensive parallel data processing owing to its SIMD processor architecture and the mote might not provide the reactiveness required for applications like robotics and MWSNs. In relation to other platforms, it can be observed from table 4.5 that the FlexEye platform provides major features for image processing at very low expense of energy along with high processing capabilities.

4.5.2 Analysis on image acquisition capabilities

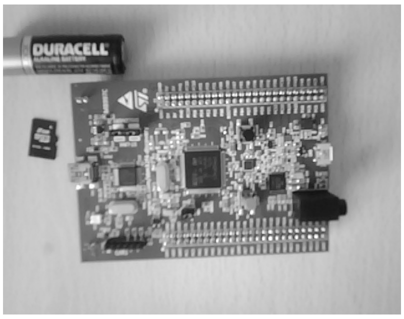
The performance assessment on MicroSD utilization as a substitute of external memory was attempted in multiple scenarios with the images of different resolutions to identify and distinguish the contribution of microcontroller, camera and MicroSD card to the latencies in image acquisition and processing. The color images of QQVGA, QVGA and VGA resolutions were successfully acquired using hardware pipelining on the STM32F407VGT6 microcontroller. The FlexEye VSM also supports grayscale image acquisition at different resolutions. The grayscale images of three different resolutions QQVGA, QVGA and VGA are extracted from the stored YUV 4:2:2 color images and are sent to MATLAB via the UART interface after isolating the Y component. The images plotted using matlab are shown in figure 4.15. The hardware architecture coupled with software libraries are capable of receiving approximately 50 color QQVGA frames per second, 17 color QVGA frames per second and 5 color VGA frames per second with minimum utilization of on-board SRAM. Though the SRAM available in the system is 196 KB, only 76.8 KB is allocated in total for both buffers (39% of the total RAM). The acquisition of QQVGA images was successfully tested at RAM utilization of 19.2 KB (9% of the total RAM) for further reducing the RAM utilization.

The libraries developed for acquisition of the images also support the acquisition of a specific portion of the images in different resolutions from the camera module. The images are cropped using DCMI peripheral of the STM32F407VGT6 microcontroller for reducing power consumption and the acquired images are shown in figure 4.16. This feature is a merit of the platform because of which the high resolution images of pyramid (used for alignment error detection as described in chapter 3) can be acquired for rapid processing.

QQVGA - 160 x 120 Pixels



QVGA - 320 x 240 Pixels



VGA - 640 x 480 Pixels

Figure 4.15. Acquired images - QQVGA, QVGA and VGA resolutions

4.5.3 Analysis on image processing capabilities

The analysis on image processing capabilities of the FlexEye platform is presented below using fundamental operations of image processing such as YUV to grayscale conversion, grayscale to binary conversion and smoothening of images. The latencies of various functionalities are

Figure 4.16. Cropped images - Acquired QQVGA and QVGA images

observed using on-chip 32-bit hardware timers operating with the resolution of 1μS. The acquired images are processed sequentially as mentioned below and processed image outputs are validated using MATLAB.

4.5.3.1 Scenario 1 - QQVGA resolution

The images of QQVGA resolution in YUV format are acquired from the camera without MicroSD card and processed to provide grayscale, binary and smoothened images. Since the total image size is 38400 bytes in YUV format, the image can be stored for further processing on RAM directly. The observed latencies are summarized in table 4.6.

Table 4.6. Timing details on image acquisition and processing - QQVGA resolution

Functionality	Latencies
YUV Acquisition time	15.894 ms
YUV to Grayscale	1.03 ms
Grayscale to binary	2.415 ms
Smoothening	3.95 ms

4.5.3.2 Scenario 2 - QVGA resolution without MicroSD card

The images of QVGA resolution acquired in YUV format should be stored in MicroSD card due to their large memory occupancy - 153600 bytes. In the present scenario, the grayscale images are generated directly from YUV data during acquisition and stored on RAM separately in an array. The grayscale images are derived in runtime by accumulating luma (Y) data of the YUV image separately using ARM core from the Buffer 1 and Buffer 2 as they are filled by DMA peripheral in circular order. The rest of data (UV in YUV) is overwritten as the new YUV data is being fetched by DMA from DCMI. The current scenario cannot facilitate color image processing and can be used as a benchmark or ideal case for comparison in acquiring QVGA images in the presence of MicroSD card which also retains data for color image processing. The results of scenario 2 are listed in table 4.7. It can be observed from table 4.7 that due to the participation of ARM processing core, the YUV to grayscale and YUV to binary conversions took approximately same time.

Table 4.7. Timing details on image acquisition and processing - QVGA resolution(without MicroSD card)

Functionality	Latencies
YUV Acquisition time	16.418 ms
YUV to Grayscale*	16.618 ms
YUV to binary*	16.647 ms
Grayscale to binary	5.188 ms
Smoothening	15.455 ms

$*$ - ARM Core active during acquisition for YUV to grayscale or binary.

4.5.3.3 Scenario 3 - QVGA resolution with MicroSD card

The QVGA images in the current scenario are directly stored in MicroSD card and are fetched later for processing. Since scenario 3 employs MicroSD card, the latencies in the acquisition, processing and generating the required data will throw light on the performance of MicroSD card as alternate memory. The acquired results are tabulated in table 4.8. The YUV to Grayscale and YUV conversion time shown in table 4.8 involves read operations from the MicroSD card memory and storing the converted image in RAM. The grayscale to binary and the smoothening operations took same time as with scenario 2 because they are sequential processes not involving microSD card and operating on the grayscale image available in the RAM.

Table 4.8. Timing details image acquisition and processing - QVGA resolution

Functionality	Latencies
YUV Acquisition time	31.551 ms
YUV to Grayscale	18.910 ms
YUV to binary	20.736 ms
Grayscale to binary	5.188 ms
Smoothening	15.455 ms

4.5.3.4 Scenario 4 - VGA resolution with MicroSD card

The scenario 4 requires the presence of MicroSD card for every operation due to the massive memory occupancy of the color VGA image - 614400 bytes and grayscale image - 307200 bytes. Since both images require memory beyond the RAM capacity, the data is operated through sequential read and writes of overlapping segments of image data from MicroSD card. The latencies observed during sequential acquisition, conversion and the processing are provided in table 4.9. The YUV to grayscale and YUV to binary conversions in table 4.9 are processed in two different methods. The first method involves run-time stripping of the 'Y' data from circular buffers using ARM core and storing the grayscale or binary images to the MicroSD card. This process varies from scenario 2 and scenario 3 because the resultant data is stored in MicroSD card instead of RAM. The ARM core is idle during acquisition in the second method and YUV image is retrieved from MicroSD card in segments and the result is stored back to the MicroSD card after the required conversion. The latencies in smoothening operation for VGA resolution image is the summation of latencies in fetching of an image in segments, conversion of a segment to grayscale in internal RAM, applying the smoothening to every pixel by fetching data of surrounding pixels for averaging and storing the converted image in MicroSD card.

Table 4.9. Image acquisition and processing - VGA

Functionality	Latencies
YUV Acquisition time	99.846 ms
YUV to Grayscale*	100.618 ms
YUV to binary*	101.129 ms
YUV to Grayscale	129.201 ms
Grayscale to binary	132.287 ms
YUV to binary	138.763 ms
Smoothening	225.372 ms

$*$ - ARM Core active during acquisition for converting YUV to grayscale or binary.

The performance of MicroSD card as an alternative to external memory can interpreted from table 4.6-4.9. The data is quadrupling with an increase in resolution and hence the latencies in the processing are bound to increase by almost same factor along with them. The time consumed for grayscale to binary and grayscale to smoothening increased almost 2.2 to 4 times from scenario 1 to scenario 2 when the processing was done completely in on-chip RAM. The acquisition time is excluded from latency calculations due to its dependency on the capabilities of both camera and

MicroSD card. Under the assumption that sufficient RAM is available for storing QVGA color image, the total latency for converting the YUV image to grayscale image of QVGA resolution excluding acquisition time can be calculated approximately for worst case as four times of row 2 in table 4.6. Hence, the latencies in process of converting the complete color image of QVGA resolution after acquisition to grayscale can be observed as 4.120 ms ideally in the worst case. In an ideal case (from scenario 2) it takes approximately 15.455 ms for smoothening a QVGA image when the image is stored in on-chip RAM. It can be estimated from the table 4.7 that approximately it takes around 61.82 ms for smoothening a VGA image in an ideal worst case scenario if the processing is done completely using on-chip RAM. The observed practical latency in the smoothening for a VGA image is provided in table 4.9 as 225.372 ms for which the image source and destination can be MicroSD card only.

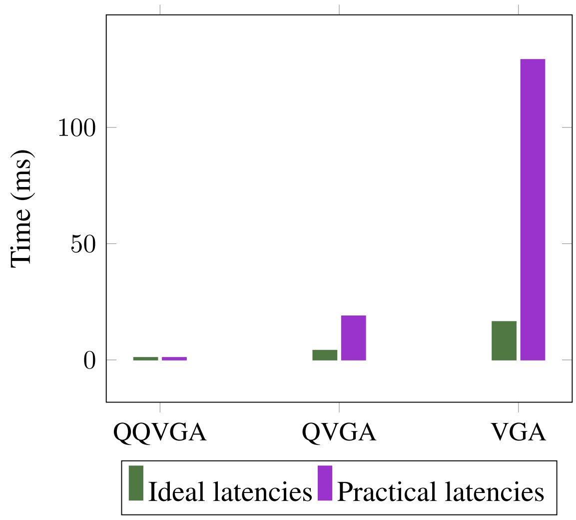


Figure 4.17. FlexEye - performance in YUV to Grayscale conversion

The multipliers for calculating the ideal worst case possible latencies in VGA resolution image processing in each image processing function is calculated from tables 4.6 and 4.7 as explained above. Latencies for ideal and practical cases are calculated as per the identified multipliers of each image processing function and are plotted in ideal and practical columns in figures 4.17, 4.18 and 4.19.

The operating frequency (25 MHz) of SDIO on STM32F407VGT6 limits the transfer capabilities available with MicroSD card. The high-speed physical layer[152] specifications for SDIO suggests data transfers at doubled clock speeds and hence if implemented, it can further reduce the latencies. Since the latencies are proportional to the number of read/write operations and decreasing the

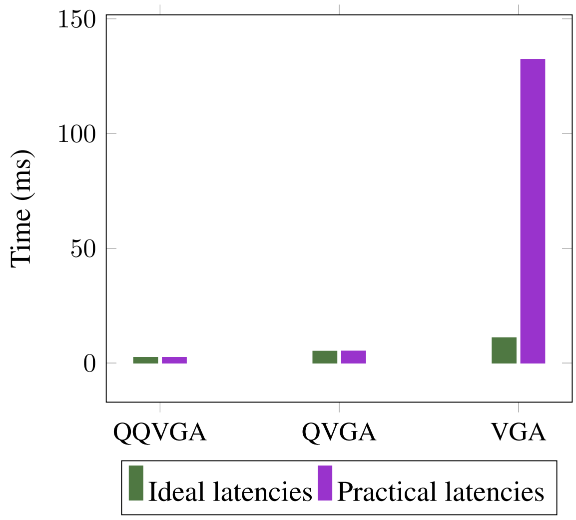


Figure 4.18. FlexEye - performance in Grayscale to Binary conversion

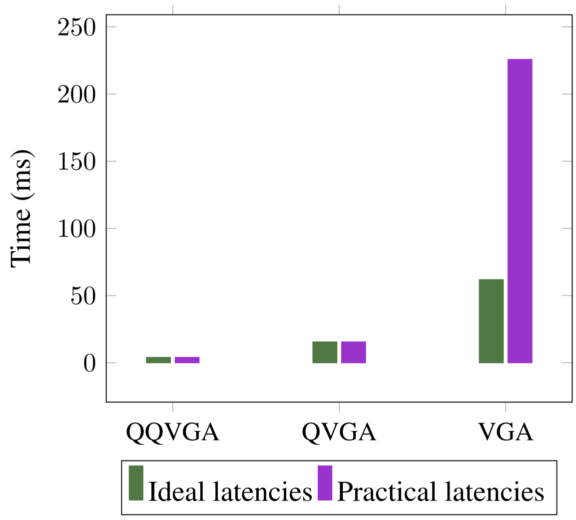


Figure 4.19. FlexEye - performance in Smoothening

same by bulk transfers etc. reduces the delays further. Though there is wide gap between ideal and practical latencies, such gaps are wide only in higher resolution images which are rarely used in practical scenarios. The latencies are also observed to be within milli-second durations and hence the FlexEye platform is sufficient to address the requirements of modular robots.

The acquisition of a VGA image in real-time is not feasible considering the limited memory available on-chip of STM32F407 microcontroller. Though it is possible to acquire via GPIO by repeatedly by the microcontroller processing core, such processes leads to bottlenecks in performance of the over all system. The implementation of novel acquisition algorithm operating on DMA of the microcontroller, the acquisition of the images of size larger than RAM boundaries is possible with utilization of less than 30% of total RAM. Such a methodology can also be used in distributed sensing as well as in navigation.

The algorithm is also capable of performing image processing operations on chunks of image during acquisition and hence saving processing time during post-acquisition. Such algorithms can be considered as an added advantage in resource constrained systems like modular robotics where the resources are shared over multiple tasks like locomotion, sensing and communication.

4.6 Summary

The details on the FlexEye platform are provided in this chapter along with analysis on power consumption and computational capabilities.

- The energy conservation capabilities FlexEye platform are analyzed at chip level using frequency scalability and it has been observed that power consumption of the microcontroller can be reduced to negligible amount at lower operational frequencies and by employing lower power modes. Since all robotic modules in the coordinating structures will not be participating in locomotion, such optimizations comes as an advantage in energy conservation.
- The hardware optimizations performed in prototyping the FlexEye for robotic designs with camera platforms can be realized from table 4.5. A custom designed FlexEye PCB board developed specifically for the purpose of image processing and wireless transmission can further reduce the power consumption by 20% by avoiding redundant physical interfaces and COTS components.
- The image processing capabilities of the platform will be an advantage for applications using modular robots like HexaMob. The autonomous docking and navigation can be facilitated using FlexEye platform in a single modular robot and by using sensor fusion, the platform can further provide improvised features for navigation in biomimetic structures.
- The performance of external memory(MicroSD card) as temporary memory buffer is analyzed for estimation of latencies. Though the performance degrades at high image

resolutions due to involvement of MicroSD card in place of external RAM, presence of external memory card aids in data storage for large-scale applications and high resolution images are seldom used for image processing on microcontrollers.

Chapter 5

Quanta - A Modular Platform for Rapid Control and Monitoring of Heterogeneous Mobile Robots

5.1 Introduction

The research in Modular robotics, MWSNs, and Swarms robotics often involves developing tailor-made hardware and software depending on the application requirements. The complexity in prototyping a large-scale application such as distributed sensing varies from testing simple tasks like remote calibration of a sensor to generation of complex coordinated movement of heterogeneous units equipped with numerous sensors and actuators in real-time. Such basic tasks often tend to be time consuming and presence of mobility in individual units of the system further curtails the advantages a researcher/developer avails from standard prototyping tools (debuggers, oscilloscopes,etc.) in real-time. In the operational phase of an application, the algorithm(s) validation requires providing controlled inputs to the application from external environment in a given test scenario and receive outputs for the same.

The varying demands of the autonomous robotic application(s) involving homogeneous/ heterogeneous mobile units with decentralized/centralized control and monitoring necessitates the researcher to develop an optimal solution for remote control and monitoring of both internal and external events while addressing the constraints in parameters like power, form factor, etc. The yields of such a solution can be further improved by increasing its adaptability to various hardware architectures/ testbeds, the rapid rate of acquisition/ monitoring, ubiquitous nature and applicability in various research scenarios. The major requirements, constraints, and infrastructure available for research and development in the domains of robotics and swarms are summarized in [153][154][155]. An elaborate description of opportunities and challenges in the development of robots in wireless environments are provided in [156].

In order to meet the requirements of remote control and monitoring for robots, a lightweight and high-speed platform referred as Quanta was developed while addressing the constraints such as power consumption and latencies with a minimal trade-off in data rates for communication. The major objectives realized from the development of Quanta platform are

- Facilitation of peer to peer and centralized communication mechanisms for remote control and monitoring of robots.
- Facilitation of concurrent execution of events in the robot, coordinated robotic system and external environment for a test scenario.
- Availability of operating system independent tools for future expansion in hardware designs and components.

This chapter of the thesis provides summary on research so far in wireless platforms in section 5.2. Section 5.3 explains on the hardware infrastructure utilized for developing quanta. The details on state machines implemented during software implementation are explained in section 5.4. The chapter is concluded in section 5.5 outlining the major outcomes from quanta platform.

5.2 Related Work

Various contributions were made in the past for facilitating remote control/monitoring of robots addressing few challenges posed by constraints such as mobility, power, and connectivity in the domain of MWSNs and swarms. An architecture(Middleware, Hardware etc.) was provided in [157] for the remote interaction with mobile robots via Internet. The hardware architecture of a teleoperated robot and the characteristics of the wireless channel are described in [158]. The architecture of a mobile robot and remote navigation system used for evaluating the RFID performance in localization is provided in [159]. A man-machine interface operating on a microcontroller with touch-based controls providing graphical programming features for controlling the robots is tested in [160]. An open-source system architecture was introduced in [161] for rapid prototyping of networked autonomous aerial vehicles. A novel architecture named 'R2P' is developed in [162] for prototyping robots using COTS by employing wired controlled area network protocol. A hormone-mimicking communication model is proposed in [51] and a multi-tier communication model is detailed in [163].

In spite of research being done in the area of modular robots for more than five decades, very little emphasis is given to development of communication strategies. Such approach is observed primarily due to the greater emphasis on the development of hardware robot models and due to lack of fully realizable applications using the modular designs. The modular robots differ in the perspective of their suitability for application in relative to swarm and MWSNs. Coordinated movements and centralized control are major traits in modular robotic application scenarios as each robotic module contributes to the placement and movement of an end effector. Such movements are realized in discrete steps through continuous control of individual robotic module in a guided manner and latencies requirements in communication vary as per the demands of the applications.

Though wired protocols are power efficient relative to wireless protocols and also lighten the load on the processing platform, the advantages derived from utilization of wireless communication are higher. Features such as distributed sensing, autonomous aggregation of individual robotic units with coordinated movement, autonomous reconfiguration, and dispersion from a biomimetic structure, remote access to various on-board resources on mobile units etc. can be realized efficiently with wireless communication. Though the wireless platforms explained so far are highly capable, they require client robots to be equipped with high-end processing boards and also the software on master/server side are platform (operating system and programming language) restricted. Hence, the platforms described above are not suitable for modular robots in which the load is shared among individual robotic modules and each module has very less processing capabilities.

The necessary processes involved in the development of a remote control and monitoring platform for control of the events/robots can be broadly categorized into

- Development of hardware for communication and control
- Development of middleware for translation of messages on different hardware platforms
- Development of human machine interface for translation of events in the scenario.

The development of hardware involves identifying the suitable communication interfaces for the robots in an application and necessary infrastructure to operate human machine interface (**HMI**) for facilitating rapid prototyping as well as executing tasks in run-time. The IEEE 802.15.4[164] and IEEE 802.11[165] protocol based wireless transceivers are standardized and widely accepted hardware radios for data transfer in robotics. The applications utilizing IEEE 802.11 radios are in general equipped with one or more PCs[166] as base station for handling heavier loads. The microcontroller boards with commercial off the shelf wireless interfaces like IEEE 802.15.4, CC2500[151], Bluetooth are used in applications of moderate complexity due to low data rates of communication. A detailed comparison of various COTS wireless interfaces is provided in [167].

Middleware in robotics plays a pivotal role of bridging the gap between the hardware abstraction libraries and the application software. It provides flexibility in the development of new algorithms while hiding the complexity of architecture underneath. Numerous middlewares have been developed for supporting various applications that require real-time guarantees (OROCOS[168], CLARAty[169], OpenRTMaist[170] etc.). Few middlewares like player/stage[171] provides easy integration and are applicable to numerous platforms due to packet oriented structure employed for control. A comprehensive study on different middleware architectures is provided in [172] [173] [174].

The development of HMI for the robotic applications is a multi-objective task with objectives such as providing optimal interface to interact with the events/ units in the experiment, simulator to test the scenario prior to hardware deployment, data logging, graphical display etc. The interfaces are often required to be implemented using standard tools like MATLAB, LABVIEW etc. due to short learning curves of the tools and well developed support libraries for centralized architectures. Many robot development environments are available in the market providing optimal HMI and simulators along with apt middleware coupled with them. The Microsoft Robotics Studio[175], Webots[176], teambots[177], CARMEN[178], Pyro[179] etc. are few robot development environments providing set of tools necessary for development of robotic applications.

Table 5.1. Demerits of robot development environments

Platform	Drawbacks
Microsoft Robotics Studio	Works only with supported OS on robot Windows OS is required on Host
Webots	Simulation only
Teambots	Simulation only Only two robots are supported
CARMEN	Only Linux Only few robots are supported
Pyro	Obsolete

As summarized in table 5.1, most of the robot development environments can support only few robotic platforms i.e. the multi-robotic simulations are possible only on the robots supported by the development environments. Most of the developments in these environments uses complex communication protocols between the host and the robots which uses resource hungry TCP/IP communication protocols and IEEE 802.11 communication interfaces. Modular robots being resource constrained devices require energy optimized communication interfaces as well software for communication that burdens the embedded platform to the least. The requirement of OS on a modular robot is minimal as the processor spends majority of time in executing locomotion with feedback control. The other activities are relatively less complex in relation to the applications for which robot development environments are designed to support and hence it is not necessary to provide an OS on a modular robot at an expense of energy conservation.

Another drawback observed in majority of the robot development environments are their dependencies on host operating system to operate on regular basis. Such feature implicitly conveys the nature of communication protocols and OS dependent features implemented in host as well as the robot. An ideal light-weight communication model that can be used for both centralized and decentralized communication is shown in figure 5.1. The centralized communication from host as well as de-centralized communication from neighboring robots can be translated by a Packet Translation layer in software of each modular robot for enabling remote monitoring and control and locomotion of a co-ordinated robotic system respectively. The robot development environments consists of a OS layer between Packet Translation layer and Hardware abstraction layer for executing the tasks. Such a feature is not a necessary requirement of modular robotics and hence can be avoided. It is also necessary that the same packet structure should be able to facilitate both centralized and de-centralized communication characteristics.

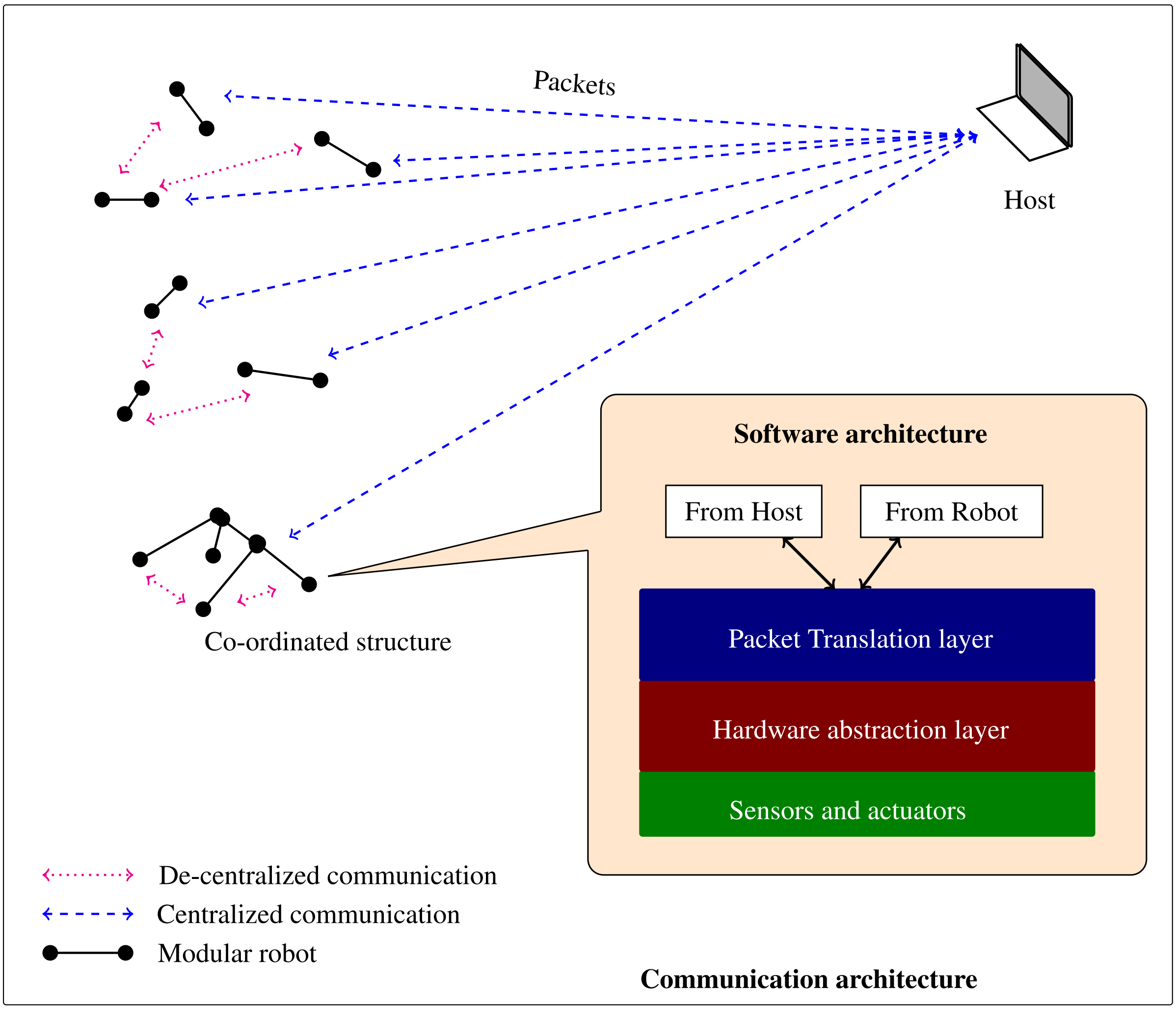


Figure 5.1. Communication and Software architecture of a modular robot

The Quanta platform development is inspired from packet oriented control of Player/Stage middleware in control architecture and teambots development environment for it's host independent nature of HMI. Though significant amount of research is done on handling the post prototyping phase of the development, no development environment has been found capable of handling the prototyping phases in robotics along with operational phases. The utilization of OS related features in the implementation of majority of middlewares makes their implementation less feasible for hardware platforms that do not require operating systems and Quanta addresses the same while making execution possible on light-weight platforms used for modular robots that generally employs 8/16-bit single core microcontrollers.

5.3 Quanta - Hardware Architecture

The software in the Quanta platform operates on FlexEye VSM described in chapter 4. Though the software is OS independent and can be implemented on any embededded platform with minimal reconfigurations, the communication on each modular robot is handled by FlexEye and hence it is optimal to add the packet translation layer to software in FlexEye instead of prototyping multi-processor platform by interfacing another embedded controller.

The host to robot communication can be handled in two approaches.

1. The host sequentially polls the robots and controls them as per the requirement
2. The host can utilize a gateway connected to it so that gateway only responds with necessary data to the host.

The major drawback of first approach is - the processes running on the host determine the latencies and the performance of the robotic system and network. Such approach can also lead to OS dependent software development and features which may be rendered obsolete in the future. The second approach provides better control over the scheduling events since a gateway is dedicated for the control and monitoring. It is easy to exploit the benefits of the COTS transceivers by controlling them via gateways instead of using bridges between the host and transceivers. Gateways can also provide features such as creating independent networks on using different channels and communicate across the networks with minimal latencies. Such features are crucial for modular robotics in which multiple modular robots might be working in a co-ordinated fashion and contending for the channel. The Quanta platform follows second approach due to its advantages in creating multiple networks and few latencies. The Quanta platform architecture is designed to be both operating system and development tool independent, light-weight over the air control and monitoring platform that can be utilized for static/mobile modular robotic units. The platform consists of the following modules.

- Base station
- Gateway node
- Client units

Quanta is capable of controlling/ monitoring tasks independently in three isolated wireless channels hence creating three separate networks (N1, N2, and N3) that can simultaneously generate tasks with interdependencies on robots in a given test scenario. Due to the server-client architecture used in Quanta, the target robot modules or hardware units are referred to as clients henceforth. The architecture of Quanta platform is shown in figure 5.2.

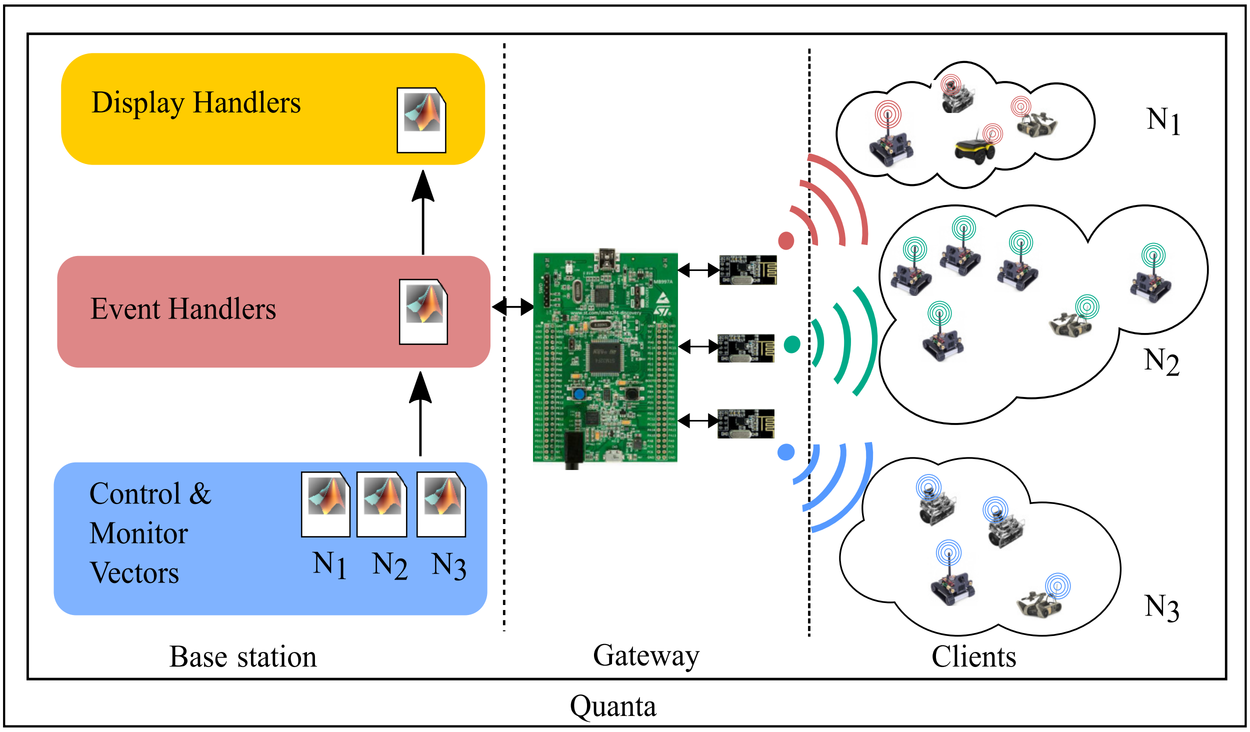


Figure 5.2. Quanta - Architecture diagram

5.3.1 Base Station

The HMI necessary for acquiring the data and controlling the clients remotely is operated from Base Station and has three components to handle the necessary activities in quanta.

1. The Control and Monitoring vectors

2. The Event handler

3. The Display handler

The Control and Monitoring vectors in the base station handles the parameters that are to be configured by the user for a control and/or monitor request. The Event handler collects all the requests to be transmitted to the networks and transmits them to the Gateway for dispatch. Upon transmission to Gateway, the Event handler waits for the data to be received from Gateway if the current transmission has a monitoring request and updates the same to the Display handlers. The Display handler updates the data to the user on the Graphical User Interfaces (**GUI**). The seamless integration of the control and monitoring process in tools like Network Simulator[180], LABVIEW etc is one of the major objectives of HMI for future inclusion of autonomous robotic events such as locomotion simulatio, network throughput analysis, power consumption analysis etc. in advanced applications. The Event and Display Handlers used in base station are implemented with very basic programming commands in MATLAB that can be easily translated to any programming languages on any operating system installed on base station. The modular approach followed in base station facilitates users in shifting operating systems and tools with minimal effort.

5.3.2 Gateway

The communication link between two nodes is the bottleneck in remote controlling and monitoring process. Though establishment of a communication link between two wireless transceivers is simple to envisage, the implementation of high-speed, low-power wireless communication that is acceptable for wide range of heterogeneous robots/clients requires detailed analysis. The Gateway plays vital role of bridging the gap between the user and the clients(robots) by employment of appropriate radio modules. The base station being a standard PC is capable of handling large volumes of data in very small amount of time and hence doesn't require rigorous analysis.

Considering heterogeneity in the hardware of clients and their capabilities, the radio modules for the control and monitoring are expected to impart minimal load on the robots to which they are interfaced. The processing of basic events like wireless channel contention, acknowledgments and retransmissions etc. in both indoor and outdoor environments are supposed to be handled optimally in order to avoid losses in terms of time and power consumption.

5.3.2.1 Gateway - Radio

Various COTS radio modules are employed for RF communications in the field of MWSNs and swarms for research while addressing the low power and high data rate constraints of wireless communication. The widely accepted COTS radio modules are Texas Instruments CC2500, XBee IEEE 802.15.4 modules[181] and standard IEEE 802.11 modules. Though the IEEE 802.11 radios are widely used because of the high data rates of communication, the power consumption is very high[182] and hence cannot be used on miniaturized robots for longer durations. The scalability of the IEEE 802.11 infrastructure without compromising the best performance is limited due to its control overhead. A recent development in the COTS radio modules by Nordic semiconductor - NRF24L01+ radio module [183] provides both high speed wireless link with lighter load on the microcontroller and hence can be chosen as alternative for soft real-time applications. Table 5.2 provides summary of the low power COTS radio modules using standard metrics. Though IEEE 802.15.4 (Zigbee) is another widely accepted protocol, the data rates of the platform are low and hence is not suitable for rapid control and monitoring. The practical data rates of IEEE 802.15.4 protocol are analyzed and provided in[184][185]. A detailed analysis on Bluetooth, Zigbee, Wifi wireless platforms are provided in [186]. The research mentioned so far indicates that the suitable choices for transceivers are CC2500 and NRF24L01+ modules for low power and high data rate wireless communication.

Table 5.2. Comparison of low power COTS radio modules

	CC2500	**NRF24L01+**	**XBee IEEE 802.15.4**
Air Data Rate(max.)	500 kbps	2000 kbps	250 kbps
Tx current	21.2 mA	11.3 mA	35 mA
Rx current	19.6 mA	13.5 mA	50 mA

The simulations on the performance of CC2500 and NRF24L01+ radio modules are not available after accounting real-time parameters like Tx to Rx and Rx to Tx switching times, payload sizes, control overhead etc. for a conclusion on the optimal choice of a transceiver for rapid control and monitoring. Table 5.3 summarizes major differences in the CC2500 and NRF24L01+

radio modules in terms of energy consumption, latencies and the packet structure. Though the NRF24L01+ radio is capable of communicating at 2 Mbaud, the maximum payload is 32 bytes - half of CC2500 radio and the switching latencies are relatively large.

Table 5.3. Comparison of CC2500 and NRF24L01+ Radio modules

	CC2500	**NRF24L01+**
Hardware parameters		
V_{Supply}	3V	3V
Operating frequency	2.4 GHz	2.4 GHz
Physical Modulation	OOK, 2-FSK, GFSK, and MSK	GFSK
Tx to Rx Latency	9.6 μs	130 μs
Rx to Tx Latency	21.5 μs	130 μs
Interface	SPI	SPI
Low power modes	Available	Available
Packet Parameters		
Preamble	2 bytes	1 bytes
Address field	1 bytes	5 bytes
Inbuilt CRC	2 bytes	1 bytes
Payload Size	64 Bytes	32 Bytes
Auto Retransmit	Not available	Available
Auto Acknowledge	Not available	Available

The state diagrams of the CC2500 and NRF24L01+ radio modules for a successful transmission and reception are provided in figure 5.3 and figure 5.4 respectively. Since the control and monitoring system is targeted for both indoor and outdoor environments, a simple communication protocol applicable for heterogeneous platforms requires an acknowledgment(**ACK**) for each packet transmission so that repetitive receptions due to reflections can be rejected. The above mentioned requirements increase control overhead and reduce the maximum throughput of communication. The real-time events and latencies and their order of occurrence that are accounted for precise throughput calculations of NRF24L01+ and CC2500 radio modules are provided in table 5.4 and figure 5.5 respectively.

The NRF24L01+ and CC2500 radio modules differ significantly in the procedure of handling the ACK packets and retransmissions. The NRF24L01+ radio is equipped with enhanced shock burst (**ESB**) technology for auto acknowledgment handling, retransmissions and discarding the multiple receptions. The absence of such technology in CC2500 transfers the load to the microcontroller interfaced to the radio for performing operations such as

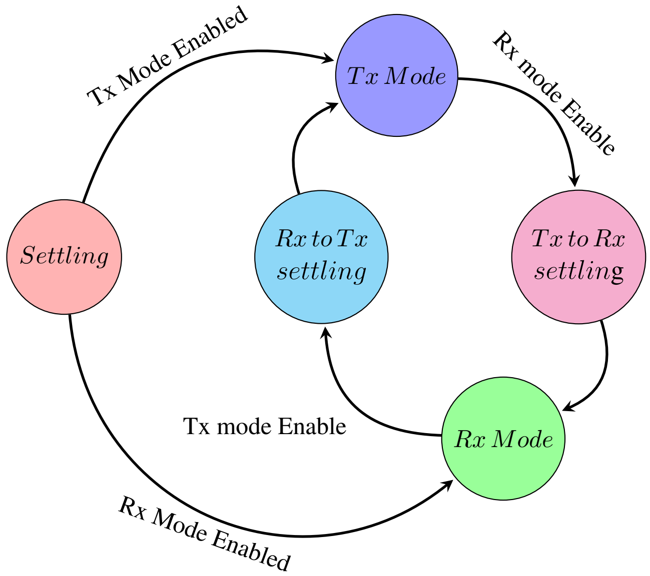


Figure 5.3. State Diagram - CC2500 Radio

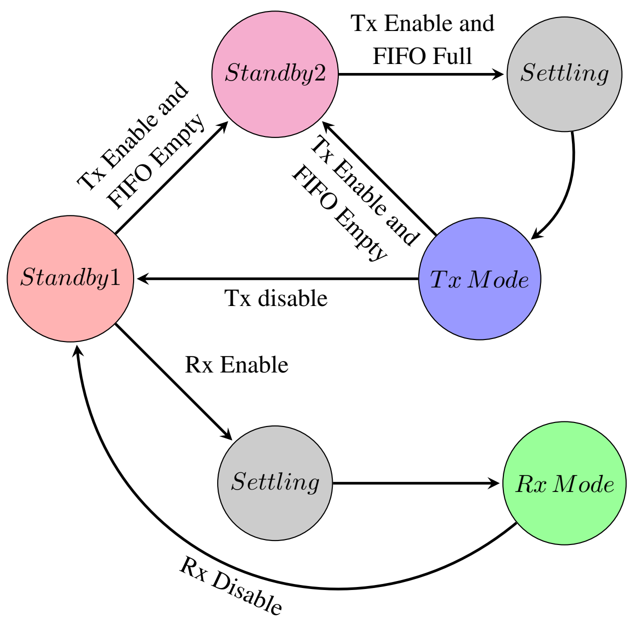


Figure 5.4. State Diagram - NRF24L01+ Radio

- Analyze the packet for Message ID
- Frame ACK packet
- Upload ACK packet
- Discard multiple receptions

Table 5.4. Real-time communication events and latencies in transceivers

Parameter	Description
UP_{data}	Time elapsed for uploading payload data to a radio $=\frac{Payload\,Size}{SPI\;Data\,rate} + DLY_{SPI} = DW_{data}$
OTA_{Data}	Time elapsed during data transmission in transmitter $= Packetsize * Datarate$ =Time spent for data reception by receiver
$Int.$	Interrupt servicing & packet processing
DW_{data}	Time elapsed for downloading data from radio
$Tx \Rightarrow Rx$	Transmit to receive switching time of radio
$Rx \Rightarrow Tx$	Receive to Transmit switching time of radio
OTA_{ACK}	Time elapsed during ACK transmission in transmitter Time spent for ACK reception by receiver $= ACK\,packet\,size * Datarate$
UP_{ACK}	Time to upload the payload ACK to radio module $=\frac{ACK\,packet\,size}{SPI\;Data\,rate} = DW_{ACK}$
DW_{ACK}	Time elapsed for downloading ACK from radio
DLY_{SPI}	Latencies due to time gap between SPI Byte transfers (calculated @ 5MHz SPI clock frequency)

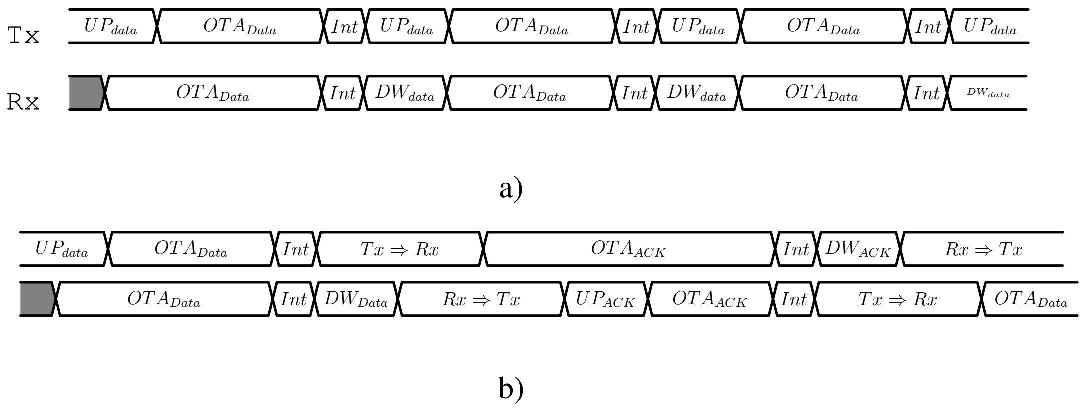


Figure 5.5. Simplex communciation Tx - Transmitter, Rx - Receiver - a) without Acknowledgement b) with Acknowledgement

- Send ACK to the previous packet in case of lost ACK.

The microcontroller interfaced to the transmitter is also burdened with the task of retransmitting the previous packet by identifying and uploading the data/ACK in case of lost data/ACK packet. The throughput and power consumption analysis including control overheads for three different scenarios were simulated in the Network Simulator tool (**NS-2**) version 2.34. The energy consumption model of the Network simulator tool was employed to identify the approximate lifetime of a radio with initial energy budget of one joule in various scenarios.

Scenario 1 and Scenario 2 Scenario 1 and scenario 2 are the point to point communication simulations in NS-2 under ideal conditions including control overheads and without the practical parameters listed in table 5.4 except OTA_{Data} and OTA_{ACK}. The ACK for transmissions are disabled in scenario 1 and enabled in scenario 2. The simulations are performed till one of the transceivers participating in the communication reached zero energy state. The energy consumption and payload delivery results from receiver side are plotted in figure 5.6 and figure 5.7 respectively. The ideal payload data rates of NRF24L01+ and CC2500 radio modules were determined by the simulation as 1684.957 kb/sec and 463.797 kb/sec respectively for the ideal conditions without ACK for each packet. The ideal payload data rates with ACK. for each transmitted packet for NRF24L01+ and CC2500 radio modules were observed from the simulations as 1344.273 kb/sec and 410.5 kb/sec respectively.

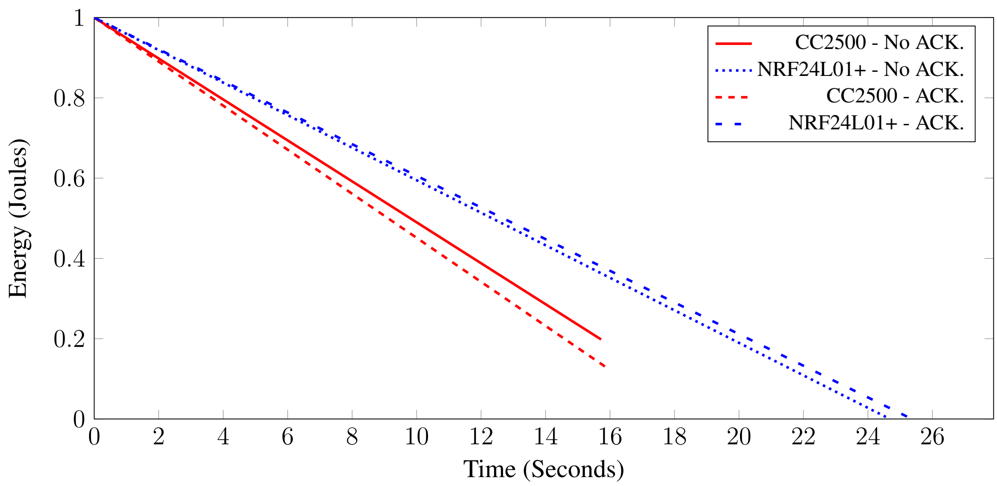


Figure 5.6. Simulation on Scenario 1 Vs. Scenario 2 - Energy consumption

Scenario 3 and Scenario 4 Scenario 3 is a simulation of point to point communication without ACK for both radios while accounting for all the real-time parameters listed in table 5.4. The results of the simulation are provided in figure 5.9. The payload delivery without ACK enabled for the NRF24L01+ and CC2500 radios as per the NS-2 simulations are 723.293 Kb/sec and 414.082 Kb/sec respectively. The scenario 4 is a simulation of point to point communication with ACK.

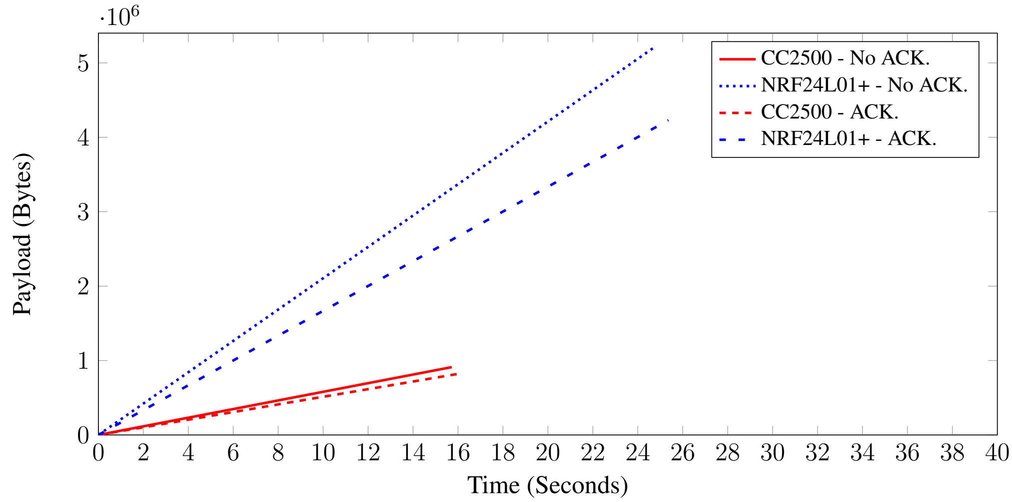


Figure 5.7. Simulation on Scenario 1 Vs. Scenario 2 - Payload delivery

including all the real-time parameters listed in table 5.4 and latencies in the order of events shown in figure 5.8. The payload delivery with ACK enabled for the NRF24L01+ and CC2500 radios as per the NS 2 simulations are 477.734 Kb/sec and 305.526 Kb/sec respectively and results are shown in figure 5.9.

The comparison provided in figure 5.10 on all the four scenarios concludes that the data throughput tend to diverge significantly from the theoretical data rates due to the hardware limitations. The NRF24L01+ modules due to it's inbuilt ESB technology imparts lighter burden on the microcontroller interfaced to them for operation and also provides much better payload delivery rates in relation to the CC2500 radios and hence an optimal choice for energy efficient RF communication in the remote control and monitoring system with heterogeneous clients.

5.3.2.2 Gateway - Processing platform

The gateway node was implemented with NRF24L01+ radios interfaced to a STM32F4 Discovery board for facilitating the remote connection between the base station and the clients. The STM32F4 discovery board acts as a network core bridging the connections between base station and clients with the help of three NRF24L01+ radios. Each radio employs different channel for communication with the clients and hence provides three virtual sets of clients for concurrent control and monitoring. The concurrent processing of control and monitoring events for three virtual networks as interpreted in figure 5.2 is performed by utilizing two DMAs present in the

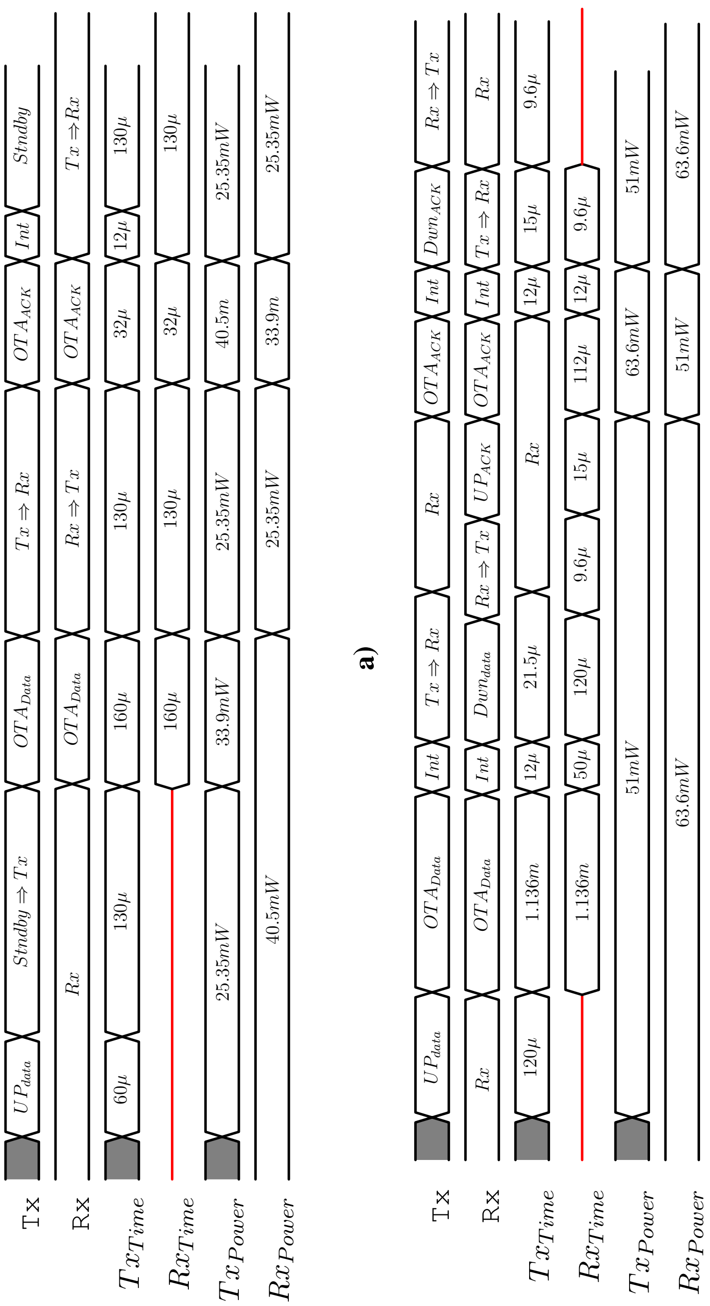
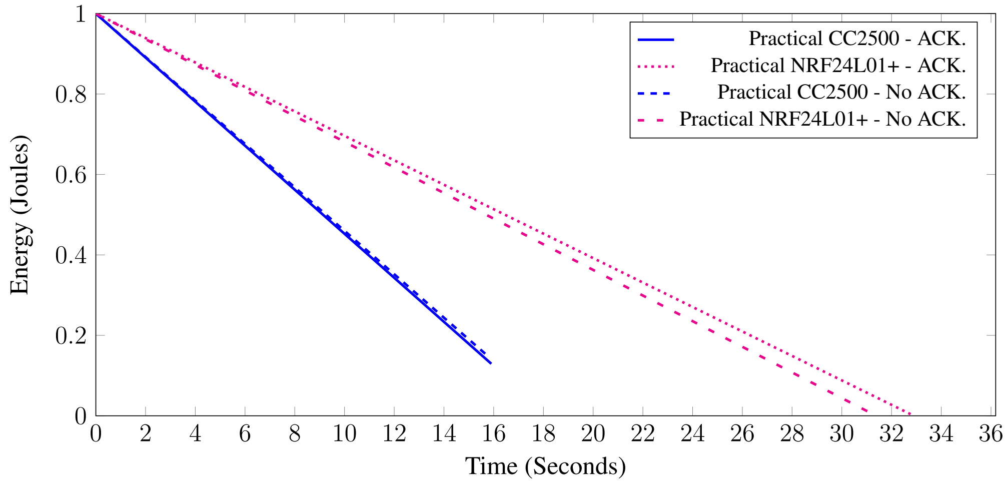
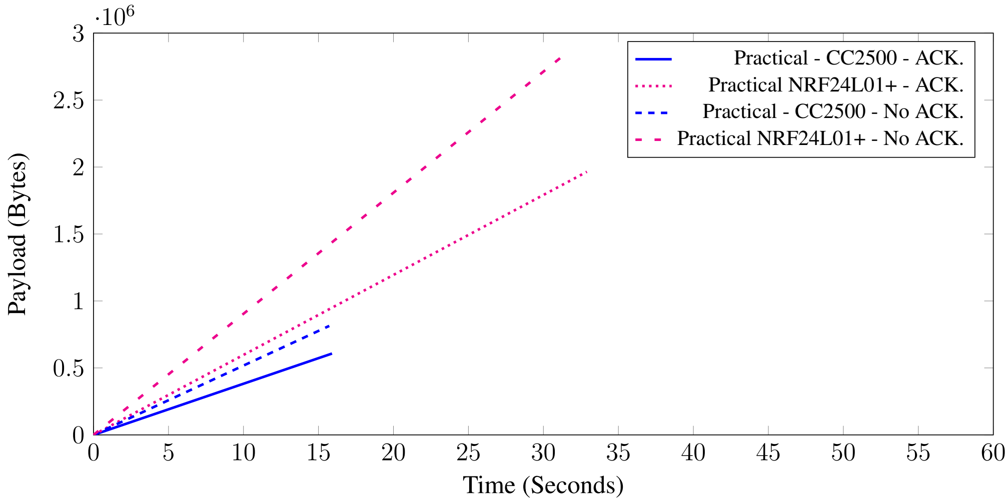


Figure 5.8. Latencies and Power consumption parameters **a)** NRF24L01+ radio module with Enhanced shock burst technology enabled for communication with ACK **b)** CC2500 radio module for communication with ACK (* Tx_{time} & Rx_{time} are sequence of latencies that occur during point to point communication in transmitter and receiver respectively. ** Tx_{Power} & Rx_{Power} are approximate power consumptions of transmitter and receiver radio modules calculated for shown duration from the manufacturer datasheets)



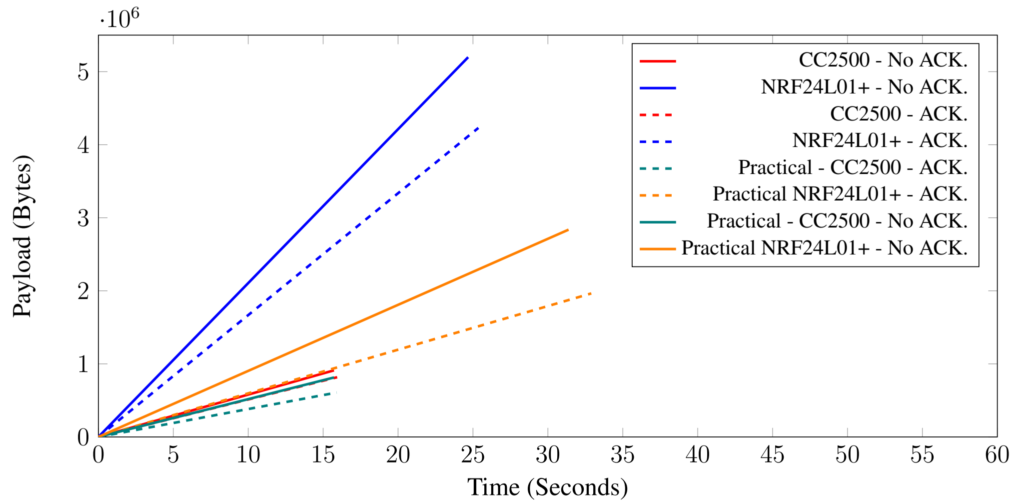
a)



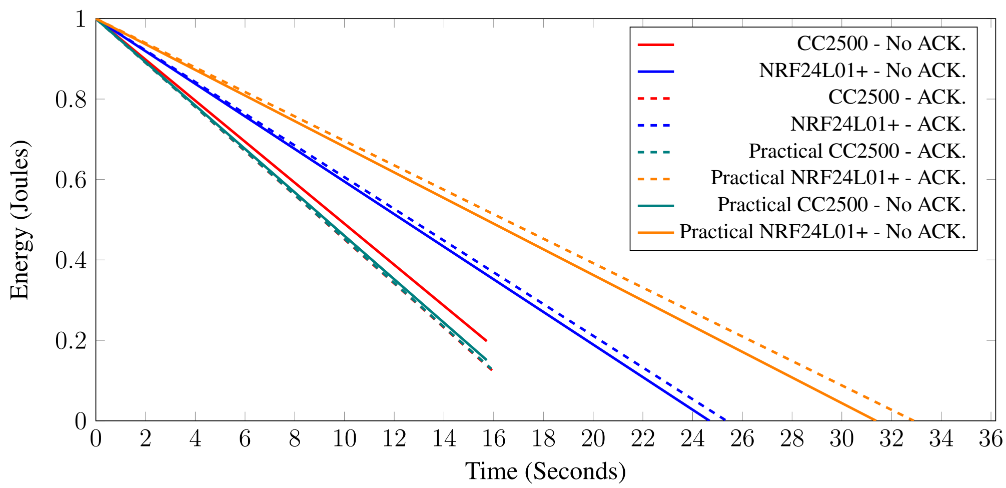
b)

Figure 5.9. Scenario 3 Vs. Scenario 4 - a) Energy consumption b) Payload delivery

STM32F407VGT6 microcontroller. The DMAs aid in minimizing latencies during sequential data upload/ reception and control instructions via SPI to the radio modules. The internal hardware architecture of gateway node is shown in fig 5.11. The Gateway is capable of storing data in large volumes and the specifications on the power consumption, performance, concurrent capabilities etc. are made available in chapter 4 of the thesis and published in [121][187].



a)



b)

Figure 5.10. Comparison of the ideal and practical communication scenarios

5.3.3 Client

The Quanta platform is designed to operate without any dependency on the hardware architecture of clients and is programmed to access the clients at one-hop distance from the gateway. For experimental purposes, the clients were equipped with the same microcontroller board employed in gateway node and also with few more sensors like OV9655 camera, motion sensor, servo motors etc. to test the best throughput and functionality of control and monitoring system.

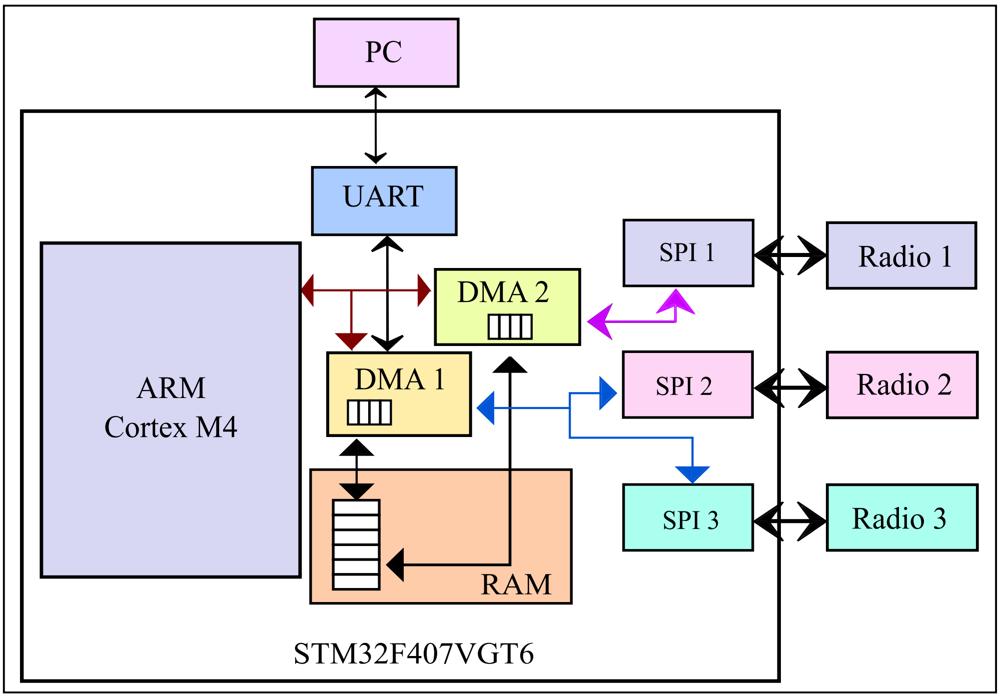


Figure 5.11. Gateway - processing platform internal hardware architecture

5.4 Quanta - Software Implementation

The communication in quanta follows server-client architecture with the base station playing the role of server and the robots/units in the experiment behaving as clients. The concurrent control and monitoring in quanta platform are feasible due to the generic packet structure used in conjunction with a strategic protocol for processing events. Every event (control and/or monitoring activity) in Quanta originates from front-end tool operated from a base station. The user has the flexibility to choose the front-end tool and the operating system as long as the packet structure (explained later) for communicating the events to the gateway is maintained. The software in quanta is operated on a specific packet structure and a protocol designed to provide numerous features for control and monitoring the events on clients.

The protocol and packet structure facilitate user with numerous optional features of concurrent control and monitoring in the same or neighboring networks. These features aid researchers to control the internal as well the external events in an experiment scenario remotely and log the results through the quanta. The software in quanta consists of three platform independent components. The components are

1. front-end interface
2. Packet Structure
3. Protocol Stack

5.4.1 Front-end interface

The control and monitor events envisaged as

- Simple read and write operations of variables in Random Access Memory(**RAM**)/registers in the client's microcontroller.
- Execution of functions written in high-level programming languages like C, C++ etc. from the client(s) program memory.

The quantity and type of such events can vary with each application. The front-end interface consists of MATLAB script files for each network consisting of a set of vectors and arrays that can be broadly categorized into Opcode, Priority, Operator and Data vectors. The four types of vectors together are referred as Control & Monitoring vectors and are employed to facilitate the user with numerous options to control the events in the clients. The hierarchy of software architecture and implementation of front-end using various vectors and arrays in Quanta is provided in figure 5.12.

The client vector in each network script file is a priority vector and is used for assigning priorities to each client ($1\, to\, m$) in a network. The Opcode vectors ($1\, to\, n$) for each client are used for conveying the parameter identity for control and monitoring events in a client. The Operator vectors (p bits for each opcode) are used for providing the details on function of the parameter being sent/requested for control/monitoring. The data vectors (p bytes for p bits in the operator vectors) contain the control information. For example, a user can remotely control the direction and angle of rotation of a specific motor by writing the custom Opcode assigned to the that motor into a Opcode vector slot, enabling a single-bit corresponding to that motor direction in a Operator vector slot and proving the data on amount of rotation in the Data vector .

The priority vectors are also present in the Event handler as shown in figure 5.12 and are used for assigning priorities to the networks using which the gateway transmits/receives packets in a

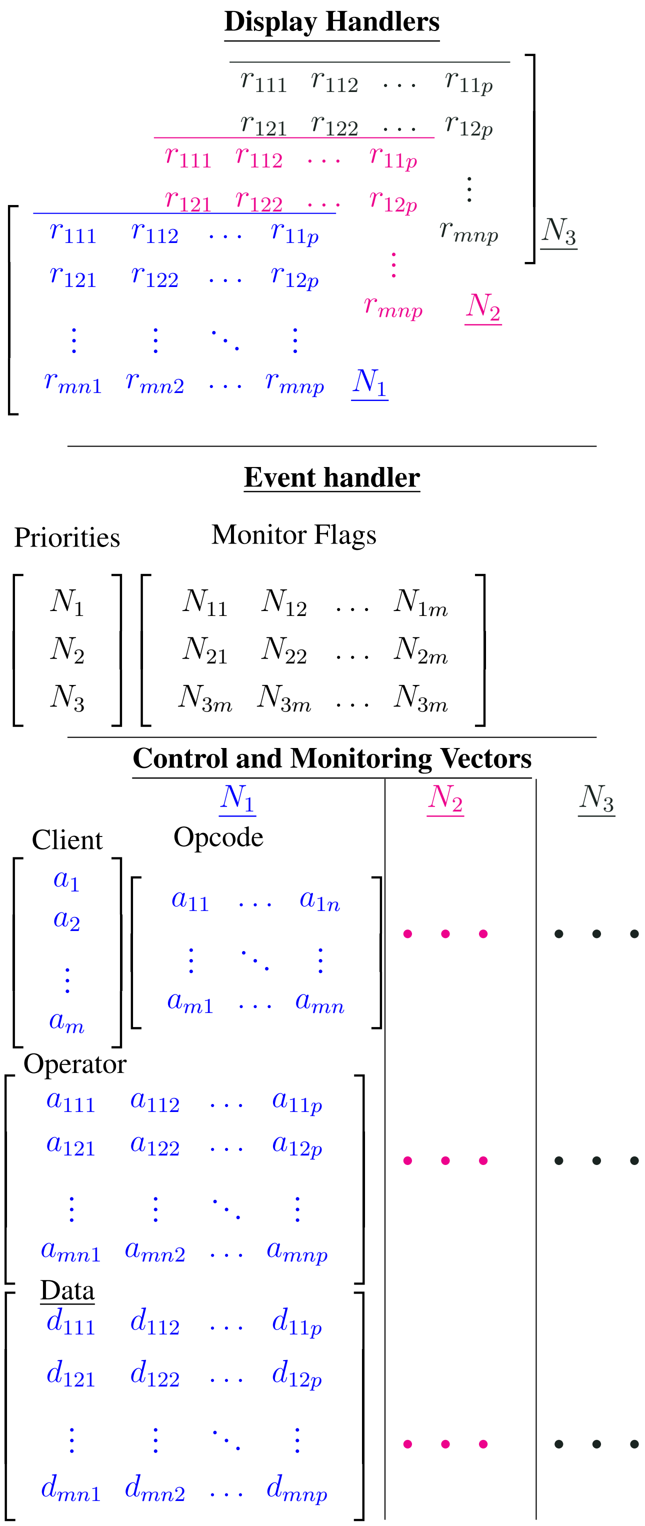


Figure 5.12. Quanta - Vectors and Front-end hierarchy

orderly fashion. The user can enable concurrency in the events of two networks by assigning same priority value to two or more networks or sequentially organize the events by assigning priorities in the increasing/ decreasing order as per the requirement. The events scheduled for transmission to each client in their respective network are processed by the Control and Monitoring vectors of the respective network and the entire matrix is forwarded to the Event handler. The Event handler analyzes events in three networks for the presence of monitoring events and updates the Monitor flags. The Event handler forwards the events to Gateway in a specific packet structure and waits until it receives data from the Gateway in case of presence of monitoring events. The data from monitoring events are updated to the vectors in Display Handlers after successful reception.

The gateway uploads packets to more than one radio in parallel or sequentially depending on the network priorities. In each network the messages are transmitted to multiple clients in the order of priorities enabled in client vectors of that specific network. The order of opcodes in each packet transmitted to each client can also be controlled by entering the high priority opcodes first and rest later in the Opcode matrix. The Opcode and Operand number assignment for a particular functionality is up to the user and can be random integer numbers chosen within the bit limits of the packet (explained in next section). The maximum number of parameters controlled/monitored for an Opcode is limited to 8 with 1 byte for each parameter in the data matrix.

5.4.2 Packet Structure

The control and monitoring over the air in Quanta is achieved using a custom designed packet structure while maintaining the platform independence features. The opcode vectors in front-end updates the opcode field in the packet that control events that are to be scheduled on clients. In the similar manner, the operand vectors enable bits in the operand byte in a packet for an opcode and data vectors provide data for the parameters mentioned in operator field for an opcode. The various combinations and features possible with an opcode, operand and data vectors are summarized in table 5.5. The opcode and operand vectors refers to user-defined functions and parameters in the clients. For example, if the DC motor control is defined by assigning an opcode and the change of velocity or direction can be conveyed using an operand. The data necessary to convey the amount of deflection, direction etc. for the DC motor are provided in the data fields of packet.

Table 5.5. Quanta - Packet structure and functionalities

Packet field Name	**Bit Nos.**	**Description**
Client_ID	0-7	Destination Client ID
No. of Packets	8-23	No. of 32 byte packets pending
No. of Opcodes	24-31	No. of operations to be performed
Opcode	32-36	code of current operation
	37-38	00 - Pre-concurrent operation 01 - Concurrent operation 11 - Post concurrent operation
	39	0 - Control event 1 - Monitoring event
Operand	40-47	No. of parameters sent in data fields for an Opcode. 1 bit for each operand.
Data	–	Parameter data sent/received to/from client respectively

The packet structure shown in figure 5.13 is commonly used for both control and monitoring of the clients. The address of client scheduled for a event is identified by gateway using the Client_ID header byte in the packet and the gateway disassembles the Client_ID from packet before transmission and appends clients transceiver address. The second and third bytes (bit no.s 8-23 in table 5.5) conveys the information regarding number of packets pending after the initial packet. The Opcode is a multipurpose field conveying information on four different parameters of a event in the scenario. The five least significant bits (**LSB**) in the Opcode byte (bits 32-36 in table 5.5) conveys information on type of event to be implemented on client's hardware (reading sensors, controlling actuators etc.). The Opcode field with the remaining bits (bit nos. 37-39) conveys numerous options like concurrency in control & monitoring a specific event. The Opcode byte is always associated with an Operand byte for identifying the type of parameter (x-data of compass, angle of servo etc.) transmitted / requested in a control / monitoring event. The enabled bits in the Operand byte have an associated byte in the Data field updated by the base station for a control

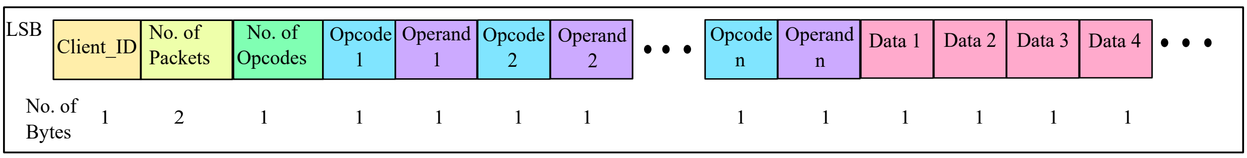


Figure 5.13. Quanta - Packet Structure

event or updated by the client for a monitoring event before transmitting the packet. The control data (bit nos. 0 - 31) are appended only in the initial packet and are followed by opcodes, operands and data fields.

5.4.3 Protocol

The opcode and priority vectors control the execution order of various events and the protocol handles timing of the events. The activities in Quanta are categorized into

- Pre-Concurrent events
- Concurrent events
- Post-concurrent events

The bits used for identifying concurrency in an opcode of a packet(37-38 in table 5.5) are disabled for Pre-Concurrent events. The Concurrent and Post-Concurrent events are updated with '01' and '11' at the same bit locations of an Opcode respectively. The Pre-Concurrent events can be control and/or monitoring activities and aids in resetting an experimental testbed to a initial state for experimentation later. The Concurrent events are primary activities of the experiment whose results are fetched for further analysis. The Post-Concurrent events are conceived as post experiment activities like acquiring results, controlling the clients to fail safe state before switching off etc.

The scheduled monitoring events on the clients in a network are organized by the Gateway through polling of each client. The state diagrams shown in figure 5.14 and figure 5.15 summarize the protocol implementation on Gateway and Client respectively. The Gateway initializes radio(s) to the address mentioned in the Client_ID byte of the packet for higher priority network(s) and transmits rest of packet(s) to the client after disassembling the Client_ID header. In case of scenarios with more than one network having equal priorities, the packets are sent in parallel from different radios with an infinitesimal delay. The minimum delay is achieved due to the employment of DMAs in the gateway controller for the SPI data transfers to the NRF24L01+ radios instead of microcontroller in uploading / downloading data and control commands.

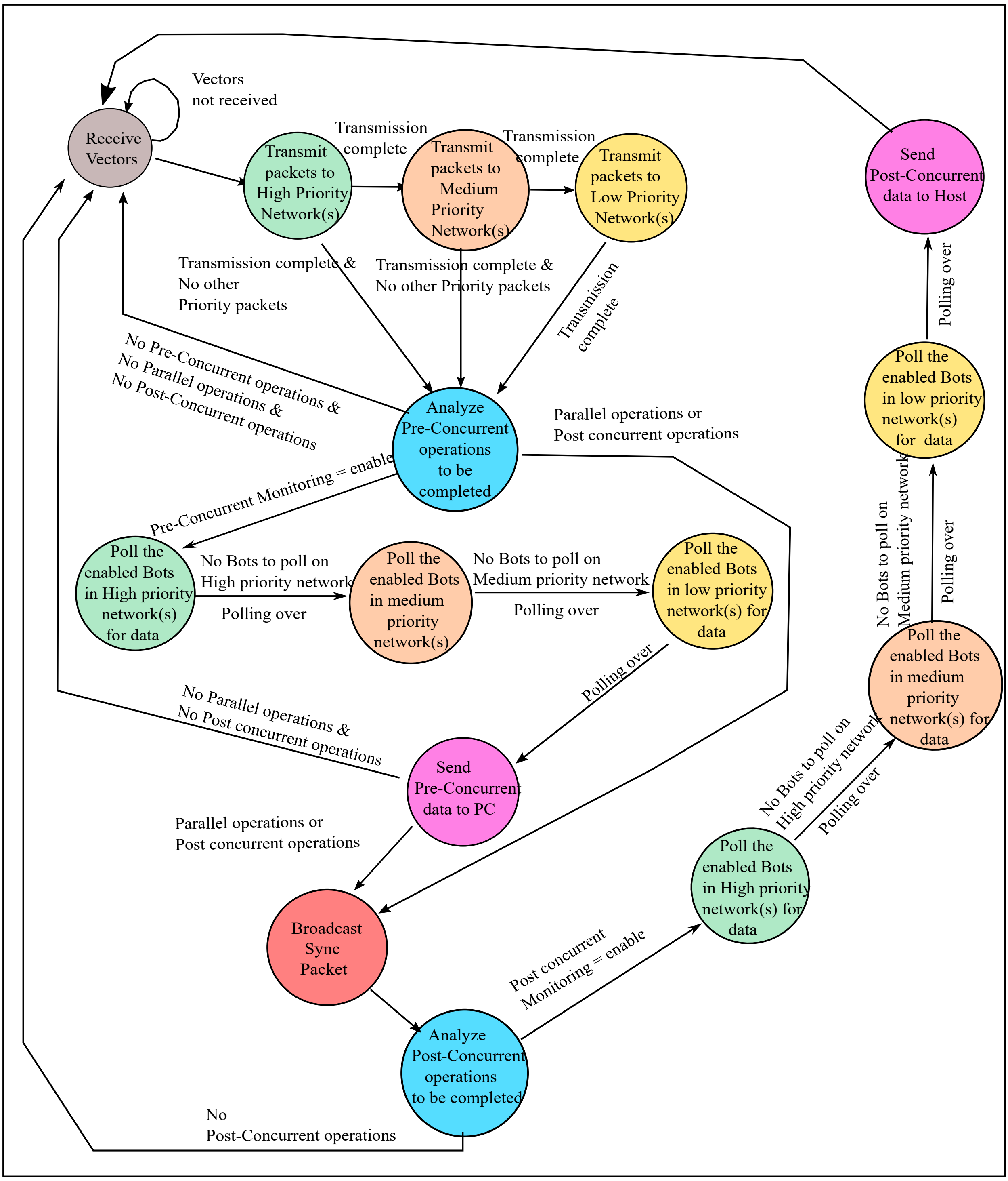


Figure 5.14. Quanta - state machine of protocol on Gateway

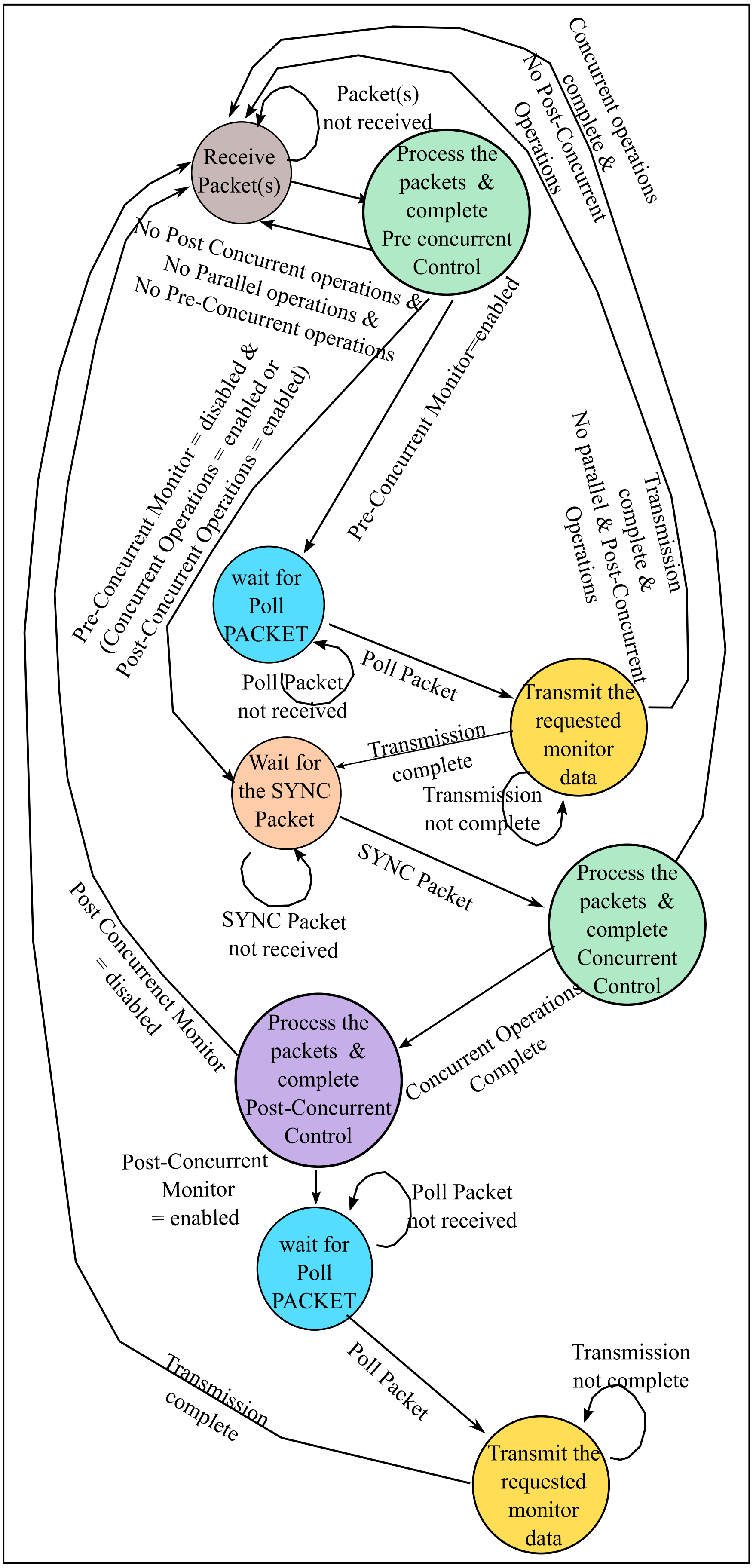


Figure 5.15. Quanta - state machine of protocol on client

The gateway segregates and generates a list of Concurrent and Post/Pre-Concurrent events from network matrix received from the base station before the dispatch of packets. Though the entire packet is dispatched to respective clients, the execution of events are partially controlled by the gateway. The clients processes control and monitoring activities in the order of opcodes sent in the packet after reception of all the packets and the pre-Concurrent activities are executed without any control of the gateway. In case of presence of an opcode with monitoring enabled in pre-concurrent event(s), the client waits for the Gateway to poll for it as shown in figure 5.14. The client transmits fetched data in the pre-concurrent activity to the gateway after receiving a single byte Poll packet. The concurrent activity in all the clients is initiated by gateway with the aid of a single byte SYNC packet. The SYNC packet is broadcasted by gateway to all clients after collecting the Pre-concurrent monitoring data from all the network(s).

The client nodes processes all the opcodes with concurrency enabled in the order arranged in packets upon reception of the broadcasted SYNC packet. Since the response times, processing time etc. of clients can be different due to heterogeneity in hardware, the transmission of SYNC packet is to be controlled through HMI and is enabled after the completion of pre-concurrent events. The concurrent events are controlling events and primary experiment events. The post-concurrent control and monitoring events are processed by the clients after the completion of concurrent events and clients waits for the Gateway to poll them in case of presence of monitoring request to the client in post-concurrent activities.

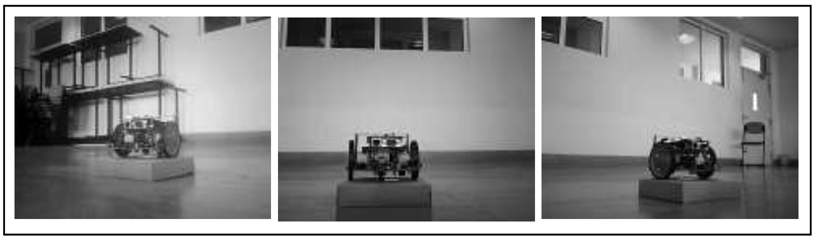
The pre-concurrent, concurrent and post-concurrent events in Quanta are designed to be optional. The user is facilitated with the flexibility of prioritizing/ ignoring the events as per the experiment requirements using suitable front-end tool in the base station. The clients require basic drivers to control the radio and a middleware to translate the packet contents to various functionalities. The Middleware of Quanta is intended to provide read/write access to variables in memory, controlling and monitoring hardware events, executing set of functions from various software stacks etc.

5.5 Results and discussions

The experiments were conducted for wireless data transfer and control using COTS NRF24L01+ radio modules. The maximum payload delivery rate of NRF24L01+ radio module is observed as 738.75Kb/sec without ACK. and 476.675Kb/sec with ACK. Experiments were conducted to validate the various combinations of priorities and opcode types. A middleware was developed for interpreting the protocol and executing the provided operations. A process was assigned to each opcode whose input parameters are controlled by the operators. A series of sequential and parallel image acquisition tasks were requested on three nodes in indoor environment with ACK enabled from separate networks via the protocol developed and acquired images are shown in figure 5.16. Each image received is of 160 x 120 resolution and 19.2KB in size. The images are split into 600 packets of 32 bytes each for over the air transmission. It has been observed that the during sequential acquisition each image was acquired in an average period of 471.3127ms and multiple receptions of the same packets was observed. The algorithm switched to acquisition of image from next network in sequential mode in 6.686 ms. During parallel acquisition it has been observed that all three images are acquired with in a total period of 472.2424ms with 133.7μsec delay in the start of reception of images from different networks. The major outcomes from the quanta platform are provided in table 5.6.

Table 5.6. Quanta platform - performance and merits

Parameter	Performance
Transceiver payload without ACK.	738.75 Kb/sec
Transceiver Payload with ACK.	476.675 Kb/sec
No. of networks	3
No. of addressable nodes in a network	> 300
Concurrent control/monitoring delay	133.7μs
Load on client	Minimal
Operating system dependencies	NA
Integration into development tools	Highly possible
Application stages	Prototyping Deployment



Right View Front View Left View

Figure 5.16. Quanta - Concurrent acquisition of Images

The NRF24L01+ radio modules though compromises data rates for reducing power consumption in relation to IEEE 802.11 radios, the energy conservation in the entire system is significant for the chosen approach to communication. The robot development environments summarized in section 5.2 employed IEEE 802.11 radio modules for data transfer and very little attention was given to the total energy consumption in the system. Though an argument such as small transmission time in IEEE 802.11 radios save power in relative to large transmission time of NRF24L01+ can be made, the clients equipped with radios in the system will be always active to service a request from server and idle listening, overhearing the channel contribute to major portion of energy consumption in the system. Hence, the total budget for energy will increase for given amount of time in case of IEEE 802.11 radios and provides less time for experimentation in case of custom-made robots with 8-bit microcontrollers and low power batteries. Moreover, Quanta packet structure facilitates organization of multiple events using minimal payload and hence the impact of transmission time on the system's performance is minimal. Since the majority of the processing is done by gateway and the transceiver, the power conservation by frequency scaling on clients explained in chapter 4 will have very minimal affect on quanta. Hence, it is feasible to implement wireless control and monitoring using quanta on 8-bit microcontroller boards or 32-bit microcontroller platforms like FlexEye with scaled down operating frequencies.

Implementation of control and data acquisition can be quite rapid and energy-efficient using Quanta in relation to conventional RDE. Three virtually separated networks created by Quanta will be significant aid for prototyping and validating modular robotic applications in which multiple active coordinated robotic systems exist. Such a feature is possible due to utilization of DMA for concurrent data transfer between gateway and the transceivers. The synchronized control mechanisms of communication and control can also facilitate real-time execution of events on clients if necessary.

Unlike the conventional RDEs which require OS on robotic modules, Quanta requires a simple packet translation layer for performing the remote monitoring and control. This is possible because the information dispatched and received can be user customized functions that are either atomic functions like reading sensors, controlling actuators etc. or context based executions like navigating to a specific co-ordinates without any specifications on locomotion or control. Such features can aid in the development of a generic software structure rather than language specific or OS specific that can be controlled as required by the user. It is possible to use the generic software structure on new robots without much modifications in Quanta as it is completely independent of robot sensor and actuator interfaces.

5.6 Summary

The major outcomes from the research on communication strategies for modular robots and development of the quanta platform are

- A platform facilitating control of events(Control and monitoring) in robots. The events can be customized as per the requirement of the application. The concurrent execution of events on multiple robotic units in a network or multiple robots in different networks can be scheduled at a minimal latency of 133.7μsec for testing modular robots in real-time.
- A hybrid packet structure and protocol that can be utilized for both peer to peer communication among robots as well as client-server model of communication between base station and robots. The implementation of centralized or distributed control and monitoring strategies is feasible in quanta. This is possible because of dependency of execution of activities on packet structure instead of the packet source.
- A platform independent front-end interface facilitating easy integration with the standard development tools like NS-2, MATLAB, LABVIEW etc. Since the control and monitoring are initiated through vectors and arrays in Quanta, it is possible to implement control logic in any development tool independent of the operating systems and their drivers. Hence, Quanta can be used in future technologies, for improvisation in HMI as per the application requirements
- Middleware in Quanta provided flexibility in addressing heterogeneity in the hardware and avoiding OS dependent features. The middleware developed for testing Quanta has a short learning curve and due to the Quanta's packet structure and it's atomic nature in controlling the events it requires minimal time for modification as per the requirements of a new application. The middleware being a light-weight design, it can be used for monitoring and controlling the prototypes for lightly-capable platforms using 8-bit/16-bit microcontrollers.

Chapter 6

Integration of systems

6.1 Introduction

The integration of various components of a multi-objective platform like modular robotics is a challenging task. Along with current requirements, it is necessary to provide scope for addition of the few interfaces such as transducers and sensors in future as per the changing application demands. Though research in modular robotic designs never extended towards localization, navigation, and co-ordination algorithms, a fully integrated system needs to address these challenges in a real-world applications in which robots are not aware of their location nor direction to proceed. Conventional wired communication strategies used in modular robotics will be of very little use in the real-world scenario where deployments are mostly random and adaptive.

Figure 6.1 provides details on various components of modular robots that are to be embedded into design, control and communication hierarchies for independent and robust operations. The design of the chassis and interfacing mechanisms for docking plays a crucial role in providing the structural adaptability to the robot as per the application requirements. Through intelligent design of a modular robot, it is possible to provide numerous DOF to the end-effector with minimal DOF on each individual robot. Such intelligent utilizes the advantages of structural symmetries and docking mechanisms.

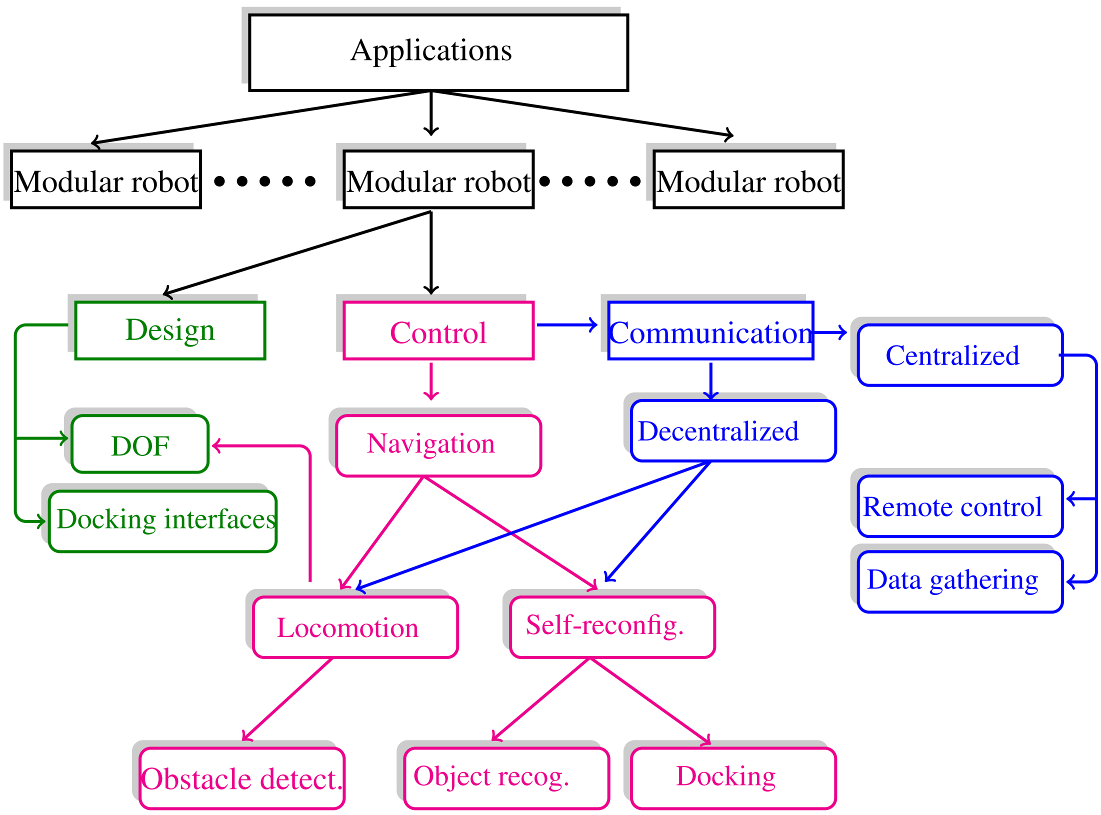


Figure 6.1. Various operational components of modular robots

After the structural design, the major emphasis is needed to be given to the control part of the design - the embedded system. In conventional designs, the embedded system is given very little importance due to diversion of research resources to structural design which is further enhanced due to lack of embedded technologies. A fully integrated embedded system requires numerous sensors to track movements, provide feedback control and communicate with necessary robots/servers for information/data transmission. Trimobot and S-bot robotic modules are examples of such integrated systems. The capabilities of embedded system used in Trimobot are limited to docking and locomotion and the S-bot capabilities using Intel XScale processor running LinuxOS and controlling directly the sound and camera interfaces are more comprehensive which includes localization, navigation, assembly and disassembly. In terms of applicability in real-world applications, it is much clearer to state that S-bot is more suitable than the Trimobot since S-bot is equipped with capable platform with more processing features.

Communication is another feature of the control system using which a remote user can monitor an event in a environment or a co-ordinated robotic structure can navigate through the same. Centralized and de-centralized communication capabilities are necessary requirements of the control system as shown in figure 6.1. The centralized capabilities of the communication system can be used by a remote user to monitor a particular robot in terms it's functions, errors etc., guide a co-ordinated robotic system into a environment, provided information regarding next structure that needs to be adapted into etc. The decentralized capabilities of communication involving robots communicating with each other for autonomous docking or exchange of information of various sensors like accelerometers, positions of various joints etc. during reconfiguration etc.

Figure 6.1 explains the operations of an embedded system in various phases of navigation. Apart from controlling the locomotion by activating the actuators at various DOF, the stability of the system needs to monitored continuously so that the immediate destination in path and the stable structure necessary for it needs to be calculated in limited amount of time. Obstacles in the path are also supposed to be recognized while navigating through the environment so that the structures need to be reconfigured accordingly. The self-reconfiguration feature of the modular robot also has to be guided by the control system by identifying the robots in the neighborhood and completing the docking process accordingly in 2D as well as 3D scenarios. Though the above mentioned tasks can be executed on a 8-bit/16-bit platforms, the performance of such systems will be considerably low and will not suit the real-time demands of the applications. The S-bot robotic design proves the same. Hence, it can concluded that a capable platform equipped with necessary intense computation support along with power optimizations is mandatory for the application.

6.2 Integration of Systems

As a part of development of a modular robot for the real-world applications capable of biomimicking, the validation of various components of the robot such as structural design, communication interfaces, docking mechanisms and embedded systems are started concurrently. The FlexEye visual camera module explained in chapter 4 was first to be finalized because of its capabilities in terms of on-chip memory, minimal power consumption, on-chip camera interface and pin-outs available for large number of sensors and actuators. It is integrated with a radio module and camera as shown in figure 6.2 for supporting communication and object recognition respectively as mentioned in figure 6.1

Figure 6.2. FlexEye Visual Sensor Mote

After validation of the capabilities of the electronics platform, the platform is integrated into a COTS mobile platform with IR sensors and encoders for completing a robotic module. The platform is shown in figure 6.3. The robot is a multi-processor configured to operate in

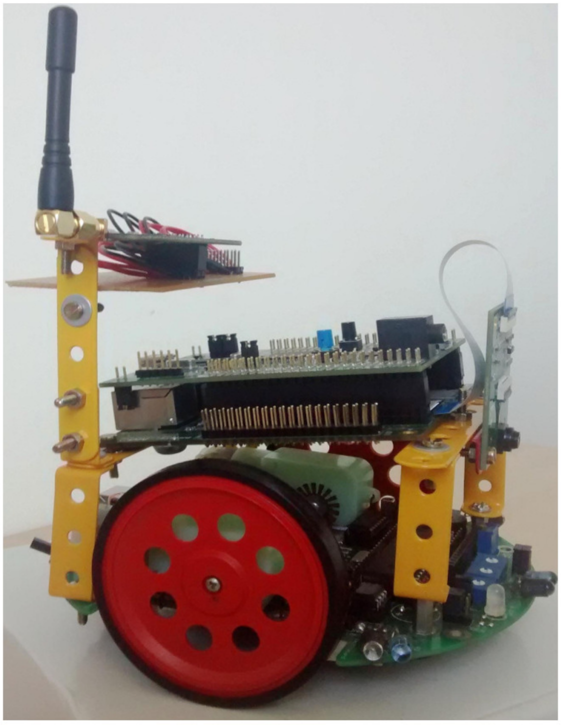


Figure 6.3. B-swarm robot

master-slave configuration. The FlexEye platform acts as a master and the ATMEGA 16A on the mobile platform connected together using UART as shown in figure 6.4. Each microcontroller is assigned with various duties so that they are loosely coupled for rapid execution of tasks. The Atmega 16A platform handles navigation control including path and obstacle detection using IR sensors. The FlexEye module handles communication between robot and provide trajectories to the slave for navigation.

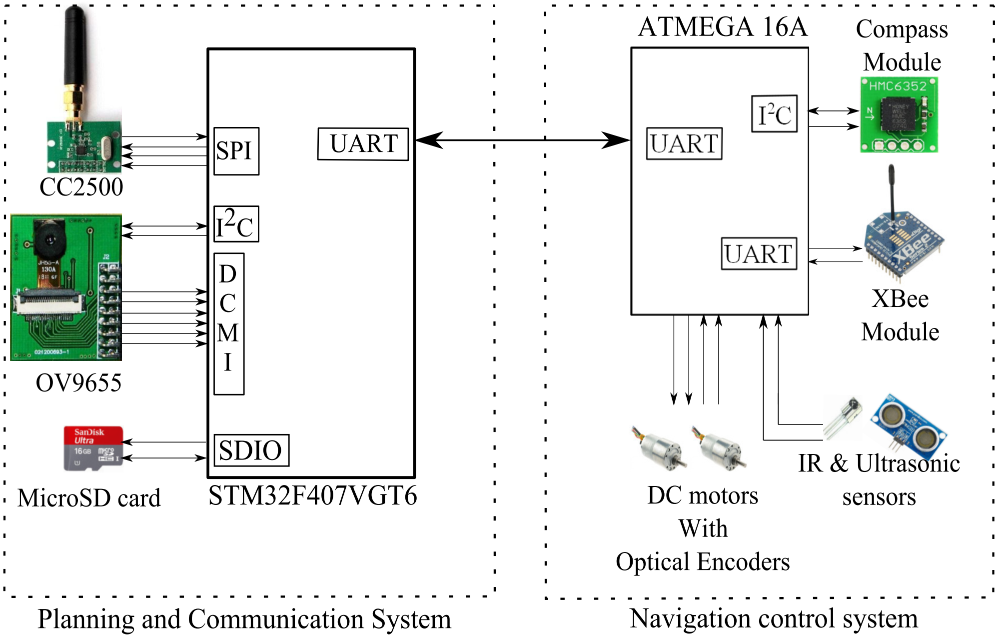


Figure 6.4. Architecture of B-Swarm multi-processor platform

The B-Swarm platform is further integrated with a claw to provide the capabilities of linking with neighboring robots as shown in figure 6.5. The design is prototyped with an intention of forming 2D-structure for the purpose of hole crossing.

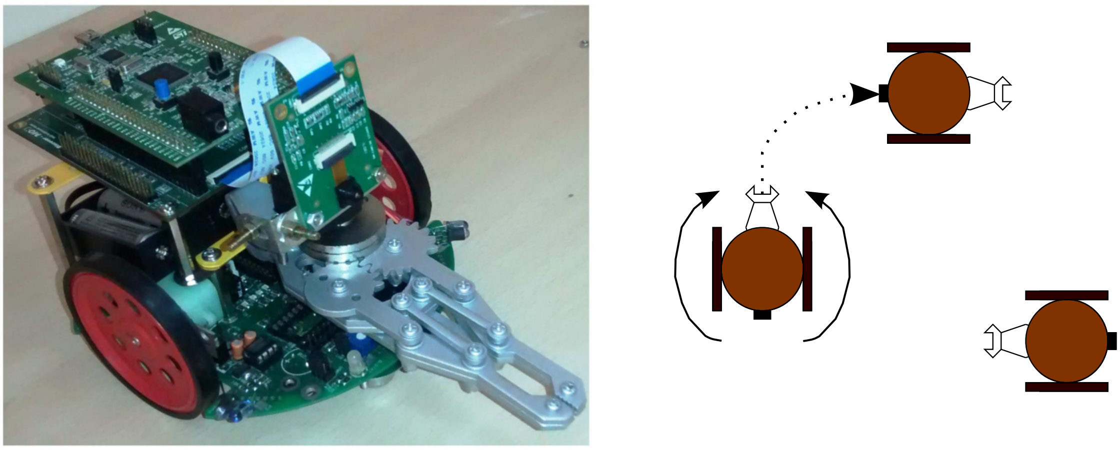


Figure 6.5. Biomimetic prototype - 2D snake

Major drawbacks observed with the biomimetic prototype is it's mass and two DOF from mobility. Though the obstacles in the path like holes can be crossed by chain formation, the purpose of the platform is limited to that single purpose. It has been also identified that master-slave mechanism can be a deterrent to the performance as it induces synchronization issues. The design of custom-robot is started to reduce the cost, form-factor, mass and improve the performance. The SQ-Bot architecture was tried out by adding an extra DOF to the biomimetic prototype shown in figure 6.5. The chain structure possible with the SQ-Bot architecture is shown in figure 6.6

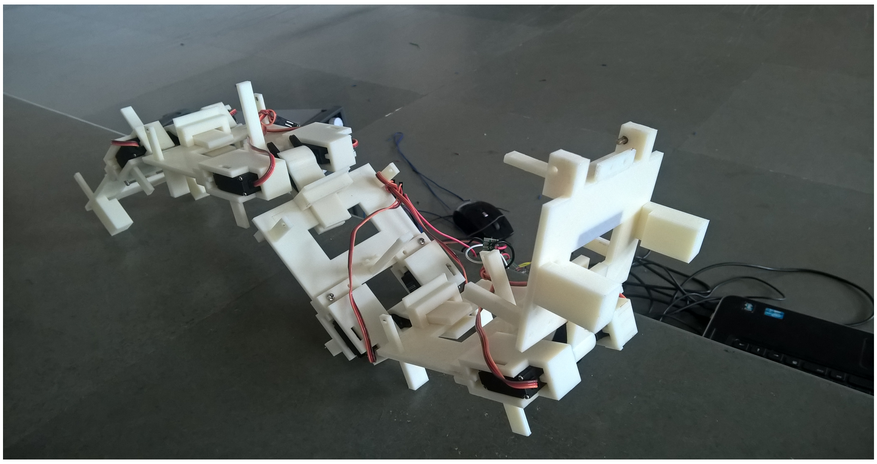


Figure 6.6. SQ-Bot - Biomimetic Chain structure

Though SQ-Bot can biomimic few more structures through an additional DOF, the drawback outweight the benefits. The actuators and claw mechanism embedded are designed to provide double torques and they consume large amount of space leading to nullification of torque improvements. The HexaMob robotic modular design shown in figure 6.7 is proposed after addition of an extra DOF to SQ-Bot prototype so that mimicking is still effective while improving torques in alternate means.

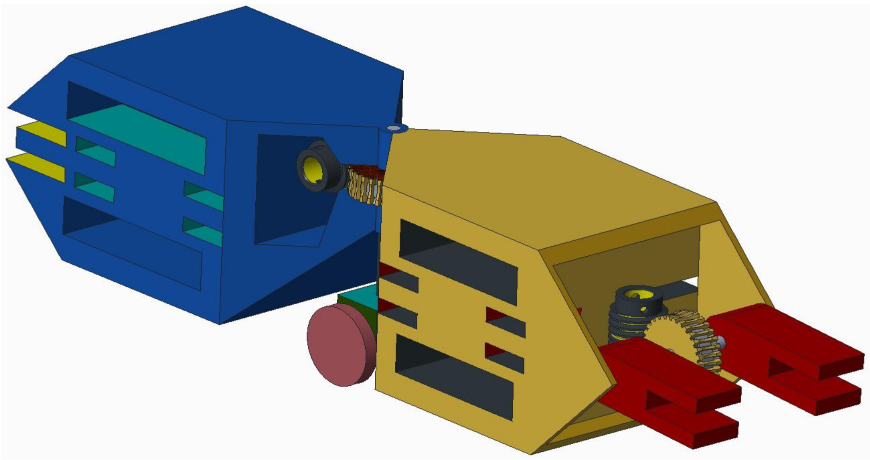


Figure 6.7. SQ-Bot - Biomimetic Chain structure

6.3 HexaMob - Hardware specifications

The total length of the HexaMob is desinged to be 16cm - 7cm for the back chassis, 7 cm for the front chassis and 2 cm for the claw. The specifications of the bevel and worm gears described in chapter 3 that are used for prototyping the HexaMob are provided in tables 6.1 and 6.2 respectively.

Table 6.1. Specifications of Bevel gears

Parameter	Measurement
Diametral Pitch	65
Circular Pitch	1.228 mm
Pressure angle	20^o
Gear ratio	1:1
No. of teeth	Gear - 25 , Pinion - 25
Pitch diameter	Gear - 9.769 mm, Pinion - 9.769 mm
Tooth thickness	Gear - 0.614 mm, Pinion - 0.614 mm

Table 6.2. Specifications of Worm gears

Parameter	Measurement
Diametral Pitch	79
Lead wheel	1.010 mm
Lead worm	1.010 mm
Lead angle	3.679^o
Worm Threads	1
No. of Teeth	60
Worm outside diameter	5.642 mm
Wheel outside diameter	20.575 mm
Pressure angle	20^o

All the gears in the HexaMob are designed to be sourced by DC Micro metal gearmotors from Pololu[188]. The micro metal gear motors from Polulu are available in eleven different torques starting from 0.144 Kg-cm to 9 Kg-cm, angular speeds from 32 rpm to 6000 rpm and three power ratings in the approximately same form-factor of 4 cm x 1.2cm as shown in figure 6.8. Rotary encoders with precision of 0.6 deg are available in small form factor for tracking motor movement in real-time. The major advantage in inculcating polulu motors into the system is that the locomotion capabilities of the system can be enhanced by replacing the motors of advanced configuration without redesigning gears or chassis.

The FlexEye VSM developed to be at the heart of control system is a prototype tested for it's image acquisition, processing and power conservation capabilities. Though the board is large in form-factor due to it's prototype nature, the compact hardware of the same hardware is available in a small form-factor (3.3 cm x 2.54 cm) as a open-source project as shown in figure 6.9, proving that possibility of miniaturizing the platform for modular robotics. It is a well established platform used for recognizing multi-object, multi-color blob tracking using which it is possible to recognize faces and objects. The 3D-printed chassis of HexaMob according to the specifications mentioned

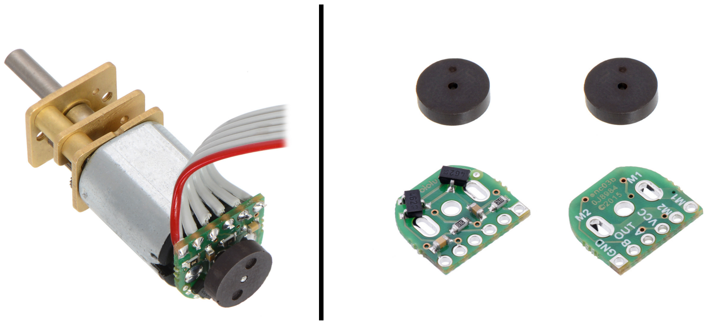


Figure 6.8. Micro metal gearmotors and magnetic encoders

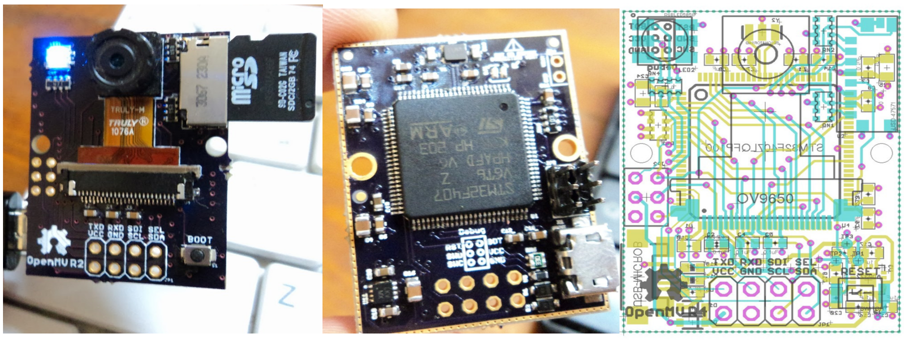


Figure 6.9. OpenMV CAM prototype and Schematic

above along with enough space for accommodating motors, electronics boards is shown in the figure 6.10

Though the electronic platform prototyped for modular robot is advanced in every specification in relation to the current modular robotic designs developed so far, the system needs few more sensors like inertial measurement units (**IMU**) to provide time to time information regarding each robot's orientation in 3D space for stable locomotion. Along with the sensor interfacing, voltage regulation and motor driver circuitry are to be incorporated for completion of embedded system.

The block diagram integrating the HexaMob, FlexEye and Quanta platforms described in chapter 3, chapter 4 and chapter 5 respectively is shown in figure 6.11. The STM32F4 discovery board is

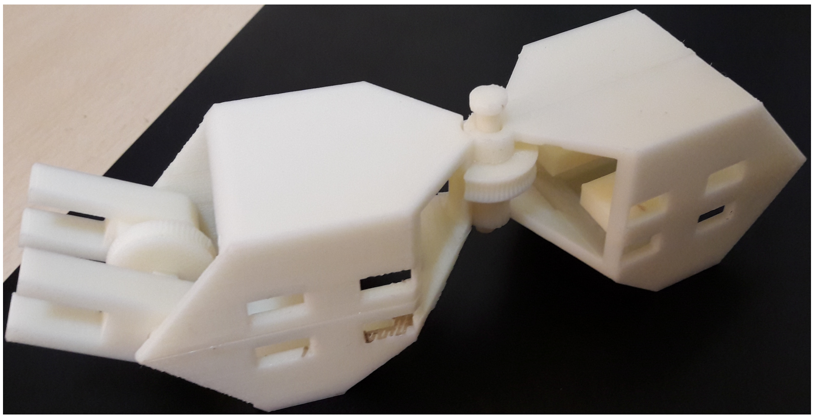


Figure 6.10. HexaMob - Final prototype

the platform used for all the operations and is interfaced to numerous sensors, actuators and drivers for realizing an application using modular robot. Though the operations are independently verified in different prototype and simulations, it is possible to realize a integrated prototype achieving the biomimetic functionalities using HexaMob, FlexEye and Quanta as their capabilities far exceed the conventional modular robotic designs.

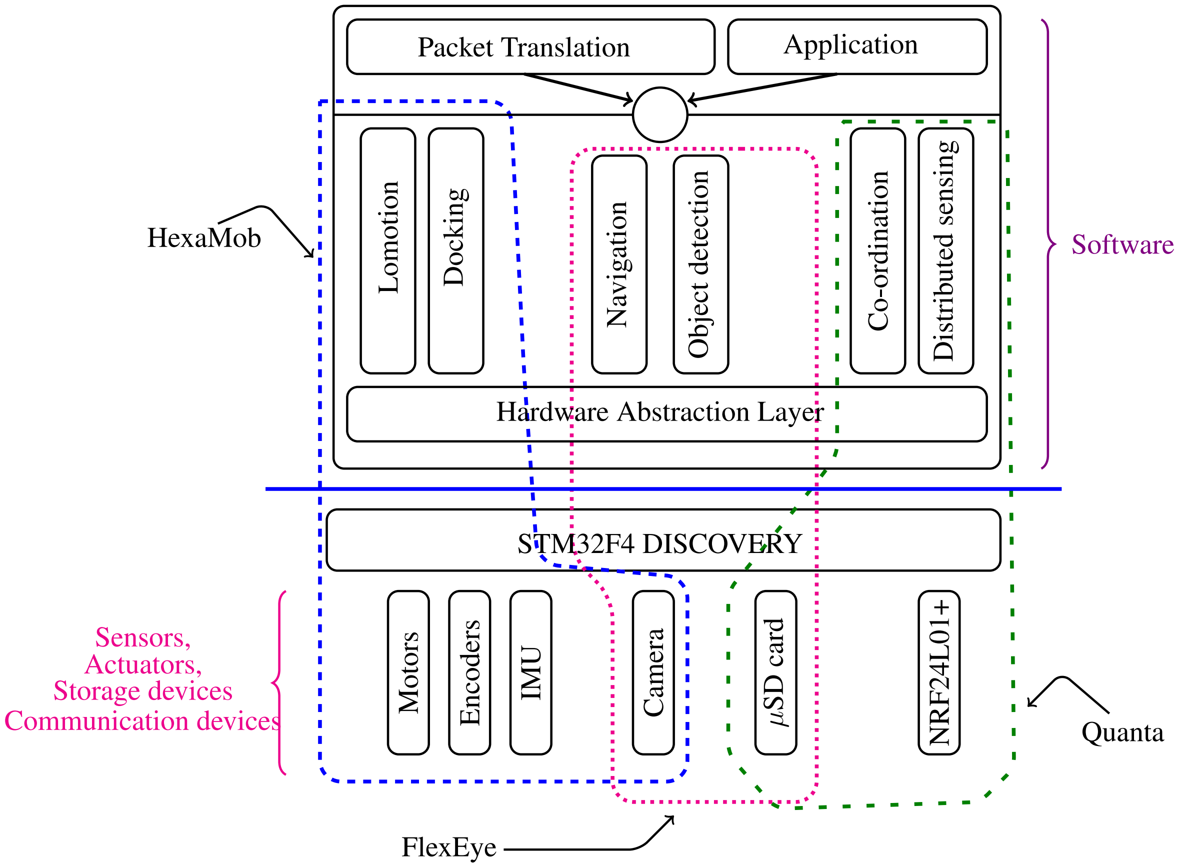


Figure 6.11. Block diagram of modular robot

6.4 Conclusions

The mimicking of biological organisms is tested out from B-Swarm bot with limited features like chain formation in 2D which were further extended to SQ-bot to chain formations in 3D. The same principle of docking and locomotion is extended to HexaMob with one more DOF and more docking interfaces and hence improving the features of HexaMob in mimicking the biological organisms.

The reconfiguration capabilities of modular robotics is attributed to number of docking interfaces to which robots can attach or detach as per the application requirements. Self-reconfiguration is an extended feature of reconfiguration in which a robot can identify another robot in the neighborhood. HexaMob platform features self-reconfiguration capabilities because of its established embedded platform capable of object recognition and numerous docking interfaces.

Chapter 7

Conclusions

In this chapter, the conclusions drawn from research work explained in this thesis are summarized and future scope of the research is presented.

7.1 Conclusions

Conventional robotics is facing setback due to the inflexible nature of the robots for wide range of applications and increasing cost(time and money). Modular robotics can provide suitable solutions to numerous applications due to their flexible nature in assembly, reconfiguration and structures. The manufacturing cost of a single modular robot is significantly less in relation to a conventional robotic design and a user can reconfigure the structures formed by such group of modular robots as per the application requirements.

The outcome of the research presented in this thesis is a design of a modular robotic system capable of facilitating sophisticated features such as distributed sensing and remote access to in-accessible terrains, while facilitating the real-time demands such as autonomous navigation and reconfiguration. The gaps in research observed in the research of modular robotics are

- Majority of the robots developed in modular robotics research are sophisticated in terms of design and they exhibited remarkable reconfiguration capabilities. But the major setback for modular robots is observed due to the inverse relationship between form-factor and capabilities. The decrease in form-factor leads to fewer number of actuators and sensors along with a reduction in torque capabilities.

- Energy conservation is another major concern in modular robots. The operational longevity of the modular robots is limited due to the employment of smaller capacity batteries. The lack of optimizations at various levels of hardware and software in modular robots is crippling the performance of the modular robots in real-time applications.

The HexaMob robotic module is designed to address major concerns in the domain of modular robotics. It is envisaged to be an integration of optimized hardware and software modules for facilitating research in the field of modular robotics. Optimizations are performed at various levels of the design hierarchy of the HexaMob for increasing its suitability in real-world applications. The utilization of generic and standardized components in electronics, as well as mechanical design, aids easy macro-sizing or down-sizing the design as per the requirement of the application.

- HexaMob - Mechanical design and optimizations
  - Docking mechanisms play critical role in deciding the form factor of the modular robots. A docking mechanism that need energy for maintaining the lock or latch generally employ additional actuators in the design explicitly for docking. The energy-less docking mechanisms using magnets also require additional actuators for undocking. A energy-less docking mechanism is proposed in HexaMob robotic module without any additional actuators. The teeth-locking mechanism controlled by precise movement of a worm gear can prove to be successful considering the lighter loads used in modular robots.
  - Majority of energy consumption of modular robotic systems is expended in locomotion. The actuators employed for providing locomotion or DOF are DC motors or servos and these actuators consume energy continuously for their operations. The accuracy in control provided by these actuators is also not satisfactory if they are employed directly for locomotion. A worm gear based mechanical design can provide implicit back-driving prevention and also significant torque improvement in a small

form-factor. The HexaMob employs two worm gear mechanisms at two DOF for improving torques during locomotion.

- The conventional IR based and magnet based alignment for docking is also avoided in HexaMob design due to their limitations. A strategy using vision sensor that is capable of recognizing errors in alignment during structural formation and reconfiguration in 2D as well as 3D scenarios is employed for facilitating alignment in docking.
- The navigational capabilities are seldom considered in modular robotics for providing the autonomous nature to individual robots. The vision sensor employed in HexaMob when coupled with odometry is capable of providing better navigation capabilities to the robot in real-world applications. The vision based algorithms can also aid in navigation of biomimetic structures when sensor fusion is employed with accelerometers, inertial measurement units, etc.
- The symmetric design of the HexaMob robotic module provides flexibility and stability during in implementing locomotions for biomimetic structures. A power and communication sharing mechanism is proposed for increasing the longevity of each robot and coordinated robotic system.

• HexaMob - Electronic design and optimizations

- Multi-processor architectures and numerous on-board sensors can lead to complexities in control and implementation. A single capable platform such as FlexEye prototyped for a robot can suffice for control and communication tasks required for modular robots as they are expected to be lightly-capable. The FlexEye embedded platform developed for HexaMob is an optimal design in terms of hardware and power consumption. The capabilities of FlexEye platform are compared with visual sensor motes which are optimized in energy consumption for research in domains like wireless sensor networks.
- FlexEye platform can acquire and process the images of VGA resolution without any need of external RAM. It also provides non-volatile memory interface for storing data in large-scale distributed applications. In spite of being a single processor platform, the features of FlexEye such as scheduling the activities like acquisition of high resolution

of color images, wireless communication and other necessary actuation features are successfully tested.

  - Energy conservation in FlexEye platform can be performed at chip level by frequency scaling and at system level by disabling peripherals and sensors. The power consumption of FlexEye at system level can be reduced to few micro-amperes when no processing is necessary by enabling sleeping modes of various devices such as camera, transceiver and sensors. The results on power consumption and other available features provided in relation to various camera platforms confirms the capabilities of the FlexEye platform.

- HexaMob - Communication strategies and optimizations

  - A novel platform referred as Quanta was developed to control and monitor the modular robots using wireless transceivers. The wireless transceivers coupled with individual capabilities of robots in navigation and mobility can provide complete autonomous nature to the modular robots.
  - Communication plays a vital role in locomotion of modular robots as well as energy conservation in robotic system as a whole. The errors in control during locomotion because of latencies in communication can lead to an unstable coordinated robotic system. The Quanta platform employs fast and efficient transceiver for the reduction of latencies. The NRF24L01+ commercial off-the-shelf radio modules with optimal energy consumption and fast data rate capabilities is chosen for the Quanta platform after simulation. The real-time parameters involved in communications such as switching times of radios, data upload, interrupt handling etc. were accounted in simulation for identification of efficient radio.
  - Quanta platform is capable of communicating to three separate networks (separate due to utilization of different frequencies for communication) and can address more than 300 nodes in each network. These features facilitate the control of external events in the environment as well as internal events in the robots for preparing, executing and resetting an experiment scenario for research.
  - A strategic packet structure employing opcodes and operands incorporated in quanta for facilitating peer to peer communication as well centralized control from base station

can be an advantage while conducting research on various communications strategies used in modular robots. Such flexibility can enable the centralized control from a modular robot participating in the coordinated robotic system instead of the base station for generating locomotions and reconfigurations.

– The organization of various events into pre-concurrent, concurrent and post- concurrent activities in the protocol for communication between client and server can be visualized as an advantage in Quanta using which a master can control a single modular robot for locomotion or many robots together in a biomimetic structure for concurrent motion.

7.2 Future scope of the work

The research work presented in this thesis proposes a novel design of a modular robot and prototyped electronic control and communication platforms. The scope of research can be further expanded by production of numerous prototypes for practical scenarios such as warehouse management where they can reconfigure as per requirements and perform tasks such as carrying the goods and sorting them in racks.

An analysis on multi-body dynamics using HexaMob robotic module for possible robotic structures can be performed before prototyping and deployment in industry grade applications. Such an approach was rarely considered due to limitations of the modular robotic designs developed so far.

The energy consumption of a individual robotic module during locomotion and the robotic modules participating in the coordinated motion can be logged and analyzed for better estimation of lifetime of the system. Such analysis can provide better adaptable strategies in dynamically changing environments.

Integration of real-time parameters involved in communication and latencies in docking into various CAD simulation tools is necessary for rapid testing and prototyping the structures. The strategies regarding centralized and decentralized decision making mechanisms can further enhance the autonomous nature of HexaMob modular robots.

Chapter 8

Summary

The current chapter of the thesis summarizes the motivation of research and relates them to the outcomes for better utilization of the work in real-world applications.

Modular robots are considered to be future robotic technologies capable of replacing the conventional robotic designs and models with homogeneous miniaturized robots that can be utilized in numerous applications. Such designs can be used from sophisticated and complex tasks such as space exploration, underwater navigation to prosthetics and adaptable domestic furniture. Energy conservation, lack of features for forming numerous structures and minimal autonomous nature are the major challenges faced in the domain of modular robotics. Numerous optimizations are necessary at the levels of design, movement, embedded system and communication for efficient operation of modular robots in real-world applications. Embedded platforms being the only intelligent part of the robotic system are expected to be capable in terms of processing for performing various sensing, processing, actuation and communication operations. It is also expected that the improvement in performance in embedded platforms cannot be achieved at a trade-off with energy conservation. Communication mechanisms and techniques constitute another vital plane of the entire software stack and hardware operations which plays crucial role in providing rapid communication along with facilitation of different applications to utilize them concurrently.

The HexaMob robotic design proposed in the thesis is modeled for energy conservation in movement while facilitating formation of numerous structures. The design is capable of mimicking gaits of different biological organisms so that the design can be used in different real-world applications with minimal modifications. The utilization of such homogeneous modules for different tasks reduce the time spent by researchers in development of unique application specific robotic designs. Back-drive restriction mechanism proposed is successfully tested and hence can be incorporated in future designs to prevent high power consumption due to stall torques and also for maintaining a fail-state structures in case of power depletion or component failures. Inculcation of vision based docking mechanisms and navigations in HexaMob robot can be an visualized as an improved feature which further enhances the autonomous nature of robots. Adaptation rate and utilization factor of designs like HexaMob are vastly great in relation to the application specific conventional robotic designs.

Embedded platform plays vital role in regular operation of robots and conventional robotic designs utilize highly capable processor platforms to multi-core platforms for implementation of the control algorithms of the robots. Incorporation of research in distributed processing along with moderately capable embedded platforms in each robot can aid in reduction in failure of complete system due to single point failures and also aids in easy replacement of the damaged components as part of immediate corrective action. FlexEye visual mote prototyped for its low power consumption feature along with improved processing capabilities can be visualized as a suitable embedded platform supporting image processing capabilities that can support distributed processing with communication. The tested features of the FlexEye platform aid in autonomous docking and navigation along with communication and control requirements of the HexaMob robot.

Communication aids in rapid execution of motion and navigation tasks in individual in both distributed or hierarchical control scenarios. Quanta platform developed for such scenarios in capable of operating on 8-bit embedded platforms with minimal load. The packet based control feature makes quanta an OS independent platform and hence it is a suitable for distributed sensing as well as for remote control in real-world applications in different robots. The platform independent communication mechanisms like Quanta, a capable embedded platform with image processing capabilities for distributed sensing and control like FlexEye and a modular robotic design like HexaMob can be integrated due to their modular and independent nature to become a complete integrated robotic platform for real-world applications.

Bibliography

[1] R. S. Lab. Anymal-quadrupedal robot. Institute of Robotics and Intelligent Systems, Zurich,. [Online]. Available: https://sirop.org/app/2e1a675c-c98e-44bd-af98-3b401a2eeee6

[2] R. Laboratory. transformable ruins search and rescue robot. Shenyang Institute of Automation. [Online]. Available: http://english.sia.cas.cn/ns/pn/201311/t20131122_113118.html

[3] WOMBAT. Kawo robot – for removing defects in pipelines. WOMBAT. [Online]. Available: http://www.wombat.cz/kawo-robot-for-removing-defects-in-pipelines/

[4] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, "Swarm-Bot: A New Distributed Robotic Concept," *Autonomous Robots*, vol. 17, no. 2/3, pp. 193–221, sep 2004. [Online]. Available: http://link.springer.com/10.1023/B:AURO.0000033972.50769.1c

[5] J. Baca, P. Pagala, C. Rossi, and M. Ferre, "Modular robot systems towards the execution of cooperative tasks in large facilities," *Robotics and Autonomous Systems*, vol. 66, pp. 159–174, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2014.10.008

[6] M. Yim, W.-M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, and E. Klavins, "Modular Self-reconfigurable Robot Systems: Challenges and Opportunities for the Future," *IEEE Robotics & Automation Magazine*, vol. 14, no. 1, pp. 43–52, 2007.

[7] M. Yim, P. White, M. Park, and J. Sastra, "Modular self-reconfigurable robots," *Encyclopedia of complexity and systems*, pp. 19–33, 2009. [Online]. Available: http://link.springer.com/10.1007/978-0-387-30440-3$_$334

[8] K. Gilpin and D. Rus, "Modular Robot Systems," *IEEE Robotics & Automation Magazine*, vol. 17, no. 3, pp. 38–55, sep 2010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5569030

[9] P. Moubarak and P. BenTzvi, "Modular and reconfigurable mobile robotics," *Robotics and Autonomous Systems*, vol. 60, p. 16481663, 2012.

[10] G. Chirikjian, "Kinematics of a metamorphic robotic system," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1994, pp. 449–455.

[11] A. P. Chirikjian, D. Stein, C.-J. Chiang, and Grigory, "Design and Implementation of metamorphic robots," in *Proceedings of The ASME Design Engineering Technical Conference and Computers in Engineering Conference*. ASME, 1996, pp. 1–10.

[12] A. Pamecha, I. Ebert-Uphoff, and G. Chirikjian, "Useful metrics for modular robot motion planning," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 4, pp. 531–545, 1997. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=611311

[13] S. Murata, H. Kurokawa, and S. Kokaji, "Self-assembling machine," in *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE Comput. Soc. Press, 1994, pp. 441–448. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=351257

[14] D. Rus, "Self-reconfiguring robots," *IEEE Intelligent Systems*, vol. 13, no. 4, pp. 2–4, jul 1998. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=708423

[15] H. Kurokawa, S. Murata, E. Yoshida, K. Tomita, and S. Kokaji, "A 3-D self-reconfigurable structure and experiments," in *Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications (Cat. No.98CH36190)*, vol. 2. IEEE, 1998, pp. 860–865. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=727308

[16] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji, "A 3-D self-reconfigurable structure," in *Proceedings. IEEE International Conference on Robotics and Automation*

(Cat. No.98CH36146), vol. 1. IEEE, 1998, pp. 432–439. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=677012

[17] C. Ünsal and P. Khosla, "Mechatronic design of a modular self-reconfiguring robotic system," in *Proceedings ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 2. IEEE, 2000, pp. 1742–1747. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=844847

[18] C. Ünsal, H. Kiliççöte, and P. K. Khosla, "Modular self-reconfigurable bipartite robotic system: Implementation and motion planning," *Autonomous Robots*, vol. 10, no. 1, pp. 23–40, 2001.

[19] E. Yoshida, S. Murata, S. Kokaji, K. Tomita, and H. Kurokawa, "Micro Self-reconfigurable Robotic System using Shape Memory Alloy," in *Distributed Autonomous Robotic Systems 4*. Tokyo: Springer Japan, 2000, pp. 145–154. [Online]. Available: http://www.springerlink.com/index/10.1007/978-4-431-67919-6$_$14

[20] E. Yoshida, S. Murata, S. Kokaji, A. Kamimura, K. Tomita, and H. Kurokawa, "Get back in shape! [SMA self-reconfigurable microrobots]," *IEEE Robotics & Automation Magazine*, vol. 9, no. 4, pp. 54–60, dec 2002. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1160072

[21] K. Hosokawa, T. Tsujimori, T. Fujii, H. Kaetsu, H. Asama, Y. Kuroda, and I. Endo, "Self-organizing collective robots with morphogenesis in a vertical plane," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 4, no. May. IEEE, 1998, pp. 2858–2863. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=680616

[22] D. Rus and M. Vona, "A physical implementation of the self-reconfiguring crystalline robot," in *Proceedings ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 2, no. April, 2000, pp. 1726–1733.

[23] J. Suh, S. Homans, and M. Yim, "Telecubes: mechanical design of a module for self-reconfigurable robotics," in *Proceedings IEEE International Conference on Robotics*

and Automation (Cat. No.02CH37292), vol. 4. IEEE, 2002, pp. 4095–4101. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1014385

[24] B. K. An, "EM-Cube: Cube-shaped, self-reconfigurable robots sliding on structure surfaces," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2008, pp. 3149–3155.

[25] J. W. Romanishin, K. Gilpin, and D. Rus, "M-blocks: Momentum-driven, magnetic modular robots," in *IEEE International Conference on Intelligent Robots and Systems*, 2013, pp. 4288–4295.

[26] J. W. Romanishin, K. Gilpin, S. Claici, and D. Rus, "3D M-Blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1925–1932. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7139450

[27] K. Gilpin, K. Kotay, and D. Rus, "Miche: Modular shape formation by self-dissasembly," in *Proceedings - IEEE International Conference on Robotics and Automation*, no. April, 2007, pp. 2241–2247.

[28] K. Gilpin, A. Knaian, and D. Rus, "Robot pebbles: One centimeter modules for programmable matter through self-disassembly," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2010, pp. 2485–2492.

[29] P. J. White, K. Kopanski, and H. Lipson, "Stochastic self-reconfigurable cellular robotics," in *Proceedings IEEE International Conference on Robotics and Automation,*, vol. 3. IEEE, 2004, pp. 2888–2893 Vol.3. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1307499

[30] P. White, V. Zykov, J. Bongard, and H. Lipson, "Three dimensional stochastic reconfiguration of modular robots," in *Proceedings of robotics science and systems*, 2005, pp. 161–168.

[31] J. Bishop, S. Burden, E. Klavins, R. Kreisberg, W. Malone, N. Napp, and T. Nguyen, "Programmable parts: a demonstration of the grammatical approach to self-organization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005,

pp. 3684–3691. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1545375

[32] P. J. White and M. Yim, "Scalable modular self-reconfigurable robots using external actuation," in *IEEE International Conference on Intelligent Robots and Systems*, 2007, pp. 2773–2778.

[33] M. White, Paul J and Yim, "Reliable External Actuation for Extending Reachable Robotic Modular Self-Reconfiguration," in *Proc. of Intl. Symp. on Experimental Robotics*, Athens, Greece, 2009, pp. 13–23. [Online]. Available: http://modlab.seas.upenn.edu/publications/white$_$iser2008.pdf

[34] M. Jorgensen, E. Ostergaard, and H. Lund, "Modular ATRON: modules for a self-reconfigurable robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 2, 2004, pp. 2068–2073.

[35] U. P. Schultz, M. Bordignon, and K. Stoy, "Robust and reversible self-reconfiguration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, 2009, pp. 5287–5294.

[36] B. Salem, "PetRo: Development of a modular pet robot," in *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, aug 2014, pp. 483–488. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6926299

[37] K. Kotay, D. Rus, M. Vona, and C. McGray, "The self-reconfiguring robotic molecule," in *IEEE International Conference on Robotics and Automation*, vol. 1, no. May, 1998, pp. 424–431.

[38] T. Fukuda and S. Nakagawa, "Method of autonomous approach, docking and detaching between cells for dynamically reconfigurable robotic system CEBOT." *JSME international journal. Ser. 3, Vibration, control engineering, engineering for industry*, vol. 33, no. 2, pp. 263–268, 1990. [Online]. Available: http://ci.nii.ac.jp/naid/130003964367/en/

[39] T. Fukuda and S. Nakagawa, "Dynamically reconfigurable robotic system," in *Proceedings.*

IEEE International Conference on Robotics and Automation. IEEE Comput. Soc. Press, 1988, pp. 1581–1586. [Online]. Available: http://ieeexplore.ieee.org/document/12291/

[40] G. Endo, K. Togawa, and S. Hirose, "Study on self-contained and terrain adaptive active cord mechanism," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No.99CH36289)*, vol. 3, no. 1, 1999, pp. 1399–1405.

[41] K. Togawa, M. Mori, and S. Hirose, "Study on three-dimensional active cord mechanism: development of ACM-R2," in *Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000) (Cat. No.00CH37113)*, vol. 3, 2000, pp. 2242–2247.

[42] M. Mori and S. Hirose, "Development of active cord mechanism ACM-R3 with agile 3D mobility," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, 2001, pp. 1552–1557.

[43] H. B. Brown, J. M. Vande Weghe, C. A. Bererton, and P. K. Khosla, "Millibot trains for enhanced mobility," *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 4, pp. 452–461, 2002.

[44] Jinguo Liu, Shugen Ma, Zhenli Lu, Yuechao Wang, Bin Li, and Jing Wang, "Design and experiment of a novel link-type shape shifting modular robot series," in *IEEE International Conference on Robotics and Biomimetics - ROBIO*. IEEE, 2005, pp. 318–323. [Online]. Available: http://ieeexplore.ieee.org/document/1708645/

[45] B. Li, S. Ma, J. Liu, M. Wang, T. Liu, and Y. Wang, "AMOEBA-I: A Shape-Shifting Modular Robot for Urban Search and Rescue," *Advanced Robotics*, vol. 23, no. 9, pp. 1057–1083, 2009. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1163/156855309X452485

[46] Dazhai Li, Hualei Fu, and Wei Wang, "Ultrasonic based autonomous docking on plane for mobile robot," in *IEEE International Conference on Automation and Logistics*. IEEE, sep 2008, pp. 1396–1401. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4636372

[47] W. Wang, W. Yu, and H. Zhang, "JL-2: A mobile multi-robot system with docking and manipulating capabilities," *International Journal of Advanced Robotic Systems*, vol. 7, no. 1, pp. 9–18, 2010.

[48] A. Lyder, R. Franco, M. Garcia, and K. Stoy, "Genderless Connection Mechanism for Modular Robots Introducing Torque Transmission Between Modules," in *ICRA 2010 Workshop on Modular Robots: The State of the Art*, 2010, pp. 77–82.

[49] M. Yim, "New locomotion gaits," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE Comput. Soc. Press, 1994, pp. 2508–2514. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=351134

[50] A. Castano, R. Chokkalingam, and P. Will, "Autonomous and Self-SuÆcient CONRO Modules for Recon gurable Robots," in *Distributed Autonomous Robotic Systems 4*. Tokyo: Springer Japan, 2000, ch. 5, pp. 155–164. [Online]. Available: file://soton.ac.uk/ude/PersonalFiles/Users/ns4g12/mydocuments/AssemblyProject/paper/Hardware/conro-modules.pdf

[51] W.-M. Shen, Y. Lu, and P. Will, "Hormone-based control for self-reconfigurable robots," in *Proceedings of the fourth international conference on Autonomous agents - AGENTS '00*, 2000, pp. 1–8. [Online]. Available: http://portal.acm.org/citation.cfm?doid=336595.336602

[52] W. M. Shen, B. Salemi, and P. Will, "Hormone-inspired adaptive communication and distributed control for CONRO self-reconfigurable robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 700–712, 2002.

[53] M. Rubenstein, K. Payne, P. Will, and W.-M. S. W.-M. Shen, "Docking among independent and autonomous CONRO self-reconfigurable robots," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 3, 2004, pp. 2877–2882.

[54] P. Dasgupta, J. Baca, S. Hossain, A. Dutta, and C. A. Nelson, "Mechanical design and computational aspects for locomotion and reconfiguration of the ModRED modular robot," in *Proceedings of the international conference on Autonomous agents and multi-agent systems*, 2013, pp. 1359–1360. [Online]. Available: http://dl.acm.org/citation.cfm?id=2485224

[55] M. Yim, D. Duff, and K. Roufas, “PolyBot: a modular reconfigurable robot,” in *IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1. IEEE, 2000, pp. 514–520. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=844106

[56] M. Yim, Y. Zhang, K. Roufas, D. Duff, and C. Eldershaw, “Connecting and disconnecting for chain self-reconfiguration with polybot,” *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 4, pp. 442–451, 2002.

[57] G. Qiao, G. Song, J. Zhang, H. Sun, W. Wang, and A. Song, “Design of transmote: A modular self-reconfigurable robot with versatile transformation capabilities,” in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, dec 2012, pp. 1331–1336. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6491153

[58] Y. Li, H. Zhang, and S. Chen, “A four-legged robot based on GZ-I modules,” in *IEEE International Conference on Robotics and Biomimetics, ROBIO 2008*, 2008, pp. 921–926.

[59] R. Moeckel, C. Jaquier, K. Drapel, E. Dittrich, A. Upegui, and A. Jan Ijspeert, “Exploring adaptive locomotion with YaMoR, a novel autonomous modular robot with Bluetooth interface,” *Industrial Robot: An International Journal*, vol. 33, no. 4, pp. 285–290, jul 2006. [Online]. Available: http://www.emeraldinsight.com/doi/abs/10.1108/01439910610667908

[60] A. Kawakami, A. Torii, K. Motomura, and S. Hirose, *SMC Rover: Planetary Rover with transformable wheels*, 2003, vol. 5. [Online]. Available: http://link.springer.com/chapter/10.1007/3-540-36268-1$_$45

[61] M. Delrobaei and K. a. McIsaac, “Design and Steering Control of a Center-Articulated Mobile Robot Module,” *Journal of Robotics*, vol. 2011, pp. 1–14, 2011. [Online]. Available: http://www.hindawi.com/journals/jr/2011/621879/

[62] M. Dorigo, “SWARM-BOT: an experiment in swarm robotics,” in *Proceedings of IEEE Swarm Intelligence Symposium,*. IEEE, 2005, pp. 192–200. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1501622

[63] R. Gro, M. Bonani, F. Mondada, and M. Dorigo, "Autonomous Self-Assembly in Swarm-Bots," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1115–1130, dec 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4020359

[64] R. Gross, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada, "Object transport by modular robots that self-assemble," in *Proceedings of IEEE International Conference on Robotics and Automation,*. IEEE, 2006, pp. 2558–2564. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1642087

[65] V. Trianni, S. Nolfi, and M. Dorigo, "Cooperative hole avoidance in a swarm-bot," *Robotics and Autonomous Systems*, vol. 54, no. 2, pp. 97–103, feb 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889005001478

[66] E. Tuci, R. Gross, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo, "Cooperation through self-assembly in multi-robot systems," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 1, no. 2, pp. 115–150, dec 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1186778.1186779

[67] R. O'Grady, R. Gross, A. L. Christensen, F. Mondada, M. Bonani, and M. Dorigo, "Performance benefits of self-assembly in a swarm-bot," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, oct 2007, pp. 2381–2387. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4399424

[68] G. Fu, A. Menciassi, and P. Dario, "Development of a genderless and fail-safe connection system for autonomous modular robots," in *EEE International Conference on Robotics and Biomimetics*. IEEE, dec 2011, pp. 877–882. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6181398

[69] M. D. M. Kutzer, M. S. Moses, C. Y. Brown, D. H. Scheidt, G. S. Chirikjian, and M. Armand, "Design of a new independently-mobile reconfigurable modular robot," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2010, pp. 2758–2764.

[70] K. C. Wolfe, M. S. Moses, M. D. M. Kutzer, and G. S. Chirikjian, “M 3Express: A low-cost independently-mobile reconfigurable modular robot,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 2704–2710.

[71] G. G. Ryland and H. H. Cheng, “Design of iMobot, an intelligent reconfigurable mobile robot with novel locomotion,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2010, pp. 60–65.

[72] D. Ko and H. H. Cheng, “Programming reconfigurable modular robots,” in *Proceedings of 2012 IEEE/ASME 8th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*. IEEE, jul 2012, pp. 160–165. [Online]. Available: http://ieeexplore.ieee.org/document/6275555/

[73] J. Davey, N. Kwok, and M. Yim, “Emulating self-reconfigurable robots - Design of the SMORES system,” in *IEEE International Conference on Intelligent Robots and Systems*, 2012, pp. 4464–4469.

[74] Y. Zhang, G. Song, S. Liu, G. Qiao, J. Zhang, and H. Sun, “A Modular Self-Reconfigurable Robot with Enhanced Locomotion Performances: Design, Modeling, Simulations, and Experiments,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 81, no. 3-4, pp. 377–393, 2016.

[75] E. Yoshida, S. Murata, A. Kamimura, K. Tomita, H. Kurokawa, and S. Kokaji, “A motion planning method for a self-reconfigurable modular robot,” in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, vol. 1, 2001, pp. 590–597. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=973420

[76] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji, “M-TRAN: self-reconfigurable modular robotic system,” *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 4, pp. 431–441, dec 2002. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1159221

[77] H. Kurokawa, A. Kamimura, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata, “M-TRAN II: Metamorphosis from a four-legged walker to a caterpillar,” in *Proc. of the Int. Conf. on*

Intelligent Robots and Systems (IROS), vol. 3, no. October, 2003, pp. 2454–2459. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1249238

[78] a. Kamimura, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata, "Distributed adaptive locomotion by a modular robotic system, M-TRAN II," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3. IEEE, 2004, pp. 2370–2377. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1389763

[79] H. Kurokawa, K. Tomita, A. Kamimura, S. Kokaji, T. Hasuo, and S. Murata, "Distributed Self-Reconfiguration of M-TRAN III Modular Robotic System," *The International Journal of Robotics Research*, vol. 27, no. 3-4, pp. 373–386, mar 2008. [Online]. Available: http://ijr.sagepub.com/cgi/doi/10.1177/0278364907085560

[80] B. Salemi, M. Moll, and W. M. Shen, "SUPERBOT: A deployable, multi-functional, and modular self-reconfigurable robotic system," in *IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 3636–3641.

[81] M. Krivokon, J. Everist, M. Rubenstein, and J. Venkatesh, "Multimode locomotion via SuperBot robots," in *Proceedings IEEE International Conference on Robotics and Automation.* IEEE, 2006, pp. 2552–2557. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1642086

[82] M. Yim, B. Shirmohammadi, J. Sastra, M. Park, M. Dugan, and C. J. Taylor, "Towards robotic self-reassembly after explosion," in *IEEE International Conference on Intelligent Robots and Systems*, no. October, 2007, pp. 2767–2772.

[83] V. Zykov, E. Mytilinaios, M. Desnoyer, and H. Lipson, "Evolved and designed self-reproducing modular robotics," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 308–319, apr 2007. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4154840

[84] S. T. S. Tang, Y. Z. Y. Zhu, J. Z. J. Zhao, and X. C. X. Cui, "The UBot modules for self-reconfigurable robot," in *ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*, no. July 2015, 2009.

[85] Y. Zhu, J. Zhao, X. Cui, X. Wang, S. Tang, X. Zhang, and J. Yin, "Design and implementation of UBot: A modular Self-Reconfigurable Robot," in *IEEE International Conference on Mechatronics and Automation,*, 2013, pp. 1217–1222.

[86] Y. Zhu, H. Jin, X. Zhang, J. Yin, P. Liu, and J. Zhao, "A Multi-sensory Autonomous Docking Approach for a Self-reconfigurable Robot without Mechanical Guidance," *International Journal of Advanced Robotic Systems*, vol. 11, 2014. [Online]. Available: ⟨GotoISI⟩://WOS:000341645300001

[87] A. Sproewitz, A. Billard, P. Dillenbourg, and A. J. Ijspeert, "Roombots-mechanical design of self-reconfiguring modular robots for adaptive furniture," in *IEEE International Conference on Robotics and Automation*. IEEE, may 2009, pp. 4259–4264. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5152613

[88] A. Sproewitz, P. Laprade, S. Bonardi, M. Mayer, R. Moeckel, P. A. Mudry, and A. J. Ijspeert, "Roombots - Towards decentralized reconfiguration with self-reconfiguring modular robotic metamodules," *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, no. ii, pp. 1126–1132, 2010.

[89] J. Neubert, A. Rost, and H. Lipson, "Self-Soldering Connectors for Modular Robots," *Transactions on Robotics*, no. submitted, pp. 1–14, 2014.

[90] J. Neubert and H. Lipson, "Soldercubes: a self-soldering self-reconfiguring modular robot system," *Autonomous Robots*, vol. 40, no. 1, pp. 139–158, jan 2016. [Online]. Available: http://link.springer.com/10.1007/s10514-015-9441-4

[91] G. Hamlin and A. Sanderson, "Tetrobot: a modular system for hyper-redundant parallel robotics," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 1, 1995.

[92] G. J. Hamlin and A. C. Sanderson, "TETROBOT: A modular approach to parallel robotics," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, 1997.

[93] V. Ramchurn, R. C. Richardson, and P. Nutter, *ORTHO-BOT: A Modular Reconfigurable Space Robot Concept*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 659–666. [Online]. Available: http://dx.doi.org/10.1007/3-540-26415-9$_$79

[94] A. Lyder, R. F. M. Garcia, and K. Stoy, “Mechanical design of Odin, an extendable heterogeneous deformable modular robot,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2008, pp. 883–888.

[95] C. H. Yu, K. Haller, D. Ingber, and R. Nagpal, “Morpho: A self-deformable modular robot inspired by cellular structure,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2008, pp. 3571–3578.

[96] D. Hjelle and H. Lipson, “A robotically reconfigurable truss,” in *2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*, 2009, pp. 73–78.

[97] J. R. Amend and H. Lipson, “Shape-Shifting Materials for Programmable Structures,” in *International Conference on Ubiquitous Computing: Workshop on Architectural Robotics*, 2009.

[98] K. C. Galloway, R. Jois, and M. Yim, “Factory floor: A robotically reconfigurable construction platform,” in *2010 IEEE International Conference on Robotics and Automation*. IEEE, may 2010, pp. 2467–2472. [Online]. Available: http://ieeexplore.ieee.org/document/5509878/

[99] H. Tokashiki, H. Amagai, S. Endo, K. Yamada, and J. Kelly, “Development of a transformable mobile robot composed of homogeneous gear-type units,” in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 2, no. October, 2003, pp. 1602–1607.

[100] S. Goldstein and T. Mowry, “Claytronics: An instance of programmable matter,” in *Wild and Crazy Ideas Session of ASPLOS, Boston, MA, October*, vol. 17, no. April 2000, 2004, pp. 2003–2003.

[101] S. C. Goldstein, S. C. Goldstein, J. D. Campbell, J. D. Campbell, T. C. Mowry, and T. C. Mowry, “Programmable Matter,” in *Computer*, vol. 38, no. 6, 2005, pp. 99–101.

[102] J. Campbell, P. Pillai, and S. C. Goldstein, “The robot is the tether: Active, adaptive power routing for modular robots with unary inter-robot connectors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2005, pp. 2960–2967.

[103] M. Shimizu, A. Ishiguro, and T. Kawakatsu, "A modular robot that exploits a spontaneous connectivity control mechanism," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, no. 1, 2005, pp. 2658–2663.

[104] M. Shimizu, T. Mori, and A. Ishiguro, "A development of a modular robot that enables adaptive reconfiguration," *IEEE International Conference on Intelligent Robots and Systems*, pp. 174–179, 2006.

[105] M. E. Karagozler, S. C. Goldstein, and J. R. Reid, "Stress-driven MEMS assembly + electrostatic forces = 1mm diameter robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems,*, 2009, pp. 2763–2769.

[106] B. R. Donald, C. G. Levey, C. G. McGray, I. Paprotny, and D. Rus, "An untethered, electrostatic, globally controllable MEMS micro-robot," *Journal of Microelectromechanical Systems*, vol. 15, no. 1, pp. 1–15, 2006.

[107] B. R. Donald, C. G. Levey, and I. Paprotny, "Planar microassembly by parallel actuation of MEMS microrobots," *Journal of Microelectromechanical Systems*, vol. 17, no. 4, pp. 789–808, 2008.

[108] A. Kamimura, S. Murata, E. Yoshida, H. Kurokawa, K. Tomita, and S. Kokaji, "Self-reconfigurable modular robot - experiments on reconfiguration and locomotion," in *Proceedings. 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1. IEEE, 2001, pp. 606–612. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=973422

[109] M. Mamei and F. Zambonelli, "Programming Modular Robots with the TOTA Middleware," in *Proceedings of the 4th International Conference on Engineering Self-organising Systems*, ser. ESOA'06. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 99–114. [Online]. Available: http://dl.acm.org/citation.cfm?id=1763581.1763591

[110] W. Hongxing, L. Shiyi, Z. Ying, Y. Liang, and W. Tianmiao, "A Middleware Based Control Architecture for Modular Robot Systems," in *2008 IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications*. IEEE, oct 2008, pp. 327–332. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4735730

[111] H. Wei, Y. Chen, J. Tan, and T. Wang, "Sambot: A self-assembly modular robot system," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 4, pp. 745–757, 2011.

[112] K. Harada, S. Russo, T. Ranzani, A. Menciassi, and P. Dario, "Design of Scout Robot as a robotic module for symbiotic multi-robot organisms," in *2011 Int. Symp. on Micro-NanoMechatronics and Human Science, Symp. on "COE for Education and Research of Micro-Nano Mechatronics", Symposium on "Hyper Bio Assembler for 3D Cellular System Innovation"*, no. August 2017, 2012, pp. 511–513.

[113] S. Russo, K. Harada, T. Ranzani, L. Manfredi, C. Stefanini, A. Menciassi, and P. Dario, "Design of a robotic module for autonomous exploration and multimode locomotion," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1757–1766, 2013.

[114] S. G. M. Hossain, C. A. Nelson, and P. Dasgupta, *Hardware Design and Testing of ModRED: A Modular Self-Reconfigurable Robot System*. London: Springer London, 2012, pp. 515–523. [Online]. Available: https://doi.org/10.1007/978-1-4471-4141-9_46

[115] C. Parrott, T.J.Dodd, and R. Gross, "HyMod: A 3-DOF Hybrid Mobile and Self-Reconfigurable Modular Robot and its Extensions," in *13th International Symposium on Distributed Autonomous Robotic Systems*, 2016.

[116] J. Liedke, R. Matthias, L. Winkler, and H. Worn, "The Collective Self-reconfigurable Modular Organism (CoSMO)," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing, AIM 2013*, 2013, pp. 1–6.

[117] W. M. Shen, R. Kovac, and M. Rubenstein, "Singo: A single-end-operative and genderless connector for self-reconfiguration, self-assembly and self-healing," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2009, pp. 4253–4258.

[118] H. Ahmadzadeh, E. Masehian, and M. Asadpour, "Modular Robotic Systems: Characteristics and Applications," *Journal of Intelligent & Robotic Systems*, vol. 81, no. 3-4, pp. 317–357, mar 2016. [Online]. Available: http://link.springer.com/10.1007/s10846-015-0237-8

[119] I. M.-t. Sfcl, D.-c. Chung, B. H. Yoo, Y.-s. Cho, B.-i. Jung, H.-s. Choi, and T.-h. Sung, "Design and Characterization of the EP-Face Connector," vol. 19, no. 3, 2009, pp. 1831–1834.

[120] C. S. Sankhar, A. Agrawal, and K. R. Anupama, "SQ-BOT - A Modular Robot Prototype for Self- Reconfiguring Structures," in *International Conference on Robotics: Current Trends and Future Challenges (RCTFC)*. IEEE, 2016, pp. 1–6.

[121] K. R. Anupama, M. V. Shenoy, and S. S. Reddy, "Platform for biomimetic swarms," in *Proceedings of the 2015 Conference on Advances In Robotics - AIR '15*. New York, New York, USA: ACM Press, jul 2015, pp. 1–6. [Online]. Available: http://dl.acm.org/citation.cfm?id=2783449.2783513

[122] K. N. Richard Budynas, *Shigley's Mechanical Engineering Design*, 10th ed., ser. McGraw-Hill Series in Mechanical Engineering. McGraw-Hill Science/Engineering/Math, 2014.

[123] T. B. Harold Rothbart, *Mechanical Design Handbook: Measurement, Analysis and Control of Dynamic Systems (Handbooks)*.

[124] C. technology. Mica 2 wireless measurement system. Crossbow technology. [Online]. Available: https://www.eol.ucar.edu/isf/facilities/isa/internal/CrossBow/DataSheets/mica2.pdf

[125] Crossbowtechnology. Mica2dot wireless microsensor mote. Crossbow technology. [Online]. Available: https://www.eol.ucar.edu/isf/facilities/isa/internal/CrossBow/DataSheets/mica2dot.pdf

[126] M. Inc. (2010) Iris professional kit. Memsic Inc. [Online]. Available: http://www.memsic.com/wireless-sensor-networks/WSN-EDU2110CB

[127] MEMSICInc. Telos mote platform. MEMSIC Inc. [Online]. Available: http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb_datasheet.pdf

[128] C. technology. Imote2 high-performance wireless sensor network node. Crossbow Technology. [Online]. Available: http://wsn.cse.wustl.edu/images/e/e3/Imote2_Datasheet.pdf

[129] W. Technologies. (2006) Mcs cricket-kit datasheet. Willow Technologies. [Online]. Available: http://www.willow.co.uk/html/cricket_mote_platform.php

[130] M. Inc. Lotus high-performance wireless sensor network platform. Memsic Inc. [Online]. Available: http://www.memsic.com/userfiles/files/Datasheets/WSN/6020-0705-01_A_LOTUS.pdf

[131] W. Technologies. Stargate: X-scale processor platform. Willow Technologies. [Online]. Available: http://www.willow.co.uk/Stargate_Datasheet.pdf

[132] M. corp. (2006) tmotesky: Ultra low power ieee 802.15.4 compliant wireless sensor module. MoteiV corporation. [Online]. Available: http://www.willow.co.uk/Stargate_Datasheet.pdf

[133] O. labs. Sun spot java development kit. Oracle corporation. [Online]. Available: http://www.sunspotdev.org/products/index.html

[134] P. Mirowski, R. Palaniappan, and T. K. Ho, "Depth camera slam on a low-cost wifi mapping robot," in *2012 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, April 2012, pp. 1–6.

[135] P. De, A. Raniwala, R. Krishnan, K. Tatavarthi, J. Modi, N. A. Syed, S. Sharma, and T.-c. Chiueh, "Mint-m: An autonomous mobile wireless experimentation platform," in *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*, ser. MobiSys '06. New York, NY, USA: ACM, 2006, pp. 124–137. [Online]. Available: http://doi.acm.org/10.1145/1134680.1134694

[136] D. Johnson, T. Stack, R. Fish, D. M. Flickinger, L. Stoller, R. Ricci, and J. Lepreau, "Mobile emulab: A robotic wireless and sensor network testbed," in *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, April 2006, pp. 1–12.

[137] E. Ertin, A. Arora, R. Ramnath, M. Nesterenko, V. Naik, S. Bapat, V. Kulathumani, M. Sridharan, H. Zhang, and H. Cao, "Kansei: a testbed for sensing at scale," in *2006 5th International Conference on Information Processing in Sensor Networks*, April 2006, pp. 399–406.

[138] J. R. Martinez-de Dios, A. Jimenez-Gonzalez, A. de San Bernabe, and A. Ollero, *CONET Integrated Testbed Experiments*. Cham: Springer International Publishing, 2014, pp. 59–73. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-01372-5_5

[139] M. Barbosa, A. Bernardino, D. Figueira, J. Gaspar, N. Gonçalves, P. U. Lima, P. Moreno, A. Pahliani, J. Santos-Victor, M. T. J. Spaan, and J. Sequeira, "ISRobotNet: A testbed for sensor and robot network systems," in *Proc. of International Conference on Intelligent Robots and Systems*, 2009, pp. 2827–2833.

[140] S. Janansefat, K. Akkaya, I. F. Senturk, and M. Gloff, "Rethinking connectivity restoration in wsns using feedback from a low-cost mobile sensor network testbed," in *38th Annual IEEE Conference on Local Computer Networks - Workshops*, Oct 2013, pp. 108–115.

[141] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, and M. Srivastava, "Cyclops: In situ image sensing and interpretation in wireless sensor networks," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, ser. SenSys '05. New York, NY, USA: ACM, 2005, pp. 192–204. [Online]. Available: http://doi.acm.org/10.1145/1098918.1098939

[142] S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "Mesheye: A hybrid-resolution smart camera mote for applications in distributed intelligent surveillance," in *2007 6th International Symposium on Information Processing in Sensor Networks*, April 2007, pp. 360–369.

[143] A. Rowe, D. Goel, and R. Rajkumar, "Firefly mosaic: A vision-enabled wireless sensor networking system," in *28th IEEE International Real-Time Systems Symposium (RTSS 2007)*, Dec 2007, pp. 459–468.

[144] A. Kerhet, M. Magno, F. Leonardi, A. Boni, and L. Benini, "A low-power wireless video sensor node for distributed object detection," *Journal of Real-Time Image Processing*, vol. 2, no. 4, pp. 331–342, 2007. [Online]. Available: http://dx.doi.org/10.1007/s11554-007-0048-7

[145] P. Chen, P. Ahammad, C. Boyer, S.-I. Huang, L. Lin, E. Lobaton, M. Meingast, S. Oh, S. Wang, P. Yan, A. Y. Yang, C. Yeo, L.-C. Chang, J. D. Tygar, and S. S. Sastry,

"Citric: A low-bandwidth wireless camera network platform," in *2008 Second ACM/IEEE International Conference on Distributed Smart Cameras*, Sept 2008, pp. 1–10.

[146] R. Kleihorst, A. Abbo, B. Schueler, and A. Danilin, "Camera mote with a high-performance parallel processor for real-time frame-based video processing," in *2007 First ACM/IEEE International Conference on Distributed Smart Cameras*, Sept 2007, pp. 109–116.

[147] N. Robotics. Fire bird v p89v51rd2 robotic research platform. Nex Robotics. [Online]. Available: http://www.nex-robotics.com/products/fire-bird-v-robots/fire-bird-v-p89v51rd2-robotic-research-platform.html

[148] St-Microelectronics. Stm32f4discovery. St. Microelectronics. [Online]. Available: http://www.st.com/en/evaluation-tools/stm32f4discovery.html

[149] St-Microelectronics. Stm32f405xx stm32f407xx data sheet. St. Microelectronics. [Online]. Available: http://www.st.com/en/microcontrollers/stm32f407vg.html

[150] Omnivision. Ov9655/ov9155 cmos sxga camera chip sensor with omnivision technology. Omnivision. [Online]. Available: http://electricstuff.co.uk/OV9655-datasheet-annotated.pdf

[151] T. Instruments. Cc2500 low-cost low-power 2.4 ghz rf transceiver. Texas Instruments. [Online]. Available: http://www.ti.com/product/CC2500

[152] S. association. Sd specifications. SD association. [Online]. Available: https://www.sdcard.org/developers/overview/sdio/sdio_spec/Simplified_SDIO_Card_Spec.pdf

[153] N. Michael, J. Fink, and V. Kumar, "Experimental Testbed for Large Multirobot Teams," *IEEE Robotics & Automation Magazine*, vol. 15, no. 1, pp. 53–61, mar 2008. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4476329

[154] S. Bensalem, M. Gallien, F. Ingrand, I. Kahloul, and N. Thanh-Hung, "Designing autonomous robots: Toward a more dependable software architecture," *IEEE Robotics and Automation Magazine*, vol. 16, no. 1, pp. 67–77, mar 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4799448

[155] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, jan 2013. [Online]. Available: http://link.springer.com/10.1007/s11721-012-0075-2

[156] S. Arumugam, R. K. Kalle, and A. R. Prasad, "Wireless Robotics: Opportunities and Challenges," *Wireless Personal Communications*, vol. 70, no. 3, pp. 1033–1058, mar 2013. [Online]. Available: http://link.springer.com/10.1007/s11277-013-1102-3

[157] A. M. Khamis, F. J. Rodríguez, and M. A. Salichs, "Remote Interaction with Mobile Robots," *Autonomous Robots*, vol. 15, no. 3, pp. 267–281, 2003. [Online]. Available: http://link.springer.com/article/10.1023/A{%}3A1026268504593

[158] E. A. Thompson, E. Harmison, R. Carper, R. Martin, and J. Isaacs, "Robot teleoperation featuring commercially available wireless network cards," *Journal of Network and Computer Applications*, vol. 29, no. 1, pp. 11–24, jan 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804504000682

[159] M. Luimula, K. Sääskilahti, T. Partala, S. Pieskä, and J. Alaspää, "Remote navigation of a mobile robot in an RFID-augmented environment," *Personal and Ubiquitous Computing*, vol. 14, no. 2, pp. 125–136, jun 2009. [Online]. Available: http://link.springer.com/10.1007/s00779-009-0238-3

[160] K.-S. Hwang, W.-H. Hsiao, G.-T. Shing, and K.-J. Chen, "Rapid Prototyping Platform for Robotics Applications," *IEEE Transactions on Education*, vol. 54, no. 2, pp. 236–246, may 2011. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5464382

[161] J. L. Sanchez-Lopez, J. Pestana, P. de la Puente, and P. Campoy, "A Reliable Open-Source System Architecture for the Fast Designing and Prototyping of Autonomous Multi-UAV Systems: Simulation and Experimentation," *Journal of Intelligent & Robotic Systems*, oct 2015. [Online]. Available: http://link.springer.com/10.1007/s10846-015-0288-x

[162] A. Bonarini, M. Matteucci, M. Migliavacca, and D. Rizzi, "R2P: An open source hardware and software modular approach to robot prototyping," *Robotics and Autonomous Systems*, vol. 62, no. 7, pp. 1073–1084, jul 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092188901300153X

[163] A. Brunete, M. Hernando, E. Gambao, and J. E. Torres, "A behaviour-based control architecture for heterogeneous modular, multi-configurable, chained micro-robots," *Robotics and Autonomous Systems*, vol. 60, no. 12, pp. 1607–1624, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2012.09.019

[164] IEEE. Ieee 802.15.4: Wireless personal area networks (pans). [Online]. Available: http://standards.ieee.org/getieee802/download/802.15.4-2015.pdf

[165] IEEE. Ieee 802.11: Wireless lans. [Online]. Available: https://standards.ieee.org/about/get/802/802.11.html

[166] G. Ferenc, Z. Dimic, M. Lutovac, J. Vidakovic, and V. Kvrgic, "Distributed robot control system implemented on the client and server PCs based on the CORBA protocol," in *Embedded Computing (MECO), 2012 Mediterranean Conference on*, 2012, pp. 158–161.

[167] J. H. AHNN, "The Robot control using the wireless communication and the serial communication," Cornell University, Tech. Rep., 2007.

[168] H. Bruyninckx, P. Soetens, and B. Koninckx, "The real-time motion control core of the Orocos project," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 2. IEEE, 2003, pp. 2766–2771. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1242011

[169] R. Volpe, I. Nesnas, T. Estlin, D. Mutz, R. Petras, and H. Das, "The CLARAty architecture for robotic autonomy," in *2001 IEEE Aerospace Conference Proceedings (Cat. No.01TH8542)*, vol. 1. IEEE, 2001, pp. 1/121–1/132. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=931701

[170] I. Chen, B. MacDonald, B. Wünsche, G. Biggs, and T. Kotoku, "A simulation environment for OpenRTM-aist," in *2009 IEEE/SICE International Symposium on System Integration: SI International 2008 - The 2nd Symposium on System Integration*. IEEE, nov 2009, pp. 113–117. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5384547

[171] G. Biggs, R. B. Rusu, T. Collett, B. Gerkey, and R. Vaughan, "All the robots merely players: History of player and stage software," *IEEE Robotics and*

Automation Magazine, vol. 20, no. september, pp. 82–90, 2013. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6544562

[172] J. Kramer and M. Scheutz, "Development environments for autonomous mobile robots: A survey," *Autonomous Robots*, vol. 22, no. 2, pp. 101–132, dec 2006. [Online]. Available: http://link.springer.com/10.1007/s10514-006-9013-8

[173] N. Mohamed, J. Al-Jaroodi, and I. Jawhar, "Middleware for Robotics: A Survey," in *2008 IEEE Conference on Robotics, Automation and Mechatronics*. IEEE, sep 2008, pp. 736–742. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4681485

[174] A. Elkady and T. Sobh, "Robotics Middleware: A Comprehensive Literature Survey and Attribute-Based Bibliography," *Journal of Robotics*, vol. 2012, pp. 1–15, 2012. [Online]. Available: http://www.hindawi.com/journals/jr/2012/959013/

[175] Microsoft. Microsoft robotics developer studio 4. Microsoft. [Online]. Available: http://www.microsoft.com/en-in/download/details.aspx?id=29081

[176] Cyberbotics. Webots. Cyberbotics. [Online]. Available: https://www.cyberbotics.com/overview

[177] T. Balch. TeamBots. Robot Control. [Online]. Available: http://www.teambots.org/

[178] F. Christopher and S. Reid. Carnegie mellon robot navigation toolkit. Carnegie Mellon university. [Online]. Available: http://carmen.sourceforge.net/doc/

[179] D. Blank, D. Kumar, L. Meeden, and H. Yanco, "Pyro: A python-based versatile programming environment for teaching robotics," *J. Educ. Resour. Comput.*, vol. 4, no. 4, pp. 1–27, 2003. [Online]. Available: http://portal.acm.org/citation.cfm?id=1047569%5Cnhttp://doi.acm.org/10.1145/1083310.1047569

[180] S. F. S. McCanne. ns network simulator (2.34). Open source. [Online]. Available: http://www.isi.edu/nsnam/ns/

[181] D. I. Inc. Xbee / xbee-pro digimesh 2.4 modules. Digi International Inc. [Online]. Available: https://www.digi.com/support/productdetail?pid=3524&type=documentation

[182] S. Chiaravalloti, F. Idzikowski, and Ł. Budzisz, "Power consumption of WLAN network elements," Telecommunications Networks Group, Tech. Rep. August, 2011.

[183] N. semiconductor. nrf24l01+ ultra low power 2.4ghz rf transceiver ic. Nordic semiconductor. [Online]. Available: https://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P

[184] Jin-Shyan Lee, "An Experiment on Performance Study of IEEE 802.15.4 Wireless Networks," in *2005 IEEE Conference on Emerging Technologies and Factory Automation*, vol. 2. IEEE, 2005, pp. 451–458. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1612712

[185] J. Edwards, F. Demers, M. St-Hilaire, and T. Kunz, "Comparison of ns2.34's ZigBee/802.15.4 implementation to Memsic's IRIS Motes," in *2011 7th International Wireless Communications and Mobile Computing Conference*. IEEE, jul 2011, pp. 986–991. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5982675

[186] J.-S. Lee, Y.-W. Su, and C.-C. Shen, "A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi," in *IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2007, pp. 46–51. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4460126

[187] K. R. Anupama, M. V. Shenoy, and S. R. CH, "BSwarm robot — A low cost mobile wireless sensor research platform using COTS products," in *2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*. IEEE, feb 2015, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7091533

[188] Polulu. Micro metal gearmotors. Polulu. [Online]. Available: https://www.pololu.com/category/60/micro-metal-gearmotors

Publications based on current work

Journal Publications

1. S. Sankhar Reddy CH., Patlolla sharath, Anita Agrawal and Anupama K.R, ”HexaMob—A Hybrid Modular Robotic Design for Implementing Biomimetic Structures”, Robotics, 2017, 6, 27.
2. S. Sankhar Reddy CH., A. Agrawal, and A. Karuppiah, “Modular Self-Reconfigurable Robotic Systems: A Survey on Hardware Architectures,” J. Robot., vol. 2017, pp. 1–19, 2017.
3. S. Sankhar Reddy CH., Anita Agrawal , Anupama K. R, ”Quanta - A Platform for Rapid Control and Monitoring of Heterogeneous Robots”, Digital communication and networks, Elsevier (communicated).

Conference Publications

1. K. R. Anupama, S. Sankhar Reddy CH., and M. V. Shenoy, “FlexEye — A flexible camera mote for sensor networks,” in 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN), 2015, pp. 1010–1015.
2. K. R. Anupama, M. V. Shenoy, and S. Sankhar Reddy CH, “BSwarm robot — A low cost mobile wireless sensor research platform using COTS products,” in 2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), 2015, pp. 1–5.
3. S. Sankhar Reddy CH., A. Agrawal, and K. R. Anupama, “SQ-BOT - A Modular Robot Prototype for Self- Reconfiguring Structures,” in International Conference on Robotics: Current Trends and Future Challenges (RCTFC), 2016, pp. 1–6.
4. K. R. Anupama, M. V. Shenoy, and S. Sankhar Reddy CH, “Platform for biomimetic swarms,” in Proceedings of the 2015 Conference on Advances In Robotics - AIR ’15, 2015, pp. 1–6.

Brief Biography of the Candidate

S SANKHAR REDDY CH. received Bachelor of Engineering degree in the Electronics and instrumentation and Master of science in Economics from BITS Pilani university, Pilani, India in 2009. He received his Master of engineering degree in Embedded systems from BITS Pilani university, Pilani, India in 2011.

He is currently working as a lecturer in Electrical and Electronics Engineering department at BITS Pilani, K K Birla Goa campus, Goa, India from 2012. His research interest lies in the applications of embedded systems such as Robotics and Network Embedded systems. He has participated in numerous workshops conducted by ATMEL, Cypress semiconductors and National Institute of Oceanography in the field of embedded systems and robotics. He is currently doing research in the domain of biomimetic robotics using modular robots and published papers in international journals and conferences..

Brief Biography of the Supervisor

K. R. Anupama is a Professor in the department of Electrical and Electronics Engineering at BITS Pilani, K K Birla Goa Campus, Goa, India. She received her Ph.D. from BITS-Pilani, Pilani Campus, Rajasthan, in 2005, on the topic of Routing Protocols for Mobile Ad Hoc Networks.

She has around 14 years of experience in teaching. Her research interest lies in fields of embedded systems and network embedded applications. She guided numerous doctoral students in the fields of communication and embedded systems. She is actively pursuing research for utilization of robotics and wireless sensor networks in pipeline monitoring. She has around 35 publications in various international journals and conferences.

Brief Biography of the Co-Supervisor

Dr. Anita Agrawal is an Assistant professor. in the department of Electrical and Electronics Engineering at BITS Pilani, K K Birla Goa Campus, Goa, India since 2004. She has received Ph.D. degree from Government College of Engineering, Amravati, 2009, on the topic of Medical diagnosis using Image processing.

She has 23 years of experience in teaching. Her research interest lies in fields of embedded systems and medical image processing. She is co-supervisor for three students working in the fields of MEMS, Robotics and Biomedicine. She is actively pursuing on employment on FPGAs in biomedicine and has over 19 publications in various international journals and conferences.