MEHHATROONIKAINSTITUUT

Mehhanosüsteemide komponentide õppetool

MHK40LT

Vladislav Babushkin

A Robust Control System For A Mobile Robot

Veakindel süsteem liikurroboti jaoks

Autor taotleb tehnikateaduse bakalaureuse akadeemilist kraadi

Tallinn 2016

AUTORIDEKLARATSIOON

Deklareerin, et käesolev lõputöö on minu iseseisva töö tulemus.
Esitatud materjalide põhjal ei ole varem akadeemilist kraadi taotletud.
Töös kasutatud kõik teiste autorite materjalid on varustatud vastavate viidetega.

Töö valmis Dmitry Shvarts juhendamisel
"20" Mai 2016 a.

Töö autor

.... allkiri

Töö vastab bakalaureusetööle esitatavatele nõuetele.

"......."....................20.…a.

Juhendaja

.... allkiri

Lubatud kaitsmisele.

.... õppekava kaitsmiskomisjoni esimees "......."....................20..…
a.

.... allkiri

TTÜ mehhatroonika instituut

Mehhanosüsteemide komponentide õppetool

BSc LÕPUTÖÖ ÜLESANNE

2015/2016 aasta **kevad** semester

Üliõpilane: Vladislav Babuškin, MAHB103653.

Õppekava MAHB02/09 – Mehhatroonika.

Spetsialiseerumine: Mehhatroonika.

Juhendaja: Teadur, Dmitry Shvarts.

LÕPUTÖÖ TEEMA:
Veakindel süsteem liikurroboti jaoks.

A Robust Control System For A Mobile Robot**.**

Töös lahendatavad ülesanded ja nende täitmise ajakava:

Nr	**Ülesande kirjeldus**	**Täitmise tähtaeg**
1.	Ülesande püstitus. Ülevaade ja analüüs.	27.03.2015
2.	Riistvara , tarkvara ja komponentide valik ja projektarvutused: korpuse ja liikuvate osade projekteerimine. Disain ja tellides tarkvara.	10.04.2015
3.	Prototüüpi valmistamine.	27.04.2015
4.	Praktilised andmed analüüsi tegemine ja süsteemi silumine.	10.05.2015
5.	Töö lõppvormistus, trükkimine ja köitmine.	19.05.2016

Lahendatavad insenertehnilised ja majanduslikud probleemid: Valmistada robotiplatvorm mis saab iseseisvalt lahendada probleemid. Mis võivad ilmneda töötamise ajal.
Täiendavad märkused ja nõuded: Kvaliteedi saavutamise lihtsate süsteemide baasil.
Töö keel: Inglise keel.

Kaitsmistaotlus esitada hiljemalt: **21.03.2015** **Töö esitamise tähtaeg: 20.05.2016**

Üliõpilane Vladislav Babuskin /allkiri/ .... kuupäev: **20.03.2016**

Juhendaja Dmitry Shvarts /allkiri/ .... kuupäev: **20.03.2016**

CONTENT

List of Abbreviations

AI - Artificial Intelligence.

MPU - Module processing unit

MD - Motor Driver.

CS - Current sensor.

DCS - Daisy chain Switch.

ASCII - American Standard Code for Information Interchange

Dictionary of communication between modules

OBS – Obstacle in enviroment

PTT – Destination point to travel

TRV – Traveled path

ANG - Angle

DST - Distance

SCM – Scanner Module

PSM – Power Supply Module

MSM – Motor Shield Module

LTT – Locomotion Tracking and Tactile Module

CM – Communication Module

CMD - Command

RSP - Response

CLL - Call

SCK – System Check

COK – Response on re-call

SOK – System check OK

SNK - System check NOK

STP – Emergency Stop.

1 INTRODUCTION

1.1 Introduction

At the moment, the most versatile, autonomous, multifunctional and intelligent form of the exploration unit is a human being. But Human is very fragile creature, and demanding of the environment condition and stand in constant need of a full range of resources. It is also unacceptable to send a man into an unknown environment for obtaining information as well as there is a significant chance of possible losses. Consequently, we need the instrument to get the necessary information in a safe way.

The robotic system already has successfully explored a variety of places such as volcanoes, deserts, and ices of Antarctica. Nevertheless, all of these incredible places with severe environmental conditions and impassable reliefs are available to the person and, if necessary, it is possible to return fairly quickly mobile platform to the laboratory and fix all malfunction. All of this can not be compared with conditions on other planets, and the availability of platforms with was sent to the other world. The situation with communication between center or station and device is not better. Time counts by hours, days or even years. In this situation becomes clearly to understand that the unit must continue to perform its core tasks while it has no communication with its center. In other words, be autonomous.

According to a recent survey, the concept of autonomy for exploratory mobile robot platforms, in my opinion, is quite limited. In general terms, the apparatus is enough to operate in a wireless mode and no need for middleware feedback from the platform. Until the task is executed with no human intervention, it can be called an autonomous unit. However, until the moment when appears error, wich one can not be self-solved. Because it is not trained for this kind of situations. In this paper represented a personal vision of possible solution in theory and practice.

The main idea of creating a more autonomous unit the apply the robust system concept on mobile robot platform that the device can independently solve problems and adapt to internal changes.

1.2 Task Definition

The main aim of this project is to design a system for mobile robot platform which will be able to continue performing prescribed tasks in case of different types of malfunction.

The system can called robust if it can continue with their prescribed task even if occurring faults of hardware, software or input data. Also, the error protection is an important pre-set for allowing systems to achieve a failsafe operation.

For the hardware level of the system is necessary to create a sufficiently flexible system to create the possibility for the device to operate without or bypassing of the faulty elements. In this work to the word "flexibility" is gave special meaning. It means that the system awarded by additional elements each of which has its special Applicability and do not duplicate and it does not copy the existing elements but under the cooperation, one or few segment can obtain the necessary data and send it to the host or change state in low-level electronics. I.e. need to improve the system to provide necessary functional and modify it to create the opportunity of multiway signals traveling.

For the Software level of the system is required to train the system to enable to itself to understand that something is wrong and teach the system how to use their hardware properly and partially to compensate the functional of the lost element. Also, the system must be prepared for extraordinary situations in the environment. For example, a collision with unregistered obstacles or correctly interpret the contradictions with different sensors.

There alway can be critical faults, which one can not solve at the place. So the success of concept can be considered as a successful if the device will work and at least with partial functional and are available for performing the primary task.

2 MECHANICAL DESING

The mechanical part present most difficult place for application of project concept. Since the breakage of mechanical elements such us reducer gear or motor, will cause serious problem to ability continue the main task.

At the stage of the first prototype is decided to organize the mechanical components in most simple form to eliminate as many as possibilities variation for failure. This decision will allow identifying faults in mechanical by AI of the robot in more detail forms for the first prototype.

2.1 Main Body

For prototype version of robot In this project was used the carcass of robot platform RP-05, which is presented on Figure 2.1 Platform carcass.

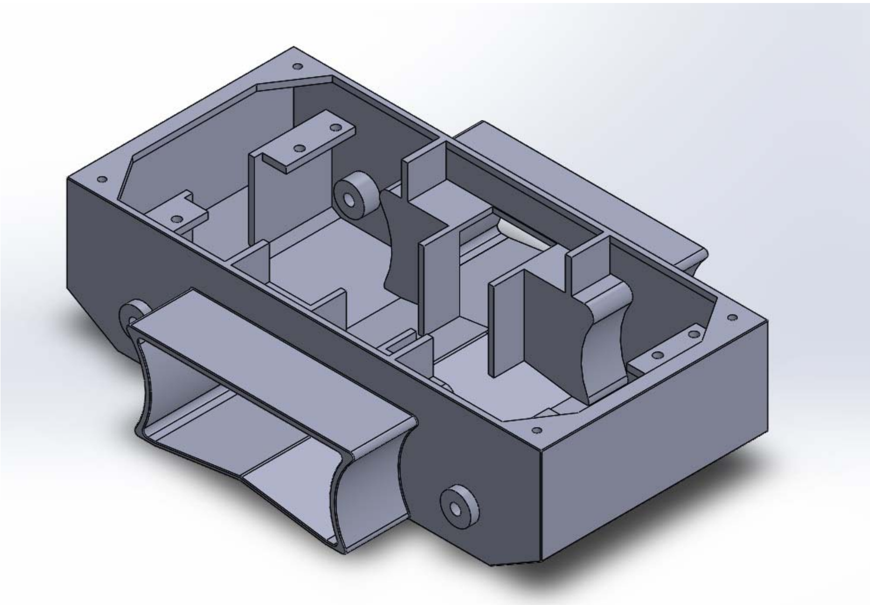


Figure 2.1.1 Platform carcass.

PR05 is a ready-made base of the mobile platform with spots for basic electronic elements of robot vehicle. The applying of ready made base has allowed beginning testing of the prototype in a shorter period. That allows to collect information about the general working process of the device and make a correction in algorithm according to the behavior of apparatus.

2.2 Apparatus Chassis

On Figure 2.2.1 Assembly of chassis, presented assembly of the motor, reducer, and chain. Desing of a frame in the final form of the first prototype is a rigidly fixed drive and driven wheel which connected through the rubber chain. Prototype frame does not contain the suspension system and represents the simplest forms of own class. However, this particular design allows a well to evaluate the results of the algorithm of self-diagnostic. The absence of all-wheel drive allows to apply feedback system and receive information from two wheel encoders and based on the data make analyze the condition of chassis.

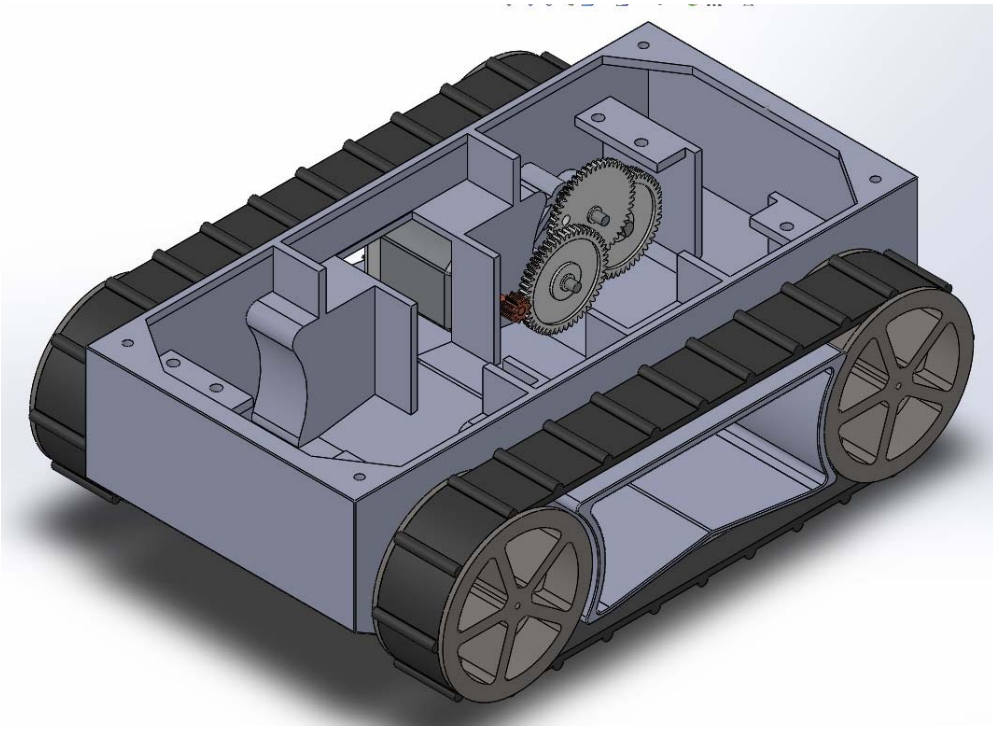


Figure 2.2.1 Assembly of chassiss.

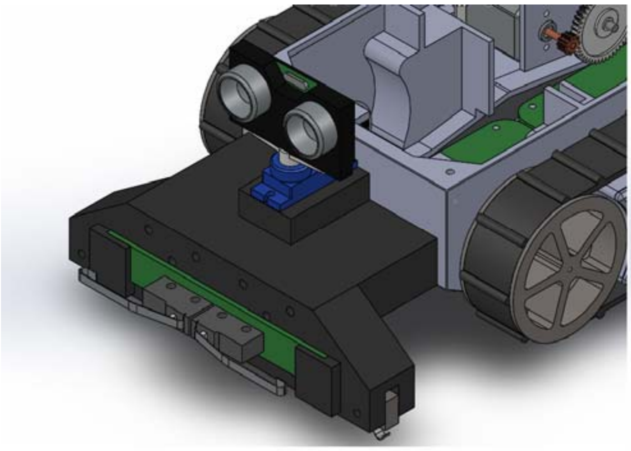
Traveling speed of robot platform at maximum efficiency of the motor can be calculating with rpm out and D (diameter) of Caterpillar wheel. Therefore, a wheel with 5 cm diameter commits 99.8 rotation per minutes. Considering that the radius of the circle L= πD, therefore, the distance covered per minute will be a πD*rpmout=1567,65 cm or 15,6 meters per minute or 26 cm/s. This value is approximate and does not take into account losses in the reducer as well as the load. The actual velocity of the vehicle will be slightly less than calculated value

2.3 Sub-construction

To interact with the environment apparatus has four rows of sensors. Each of which performs own task and simulating the senses. For example sight, touch and the vestibular apparatus. On the main body added a three additional extension for placing the sensors.

2.3.1 The holder of the front sensors

Front extension intended for fixation radar, two front and horizontal micro of switches. Placement of sensors shown in Figure 2.3.2 Front body extension.



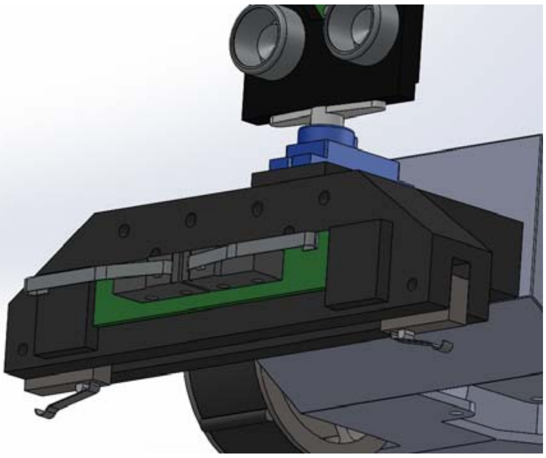


Figure 2.3.2 Front body extension.

The front two vertical micro switch looks like feelers of insect. It performs a similar function, detect objects and prevent collisions with obstacles. The horizontal microswitch provides a tactile sense of the platform chassis. Vertical micro-switch provides a tactile sense of the platform frame. Activation of at least one means that the robot locates near gap or bottom does not abut to the working surface. This State effect on working algorithm. Also on the front extension is spotting radar, which one provide the vision of environments for the device.

2.3.2 The holder of the lower sensors

The bottom extension is holding two optic sensors. The construction and placement of bottom extension presented in Figure 2.3.2

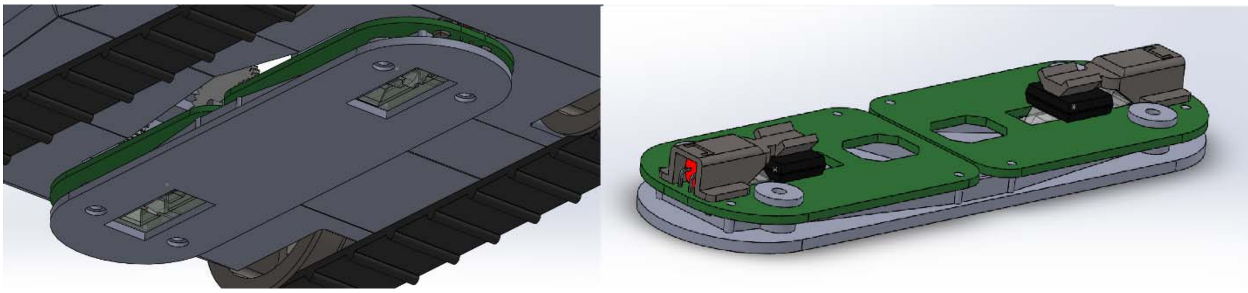


Figure 1.3.2 Optic sensors box

Their task is registering the shifting of the platform. The optic sensor box, in some sense, can be called the vestibular apparatus, but it operates only in two dimension space. However, with the participation of the horizontal microswitch can say that it measures in the third dimension also. But only two states.

2.3.3 The holder of the rear sensors

On the rear extension spotting second part of vertical and horizontal microswitches. Holder for rear sensors presented in Figure 2.3.3 Rear body extension.

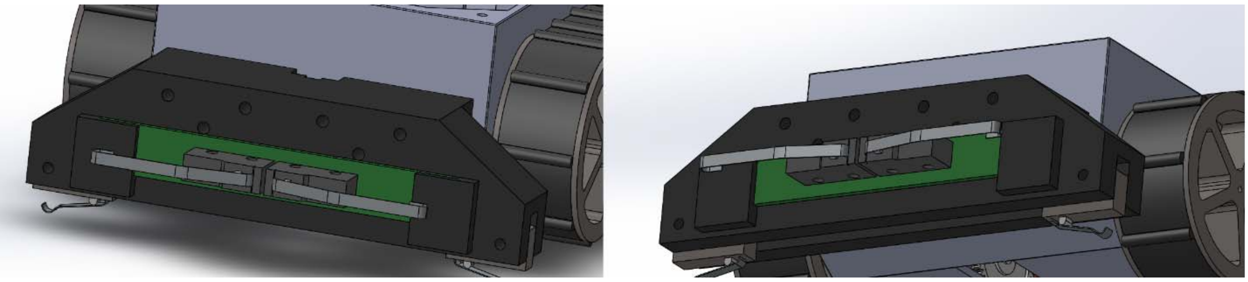


Figure 2.2.3 Rear body extension

The rear sensors group have the same task as front sensor group. Since radar have only 180 vision sector, there is no opportunity to check backward obstacles. Therefore, rear feelers prevent collisions with obstacles in case of maneuvering in reverse mode.

2.4 Battery Compartment

Since the device needs to be autonomous, it must have an own power source. Rechargeable and with enough capacity to work estimated time. Designed battery box presented in Figure 2.4.1 Battery bay.

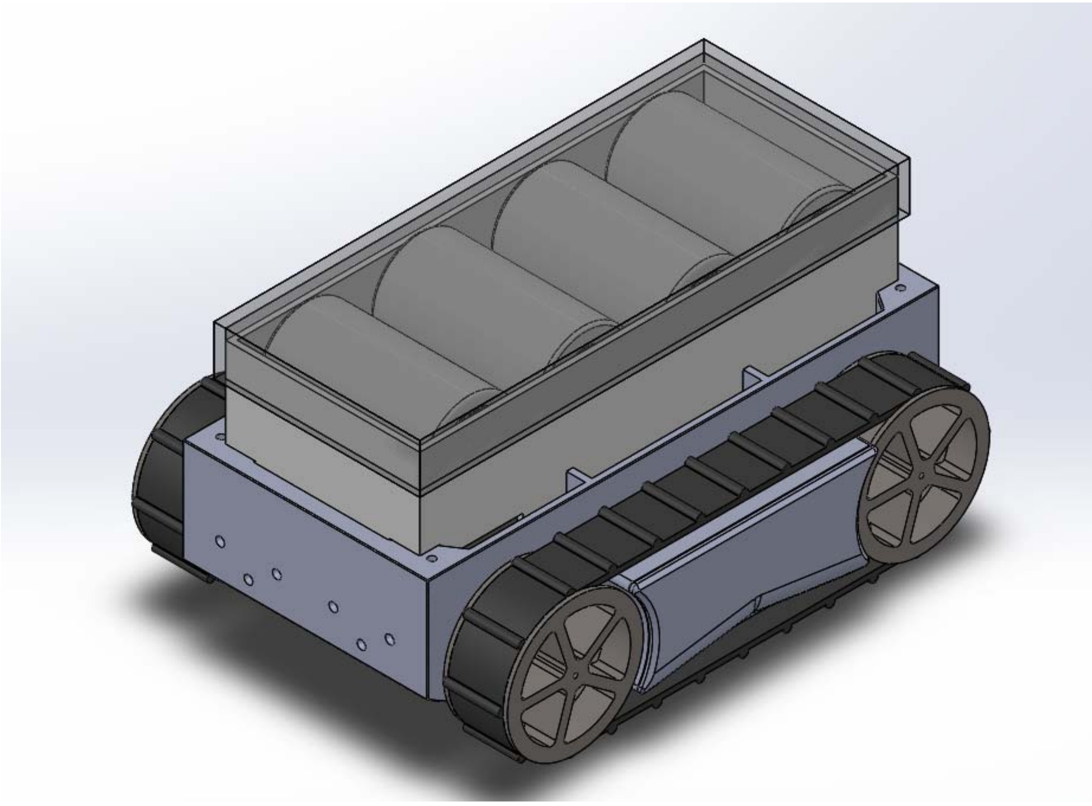


Figure 2.4.1 Battery bay

As power source is using the serial connection of four er34615m_eve lithium batteries 3.6 volts in D pack. The nominal capacity of each element is 19.0 Ah, and a nominal voltage of 3.6 V. Therefore, the main specification of PSM are 14.4 rated voltage and 19 Ah nominal capacity.

While charging process, the load on linear regulator rapidly increase, since loading segment have a current peak at 1Ah while charging one of the battery cells of PSM. To eliminate burnout of the system the voltage distribution layer put beyond the internal structure of linear regulators are equipped with a powerful heatsink with fan. Heatsink with fan presented at Figure 2.4.2 Heatsink.

Figure 2.4.2 Heatsink

The heatsink was uninstalled from the old computer video card and use 12 volts fan. Using fan online all time is not necessary because voltage regulators in normal mode operating do not emit significant amounts of heat and additional cooling by the fan only need is the case of overheating. For this purpose in heatsink installed temperature sensor and fan activating when the temperature reaches necessary value.

As a result, the primary energy distributor transferred from the internal structure to the top of PSM and shown at Figure 2.4.3 as gray box and voltage regulators installed into the metal frame of the heatsink. The final version of PSM design presented at Figure 2.4.3 PMS Overall Construction.

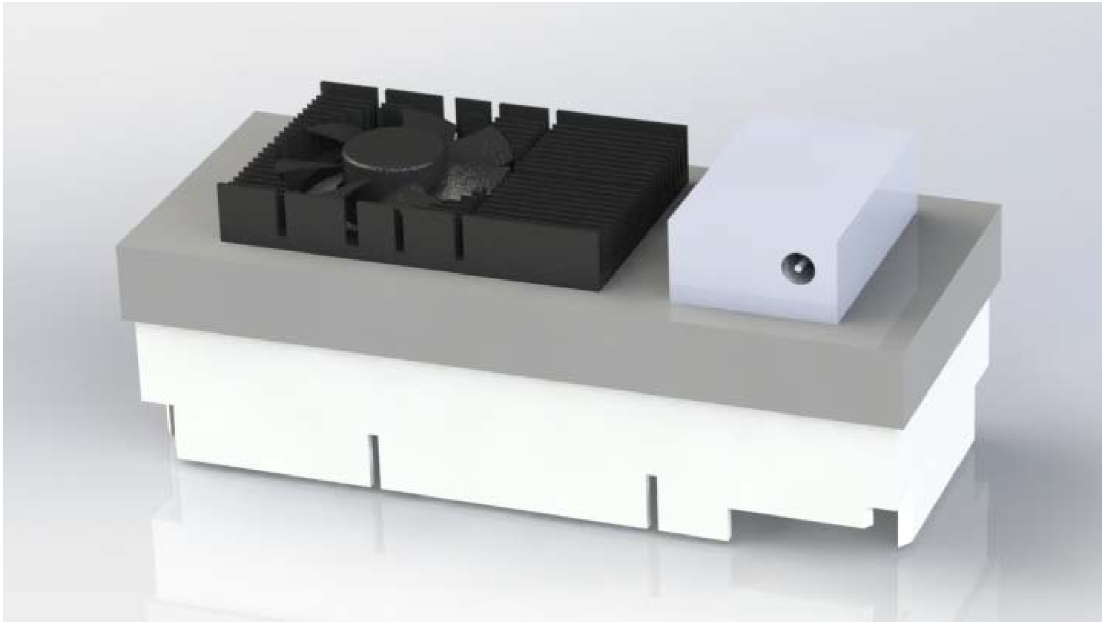


Figure 2.4.3 PSM Overall coonstuction

3 HARDWARE LEVEL

3.1 System analysis

Firstly considers the sufficient block diagram for the mobile platform to perform the task as orientation in environmental and movement in the workspace. Block diagram presented in Figure 3.1.1 Simple block diagram.

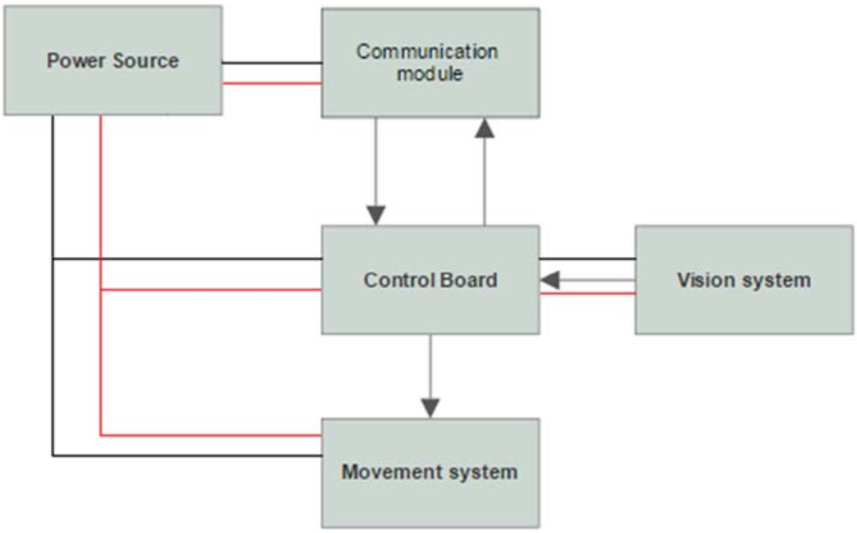


Figure 3.1.1 Simple block diagram

As defined by robust system concept the device must continue performing the prescribed task in case of malfunction mentioned above. However, this system is not tolerant to disruptions of hardware and software. To achieve the match with the concept of the robust system the device need to upgraded.

The first step in the evolution of hardware in frames of this paper is a separate main control board to four different modules. Also separate whole functional of mobile robot platform between modules. The new block diagram of scrutiny board presented in Figure 3.1 Upgraded block diagram.

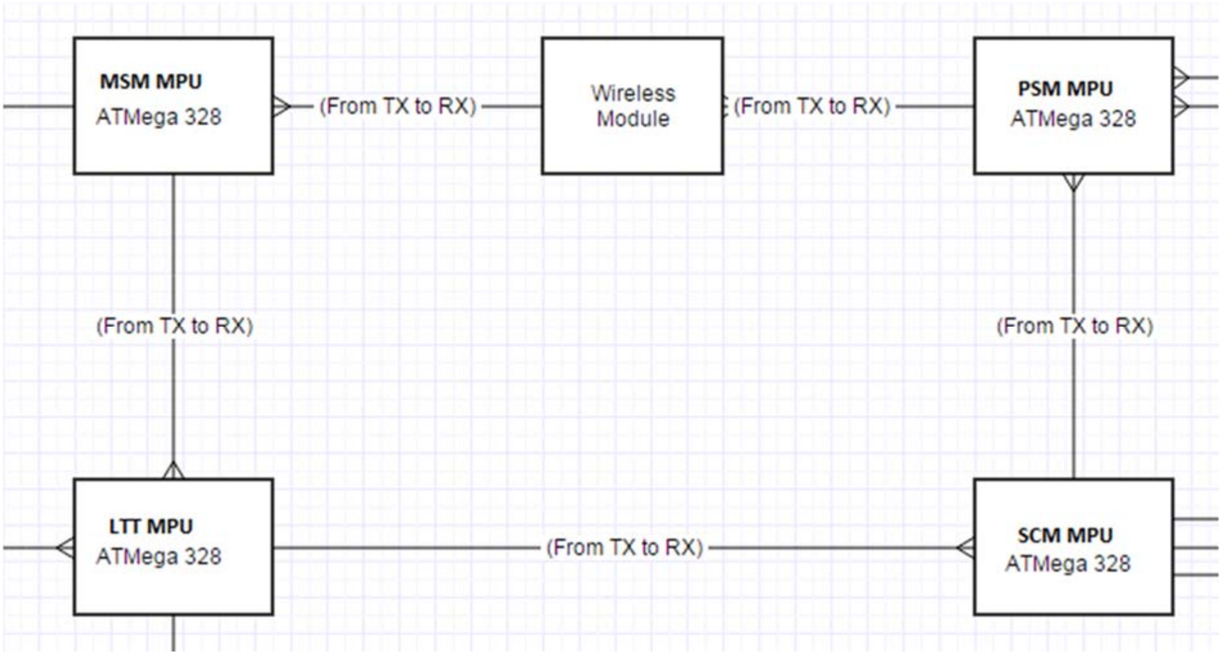


Figure 3.1.2 Upgraded block diagram.

The advantage of this decision is the possibility to arrange parallel processes, which will increase the overall performance of the device. The result of this step is that if fails one of the processors the machine will lose only part of their functional but others modules will be still able to perform an own task. Also, it this case will interrupt intercom and performing of overall device tasks become impossible. Therefore, it must take few more updates. The advantage obtained at this stage is that the device will be awake. Next update presented at Chapter 3.3 Daisy Chain switch.

3.2 Modules

Separation of the central board has allowed creating a more flexible system that allows adjusting internal processes separate from the main algorithm. Also, expand the internal functional of module unit and implemented the self-test process. Each module was also improved to meet the general concept.

Self-test can identify the fault at the stage of its inception, and make some action that the system does not get more damage while operating in failure mode.

3.2.1 Module CPU

As the core of each module used High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller ATMEGA328P-PU in a dual inline package. In Figure 3.2.1 presented the ATMega328 pinout.

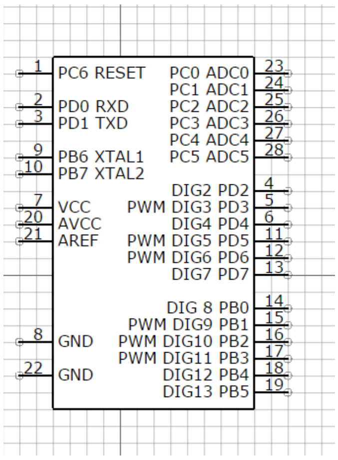


Figure 3.2.1.1 ATMega328P-PU.

In each module, ATMega used in similar standard assembly presented in Figure 3.2.1.2. For operating it need quartz external crystal 16 Mhz which sets the frequency of the processor, and the power supply. All that what installed over of presented structure relates to the module segments. Module Processing unit (MPU) a connected in serial connection.

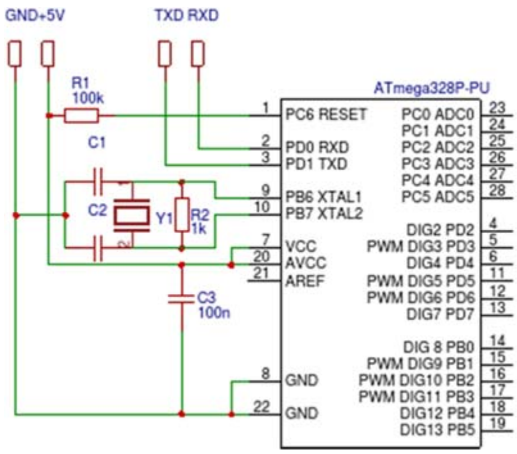


Figure 3.2.1.2 ATMega Conntroller Assembly.

To facilitate orientation in this paper the above assembly named MPU, what means Module Processing Unit.

3.2.2 Power Supply

The Power Supply Module contains four segments with considering MPU. The first one is PSM lithium battery charger with current sensor presented at Figure 3.2.2.1 Charging segment

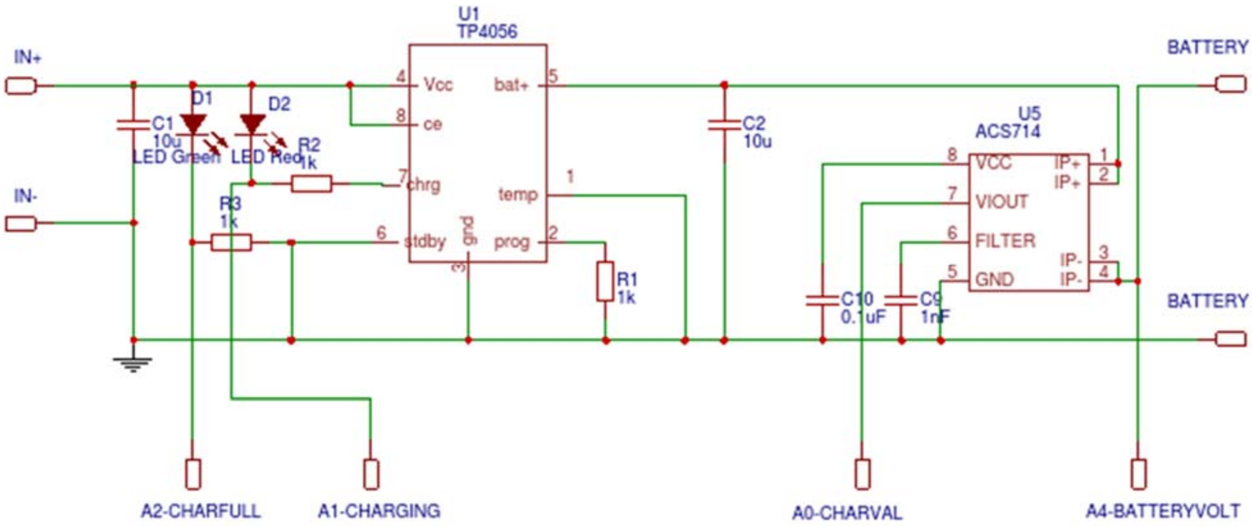


Figure 3.2.2.1 Charging segment

This segment is a combination of charging element on the base of the TP4056 controller and current sensor ACS714. When the external power source is connected to begin charging process appears an opportunity to supply device by energy without the participation of internal energy elements. In circuit mounted few additional points to track charging process. A0 and A4 Measure Current and Voltage on output. A2 and A1 retranslate data from TP406 to MPU of PSM.

Inasmuch as battery cells charging by one it need additional segment that allows extracting battery cells from battery bay. In this purpose was designed switch presented in Figure 3.2.2.2 PSM pack switch.

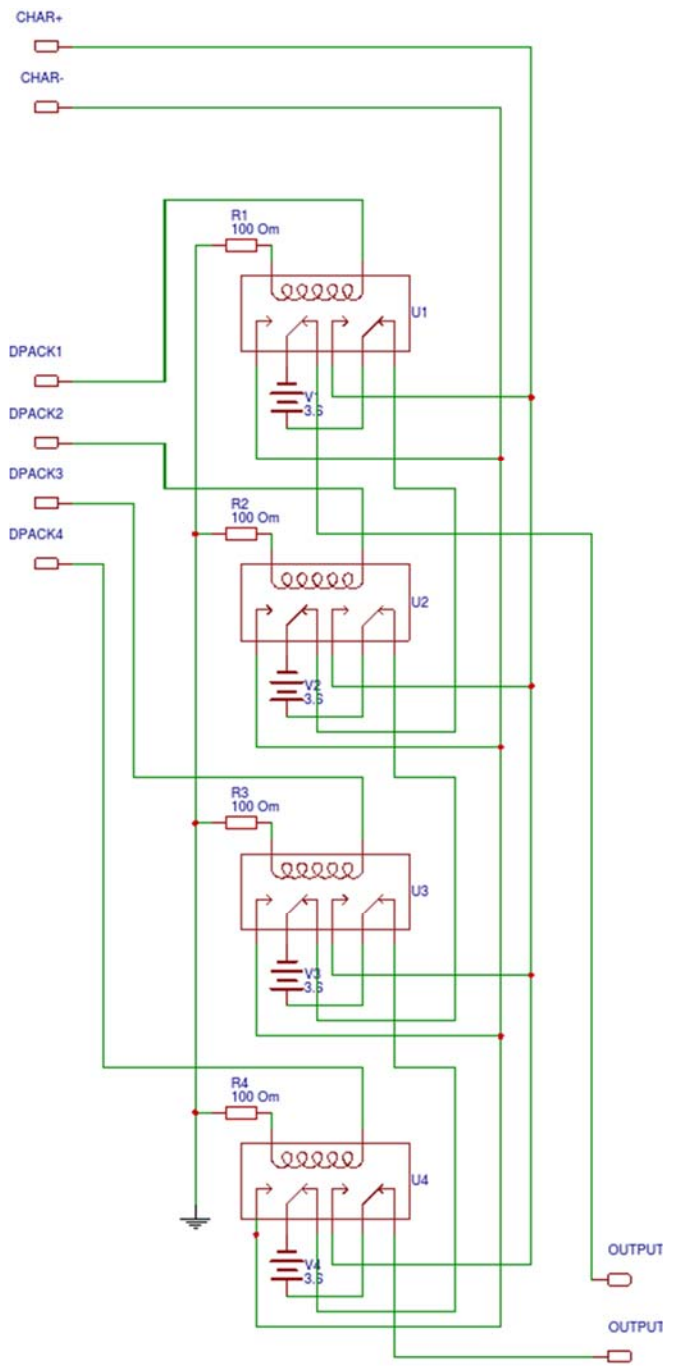


Figure 3.2.2.3 PSM pack switch

Two channel relay switch in NC keeps cells in serial connection and when one of switch change state the serial connection is interrupted. After that moment device, supplied by external source and PSM voltage segment is re-energized by external energy supply. PMS voltage part is shown in Figure 3.2.2.4 PMS voltage distributor segment.

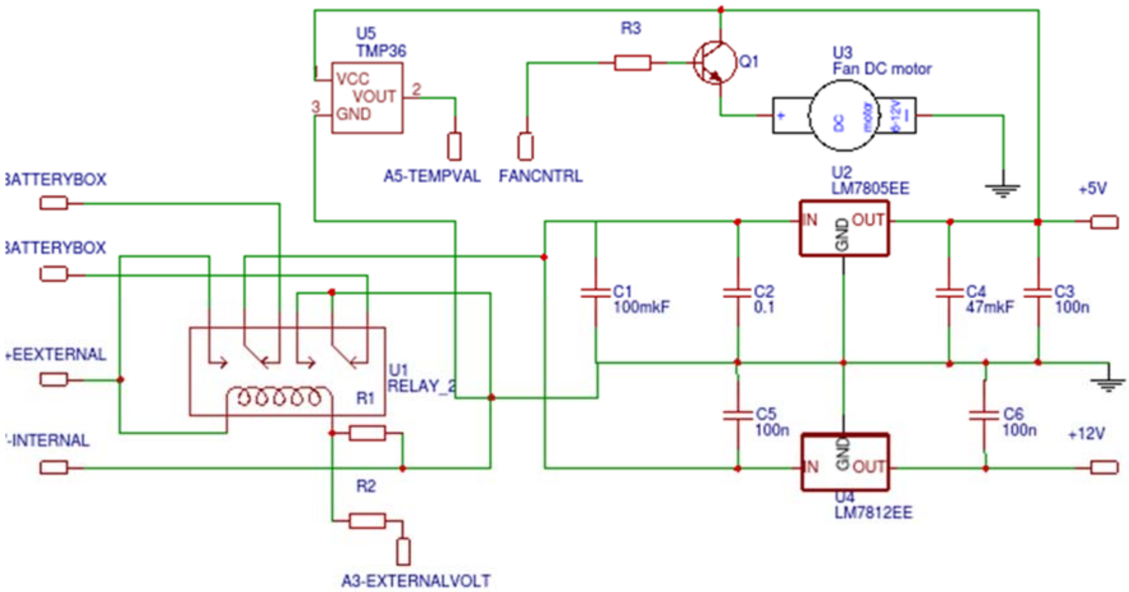


Figure 3.2.2.4 PMS voltage distributor segment

To ensure device with all necessary level of voltage battery bay output connected to output voltage distributor segment. It contains two linear voltage regulators. The LM7805 for 5 Voltage supply and LM7812 for 12 volts consumers. Linear voltage regulators spotted on the heatsink to avoid overheating of regulators. According to the value of temperature sensor the fan turning on to supply additional cooling. If the temperature is normal, the fan is inactive.

Above mentioned segments connected with MPU. The primary task of PSM is ensuring the device by energy supply. The PSM circuit organized in such a way that to provide energy for apparatus there is no need to change anything. In other words, the position of the active elements in the normal mode provides a connection to an internal power source.

In the case of charging process, self-test and energy consumption analysis there are various manipulations with batteries. Consequently, the chain in the internal power supply is interrupting, and the unit is de-energized. To not de-energize the unit and avoid the interaction of inner and outer source of energy the input of external and internal supply lines separated by the switch. When the voltage on the external source line raise the switch, interrupt the battery box and voltage regulators segment.

In a case of malfunction, the module will lose some functionality but the voltage supplying will still running. MPU provides the charging process, temperature and energy consumption monitoring and data retranslating. Charging process can delegated to floating Core. The data are retranslating solving by exception PSM from daisy chain. As a result, the only temperature monitoring and will lose, but the fan can activate from NSA. If fan will running always, there is no need to track temperature.

3.2.3 Scanner

Scanner module contains three segments. HC-SR04 ultrasonic sensor, servo and also MPU. Module task is reading distance to obstacles and angle measurement of barriers relatively actual position of apparatus.

HC-SR04 sensor emits an ultrasound at 40 kHz which to obstacle then bounce back to the sensor. According to the travel time and the speed of the sound can calculate the distance. The HC-SR04 has four pins, Ground, VCC, Trig, and Echo. The VCC and GND connected to supply network of the device and the Trig and Echo pins to Digital I/O of MPU. The connection of SC-SR04 and MPU and presented at Figure 3.2.3.1 HC-SR04 pinout.

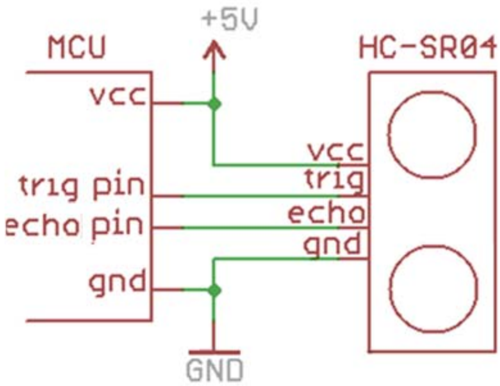


Figure 3.2.3.1 HC-SR04 pinout

After that the state of trig is set to HIGH for ten µs will be sent out an eight cycle sonic burst it will receive in the Echo pin. The Echo pin will output the time in microseconds the sound wave traveled. Visualization of the process presented in Figure 3.2.3.2 HC-SR04 Timing chart.

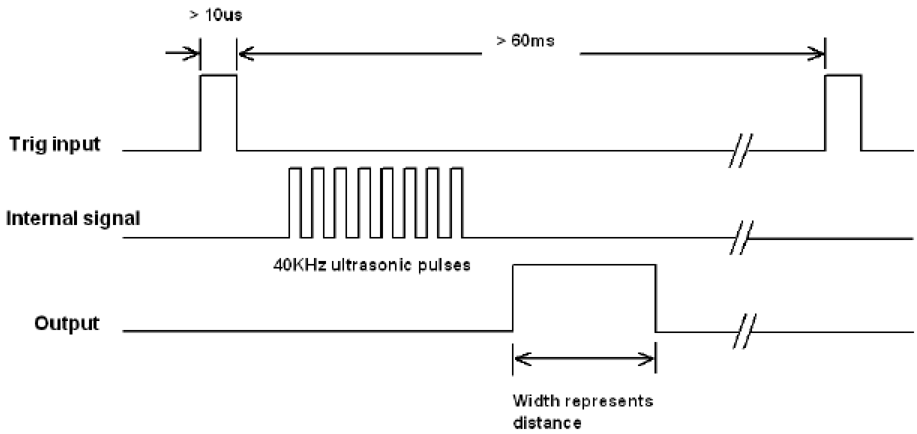


Figure 3.2.3.2 HC-SR04 Timing chart

To ensure multiple direction scanning need to rotate Ultrasonic sensor. To control rotation is used tiny and lightweight servo motor SG90 with high output power. The servo can provide guidance from 0 to 180 degrees relatively to apparatus position, and works like the standard servo.

Figure 3.2.3.3 SG90.

Radar use five position to scan enviroment. Position 0,1,2,3,4 to -90,-45,0,45,90 degree direction respectively.

Combining the above two segments in enough to create a radar which provides the vision of environment around of the device. The required processing power is not sufficient to highlight the radar as a separate module. Nevertheless, this was done intentionally because according to the overall design of mobile platform the Scanner module contains the initial position of the Floating Core.

3.2.4 Locomotion Tracking & Tactile

Movement tracking ensured by optic sensor segment of LTT Module. The circuit diagram is shown at Figure 3.2.4.1 Optic sensors.

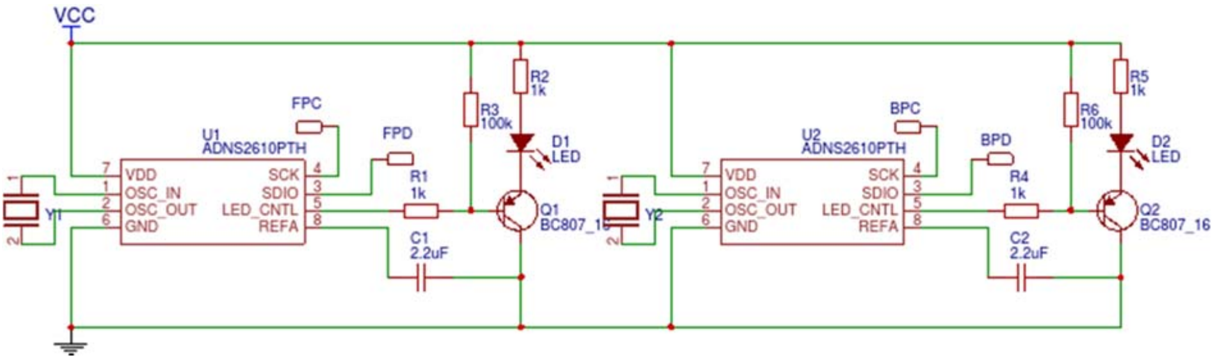


Figure 3.2.4.1 Optic sensors

Locomotion tracking device designed on the base of two similar optical mouse sensors ADNS-2610. It is used to implement a noun- mechanical tracking engine for the mobile platform to track shifting of work surface relatively to apparatus. Each of optic sensor detects delta of x and y-axis. About each other the sensors strictly fixed in bottom body extension. Also, the central point between sensors in coincides with the focal point of apparatus. Using geometric data of the sensors position on the body and change in coordinate allows calculating the rotation angle of the platform around z- axis in a workspace.

In practice, the points are attached to overall apparatus geometry and physically fixed in. However, in virtual space the situation is opposite. In program sensors expressed as two floating points which move synchronously relative each other.

It is also possible of non-realistic mapping of sensor points in virtual space. After necessary calculations of the distance traveled and the angle of rotation all false data can detecting by simple test – count distance between points. Given that the device has high accuracy, abnormalities may be caused by violation of a predetermined distance between the working surface and the bottom of the unit. The data from tactile segment can confirm or refute this cause. Tactile and feelers segment presented on Figure 3.2.4.2 Circuit diagram of a tactile segment.

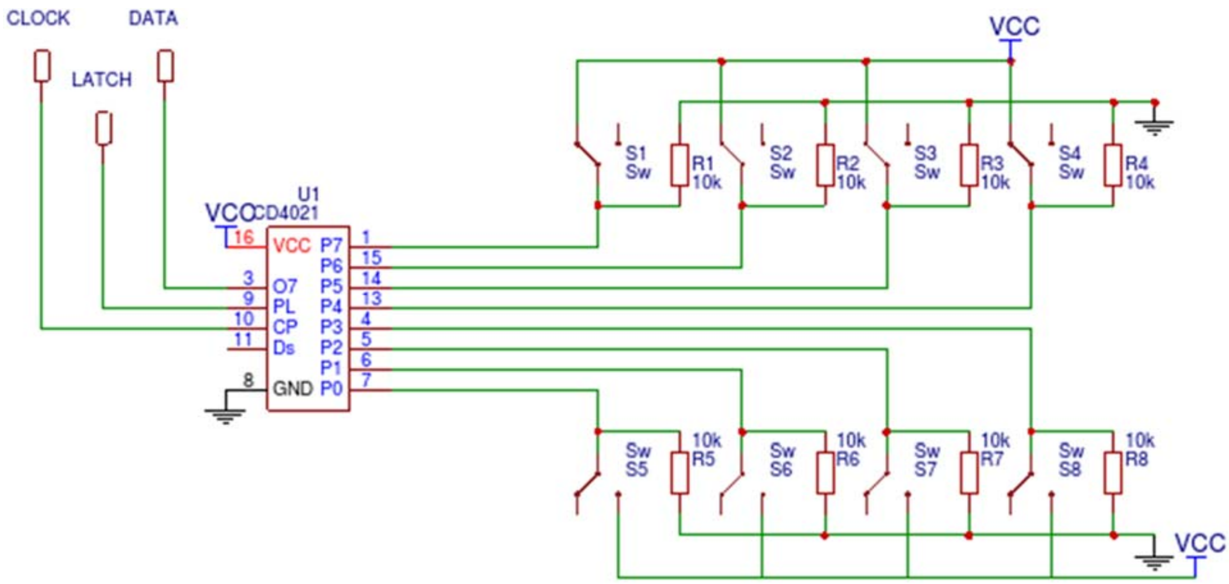


Figure 3.2.4.2 Circuit diagram of Tactile segment

Sensors of the tactile segment can divide into two groups. The first group is vertical microswitches located on the perimeter of the device it can call feelers. This group used NO port of the switch. Obstacle detection was originating by activation one or few of feelers. The second group also located on the perimeter of apparatus but positioned on the bottom. The second group used NC ports of the switch and activating in case if device chassis lose direct contact with the surface.

Activation on of the sensors from the second group also counts as obstacle detection. In this case, mobile platform AI will not use data from optic sensors even if calculation passed the self check. Since feelers and chassis tactile sensors located in device comfort zone hence next priority step is the reverse action of last one. This action should return the state of sensors in a tactile segment in normal position.

3.2.5 Motor Shield Module

To actuate the mobile platform it is necessary to ensure it by the modules that are capable of producing rotation and as a consequence – movement

As motion source in this project decided to use standard 280 DC motor. 280 is a high speed, high power, high Torque DC motor that can also work as an electric generator. Using this feature is planned for later versions of the prototype. Rated for six Volts DC, it can operates in the range of 5 up to 16 Voltage. More details about the motor shown in Figure 3.2.5.1 Motor specification.

MODEL	VOLTAGE		NO LOAD		AT MAXIMUM EFFICIENCY				STALL	
	OPERATING RANGE	NOMINAL	SPEED	CURRENT	SPEED	CURRENT	TORQUE	OUTPUT	TORQUE	CURRENT
			rpm	A	rpm	A	g cm	W	g cm	A
LD-280FC-18180	9.0～16	12	10400	0.15	8667	0.49	50	4.40	75	300

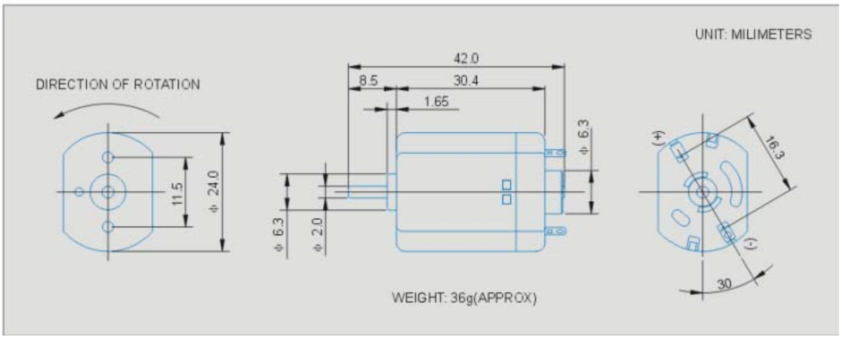


Figure 3.2.5.1 280 DC motor specification.

Since used DC permanent magnet motors, which can make noise by arcing caused by the commutation of the brushes. Noise effect on PWN signals in the device. To eliminate it used an LC low pass filter. In filter configuration, an inductance is placed in series with the motor driver and the brush and a capacitor is placed from the power feed to ground. Circuit diagram presented at Figure 3.2.5.2 LC low pass Filter.

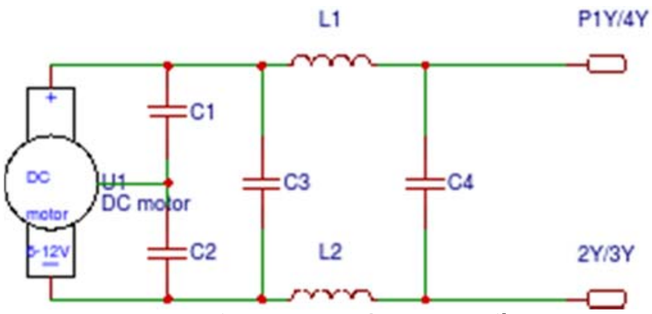


Figure 3.2.5.2 LC Low pass Filter.

Presented filter topology is useful at filtering brush noise when the motor is running. However, due to the capacitance from the switched power feed to the ground and the associated inrush current, this filter configuration can create excessive stress on the unit used to control the motor.

To avoid unwanted loan on MPU was applied the motor driver. To perform these functions selected L293D motor driver. The L293D are quadruple high-current half-H drivers. L293DNE designed to provide a bi-directional flow of acceptable current of 1 A, with a voltage range from 4.5 to 36 Volts. All outputs of the motor driver are compatible with TTL logics.

Unlike from stepper motors, the dc motors do not have the ability to perform the required number of rotation. Hence, it is continuous DC motor. Therefore, it is necessary to integrate feedback and make stop rotation when a required rotation number was done. For the purpose of chassis, as feedback applied Optical encoder. It also requires an additional driver for calculations. Since the role of computing driver carrying by MPU of movement module, there no need for additional encoder drivers. This allows connecting the encoder directly with a microcontroller. This solution will simplify the circuit diagram and mechanical design.

Figure 3.2.5.3 Trans missive encoder Sensor..

The encoder has four ports and operates two major provisions under which the contacts are closed and open. Upon rotation of the encoder in the direction of clockwise or counterclockwise going short-circuit of one pair and then a second. The order of pairs determines directions of rotation. Considering presence the motor driver through which MPU gives the direction of rotation it is possible to use as a simple tachometer. For the manufacture of the tachometer for the motor, modules will be sufficient to use only one port of the encoder. Wiring of the encoder is shown in Figure 3.2.5.4 Encoder circuit diagram.

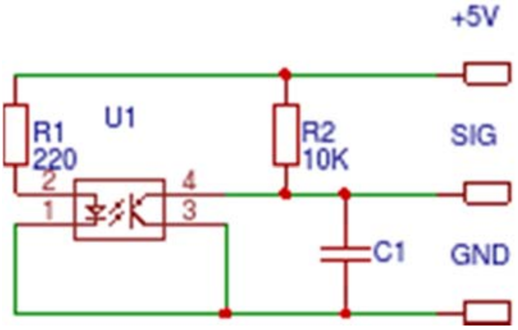


Figure 3.2.5.4 Encoder circuit diagram.

Thus by reading the number of pulses while movement, the rotation speed can be determined by the frequency of signal change and the direction identified by the motor driver with activating channel of the engine control (1A/4A or 2A/3A). This solution allows creating a full feedback without duplication of elements or functionality.

Assembly of the motor segment of MSM presented at Figure 3.2.5.5 Motor segment.

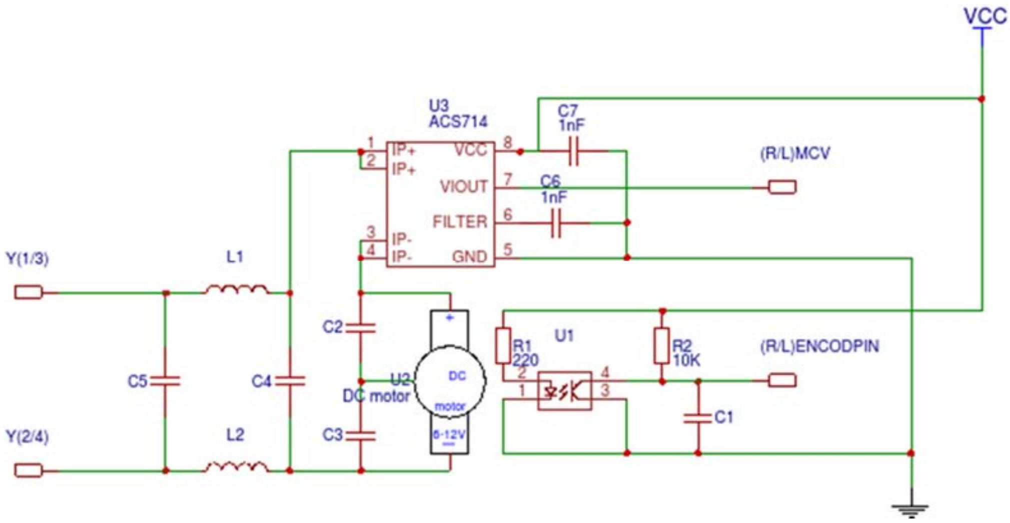


Figure 3.2.5.5 Motor segment

At presented schematic diagram is shown designed assembly of the motor segment for this project. The motor connected to the motor driver and on one of the channel installed the Current sensor. The task of the current sensor is tracking current level at motors. At this project stage, the value of CS is using while the module was performing a self-test function. But it also applied space for updates on future work. The left, and right motor segments are identical and connected to Motor control section trough Y(1-4) channels. The schematic diagram of Motor control section presented at Figure 3.2.5.5 Motor control segment.

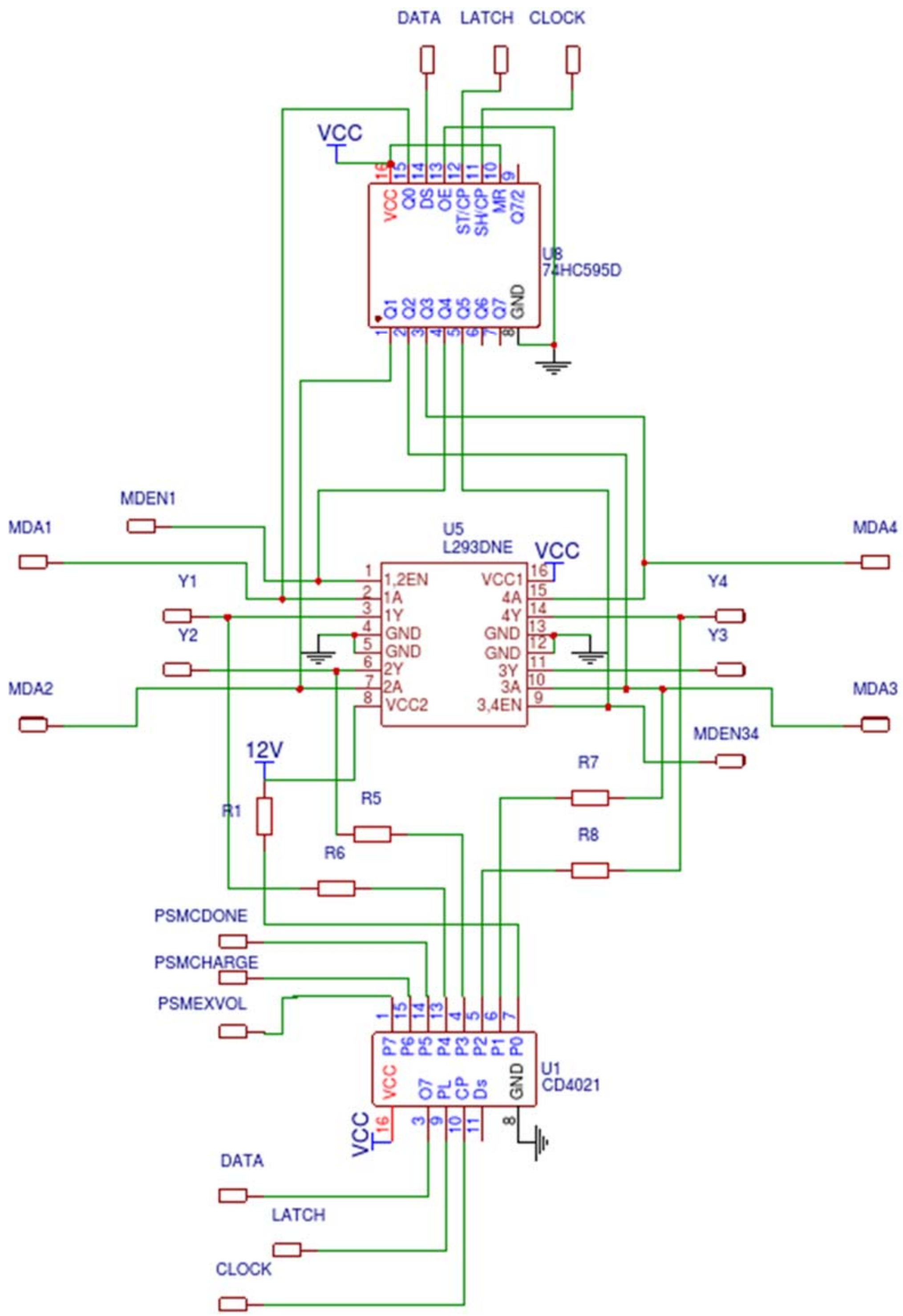


Figure 3.2.5.6 Motor control segment.

The motor control segment contains each element of NSA line and motor driver. The shift out register is not active in the normal mode of operating. This register is provided remote access to the motor driver for others MPU if case of necessary. The change in register is also inactive, and he offers a vision of actual acting between motor and motor control segments. This is necessary data for self-test in emergency and normal mode of operation.

3.2.6 Communication Module

The start and end of daisy chain inside of the unit are the communication module. As a communication module acts the Xbee S1 device presented at Figure 3.2.6.1 Xbee S1. As the communication module, it performs data transmission and receives from the host. Xbee also has additional ports what allows implementing other functions. For example, transmit data directly to the control center. Also, organize directly monitor the machine in the event of partial system failure. In the case of unsuccessful initialization involved changes in the data flow.

In other words, Xbee is only communication module but with the support of the control center can act as a core support.

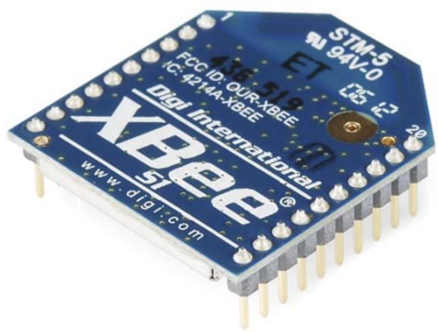


Figure 3.2.6.1 Xbee S1

Multipoint wireless networking RF XBee module do not need any configuration out-of-the-box RF communication. Within the framework of this paper, this characteristic is a significant advantage as the reduce the required amount of time before the incarnation of the prototype in the life and the beginning of the first field test.

An important issue is the speed of data transmission. After completion of the preliminary data, conversion is necessary to pass it at a sufficient rate for a comfortable operating of the module. In data specification, the given RF data rate is 250kbs. That is adequate for the transmission of data packets.

3.3 Daisy chain switch

The entire system consists of individual modules, each of which has its functions and is responsible for their task. The system needs communication link. The traditional method for the line organization is the serial connection between modules. In another word, it needs to connect RX port from previous board to TX port to next board. However, in the case of malfunction one of any MPU, the chain will be broken and data transmitting process will be disrupted. To avoid this situation was constructed switch matrix, which allows transmitting data were bypassing particular MPU. Circuit diagram of switch matrix presented at Figure 3.3.1 Switch Matrix circuit diagram.

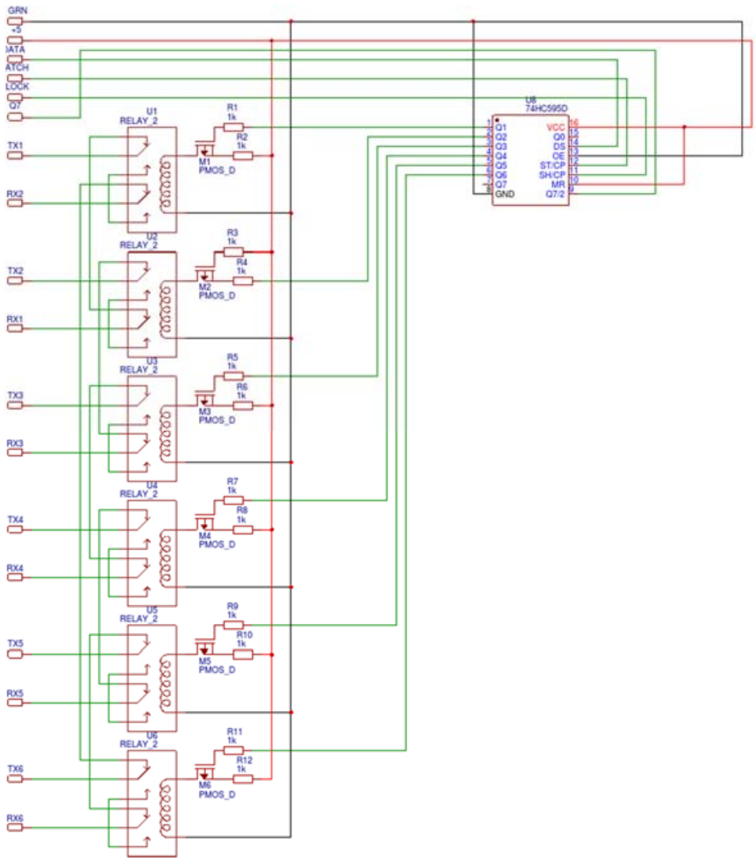


Figure 3.3.1 Switch Matrix circuit diagram.

Switch Matrix (SM) implemented by 8-bit shift registers. The ’HC595 devices contain an 8-bit serial in, the parallel out shift register that feeds an 8-bit D-type storage register. What allows to control a large number of digital ports by using a limited resource.

The new view of hardware structure presented at Figure 3.3.2 Hardware block diagram.

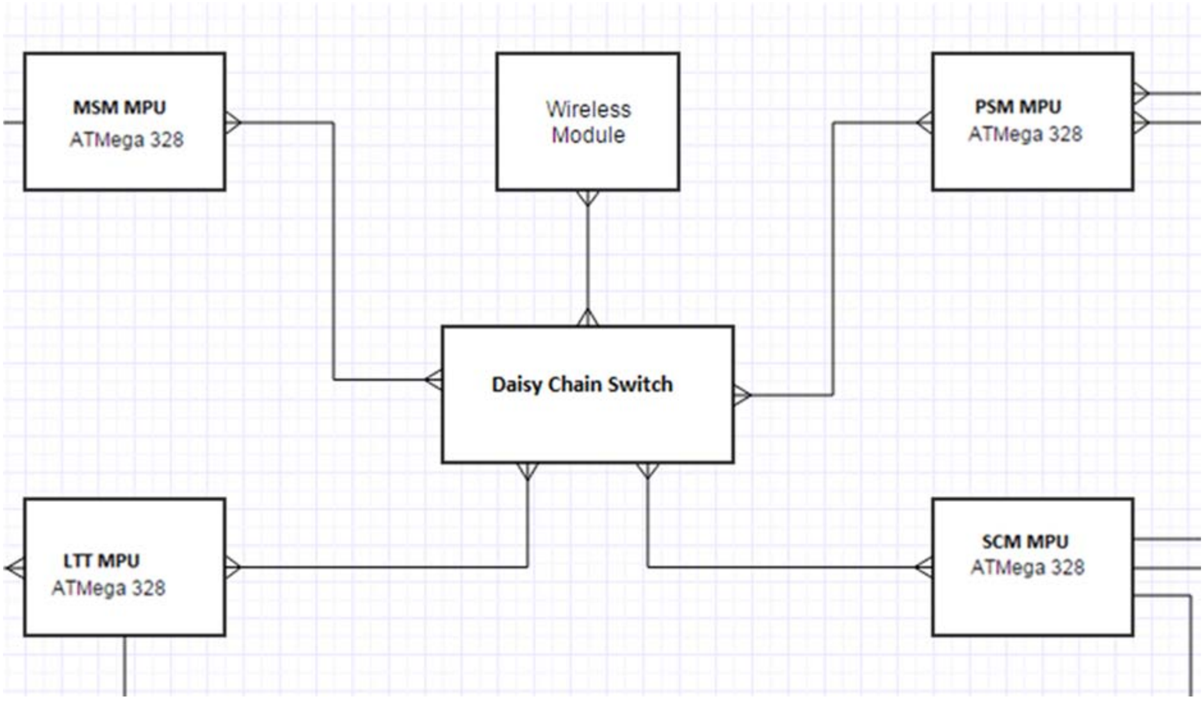


Figure 3.3.2 Hardware block diagram

Thereby becomes possible to restore the data flow with bypass of nonresponsible , damaged or wrongly operating modules. In the design of the Daisy Chain switch applied the same principles in PSM with battery cell connection method. It means that the device in normanl mode connect all modules in a serial connection as it was before switch matrix integration. Thereby in the case of any element in this module will not interrupt chain in the device. It contrast to the mentioned above modules, the Daisy chain switch do not contain own MPU. The control of DCS keeps floating core algorithm by NSA system.

In short, this upgrade compensates for one of the weaknesses described at the end of chapter 3.1 system analysis. Now the unit can stay in the mind but also have an opportunity to restore intermodular message. To compensate the second shortcomings of the first update, namely, access to the lower level of electronics. From already designed modules have been allocated existing registers in a separate theoretical group. After several improvements, this group evolved into a some analog of the human nervous system. From which got its name.

3.4 Nervous system analog (NSA)

At this moment have been done a few modification, which aimed to save the integrity of module cooperation inside of apparatus. In the case of fault one of the module, the system will self-rebuild to bypass nonresponsible module. However, the low level electronic of error module will stay without control. The whole system has several important points that must interact to perform at least partial functional which is sufficient for performing prescribed tasks. In other words, the emergency minimum of functional to do the primary task in case of malfunction. To create such an opportunity is designed the unique sub-system with the conditional name Nervous system analog.

Nervous system analog contains to lines of shift registers. First, one is serial connections of the 74HC595 shift register. The line of 74HC595 shift registers proposes to control all low-level electronics to provide an emergency minimum of functional. And the CD4021BE provides reading all available states of critical points in the device. Each line does not have own module or structure. Registers integrated into the necessary module because of ergonomics and amount of flexible wire. It is more optimal to operate only by control channels (Clock, Latch, and Data) and connect them to the network. I.e. the allocation of NSA as a solid module would raise the need to make at least 35 additional flexible connections with modules that will weighty complicate the service of the system and adversely affect the durability of the overall system.

RX of NSA contains two Shift in register CD4021BE. In a normal mode of operating the ShiftIn registers read-only feelers data (chassis tactile sensors) i.e. from eight micro switches. The indication from these sensors is critical because four of them prevent the falling from the work surface, and the other four are the latest orientation tool in space. For this reason, primary access to the shift registers belongs to TTL module. Integrated elements of the RX NSA presented in Figure 3.4.1

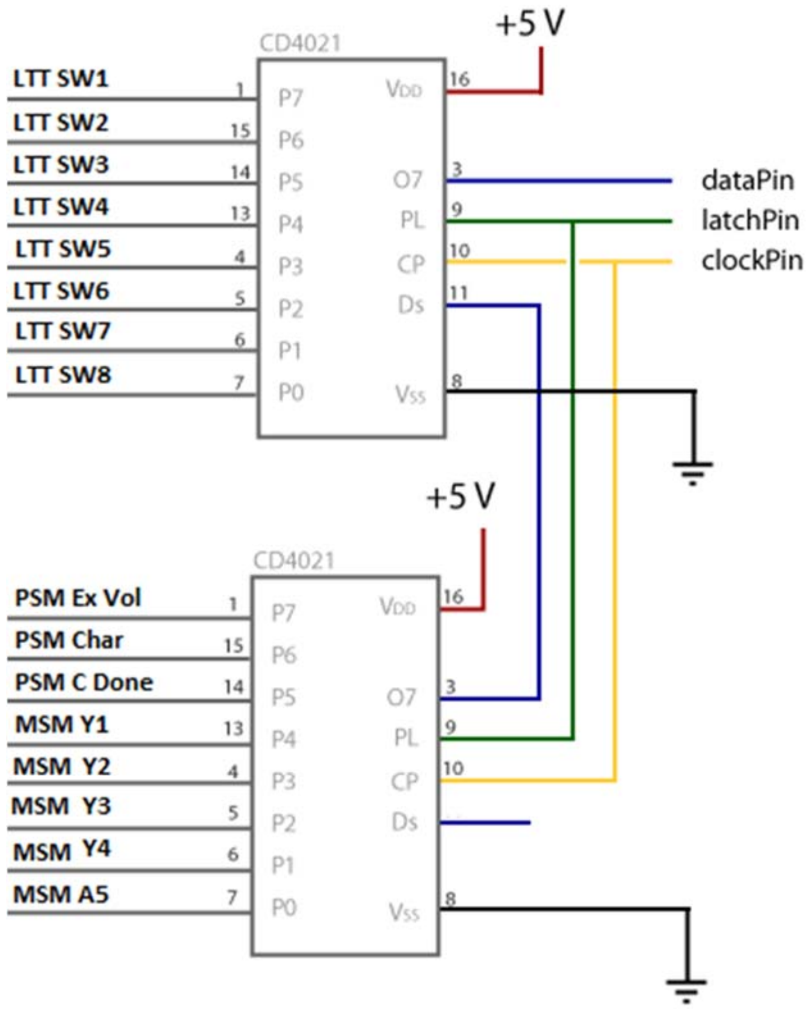


Figure 3.4.1 RX of NSA.

TX NSA contains three ShiftOUT register 74HC595. Each of which integrated into the Communication switch matrix, Motor Shield, and Power Supply module. Integrated elements of RX NSA presented at Figure 3.4.2.

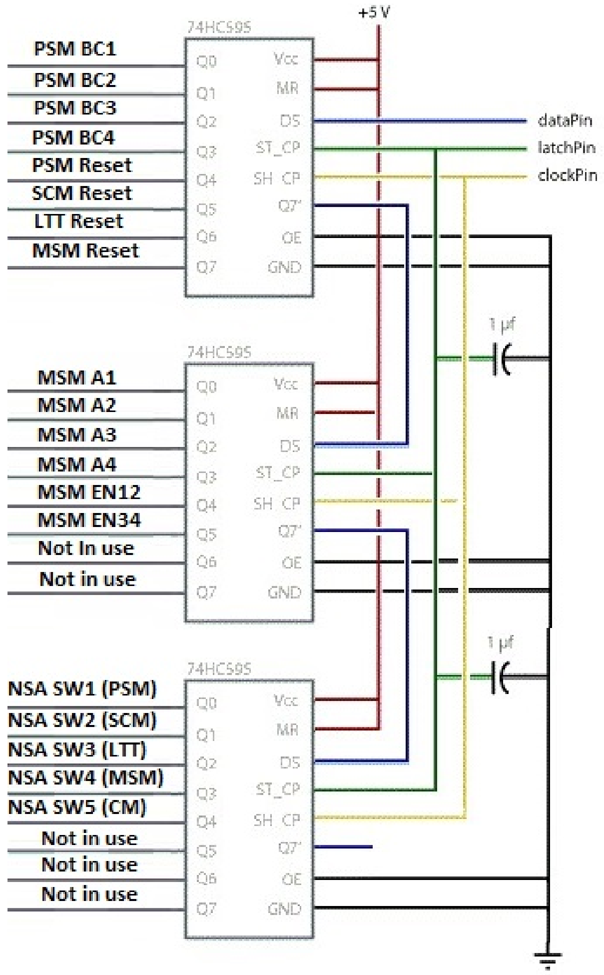


Figure 3.4.2 RX of NSA.

This way the NSA can carry of SW module, the charging process without MPU in PSM and also steering of chassis without MPU of MS. In a normal mode of operating ShiftOut registers is under control by Radar module and are using only for connection change between modules i.e. active shift register is only in SW. Others values of TX NSA which are not using in normal mode fo operation equals to zero. Thus, the interference on working process of other modules will not be.

3.5 Overall Structure

General connection diagram showed on Figure 3.5.1. By blue color is outlined the main channel of communication between modules. Each TX and RX ports connected in the switch matrix. Thus, a way has implemented the ability to change the way of data traveling.

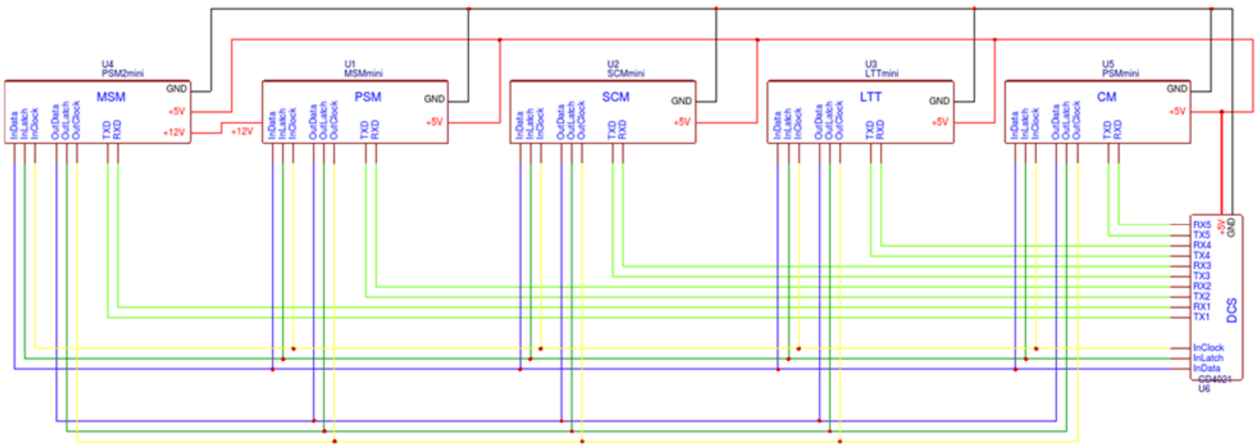


Figure 3.5.1 General connection diagram.

The NSA network is outlined by orange color, according to a setup of this paper the access to NSA is available for each module. Theoretically, the output of NSA system may be optimized. At this moment, the system uses six port of each MPU but two ports of each line, the latch and clock pins may bay be combined and as a results NSA output will count four ports, and only data outputs will be individual. However, this decision makes it impossible to use lines of NSA separately. In normal mode op operation, the ShiftIn line is employed by TTL Modula and ShiftOut line is used by Movement module. And compile of lines will decrease the flexibility of the device.

In the own final form of hardware, it vaguely reminiscent of the neural network. Entire block diagram p presented at Figure 3.5.3 Final hardware block diagram. Exactly by this is implemented the concept of robust system on hardware lever. Since each command or data from the sensor can reach the host in multiple ways. Enev if by some reason the functionality of any segment is no longer available the whole system designed so that the lost one segment functionality can replaced by using others sensors in another sense.

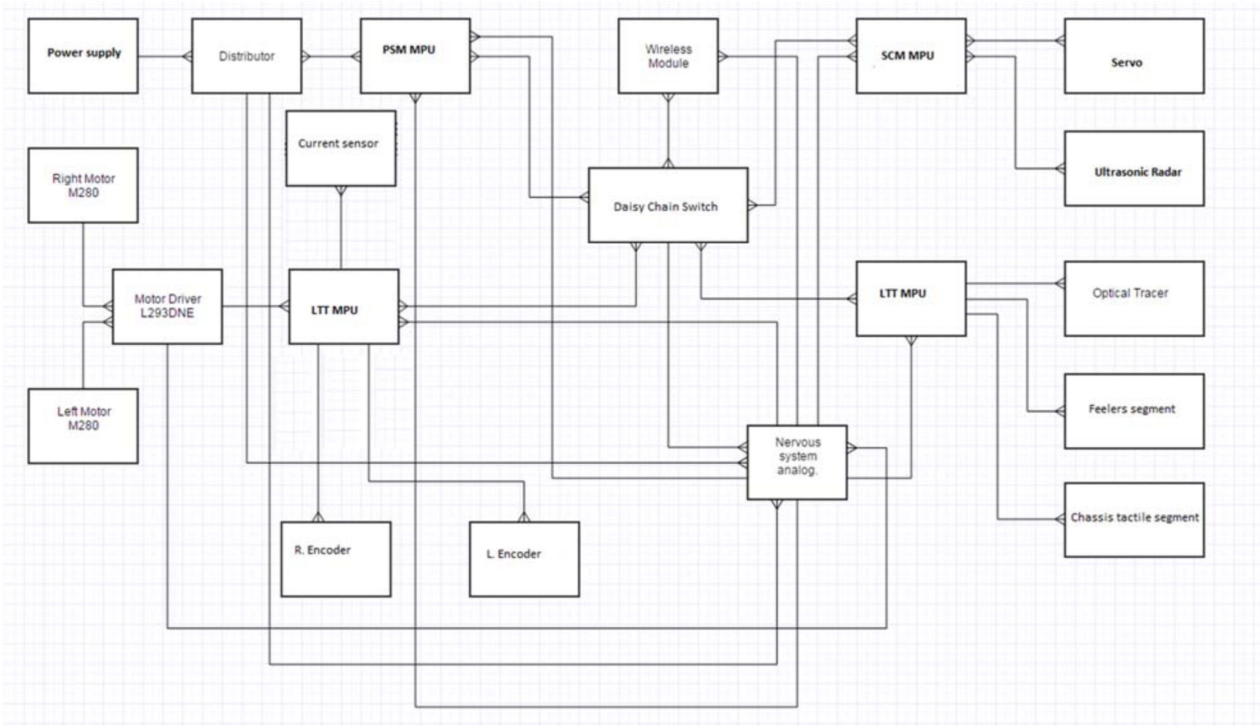


Figure 3.5.2 Final hardware block diagram.

4 SOFTWARE LEVEL

In the following chapter, each module of the project reviewed as a separate segment. Implementation the different functions for each module and tasks is subject to this chapter to identify the better method for each task performing to reach objectives of this paper.

The prototype the model with the necessary characteristics, segment and sensors placement firstly need to be designed in a virtual environment. The main program on this project for the prototyping is SOLIDWORKS Education Edition 2014-2015.

To develop the concepts of various shields used a basic circuit scheme and modified it until to the required functionality. Construction of circuit diagrams for modules performed in Web and Cloud-based EDA tool suite EasyEda. EasyEDA is a free application with integrating powerful schematic capture, mixed-mode circuit simulator and PCB layout in a seamless cross-platform browser environment, for electronic engineers, educators, students, and hobbyists.

Beside of the self-designed and assembled modules also use ArduinoBoards and also the assemblies with various sensors and functions as Xbee module, ultrasonic radar, servo motor, etc.

For the creation and implementation of the operating algorithms uses several different programs, but the main environment is Arduino IDE Software. The open-source Arduino Software makes it easy to write code and upload it to the board. The environment is written in Java and based on Processing and other open-source software.

4.1 Module overall code structure

The each Module can consider as a standalone segment of mobile robot platform. Except for rare exceptions the modules interact with each others by messages. Nevertheless, each of them operates on the general principle designed in this work and has a similar base in algorithms action. General view of the code presented at Figure 4.1.1 Template code of MPU.

```
// Include the required libraries
#include "math.h"
//----------------------------
// Binding of physical ports to virtual values
int INlatchPin = 8;  // NSA Shift in line
int INdataPin = 9;
int INclockPin = 10;
int OUTlatchPin = 11; //NSA Shift out line
int OUTdataPin = 12;
int OUTclockPin = 13;
//----------------------------
// Set of massives
boolean SHINdata[24]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
boolean SHOTdata[24]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
//----------------------------
void setup() {
// Set of pin modes
Serial.begin(9600);

pinMode(INlatchPin, OUTPUT);
pinMode(INclockPin, OUTPUT);
pinMode(INdataPin, INPUT);
pinMode(OUTlatchPin, OUTPUT);
pinMode(OUTclockPin, OUTPUT);
pinMode(OUTdataPin, INPUT);

}
//----------------------------***
void MainProcess(){      }
//----------------------------***
int ReadData(){        }
//----------------------------***
int CleanData(){       }
//----------------------------***
int UpdateData(){      }
//----------------------------***
// Loop of module program
void loop() {
while(Serial.available()==0) {
  MainProcess();}           // *** - short name of module.
while(Serial.available()!= 0){
  ReadData();}
            }
```

Figure 4.1.1 Template code of MPU.

The loop of code organized in such way that if serial available equals to zero, this means that no one does not transmit the data then MPU performing prescribed task. When activity on RX port detected the whole process aimed to read and interpretive message. More details about reading process located in chapters 4.2 and 4.3.

In addition to the loop, the template contains a number of additional functions. Their number depends on the module functional and his load. About ReadData function already was mentioned above. Also, each MPU program will include such function as CleanData and UpdateData. Clean data is intended to clean the intermediate data after the calculation period. Since all values are global and each function performs its conversion after calculation session the data must reset. Performed by the previously mentioned function.

As general function for the modules also are SystemCheck and CoreControl. SystemCheck is intended for testing of mechanical and hardware level of the device. The method of this function lines in few repeating of same actions, and if the output of the sensor is stable and same, it means that the segment was working properly. The CoreControl function contains simplified algorithms for each element and works only with NSA lines.

During the evolution process of the first prototype is possible that some of the basic functions may be changed or separated for example, for partial purification of intermediate data

4.2 Data format

During chatting session is quite important to exclude damaged data pack on incorrect input. To achieve this aim, each of the modules has a reading algorithm. ReadData function can determinate the destination point of data pack, and if the message is intended for a module, the function will extract necessary data and execute pack from data "train". In the case of the address mismatch, the message will transferred to the next module.

Could say that debris or damaged data pack is also not consistent with the expected type and retranslating such message will spam data link. This is undesirable for a network of apparatus. To ensure the purity of the channel was developed a particular message format. If the message does not match the format, then the damaged part is removed from the data line.

Each module has a short name and this name serve as module address. For example, the short name of Locomotion Tracking and Tactile Module is LTT. If a host needs to check system condition of LTT, it will send a message to serial as <<LTT<CMD:SCK>>>. This message interpreted as "System check of Tracking and Tactile Module". Also, there are untargeted messages. Their presence justified by standard format and the final destination of it is a host of the device. There are a three types of untargeted data:

1. <<OBS<ANG:value><Disvalue>>>
2. <<PTT<ANG:value><DST:value>>>
3. <<TRV<ANG:value><DST:value>>>

The destination point of these three type is also the host of the device. First, two of them are born by Scanner Module. OBS data can also origin in TTL by activation of feelers. Nevertheless, they all going to host for virtual map building (future work). The DST data format is the decision of Scanner module about traveling direction. Passing through Movement Module, it leaves there a task to perform – reach the destination point. Then forwarded through the network to the host. The last type was starting from LTT and report to host about the done path. Thus, the full list of messages can be written as:

First group: <<(1)<(2):(3)>>>

1. Address : SCM / PSM / MSM / LTT
2. Action: CMD (Command) / RSP (Response)
3. Status: SCK (System Check) / Call correct (COK) / System correct (SOK)

Second group:

<<(*) <ANG:value><DIST:value>>> .

* - OBS (Obstacle) / DST (Destination) / TRV (Done path)

4.3 Chatting between modules.

Described types of messages in the previous chapter formed a kind of tree branching variations of a possible incoming message. To read and follow the classification of messages and withdrawal of the necessary data from incoming data flow was developed the unique algorithm and presented in paper attachment as Chatting algorithm.

Given the specifications of the modules, namely, the expectation of commands and origin of messages the Daisy Chain of MPU organized in particular order to eliminate the necessary of double circling of data package trough device internal network line.

Least of all in the interaction between modules is also busy the PSM. According to the reason mentioned above, the place of this module in the chain does not matter. Nevertheless, he takes the first place in the chain. Such a decision based on the fact that in a case of contact loss with the host and switch to offline mode will close the chain in the inside of the module and all traffic will go to the second round. The network will fill by the unwanted load. To eliminate this need to equip each data analyzer module or to install the filter in the first module. However, not excluded the failure of the PSM and together with him the filter. Therefore, it is necessary to provide the system by offline mode messages. However, to preserve the possibility for both updates, the PSM will keep his place.

SCM module provides most of the messages, five of six message in pack intended to the host. As a result, it will be logical to place this module to the last place in the chain. Nevertheless, the sixth message contains data of next point of destination. Hence, it must be delivery directly to the MSM to increase performance. However, given that the modules operate in two modes. Data flow and work process. To not cause delays in the process decided to start processing with obstacle data, the first five messages, then sending destination data. This will help to synchronize the overall process.

Next place in the chain takes the MSM because of the destination message. It must travel directly to the recipient. Also, the last module (LTT) is designed for the movement analysis. After the fact of shifting. Hence, the last place is occupied by LTT.

With this modules order in the chain, the total chatting algorithm does not create the need to submit results to the second round. This will remove the channel of communication from unnecessary load.

4.4 Module Program

This chapter describes the main processes in the each module. Subsequently of which is combined the core algorithm.The description of the module functions on this section I quite limited. Since the full version of module codes has a lot of same segments and some of them has been a little bit modified individually for each module. A more complete version of the codes stored in attachment to this paper.

4.4.1 Power Supply

While performing exploration by mobile platform, the program only keeps tracking of energy consumption by the analog port. If more than 80% of the energy has spent then the program informs the HOST that it is time to return. It believed that 20% would be enough to return to the center by the short path. Implement separate charging of each battery occurs by comparison values of Voltage and Current values with prescribed charging process. When Voltage equals to 4.2 Voltage and Current value while charging equals to zero. The algorithm switches to another cell. The diagram of charging process presented at Figure 4.4.1.1. Charging process diagram.

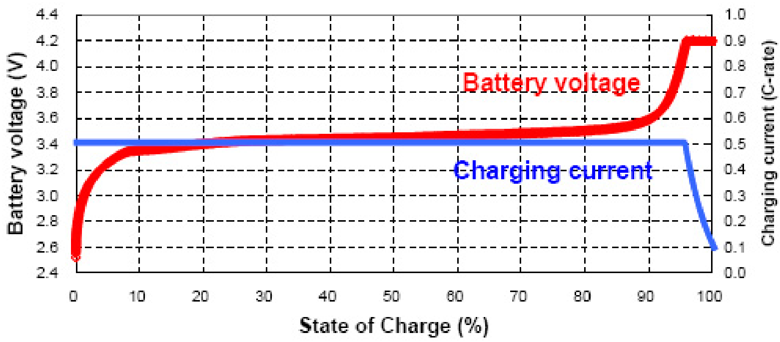


Figure 4.4.1.1 Charging process diagram.

According to values of Figure 4.4.1.1 can define the stage of charging. Nevertheless, the A.I. of the device have interest only about results. At this moment, the switch change battery only in case if current and voltage reach necessary value. However, the definition of the stage can be applied in next upgrade of the prototype.

Since MPU operate in two modes, chatting or work it is highly undesirable to use the statements with the delay or an internal loop as while(). Because the algorithm will not able to finish subloop until the end of the inner loop appears and the program return to the main loop to check incoming messages. Thus charging function should organize without internal loops. This problem solved by code structure presented at Figure 4.4.1.2 Charging Function.

```
int Charging(){
 if( CharPackStt[0]==0 || CharPackStt[1]==0 || CharPackStt[2]==0 || CharPackStt[3]==0){
 if( CharPackPin[0]==0 && CharPackPin[1]==0 && CharPackPin[2]==0 && CharPackPin[3]==0){ //Check For cell activation
  for (int i=0; i <= 3; i++){if (CharPackStt[i]==0){j=i;}}                          // Battery status check
  CharPackPin[j]=1; CharPackStt[j]=1; SetOutputs();}                               // Set to uncharged unit
/*delay(200);*/ CheckInputs();                                                     // Delay for relay activation
  if( vall<=500 && CharFull == 1 && CharProc == 1){CharPackPin[j]=0;SetOutputs();}  // Turn back cell if charging complete
/*delay(200);*/           }else{Serial.println("<<PSM<RSP:CPD>>>");CharStatus == 0;} // CPD - Charging procces done
}                                                      // Delay for relay activation
```

Figure 4.4.1.2 Charging Function

The first step of the algorithm is check activation one of the relays. If one is active, it means that charging already going and then do nothing. If all of them inactive then test the status of battery cells. When the uncharged element is defined it connect to charge module and one of battery pin, become active, and the first part of the code is always skipping.

Until values reach a necessary point, then battery pin go low, but battery status stay equals to 1. During next status check, the battery with status one will not count. Until all battery status equals to one.

Self-diagnosis function organized by switching one by one of battery cells and compare values with the template. Code presented at figure 4.4.1.3 Self-test function.

```
int SelfCheckFunction(){
   for (int i=0; i <= 3; i++){CharPackPin[i]=0;} SetOutputs();//null inputs before check
   digitalWrite(CharPack1Pin,HIGH); val5 = analogRead(BatteryVal);delay(200);
   if (val5 >= 600 ){SelfCheck[0]=1;}else{SelfCheck[0]=0;} digitalWrite(CharPack1Pin,LOW);delay(200);
   digitalWrite(CharPack2Pin,HIGH); val5 = analogRead(BatteryVal);delay(200);
   if (val5 >= 600 ){SelfCheck[1]=1;}else{SelfCheck[1]=0;} digitalWrite(CharPack1Pin,LOW);delay(200);
   digitalWrite(CharPack3Pin,HIGH); val5 = analogRead(BatteryVal);delay(200);
   if (val5 >= 600 ){SelfCheck[2]=1;}else{SelfCheck[2]=0;} digitalWrite(CharPack1Pin,LOW);delay(200);
   digitalWrite(CharPack4Pin,HIGH); val5 = analogRead(BatteryVal);delay(200);
   if (val5 >= 600 ){SelfCheck[3]=1;}else{SelfCheck[3]=0;} digitalWrite(CharPack1Pin,LOW);delay(200);
   for (int i=0; i <= 3; i++){if (SelfCheck[i]==1){SelfCheckStatus = SelfCheckStatus + 1;}}
   if (SelfCheckStatus == 4){Serial.println("<<PSM<RSP:SOK>>>");}else{Serial.println("<<PSM<RSP:SNK>>>");}
   SelfCheckStatus=0;           }
```

Figure 4.4.1.3 Self-Test Function

In contrast to the charging function, the self-test function has fewer requirements. It allows to use delay hence system will wait for the response. So SelfTest can take so much time as it needs. The logic of this function was also simpler. Switching the cells one by one the MPU checks the voltage at the input of the battery charger element. If the value is within the specified framework, it means that changer-switch changes it position and battery create the voltage, detected by the A4 analog port of MPU.

The minimal variation of PSM codes based on evidence of three states. Firs are detecting voltage on charging network. Second and third is state of charging module indicators, the charging, and full charge.

4.4.2 Scanner

Scanner construction is made of two elements, whose tasks are distance and angle measurement. As distance reader act the HC-SR04 ultrasonic sensor. The angle of radar depends on the angle of rotation of the servo. Thus, the angle is set by PWM signal to servo and distance is measured by counting pulses.

The primary process of SCM is scan environment once in a work cycle and send results to device network. By simple algorithm presented at Figure 4.4.2.1 Scanning algorithm ultrasonic radar measure distance in five directions.

```
//-----------------------------***
int Scan() {
  for (j=0; j <= 4; j++){
  delay(150);
      RadServo.write(SG90Angl[j]);
      delay(250);
      dstn[j]=Radarecho()/58.2;}
RadServo.write(SG90Angl[2]);}
//-----------------------------***
float Radarecho() {
 float value=0;
 digitalWrite(trigPin, LOW);
 delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin, LOW);
 value = pulseIn(echoPin, HIGH);
 return value;}
 //-----------------------------***
```

Figure 4.4.2.1 Scanning algorithm..

Obtained value saved in massive dstn[j]. However, the reading values occurred by radar that has offset about the center of the device. Consequently, the obtained data is also necessary to convert about the center of apparatus. This task performed by the Calculation() function presented in Figure 4.4.2.3.

To calculate transferring data are using 4.4.2.1 and 4.4.2.2 formulas. Visualization of calculation shown in figure 4.4.2.2

$$Dist_2 = \sqrt{((\cos\alpha * dist_1)^2 + ((\sin\alpha * dist_1) + CS)^2)} \quad \text{Formula 4.4.2.1}$$

$\beta = \cos^{-1}(\frac{dist_1 * \cos\alpha}{dist_2})$ Formula 4.4.2.2

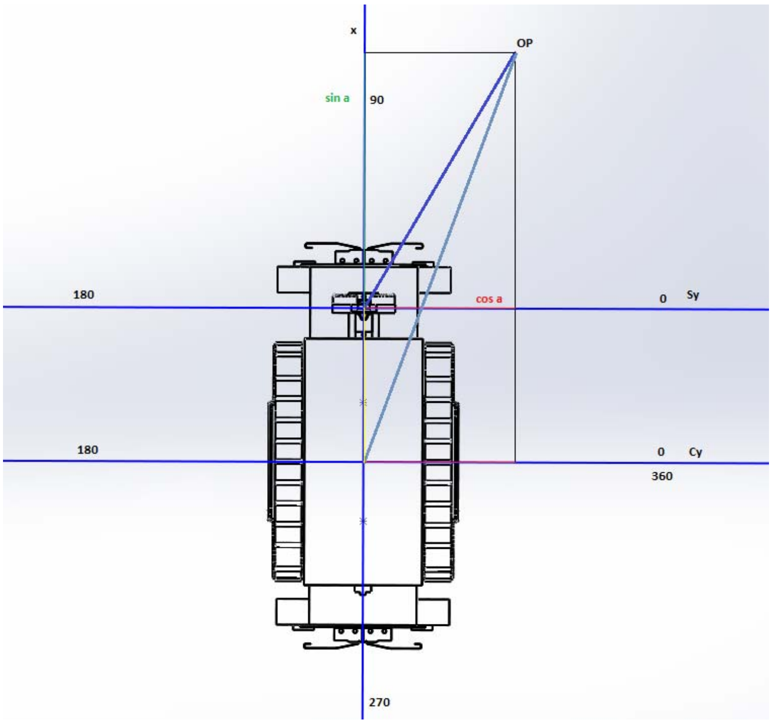


Figure 4.4.2.2

```
//------------------------------***
int Calculation() {
for (j=0; j <= 4; j++){
temp1=(dstn[j]*cos(RLAngl[j]/rad))*(dstn[j]*cos(RLAngl[j]/rad));
temp2=((dstn[j]*sin(RLAngl[j]/rad))+cntrshft)*((dstn[j]*sin(RLAngl[j]/rad))+cntrshft);
TransfDstn[j]=sqrt(temp1+temp2);
temp3=((dstn[j]*cos(RLAngl[j]/rad))/TransfDstn[j]); TransfAngl[j]= acos(temp3)*rad;
CleanData();     }
  for (j=0; j <= 4; j++){
    if ( TransfDstn[j] >= PTTdst ) {
      PTTdst=TransfDstn[j];
      PTTang=TransfAngl[j];         }
                              }
     PTTdst=PTTdst*(0.75); }
//------------------------------***
```

Figure 4.4.2.3. Calculation of transfering data

Self-test is going almost in a same way as the primary process but without coordinate conversation. The process repeated several times, and then compare the data. If the results are identical, hence the segment is serviceable and stable. Algorithm presented at Figure 4.4.2.3 SelfTest of SCM.

```
//-----------------------------***
int SystemCheck(){
Scan();for (j=0; j <= 4; j++){SelfCheckData[j]=SelfCheckData[j]+dstn[j];}
Scan();for (j=0; j <= 4; j++){SelfCheckData[j]=SelfCheckData[j]+dstn[j];}
Scan();for (j=0; j <= 4; j++){SelfCheckData[j]=SelfCheckData[j]+dstn[j];}
for (j=0; j <= 4; j++){
  SelfCheckData[j]=(SelfCheckData[j]/3);
  SelfCheckDataNEG[j]=SelfCheckData[j]-(SelfCheckData[j]/10);
  SelfCheckDataPOS[j]=SelfCheckData[j]+(SelfCheckData[j]/10);}
Scan();for (j=0; j <= 4; j++){
if (SelfCheckDataNEG[j] <= dstn[j] && SelfCheckDataPOS[j] >= dstn[j] ){ c=c+1; }
  }
if(c==4){ Serial.print("<<SCM<RSP:SOK>>>");}else{ Serial.print("<<SCM<RSP:SNK>>>");}
}
//-----------------------------***
```

Figure 4.4.2.3 SelfTest of SCM

Radar measure destination if five directions three times. After what calculating the average measurement values and make four comparations. If the readings are identical, taking into account the 10% deviation then test passed, and it can argue that the system is ok.

Unlike other modules, the SCM do not have a minimal version of the algorithm. Incarnation through the NSA is possible but excessively load the core program and on the NSA. It is not necessary for such a fundamental decision due to presence the ability to replace the functional of the scanner by feelers. Weak choice but it works, in the case of emergency.

4.4.3 Locomotion Tracking & Tactile

The main task of LTT module is track shifting of device trough two bottom optical sensors and send obstacle coordinates in the case of activation one of the fellers. The main loop of entire algorithm organized in the same way as others and presented at Figure 4.4.3.1 LTT Loop.

```
//------------------------------***
void loop(){
while(Serial.available()==0) { ReadCord();checkIn();FeelerOBS();}
while(Serial.available() > 0){ ReadData();}
}
//------------------------------***
```

Figure 4.4.3.1 LTT Loop.

A chatting period occurs while the device is standing. At this time no need to track shifting. This allows separate the processes between two positions without any problems. The checking the status of the feelers has left in first “while“ of serial.Available logic but was divided from ReadCord because of the importance of the activity in this segment and it requires an instant message to MSM. Algorithm of feelers detection presented on Figure 4.4.3.2 FeelersOSB.

```
void FeelerOBS(){
  for (i=0; i<=7; i++){
      if(data[i]==1&&FLRdata[i]==0){
      Serial.print("<<OBS<ANG:"); Serial.print(FLRANG[i]);
      Serial.print("><DST:"); Serial.print(FLRDST[i]);
      Serial.print(">>>"); FLRdata[i]=1;
       Serial.print("<<MSM<CMD:STP>>>");
      }else if(data[i]==0&&FLRdata[i]==1){FLRdata[i]=0;}   }}
//------------------------------***
```

Figure 4.4.3.2 FeelersOBS

To realize the desired result - a one-time message sending, applied an additional massive. When the switch changes own state the data[i] become equals to 1, and they LTT send the message about detected obstacle and also send a command to MSM to stop moving. To exclude the re-message sending the program also set the FLRdata[i] to equals 1. When program re-checking state and see that the message has been already sent. Then it waits of the switch deactivation. Program remove FLRdata[i] only in case if feelers deactivated.

The segment of code for the calculation of the patch represented in Figure 4.4.3.3 ReadCord. To carry out this process using three libraries. The ADNS2610, to work with the optic sensor with the same name. The math.h, to perform calculations and MsTimer2. The Timer used to avoid data sending in case of short pause in device movement.

```
#include "ADNS2610.h"
#include "math.h"
#include <MsTimer2.h>
ADNS2610 FOptical1 = ADNS2610(FSCLK, FSDIO);
ADNS2610 ROptical1 = ADNS2610(RSCLK, RSDIO);

//------------------------------***
void Activation(){
Calculation();
Report();
ClearCord();
MsTimer2::stop();}
//------------------------------***
void ReadCord(){
   PFy = FOptical1.dx(); Fy -= PFy; if (PFy != 0) {MsTimer2::start();}
   PFx = FOptical1.dy(); Fx -= PFx; if (PFx != 0) {MsTimer2::start();}
   PRy = ROptical1.dx(); Ry += PRy; if (PRy != 0) {MsTimer2::start();}
   PRx = ROptical1.dy(); Rx += PRx; if (PRx != 0) {MsTimer2::start();}   }
//------------------------------***
void Calculation(){
NVOX=(Rx*0.85)-Fx;
NVOY=(Ry*0.85)-Fy;
fi=(((Fy-Ry*0.85)*180)/(2*pi*r))+90;
dstn=sqrt(pow(((Fx+Rx)/2),2)+pow(((Fy+Ry)/2),2)); }
//------------------------------***
```

Figure 4.4.3.3 ReadCord

After movement starting the function ReadCord() accumulates a total coordinates change of the front and rear sensor points and also regularly starts the timer when the value of P(F/R)(x/y) is not equal to zero. Once the robot has stopped and waiting for the next command, the restart of the timer has been stopped. When the value of timer reaches necessary value, it triggers the Activation function. Then comes the calculation of the angle of rotation through the length of the circle sector and distance by the calculating of the distance between the start and end center point of apparatus in the workspace.

When all calculation is done the program send angle and distance values in unique data format described in Chapter 4.2 Data format. After this acting, the ClearCord function cleaning all intermediate calculations and the accumulated coordinates change.

Self-test of TLL module occurs by checking last projection of device movement vector on the x-axis (NVOX). If NVOX value equals with prescribed distancy between the front and rear points of optic sensors, then system check is passed. Another way it will fail ant that means that one of the optical sensors do not send data of the device is not standing right on the work surface. The second state can check by feelers. In this case, the data from TLL will not count in overall interaction.

4.4.4 Motor Shield

Depending on the mode of movement management is carried out by two types of algorithms. In the first mode, the path is set by the host. In the case of the absence of data about the path from the host the machine moves within sight of the radar, this is the second mode. Transferring data of angle and distance presented at Figure 4.4.4.1 Transfering.

```
//------------------------------***
int TRVCalculation() {
  MSMAng = AngDstTemp[0] - 90;
  MSMDst = AngDstTemp[1] / onepulse;
  ANGpulses = abs(MSMAng / onedegree) * AngCallib; // for turn
  DSTpulses = abs(MSMDst) * DstCallib;
  Step1 = 1;
  AngDstTemp[0] = 0; AngDstTemp[1] = 0;}
//------------------------------***
```

Figure 4.4.4.1 Transfering.

By using a simple mathematic, the program calculates the necessary amount of movement according to the pulse count from encoders to perform rotation and direct movement of the platform. In this case data from encoders serves as motion register. The primary process of The MSM presented at Figure 4.4.4.2 Movement performing.

The motion of device includes two steps. First, one is turning and the second is direct traveling. Both of them operate on same feedback and use one driver. Also, it is necessary to apply the same principle as was used on the PSM to perform movement without inner loops. For this purpose added a few int in the code to separate steps.

Firstly will complete the rotation. To activate this step the Step1 must be active and Step2 with AngCal inactive. Then algorithm decides in wich side need to turn and set motor driver in the necessary state. After what int AngCal set active and on the next lap this step will be skipped.

When the first step is done, the states of itns set for performing the rotation. At this moment on each next lap will be active only this segment of code while encoders are accumulating necessary value. Which one was precalculated and marked as ANGpulses. Then AngPer will equal to 1. This mean that performing done and now allowed to perform the direct motion.

Direct motion performance organized in the same way as rotation algorithm. The differents only in set for motor driver and as comparison value now used DSTpulses.

When all steps of motion complete the MovementPerform function clear all temporary values and become ready to next command.

```
void MovementPerform() {
  SetMDpins();

        if (AngCal == 0 && Step1 == 1 && Step2 == 0) {
        if (MSMAng > 0) {LeftMotor(1);RightMotor(-1);}
    else if (MSMAng < 0) {LeftMotor(-1);RightMotor(1);}
    else {Step1 = 0;Step2 = 1;} AngCal = 1;}

  SetMDpins();

           if (DstCal == 0 && Step1 == 0 && Step2 == 1) {
           if (MSMDst > 0) {LeftMotor(1);RightMotor(1);}
       else if (MSMDst < 0) { LeftMotor(-1);RightMotor(-1);}
       else {LeftMotor(0);RightMotor(0);} DstCal = 1; }

  SetMDpins();

  if (Step1 == 1 && Step2 == 0 && AngCal == 1 && AngPer == 0) {
    if (LEncodPulse > ANGpulses) {LeftMotor(0);SetMDpins();}
    if (REncodPulse > ANGpulses) {RightMotor(0);SetMDpins();}
    if (LEncodPulse > ANGpulses && REncodPulse > ANGpulses) {
      RightMotor(0); LeftMotor(0); SetMDpins();
      Step1 = 0; Step2 = 1; AngPer = 1;}}

  if (Step1 == 0 && Step2 == 1 && DstCal == 1 && DstPer == 0) {
    if (LEncodPulse > DSTpulses) { LeftMotor(0);SetMDpins();}
    if (REncodPulse > DSTpulses) { RightMotor(0);SetMDpins();}
    if (REncodPulse > DSTpulses && LEncodPulse > DSTpulses) {
      Step1 = 0; Step2 = 0; DstPer = 1;}}

  if (Step1 == 0 && Step2 == 0 && AngCal == 1 && AngPer == 1 && DstCal == 1 && DstPer == 1) {
    RightMotor(0); LeftMotor(0);
    SetMDpins();
     Serial.print("<<MSM<RSP:DSP>>>");
    REncodPulse = 0;LEncodPulse = 0;
    AngCal = 0; AngPer = 0;
    DstCal = 0; DstPer = 0;
    Step1 = 0; Step2 = 0;
    MSMAng = 0;MSMDst = 0;
    ANGpulses = 0;DSTpulses = 0;}
}
```

Figure 4.4.4.2 Movement prerforming

Like others modules, the MSM also have call module and self-test function. The call module function in absolutely same. If module read data function work properly then it can give the answer to others modules. The self-test function is based on the same principle as the test of others modules and presented on Figure 4.4.4.3 MSM Self-test.

```
//-----------------------------***
int SystemCheck() {
  int TLMCV; int TRMCV;
   EmergencyStop();
   RightMotor(1); LeftMotor(1);
   SetMDpins();
   delay(500);
   TLMCV=analogRead(LMCV);
   TRMCV=analogRead(RMCV);
   delay(200);
   RightMotor(0); LeftMotor(0);
   SetMDpins();
 if(REncodPulse==LEncodPulse && TLMCV > 450 && TLMCV < 550  && TLMCV > 450 && TLMCV < 550){
   Serial.print("<<SCM<RSP:SOK>>>");
 }else{Serial.print("<<SCM<RSP:SNK>>>");}
   EmergencyStop();}
//-----------------------------***
```

Figure 4.4.4.3 MSM Self-test

The workability of MSM is testing by direct motion with given period (800 ms). If after the end of motion the pulses count of the left and right encoders are identical this indicates a stable operability of the system. Such a simple test is enough to check the whole module system since the input set at the one end of the segment, and the output is on the other end. Also, can be added the test of the traveled distance and then compare it with the sample. However, this is not required since any movement of the device will be registered by TLL module.

After receiving the data of the final destination point, the machine will not stop until it reaches mounted point. But while motion performed there is also possible to collide with an unregistered obstacle or falling from the working surface. Of course, the feelers (TTL) can detect those types of barriers and send a warning to the MSM. But only thanks to absent of inner loops the Motor supply module can react to this message and stop motion performing.

4.5 Floating Core

At the start of this project, the Floating core was designed as one solid code which one is activating in the case of long silent in the device chat. Then the device will continue working in an emergency only on one active MPU. However, such a solution is too radical and eliminates the possibility of a full-fledged operating in case of minor malfunction. To create a more flexible system the core has divided into segments, each of which can replace the specific module program. Also, has been added functionality that will try to resume normal mode of operation by rebooting the unresponsive module. And only if the restart does not help the Core is activate one of the segments to replace failed MPU functionality. At least a part of full functional. The full version of The Core presented at attachment to this paper.

The main tool of the core to interact with whole device low level electronic is the NSA. For reading the values The Core is using ReadPins() function which based on shiftIn function and for pin controls is using SetPins() function which based on shiftOut function. ShiftIn and shiftOut function taken from Arduino web page in the learning tutorials.

Core implementation in each MPU loop() looks identical and presented at Figure 4.5.1 Core implementation.

```
MsTimer2::set(300000, CoreActivation);
}
//------------------------------***
void loop() {
  while (Serial.available() == 0) {
    MovementPerform(); Core();   }       // MSM example
  while (Serial.available() > 0) {
    ReadData(); MsTimer2::Start();}
}
//------------------------------***
```

Figure 4.5.1 Core implementation.

If in device chat already long time persist the silent it could mean two variations. The first one says that apparatus is standing in sleep mode and charging an battery. But this situation can be detected by voltage value on the external voltage power supply port. In this case Core activation() function will be triggered, but the activity of the functions will be limited only by modules rollcall.

The Second possible case is that the one of module stops retranslation and. In the second case, the Core algorithm do not have any limits. The Activation process of the core is presented at Figure 4.5.2 Core activation.

```
//-----------------------------***
void CoreActivation(){
  if(ActStart==0){
  for (int j=0; j<=3; j++){
  ModulesState[j]=0;}
   ActStart=0;}

  if(ReCall==0){ModulesReCall();ReadPins();}
  if(ReCall==1){ChekcModulesStatus();}
  if(ModStat==0&&SHRGIN[8]==0){ModuleRestart();} // in case if ext vol absent

  if(ModulesState[0]==0&&RestartedState[0]==1){
    if(ModuleDiactivation[0]==0) {
      SHRGOUT[17]=1;SetPins();                    // 18 Pin PSM
      ModuleDiactivation[0]=1;
      CoreFragment[0]=1;}
    }
  if(ModulesState[1]==0&&RestartedState[0]==1){
    if(ModuleDiactivation[0]==0) {
      SHRGOUT[18]=1;SetPins();                    // 19 Pin SCM
      ModuleDiactivation[1]=1;
      CoreFragment[1]=1;}
  }
  if(ModulesState[2]==0&&RestartedState[0]==1){
    if(ModuleDiactivation[0]==0) {
      SHRGOUT[19]=1;SetPins();                    // 20 Pin LTT
      ModuleDiactivation[2]=1;
      CoreFragment[2]=1;}
    }
  if(ModulesState[3]==0&&RestartedState[0]==1){
if(ModuleDiactivation[0]==0) {
      SHRGOUT[20]=1;SetPins();                    // 21 Pin MSM
      ModuleDiactivation[3]=1;
      CoreFragment[3]=1;}
    }
}
//-----------------------------***
```

Figure 4.5.2 Core activation

For the organization of The Core working process was applied already described before two methods. Avoiding long inner loops and step by step acting to work with only necessary segments on each new lap.

The first phase of CoreActivation() function is nullified modules state and then send the call to each module. Each next step in CoreActivation will perform when timer triggers it again. Thus, there is enough time to earn the response from all modules. The third phase is restarting of the non-responsible module if they are present.

The rest part of the code will skip until the device on the charge. Of course, the skip line can be moved to the top of the CoreActivation or to remove it completely but in this version of the prototype, it does not make sense since the device not provided by an opportunity of charging during main task performing. For example, as the primary task can be an exploration of unknown environment and as a voltage source for the charge can be solar batteries.According to results of previous actions the non-responsible module excluded from daisy chain by an activating the switch in the daisy switch matrix by the NSA. Thus, the internal messaging is re-established and the lost part of module functionality is restored by another MPU remotely. When the system analysis is done by CoreActivation() function, the program returns to the main loop and program start also performing the main task with the partial participation of Core() function as presented on Figure 4.5.1. The Core code presented at Figure 4.5.3 The Core.

```
//-----------------------------***
void Core(){
if(CoreFragment[0]==1) {      //  PSM
PSMCoreFragment();}
if(CoreFragment[1]==1) {      //  SCM
SCMCoreFragment();}
if(CoreFragment[2]==1) {      //  LTT
LTTCoreFragment();}
if(CoreFragment[3]==1) {      //  MSM
MSMCoreFragment();}
}
//-----------------------------***
```

Figure 4.5.3 The Core.

Practically the core is always active and cycled with the main module program. But it has inactive segments. Also, it exists at all modules, as it mentioned before and at the beginning of work process it is inactive in every module. Thanks to the properties of The core it was called The Floating Core by the author of this paper. Given that in the last update of core algorithm, it was divided into a few segments, and it is possible that different segment of the core is active in different modules. For example, the PSM task performing the SCM and LTT task is carried out by MSM. So actually it can also call as Core with superposition. Due to the possibility of division into different parts, the presence of each part in the different modules and together it is still the one whole process.

5 DEVICE TEST AND PRACTICAL RESULTS

5.1 The mechanical component

The selected body for this project showed a good and also bad side of this choice. As the negative results can be noted the accuracy of motion performing. The quality of reducers and track wheels is too low to provide predictable and stable movement. Thanks to the well-designed feedback of chassis it becomes able to ensure more accurate motion and the device reaching the destination point with low deviation. As the positive side of the body design, is the good interior ergonomics, a good balance between the overall weight of the machine and the traction force, the overall shape of the body and durable construction. The good interior ergonomics allows placing all designed hardware segments in the inner space without removing any existing elements. The overall shape of the body is very tolerant to adding additional attachments, i.e. it was not required to make global modifications of the body to set other structures as feeler holders, SCM bay, and PSM module.

5.2 Power Supplu Module

The total weight of PSM is over than half of one kilogram. I might do not sure how to mark this characteristic, as the good one or as the bad one. It is a necessary to ensure a sufficient amount of energy. At this moment, PSM provides almost 20 Ah. Nevertheless, big weight PSM create a positive effect. It creates additional pressure to the chassis and increases the clutch with a working surface. If we speak about the battery capacity, I can make one negative note. Even if the battery has the perfect condition, the charge cycle is very long. The charge algorithm change cells if the necessary value of charge is reached. As a result, it's hard to track of non-chargeable units. To solve this problem need to apply extended internal clock (to counting hours). It is planned to implement in the second version of the prototype. But in this version of the prototype, this problem is solved by charging the battery at a high current. As for the rest, the PSM is fully coped with the tasks assigned to him. Successfully combines a stable participation in the device chat even when the module is busy by charging process.

It can also say that the failure of the MPU in PSM is a minor malfunction. Since in the offline mode, it provides the energy to the whole device. In this version, the charging function is able only at the station. But if the machine reaches the home it will be able to receive the maintenance. It turns out that in the case of PSM failure only lose the ability to track power consumption, but it is not critical since when it detects any problems in the PSM system, the machine must return to the base. To this point, the machine decides to go back if the energy left only for the return journey or absent of any charge data.

5.3 Scanner Module

If consider the normal operation mode the notes can be made only to the module components. In the technical characteristics of the ultrasonic radar is indicated a relatively high accuracy by the manufacturer. However, in practice, the values are measured correctly, but there is serious instability. Sometimes the value have too large deviation. This can be explained by of the structural features. To earn required accuracy, it is desirable to use the average value of several measurements. Nevertheless, even with as unacceptable deviation the unit handled. In this case helped the security system (feelers) and that fact that the SCM data not affecting the already made movement data (PTT and TRV). Advantage of the created system. If the one module mada a mistake, there is second module to check state of first one. The second element of the SCM (servo) also does not comply with the specified characteristics. The actual angle does not match with an angle which is given by device software. To address this deficiency in the program introduced additional data. One data is for the servo, and the other data is for data translating into the device chat. Nevertheless this was one of the engineering problems - to create a functional system based on cheap items.

Concerning the test results of SCM as part of the robust system. In the case of the MPU failure of the SCM module, the device will permanently lose the vision ability. This situation can not designate as a success of the robust system. But the primary task of the SCM is to detect obstacles. Fellers can perform the same function. However, the vision zone will be reduced weighty. That corresponds to the objectives of this paper - recovering at least partially functional.

Can be a positively noted the self-test of SCM. The MPU of SCM can determine the health of radar and servo segments. In the event of a mechanical failure of the servo or radar segment the radar and the servo elements will still work, but measurements will be invalid. The program is designed in such a way that it can determine this type of malfunction and exclude the SCM data from the overall data processing.

5.4 Motor Shield Module

Thanks to well-designed program structure (lack of internal dead loops and step by step execution) the MSM handle with the execution of multiple processes simultaneously and accurately performs tasks. Regularly read the encoders data, controls the motor driver, and participates in the device chat. The software part operates according to the design and meets the requirements set by this project. But as has been aforementioned the execution of the motion by hardware needs additional work to be done . Namely, the precision in the movement must be increased.

5.5 Locomotion Tracking & Tactile module

This module considers two basic segments. The locomotion tracking segment and tactile segment. The locomotion tracking segment (two optical sensors) showed himself very well in action. It provides a precise data abou performed motion. One disadvantage of this segment strictly fixed distance from the working surface. If the distance is increased, the data are incorrect or missing. This type of error preserves the tactile segment of this module. In the case of loss of the functionality of this segment of the system will start to use the PTT-type of data. It will decrease the overall quality of task performing. This is permissible in the case of an emergency operating mode.

The tactile segment also works as it was intended. The disadvantage of this feelers version is physical contact with the work surface. This can create an obstacle for the movement. This problem can be solved for example, by nonphysical contact sensors such as optics which measures the distance to the work surface. In the case of failure of the MPU, the access to the tactile segment also has the NSA.

In this case, the function will be not lost. In the event of failure of the feelers segment, the functional can be partially restored by radar section of SCM module. But if it refuses the tactile layer and the fall of apparatus from the work surface cannot be avoided. Since it is assumed to return by the already traversed path, it is likely to prevent the fall. However, it is a fragile argument.

5.6 Daisy chain switch

The daisy chain switch does not have own MPU and it controls by NSA. The DCW module is designed as a supplement to increase the robust concept of the system. However, it is possible that the DCS may also break. In this case, will help the same principle as in the PSM. In disabled mode, it keeps daisy chain in original position.

5.7 Nervous system analog

Same as the DSW, the NSA is designed as a supplement to increase the robust concept of the system. The tests confirmed the possibility of remote control of each module segments by other MPU via the NSA. The only disadvantage of NSA design is that the at restart of MPU in shift registers on the TX line for a brief moment exhibited a random value on each port. As a result of this sometimes the device can make twitches or occurs arbitrary switch shifting in DSW. Since the end of the restart, the output values in TX line returning to normal positions, and it does not harm the system.

Only fear that during the described above process the false operation may occur in PSM. And as a result, the one of cell switches may be activated, and that causes the de-energization of the whole system if the delay is long enough. The apogee of this case will be a reboot of the entire system. In the other words, the device wanted to restart only one of the modules but restarted all modules. However the final result is same – wanted module is rebooted. Can this be considered as one of self-solving the unforeseen problem? Maybe. But this is questionable.

5.8 Note

In the chapter number, 5 Device test and practical results describe some theoretically possible and occurred error and explains how they solved by using another segment or the logic of device program. Rises the next question. What happend if a few malfunction will happen at one time? Maybe some combination of errors is compatible with the life of the apparatus. But the most combination is not capable of the adequate operating of the mobile platform system. It believed that the fault occured by one per each time. As a human`s diseases.

6 DISCUSSIONS

6.1 Project success

The aim of this project was the creations of the system for the mobile robot platform which could be called The Robust System. For this purpose, the author did personal research to find out what exactly means the word „robust". The most likely definition of this term I found in evolution (also called biological or genetic robustness). On the base on the definition of „ Robustness of a biological system" was developed the concept of a robust system for mobile robot platform and applied it to this project. As the starting point as the basis of this project was taken Robotic Platform RP-05 and upgraded with modification to create the ability to resist a few types of errors, reach the opportunity to continue the operating process taking into account the possible degradation of the system, and train the system to adapt to internal changes. By what criteria can be judged the success of the project?

At the stage of designing were considered the failure cases of any element in apparatus and set the path to bypass the faulty segments or entire modules. Also, the system was taught to use bypasses. Testing has shown that the concept relly works, and the goal reached. The device can operate in the case of expected malfunctions. This allows to say that the project is successful.

Moreover, the real success is that during the test period of the device was observed independent solution by device system of unforeseen situations by natural way. For example, the unplanned disconnection from the external power supply or the reboot of the whole device when the system wanted to restart only one. The reason for the latter one is in the NSA defect, but the aim of action reached. Also in the early stages of testing occurred overload of chat channel due to the lack of sorting of data retranslation in different cases but the main problem is still carried out. As it turned out during the investigation chat was filled with repetitive messages. It was self-solved thanks to the step by step method and each time when module receive the same message the main program did not react on it until the main function were complete the first command.

So with some time, the repetitive message was absorbed without any activation or damaged due limited buffer size and late response of ReadData() function in the main loop and also several previously described minor malfunctions in this paper.

Perhaps these side effects are minor and is a good reason for debugging. Nevertheless, I as the author, see those side effect as seeds for a huge potential.

6.2 Conclusion

In this paper designed robust control system for mobile robot platform to provide the continuous performance of tasks. Developed concept of the robust system is intended for exploration platforms which explore inaccessible or danger places where the loses of the device facing the loss of important data and without possibility return the unit with the side help.

The first prototype made by the set of Robot platform expansion kit for RP05/06, Arduino/Genuine Starter kit, computer mouses, the composite set of sensors and actuators, and different sets of electronics and microchips. Kits allowed beginning a testing period of the device and as well the concept idea in a shorter time. Testing period showed that there is still needs some improvements but also proved that the idea is viable and has the great possibility.

Also were examined the Shift Registers, which allows controlling a lot of port by using only three ports of MPU for each line and develop the NSA. The timer library the timer as the internal interrupt. As well was examined a numerous types of the sensor as optic and ultra ultrasonic radar. By which made several segments responsible for interaction with the environment. Likewise was grown advanced skills in code structuring to organize a step by step and avoiding of internal loops methods. That make possible to realize stable program operation based on simple statements. Also were grown skills in circuit mounting, component placement, design calculation and the creation of electrical circuits. Also increased understanding of the overall process of designing and debugging the system as a solid unit. Also studied a character-encoding scheme as the specific language of communication based on the ASCII was developed to arrange pitch error messages and designed by ASCII the general algorithm for intermodular messages.

In the future, the developed robust system concept can be applied on machines, devices, and vehicles whose primary task is to achieve the goal at any cost, such as commercial flights, or in the military industry, where highly appreciated the vitality of combat units and replacing existing systems with the same purpose.

6.3 Future Work

Regarding the future work of the developed apparatus is as follows:

- Optimize the overall intermodular message algorithm and translate data format into machine code to improve performance.

- Replace multiple rows of sensors to more advance sensors. For example, replace ultrasonic radar on a 360 RPlidar laser scanner. This will increase the viewing angle from 180 to 360 degrees, and the number of measurements per second with 1 to 2000 samples per second. That allows creating a more accurate picture of the environment. Replace tactile segment of the non-contact optical sensors. Deleting a contact of mechanical switches will increase the reliability of the system and removes the obstacles to the movement. Replace the optical segment to digital accelerometer and gyroscope elements. This will reduce the requirement for work surface and translates the measurement from 2D to 3D of virtual space.

- Debug the NSA and DCS

- Expansion the data transmission capabilities. I.e. to create the opportunity to request any int value in the test mode.

Regarding the future work of the developed concept is as follows:

- Taking into account the results of the done work. Highlight the concept as an independent method and develop a universal technology for the application.

- Apply technology to more sophisticated devices with a large number of limbs (Hex legs or mecanum wheels platform) or segmental structure of the main body (Self-reconfiguring modular robots). To develop the Robust concept (more like the evolution definition) for a mechanical level.

6.4 Summary

The main aim of this project is to design a system for mobile robot platform which will be able to continue performing prescribed tasks in case of different types of malfunction. The system could call Robust if it were able to continue with their prescribed work even if occurring faults of hardware, software or input data. Also, the error protection is important pre-set for allowing systems to achieve a failsafe operation.

In the project have been successfully solved following problems. The device has a lot of sensors and performs a high number of processes. For proper execution of all operations, all of the apparatus elements were divided into groups, each with the computing element. This helps distribute the load evenly of the overall system and perform the multi-processes. Significantly increasing the performance of the device system.

For cooperation, all MPU were united in the daisy chain. The specific language of communication-based on the ASCII developed to arrange intermodular messages. This module can share the final results of calculations, and it would greatly facilitate the final data processing by the host. It also designed in the way to understand the language by user or observer. That helps in the process of debugging the overall algorithm. To create a connection invariance between the modules has been developed the DCS. This solved the problem of communication channel interrupt when a dead module appeared.

To create an opportunity to take control over the low-level electronics of dead modules built NSA. This allows fo device to restore the lost functionality partially.

The system betrothed to the use of embedded options. As well System was taught to analyze their condition and detect severe or minor faults. Depending on the type of malfunction the machine decides to return to the base, or the continuation of the implementation of the primary objectives.

If possible, each of the modules has been designed so that when the power is turned off the main active elements take the starting position providing the restore of minimum functional in a natural way.

The test results are positive. However, not without exceptions. In addition to the foreseen problems the unit also dealt with a few of unexpected failures by natural way. More detail can found in chapter 5 DEVICE TEST AND PRACTICAL RESULTS.

The future work is optimization of the overall intermodular message algorithm and translate data format into machine code to improve performance. Also, replace multiple rows of sensors on more advanced sensors. Finalize the NSA and DCS. Expansion the data transmission capabilities. In Other words, create the opportunity to request any value in the test mode.

Regarding the future work of the developed concept is Highlight the idea as an independent method and develop a universal technology for the application. Then apply technology to more sophisticated devices with a large number of limbs or segmental structure of the main body.

6.5 Kokkuvõte

Selle projekti peamine eesmärk on liikurroboti platvormile süsteemi kujundamine, mis on võimeline ettenähtud ülesannete täitmist jätkata erinevate rikete puhul. Süsteemi võiks nimetada veakindluks, kui see oleks võimelike jätkama ettenähtud tööga isegi kui esinevad riistvara, tarkvara või andmesisestusvead. Veel on veakaitse oluline eeldus, et võimaldada süsteemidel töökindlat toimimist saavutada.

Projektis on edukalt lahendatud järgnevad probleemid: seadmel on palju andureid ning soooritab arvukaid protsesse. Kõikide toimingute korralikuks täideviimiseks on kõik seadme elemendid jaotatud gruppidesse, igas arvutitehnika element. See aitab koormust ültlaselt jaotada kogu süsteemis ning täita multi-protsesse, oluliselt suurendades seadme talitlussüsteemi.

Koostöö nimel on kõik MPU'd ühendatud juhmestikus. See konkreetne suhtluskeel põhineb ASCII-l, arendatud intermodulaatsete sõnumite korrastamiseks. Moodul saab jagada kalkulatsioonide lõpptulemusi ning see hõlbustaks suuresti lõplikku andmetöötlust peremees-süsteemi poolt. Veel on see kujundatud kasutaja või vaatleja keelt mõistma. See aitab üldise algoritmi vigade kõrvaldamise protsessi. DCS on arendatud, et luua invariantsusühendust moodulite vahel, mis lahendas ühenduskanali katkestamise probleemi „surnud“ mooduli ilmumise puhul.

NAS on ehitatud, et luua võimalus surnud moodulite madala taseme elektroonika üle kontroll võtta, võimaldades seadmel osalt kaotatud funktsionaalsus taastada.

Süsteem kasutab sisestatud võimalusi ning on veel õpetatud oma olukorda analüüsima ning raskemaid või väiksemaid vigu avastama. Sõltuvalt rikke tüübist, otsustab masin baasi naasta või jätkata peamiste eesmärkide teostamist.

Iga moodul on kujundatud viisil, et kui toide on välja lülitatud, võtavad peamised aktiivsed elemendid algpositsiooni, tagades loomulikul viisil minimaalse funktsioneerimise taastamise.

Testi tulemused on positiivsed kuid mitte ilma eranditeta. Lisaks ette nähtud probleemidele, käsitles üksus loomulikul viisil ka mõne ootamatu tõrkega. Rohkem üksikasju võib leida peatükis 5 – SEADME TEST JA PRAKTILISED TULEMUSED.

Eesootav töö – üldise intermodulaarse sõnumi algoritmi optimeerimine ning andmete tõlkimine masinkoodi, et toimimist parandada. Veel tuleks asendada mitmed read andureid rohkem arenenud anduritega. Lõpule viia NSA ja DCS. Andmeedastusvõime laiendamine. Teiste sõnadega – luua võimalus test režiimis nõuda mistahes väärtust.

Seoses edasise tööga väljaarendatud kontseptiga, tuleks esile tõsta idee kui sõltumatu meetod ning arendada universaalne tehnoloogia selle rakendamiseks. Siis rakendada tehnoloogia keerukamatele seadmetele suurema arvu jäsemete või segmentaalstruktuuriga peakorpusega.

7 REFERENCE

[1] Dr. H. Kanwar. Classical Mechanics

[2] Valery Vodovozov. Electric Drive Systems and Operation

[3] D. C. Kulshreshtha. Basic Electrical Engineering 1 Edition

[4] Dale R. Patrick. Electronic Digital System Fundamentals

[5] Michael Margolis. Make an Arduino-Controlled

[6] Raivo Sell, Mikk Leini, Rain Ellermaa .AVR mikrokontrollerid ja praktiline robootika

[7] Massimo Banzi. Getting Started with Arduino

[8] Alex Allain . Jumping into C++

[9] Steve Oualline. Practical C++ Programming

[10] Gamini Dissanayake, Stefan B. Williams, Hugh Durrant-Whyte, Tim Bailey. Autonomous Robots

[11] Juan-Antonio Fernández-Madrigal. Simultaneous Localization and Mapping for Mobile Robots: Introduction and Methods.

[12] Soren Riisgaarn, Morten Rufus Blas .SLAM For Dummies

[13] Alexander J. B. Trevor, John G. Rogers III, Henrik I. Christensen . Planar Surface SLAM with 3D and 2D Sensors.

[14] Houxiang Zhang; Wei Wang; Zhicheng Deng; Guanghua Zong & Jianwei Zhang .A Novel Reconfigurable Robot for Urban Search and Rescue.

[15] Cyrill Stachniss Wolfram Burgard. Exploring Unknown Environments with Mobile Robots using Coverage Maps.

[16] SEBASTIAN THRUN, SCOTT THAYER, WILLIAM WHITTAKER, CHRISTOPHER BAKER, WOLFRAM BURGARD, DAVID FERGUSON, DIRK HÄHNEL, MICHAEL MONTEMERLO, AARON MORRIS, ZACHARY OMOHUNDRO, CHARLIE REVERTE, AND WARREN WHITTAKER. Autonomous Exploration and Mapping of Abandoned Mines.

[17] http://www.open-electronics.org/arduino-isp-in-system-programming-and-stand-alone-circuits/ (10.01.2016)

[18] https://www.arduino.cc/en/tutorial/ShiftOut (10.05.2016)

[19] https://www.arduino.cc/en/Tutorial/ShiftIn (10.05.2016)

[20] http://pastebin.com/YpRGbzAS (10.05.2016)

[21]http://f1p.ucoz.ru/blog/intellektualnye_sensory_kompleksnye_reshenija_ardu_mysh_vysokoskorostnoj_opticheskij_sensor/2011-10-02-95-987 (12.05.2016)

[22] http://cxem.net/pitanie/5-295.php (12.05.2016)

[23] https://www.udacity.com/course/artificial-intelligence-for-robotics--cs373 (In process) (10.05.2016)

[24] http://examples.digi.com/get-started/basic-xbee-802-15-4-chat/ (13.05.2016)

[25] http://howtomechatronics.com/projects/arduino-radar-project/ (13.05.2016)

[26] https://github.com/AerospaceRobotics/RPLidar-SLAMbot/blob/master/slambotgui_source/slambotgui/comms.py (13.05.2016)

[27] https://www.arduino.cc/en/Reference/SoftwareSerial (14.05.2016)

[28] https://www.indiegogo.com/projects/ockel-sirius-b-the-powerful-windows-10-pocket-pc#/story (14.01.2016)

[29] http://www.seeedstudio.com/wiki/4WD_Mecanum_Wheel_Robot_Kit_Series#Part_List (14.01.2016)

[30] http://www.seeedstudio.com/document/pics/introduction1.jpg (15.05.2016)

[31] https://www.arduino.cc/en/Main/ArduinoBoardMicro (15.05.2016)

[32] http://pighixxx.tumblr.com/post/43005706561/arduino-micro-pinout (15.05.2016)

[32]http://hardwarefun.com/tutorials/use-arduino-as-an-isp-programmer-to-program-non-arduino-avr-microcontrollers (15.05.2016)

[33] http://www.instructables.com/id/Arduino-Wireless-Programming-with-XBee-Series-1-or/ (16.05.2016)