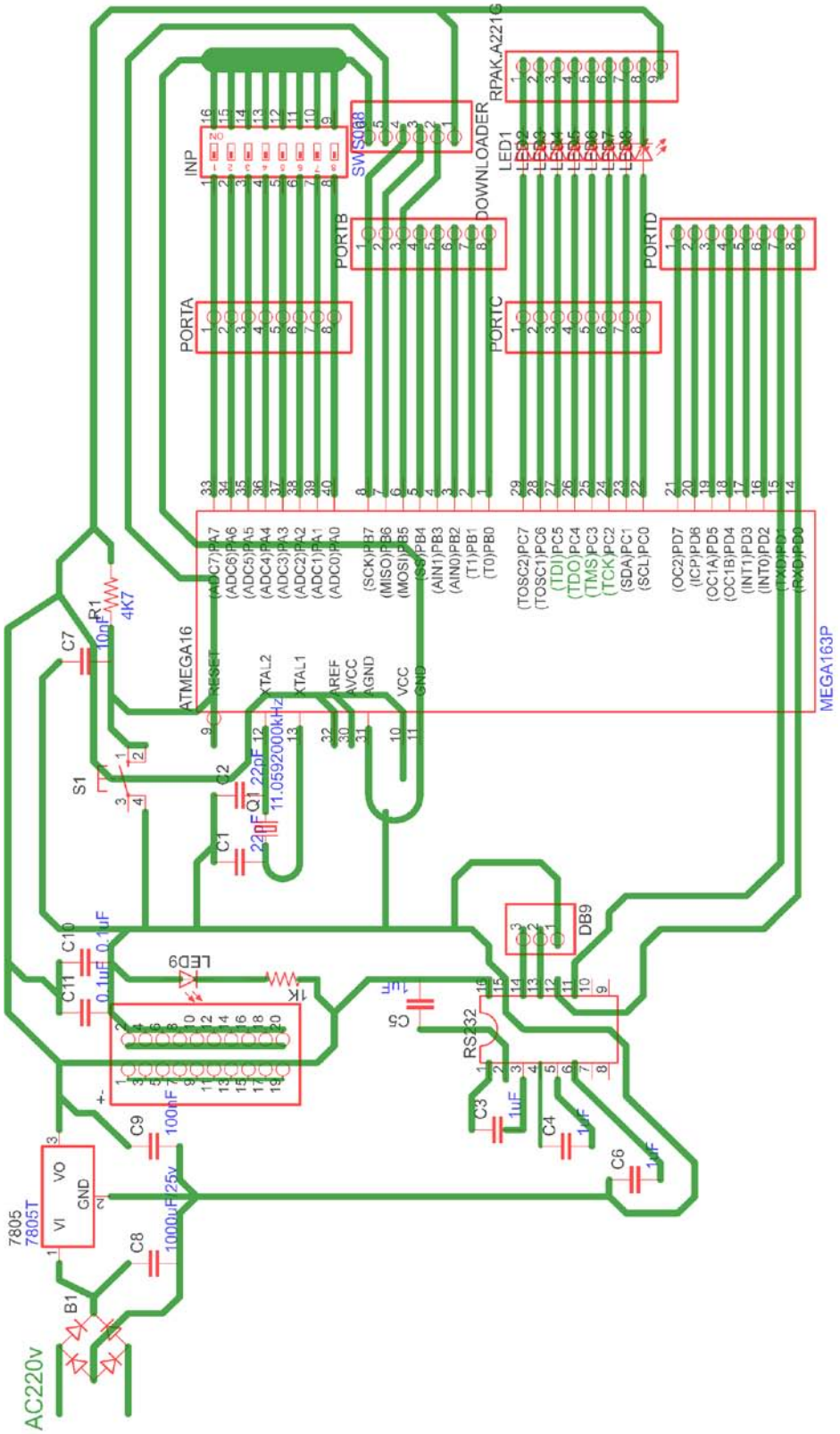
LAMPIRAN A

DATA SHEET

Data Sheet Modul ATMEGA16



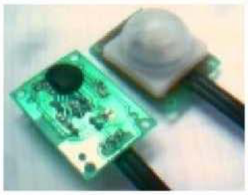
Sensor PIR KC7783R

KC7783R
PIR Module
Low Cost version

This is a low cost version for PIR module series from COMedia Ltd. It is designed for cost sensitive consumer product. Except the IC package format, all the mechanical and electrical spec is same as KC7783.

Features:

- IC soft package by dice bonding technique
- Small size: 25 x 35mm
- Ball lens is included as standard configuration
- 3 leads flat cable for easy connection
- 4 mounting holes on board
- High Sensitivity
- High immunity to RFI
- Power up delay to prevent from false triggering
- Output High for direct connect to control panel

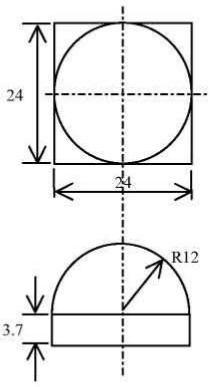


Specification

	Min	Typ	Max	Unit
Operation Voltage	4.7	5	12	V
Standby Current (no load)		300		μA
Output Pulse Width	0.5			Sec
Output High Voltage		5		V
Detection Range		5		M
Operation Temperature	-20	25	50	°C
Humidity Range			95	%

Note: 1. All other features and specification, please refer to KC778B
2. Minimum output pulse width can be customer specified.

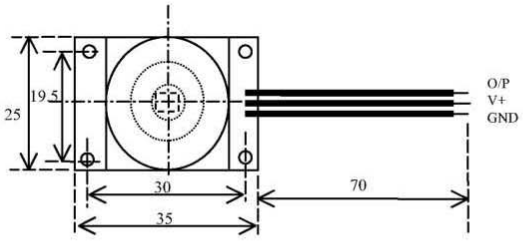
Lens Dimension (unit in mm)



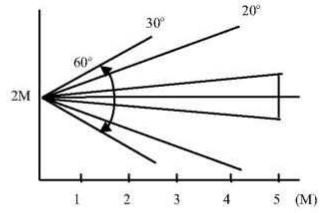
Standard Configuration

PIR controller	KC778B in dice form
PIR Sensor	RE200B by NICERA
Lens	Ball lens of 60° detection angle
Connector	3 leads flat cable, Power, GND, O/P

Mechanical Dimension



Vertical View Pattern



Application Note:

1. The PIR sensor is sensitive to the temperature change and therefore to prevent from operating the module in rapid environmental temperature changes, strong shock or vibration. Don't expose to the direct sun light or headlights of automobile. Don't expose to direct wind from heater or air conditioner.
2. This module is designed for indoor use. If using in outdoor, make sure to apply suitable supplemental optical filter and drop-proof, anti-dew construction
3. Detection range might be varied in different environmental temperature condition.

Sensor PIR #555-28027

PARALLAX

Web Site: www.parallax.com
Forums: forums.parallax.com
Sales: sales@parallax.com
Technical: support@parallax.com

Office: (916) 624-8333
Fax: (916) 624-8003
Sales: (888) 512-1024
Tech Support: (888) 997-8267

PIR Sensor (#555-28027)

General Description

The PIR (Passive Infra-Red) Sensor is a pyroelectric device that detects motion by measuring changes in the infrared levels emitted by surrounding objects. This motion can be detected by checking for a high signal on a single I/O pin.

Features

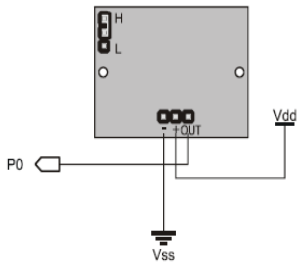
- Single bit output
- Small size makes it easy to conceal
- Compatible with all Parallax microcontrollers
- 3.3V & 5V operation with <100uA current draw

Application Ideas

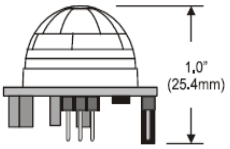
- Alarm Systems
- Halloween Props

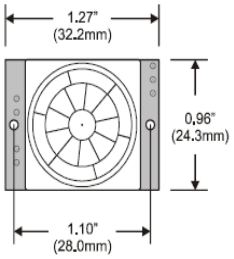
Quick Start Circuit

Note: The sensor is active high when the jumper (shown in the upper left) is in either position.



Module Dimensions





PIR 325 FL65

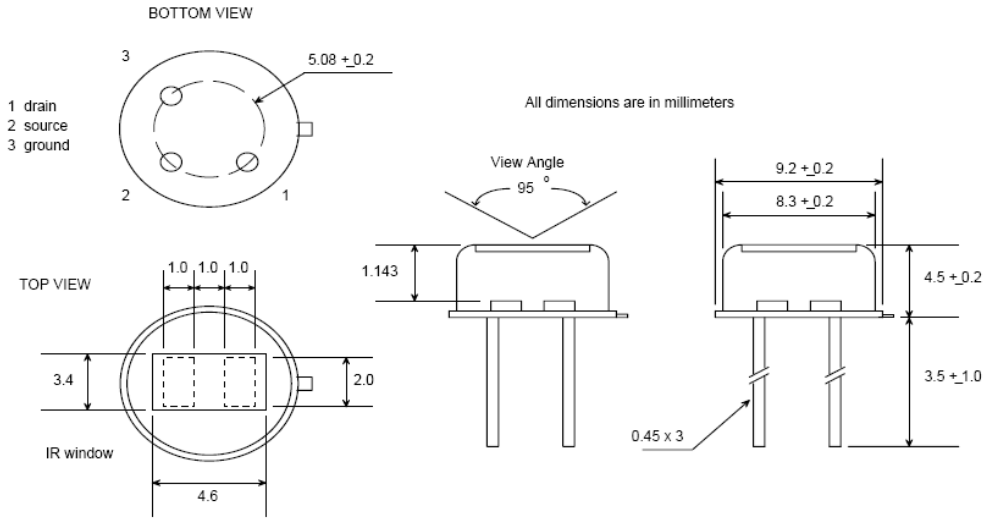
PIR325 Infrared Pyroelectric Sensor

Specifications and Dimensions

PIR325	
ELEMENT SIZE	2 x 1, 2 elements
SPECTRAL RESPONSE µm (1)	5 ~ 14
NOISE µVpp	20
OUTPUT mv pp (2)	3900
OFFSET VOLTAGE volts (3)	1.0
SUPPLY VOLTAGE volts (4)	2.5 ~ 15
OPERATING TEMPERATURE ° C	-30 ~ 70
STORAGE TEMPERATURE ° C	-40 ~ 80

NOTES:

1. With built-in window filter
2. After 72 db amplifier gain
3. At source pin 2 with 5 volts and 100K load
4. Well filtered power supply



Spesifikasi TLP 434A

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Vcc	Operating supply voltage		1.5	-	12	V
Icc	Peak Current		-	5	9	mA
Vin	Input High Voltage	IData=100Ua (High)	Vcc-0.5	-	Vcc	V
Vii	Input Low Voltage	IData=0Ua (Low)	-	-	0.3	V
Fo	Absolute Frequency		314.8	315	315.2	MHz
ΔFO	Relative To 433.92MHz			+/-150	+/-200	KHz
Po	RF Out Power Into 50ohm		-3	0	+2	dBm
	Modulation Bandwidth	External Encoding	-	5	-	KHz
Tr	Modulation Rise Time		-	-	100	uS
Tf	Modulation Fall Time		-	-	100	uS

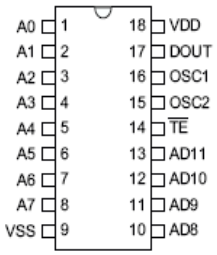
Spesifikasi RLP 434A

Symbol	Parameter	Conditions	Min	Typ	Max	
Vcc	Operating supply voltage		4.9	5	5.1	
ITot	Operating supply voltage		-	4.5	-	
I Data	Data Out	Data = +200 uA (High)I	Vcc-0.5	-	Vcc	V
		I Data = -10 uA (Low)	-	-	0.3	V

Electrical Characteristics					
Characteristics	SYM	Min	Typ	Max	Unit
Operation Radio Frequency	FC	300 - 434			MHz
Sensitivity	Pref			-106	dBm
Channel Width		+-500			KHz
Noise equivalent BW	NEB		5	4	KHz
Baseboard data rate				3	Kb/s

HT12E

8-Address
4-Address/Data



HT12E
–18 DIP

Pin Description

Pin Name	I/O	Internal Connection	Description
A0~A7	I	CMOS IN Pull-high (HT12A)	Input pins for address A0~A7 setting These pins can be externally set to VSS or left open
		NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	
AD8~AD11	I	NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	Input pins for address/data AD8~AD11 setting These pins can be externally set to VSS or left open
DOUT	O	CMOS OUT	Encoder data serial transmission output
$\overline{TE}$	I	CMOS IN Pull-high	Transmission enable, active low (see Note)
OSC1	I	OSCILLATOR 1	Oscillator input pin
OSC2	O	OSCILLATOR 1	Oscillator output pin
VSS	I	—	Negative power supply, grounds
VDD	I	—	Positive power supply

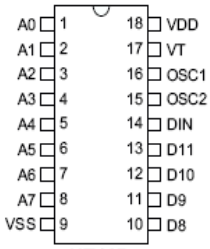
HT12E Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{DD}	Operating Voltage	—	—	2.4	5	12	V
I_{STB}	Standby Current	3V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I_{DD}	Operating Current	3V	No load f_{OSC}=3kHz	—	40	80	μA
		12V		—	150	300	μA
I_{DOUT}	Output Drive Current	5V	V_{OH}=0.9V_{DD} (Source)	–1	–1.6	—	mA
			V_{OL}=0.1V_{DD} (Sink)	1	1.6	—	mA
V_{IH}	"H" Input Voltage	—	—	0.8V_{DD}	—	V_{DD}	V
V_{IL}	"L" Input Voltage	—	—	0	—	0.2V_{DD}	V
f_{OSC}	Oscillator Frequency	5V	R_{OSC}=1.1MΩ	—	3	—	kHz
$R_{\overline{TE}}$	$\overline{TE}$ Pull-high Resistance	5V	$V_{\overline{TE}}$=0V	—	1.5	3	MΩ

HOLTEK

HT12D/HT12F
2^{12} Series of Decoders

8-Address
4-Data



HT12D
−18 DIP-A

Pin Description

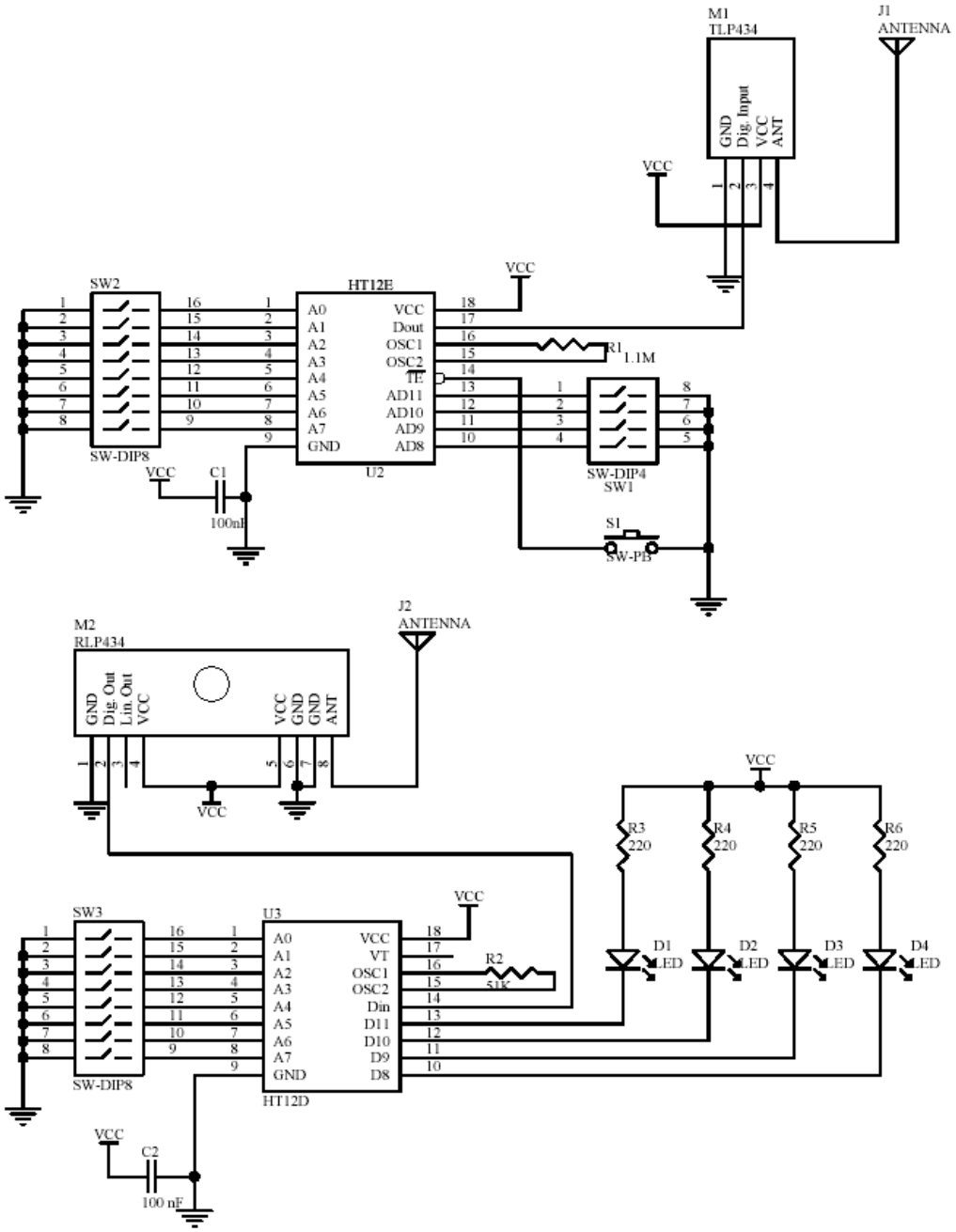
Pin Name	I/O	Internal Connection	Description
A0~A7 (HT12D)	I	NMOS Transmission Gate	Input pins for address A0~A7 setting These pins can be externally set to VSS or left open.
D8~D11 (HT12D)	O	CMOS OUT	Output data pins, power-on state is low.
DIN	I	CMOS IN	Serial data input pin
VT	O	CMOS OUT	Valid transmission, active high
OSC1	I	Oscillator	Oscillator input pin
OSC2	O	Oscillator	Oscillator output pin
VSS	—	—	Negative power supply, ground
VDD	—	—	Positive power supply

Electrical Characteristics

Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{DD}	Operating Voltage	—	—	2.4	5	12	V
I_{STB}	Standby Current	5V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I_{DD}	Operating Current	5V	No load, f_{OSC}=150kHz	—	200	400	μA
I_O	Data Output Source Current (D8~D11)	5V	V_{OH}=4.5V	–1	–1.6	—	mA
	Data Output Sink Current (D8~D11)	5V	V_{OL}=0.5V	1	1.6	—	mA
I_{VT}	VT Output Source Current	5V	V_{OH}=4.5V	–1	–1.6	—	mA
	VT Output Sink Current		V_{OL}=0.5V	1	1.6	—	mA
V_{IH}	"H" Input Voltage	5V	—	3.5	—	5	V
V_{IL}	"L" Input Voltage	5V	—	0	—	1	V
f_{OSC}	Oscillator Frequency	5V	R_{OSC}=51kΩ	—	150	—	kHz

Rangkaian RLP.TLP 434A



LAMPIRAN B

LISTING PROGRAM

Program Pengontrol

```
/*****************************************************
This program was produced by the
CodeWizardAVR V2.03.3 Evaluation
Automatic Program Generator
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com
Project :
Version :
Date    : 02/07/2010
Author  : Freeware, for evaluation and non-commercial use only
Company :
Comments:
Chip type           : ATmega16
Program type        : Application
Clock frequency     : 11,059200 MHz
Memory model        : Small
External RAM size   : 0
Data Stack size     : 256
*****************************************************/
#include <mega16.h>
#include <delay.h>
// Standard Input/Output functions
#include <stdio.h>
// Declare your global variables here
void main(void)
{
// Declare your local variables here
unsigned char datarx;
// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=P State6=P State5=P State4=P State3=P State2=P State1=P State0=P
PORTA=0xFF;
DDRA=0xFF;
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x00;
// Port C initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
PORTC=0x00;
DDRC=0xFF;
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
```

```
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x47;
// Analog Comparator initialization
```

```
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;
#asm ("nop")                    // Persiapan
while (1)
      {
      // Place your code here
      datarx=getchar();
      if(datarx=='p') //stop
         {
         PORTA.0=0;
         PORTA.1=0;
         PORTA.2=0;
         PORTA.3=0;
         PORTC.0=0;
         PORTC.1=0;
         PORTC.2=0;
         PORTC.3=0;
         };
      if(datarx=='w') //maju
         {
         PORTA.0=0;
         PORTA.1=0;
         PORTA.2=0;
         PORTA.3=1;
         PORTC.0=0;
         PORTC.1=0;
         PORTC.2=0;
         PORTC.3=1;
         };
      if(datarx=='s') //mundur
         {
         PORTA.0=0;
         PORTA.1=0;
         PORTA.2=1;
         PORTA.3=0;
         PORTC.0=0;
         PORTC.1=0;
         PORTC.2=1;
         PORTC.3=0;
         }
      if(datarx=='a') //kiri
         {
         PORTA.0=0;
         PORTA.1=0;
         PORTA.2=1;
         PORTA.3=1;
         PORTC.0=0;
         PORTC.1=0;
         PORTC.2=1;
         PORTC.3=1;
         delay_ms(1000);
         PORTA.0=0;
         PORTA.1=0;
```

```
    PORTA.2=0;
    PORTA.3=0;
    PORTC.0=0;
    PORTC.1=0;
    PORTC.2=0;
    PORTC.3=0;
    };
if(datarx=='d') //kanan
    {
    PORTA.0=0;
    PORTA.1=1;
    PORTA.2=0;
    PORTA.3=0;
    PORTC.0=0;
    PORTC.1=1;
    PORTC.2=0;
    PORTC.3=0;
    delay_ms(1000);
    PORTA.0=0;
    PORTA.1=0;
    PORTA.2=0;
    PORTA.3=0;
    PORTC.0=0;
    PORTC.1=0;
    PORTC.2=0;
    PORTC.3=0;
    };
if(datarx=='2') //up cam
    {
    PORTA.0=0;
    PORTA.1=1;
    PORTA.2=0;
    PORTA.3=1;
    PORTC.0=0;
    PORTC.1=1;
    PORTC.2=0;
    PORTC.3=1;
    delay_ms(1000);
    PORTA.0=0;
    PORTA.1=0;
    PORTA.2=0;
    PORTA.3=0;
    PORTC.0=0;
    PORTC.1=0;
    PORTC.2=0;
    PORTC.3=0;
     };
if(datarx=='5') //posisi awal 45derajat
    {
    PORTA.0=0;
    PORTA.1=1;
    PORTA.2=1;
    PORTA.3=0;
    PORTC.0=0;
    PORTC.1=1;
```

```
        PORTC.2=1;
        PORTC.3=0;
        };


    if(datarx=='8') //up cam
        {
        PORTA.0=0;
        PORTA.1=1;
        PORTA.2=1;
        PORTA.3=1;
        PORTC.0=0;
        PORTC.1=1;
        PORTC.2=1;
        PORTC.3=1;
        delay_ms(1000);
        PORTA.0=0;
        PORTA.1=0;
        PORTA.2=0;
        PORTA.3=0;
        PORTC.0=0;
        PORTC.1=0;
        PORTC.2=0;
        PORTC.3=0;
        };
    if(datarx=='o') //sensor on
        {
        PORTA.0=1;
        PORTA.1=0;
        PORTA.2=0;
        PORTA.3=0;
        PORTC.0=1;
        PORTC.1=0;
        PORTC.2=0;
        PORTC.3=0;
        };
    if(datarx=='z') //robot off
       {
       PORTA.0=1;
        PORTA.1=0;
        PORTA.2=0;
        PORTA.3=0;
        PORTC.0=1;
        PORTC.1=0;
        PORTC.2=0;
        PORTC.3=0;
        };
    };
}
```

Program Simulasi Robot

```
/*****************************************************
This program was produced by the
CodeWizardAVR V2.03.3 Evaluation
Automatic Program Generator
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com
Project :
Version :
Date    : 02/07/2010
Author  : Freeware, for evaluation and non-commercial use only
Company :
Comments:
Chip type           : ATmega16
Program type        : Application
Clock frequency     : 11,059200 MHz
Memory model        : Small
External RAM size   : 0
Data Stack size     : 256
*****************************************************/
#include <mega16.h>
#include <delay.h>
#define servokiri   PORTB.7
#define servokanan  PORTB.6
#define servotangan PORTB.5
#define blow        PORTB.3
#define off         PORTB.2
#define RX1         PINA.4
#define RX2         PINA.5
#define RX3         PINA.6
#define RX4         PINA.7
#define PIR1        PINA.2
#define PIR2        PINA.3
// Declare your global variables here
   int i;
   int j;
//PIR1=PIR Kiri
//PIR2=PIR Kanan
void main(void)
   {
   // Declare your local variables here
   // Input/Output Ports initialization
   // Port A initialization
   // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
   // State7=P State6=P State5=P State4=P State3=P State2=P State1=P State0=P
   PORTA=0xFF;
   DDRA=0x00;
   // Port B initialization
   // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
Func0=Out
   // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
   PORTB=0x00;
   DDRB=0xFF;
   // Port C initialization
```

```
    // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
Func0=Out
    // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
    PORTC=0x00;
    DDRC=0xFF;
    // Port D initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
    PORTD=0x00;
    DDRD=0x00;
    // Timer/Counter 0 initialization
    // Clock source: System Clock
    // Clock value: Timer 0 Stopped
    // Mode: Normal top=FFh
    // OC0 output: Disconnected
    TCCR0=0x00;
    TCNT0=0x00;
    OCR0=0x00;
    // Timer/Counter 1 initialization
    // Clock source: System Clock
    // Clock value: Timer 1 Stopped
    // Mode: Normal top=FFFFh
    // OC1A output: Discon.
    // OC1B output: Discon.
    // Noise Canceler: Off
    // Input Capture on Falling Edge
    // Timer 1 Overflow Interrupt: Off
    // Input Capture Interrupt: Off
    // Compare A Match Interrupt: Off
    // Compare B Match Interrupt: Off
    TCCR1A=0x00;
    TCCR1B=0x00;
    TCNT1H=0x00;
    TCNT1L=0x00;
    ICR1H=0x00;
    ICR1L=0x00;
    OCR1AH=0x00;
    OCR1AL=0x00;
    OCR1BH=0x00;
    OCR1BL=0x00;
    // Timer/Counter 2 initialization
    // Clock source: System Clock
    // Clock value: Timer 2 Stopped
    // Mode: Normal top=FFh
    // OC2 output: Disconnected
    ASSR=0x00;
    TCCR2=0x00;
    TCNT2=0x00;
    OCR2=0x00;
    // External Interrupt(s) initialization
    // INT0: Off
    // INT1: Off
    // INT2: Off
    MCUCR=0x00;
    MCUCSR=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;
#asm ("nop")
blow=0;
off=0;
while (1)
      {
      // Place your code here
      PORTC=PINA;
      if (RX1==1 && RX2==0 && RX3==0 && RX4==1) //OFF
         {
          off=1;
         };
      if (RX1==0 && RX2==0 && RX3==0 && RX4==0) //berhenti
         {servokiri=0;
         servokanan=0;
         servotangan=1;
         delay_us(600);
         servokiri=0;
         servokanan=0;
         servotangan=0;
         delay_us(600);
         servokiri=0;
         servokanan=0;
         servotangan=1;
         delay_us(600);
         servokiri=0;
         servokanan=0;
         servotangan=0;
         delay_us(600);
         servokiri=0;
         servokanan=0;
         servotangan=1;
         delay_us(17600);
         };
       if (RX1==0 && RX2==0 && RX3==0 && RX4==1) //maju
         {servokiri=1;
         servokanan=1;
         servotangan=1;
         delay_us(600);
         servokiri=1;
         servokanan=0;
         servotangan=0;
         delay_us(600);
         servokiri=1;
         servokanan=0;
         servotangan=1;
         delay_us(600);
         servokiri=1;
         servokanan=0;
```

```
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(17600);
    };
  if (RX1==0 && RX2==0 && RX3==1 && RX4==0)  //mundur
    {servokiri=1;
    servokanan=1;
    servotangan=1;
    delay_us(600);
    servokiri=0;
    servokanan=1;
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=1;
    servotangan=1;
    delay_us(600);
    servokiri=0;
    servokanan=1;
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(17600);
    };
  if (RX1==0 && RX2==0 && RX3==1 && RX4==1)  //kiri
    {servokiri=1;
    servokanan=1;
    servotangan=1;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(17600);
    };
  if (RX1==0 && RX2==1 && RX3==0 && RX4==0)  //kanan
    {servokiri=1;
    servokanan=1;
    servotangan=1;
```

```
    delay_us(600);
    servokiri=1;
    servokanan=1;
    servotangan=0;
    delay_us(600);
    servokiri=1;
    servokanan=1;
    servotangan=1;
    delay_us(600);
    servokiri=1;
    servokanan=1;
    servotangan=0;
    delay_us(600);
    servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(17600);
    };
if (RX1==0 && RX2==1 && RX3==1 && RX4==0)  //undergroundcam
    {servokiri=0;
    servokanan=0;
    servotangan=1;
    delay_us(1900);
    servokiri=0;
    servokanan=0;
    servotangan=0;
    delay_us(18100);
    i=8;                          //45 derajat
   };
 if (RX1==0 && RX2==1 && RX3==1 && RX4==1)  //upcam
    { i=i-1;                 // +10derajat
     };
 if (RX1==0 && RX2==1 && RX3==0 && RX4==1)  //downcam
    {i=i+1;                  //-10derajat
    };
 if(i==8)                              // 45 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1900);
       servokiri=0;
       servokanan=0;
       servotangan=0;
       delay_us(18100);
       };
    };
 if(i==7)                              //55 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1800);
       servokiri=0;
       servokanan=0;
```

```
        servotangan=0;
        delay_us(18200);
        };
    };
if(i==6)                          //65 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1700);
       servokiri=0;
       servokanan=0;
       servotangan=0;
       delay_us(18300);
       };
    };
if(i==5)                          //75 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1600);
       servokiri=0;
       servokanan=0;
       servotangan=0;
       delay_us(18400);
       };
    };
 if(i==4)                          //85 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1500);
       servokiri=0;
       servokanan=0;
       servotangan=0;
       delay_us(18500);
       };
   };
 if(i==3)                         //95 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
       servokanan=0;
       servotangan=1;
       delay_us(1400);
       servokiri=0;
       servokanan=0;
       servotangan=0;
       delay_us(18600);
       };
    };
 if(i==2)                          //105 derajat
   {for(j=0;j<5;j++)
      {servokiri=0;
```

```
      servokanan=0;
      servotangan=1;
      delay_us(1300);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(18700);
      };
  };

if(i==1)                                  //115 derajat
  { for(j=0;j<5;j++)
     {servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(1200);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(18800);
      };
   };
if(i==0)                                   //125 derajat
  {for(j=0;j<5;j++)
     {servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(1100);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(18900);
      };
   };
if (RX1==1 && RX2==0 && RX3==0 && RX4==0)  //sensor on
   {goto mulai;
   };
mulai:
  {if(PIR1==1 && PIR2==1)
      {   for(j=0;j<100;j++)                //putar kiri       0
          {servokiri=1;
           servokanan=1;
           servotangan=1;
           delay_us(600);
           servokiri=0;
           servokanan=0;
           servotangan=0;
           delay_us(600);
           servokiri=0;
           servokanan=0;
           servotangan=1;
           delay_us(600);
           servokiri=0;
           servokanan=0;
           servotangan=0;
```

```
 delay_us(600);
 servokiri=0;
 servokanan=0;
 servotangan=1;
 delay_us(17600);
}
 if(PIR1==1 && PIR2==1)            //1Y
    {blow=1;
     for(j=0;j<200;j++)              //putar kanan      10
            {servokiri=1;
            servokanan=1;
            servotangan=1;
            delay_us(600);
            servokiri=1;
            servokanan=1;
            servotangan=0;
            delay_us(600);
            servokiri=1;
            servokanan=1;
            servotangan=1;
            delay_us(600);
            servokiri=1;
            servokanan=1;
            servotangan=0;
            delay_us(600);
            servokiri=0;
            servokanan=0;
            servotangan=1;
            delay_us(17600);}
      if(PIR1==1 && PIR2==1)          //11Y
            { blow=1;                 //12
              delay_ms(1000);
              blow=0;
              delay_ms(1000);
               for(j=0;j<100;j++)              //putar kiri       15
                 {servokiri=1;
                  servokanan=1;
                  servotangan=1;
                  delay_us(600);
                  servokiri=0;
                  servokanan=0;
                  servotangan=0;
                  delay_us(600);
                  servokiri=0;
                  servokanan=0;
                  servotangan=1;
                  delay_us(600);
                  servokiri=0;
                  servokanan=0;
                  servotangan=0;
                  delay_us(600);
                  servokiri=0;
                  servokanan=0;
                  servotangan=1;
                  delay_us(17600);
```

```
                }
               if(PIR1==1 && PIR2==1)
               {blow=1;                          //17
                delay_ms(1000);
                blow=0;
                delay_ms(1000);
               }
               else
               {goto mulai;
               }
          }
     if(PIR1==1 && PIR2==0)          //11N
          {for(j=0;j<100;j++)              //putar kiri        13
               {servokiri=1;
                servokanan=1;
                servotangan=1;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=0;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=1;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=0;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=1;
                delay_us(17600);
               }
          blow=1;                    //14b
          delay_ms(1000);
          blow=0;
          delay_ms(1000);
          goto mulai;
          }
   }
if(PIR1==0 && PIR2==1)          //1N
   {  for(j=0;j<100;j++)              //putar kanan      3
      {servokiri=1;
      servokanan=1;
      servotangan=1;
      delay_us(600);
      servokiri=1;
      servokanan=1;
      servotangan=0;
      delay_us(600);
      servokiri=1;
      servokanan=1;
      servotangan=1;
      delay_us(600);
```

```
servokiri=1;
servokanan=1;
servotangan=0;
delay_us(600);
servokiri=0;
servokanan=0;
servotangan=1;
delay_us(17600);}
if(PIR1==1 && PIR2==1)        //4Y
{blow=1;                      //5
 delay_ms(1000);
 blow=0;
 delay_ms(1000);
 for(j=0;j<100;j++)           //putar kiri       8
     {servokiri=1;
      servokanan=1;
      servotangan=1;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(17600);
     }
 if(PIR1==1 && PIR2==1)       //9Y
       {blow=1;               //14a
        delay_ms(1000);
        blow=0;
        delay_ms(1000);
       goto mulai;
       }
 else                         //9N
       {goto mulai;           //14b
       }
}
if(PIR1==1 && PIR2==0)        //4N
{for(j=0;j<100;j++)           //putar kiri       6
     {servokiri=1;
      servokanan=1;
      servotangan=1;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=0;
```

```
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=1;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=0;
                delay_us(600);
                servokiri=0;
                servokanan=0;
                servotangan=1;
                delay_us(17600);
               }
            blow=1;                              //7
            delay_ms(1000);
            blow=0;
            delay_ms(1000);
            goto mulai;
            }
         }
}
if(PIR1==1 && PIR2==0)   //kiri
{for(j=0;j<100;j++)
     {servokiri=1;
      servokanan=1;
      servotangan=1;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=0;
      delay_us(600);
      servokiri=0;
      servokanan=0;
      servotangan=1;
      delay_us(17600);
      }
blow=1;
delay_ms(1000);
blow=0;
delay_ms(1000);
}
if(PIR1==0 && PIR2==1) //kanan
{for(j=0;j<100;j++)
      {servokiri=1;
      servokanan=1;
      servotangan=1;
```

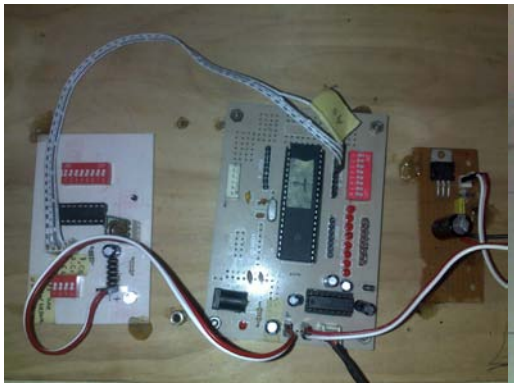
```
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=0;
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=1;
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=0;
        delay_us(600);
        servokiri=0;
        servokanan=0;
        servotangan=1;
        delay_us(17600);
        }
blow=1;
delay_ms(1000);
blow=0;
delay_ms(1000);
}
if(PIR1==0 && PIR2==0) //bebas
{for(j=0;j<500;j++)
      {servokiri=1;                         //kanan
        servokanan=1;
        servotangan=1;
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=0;
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=1;
        delay_us(600);
        servokiri=1;
        servokanan=1;
        servotangan=0;
        delay_us(600);
        servokiri=0;
        servokanan=0;
        servotangan=1;
        delay_us(17600);
        }
if(PIR1==1 || PIR2==1)
        {goto mulai;
        }
if(PIR1==0 && PIR2==0)    //tetap ga da orang
      {for(j=0;j<500;j++)
            {servokiri=1;                       //maju
             servokanan=1;
             servotangan=1;
             delay_us(600);
```

```
                    servokiri=1;
                    servokanan=0;
                    servotangan=0;
                    delay_us(600);
                    servokiri=1;
                    servokanan=0;
                    servotangan=1;
                    delay_us(600);
                    servokiri=1;
                    servokanan=0;
                    servotangan=0;
                    delay_us(600);
                    servokiri=0;
                    servokanan=0;
                    servotangan=1;
                    delay_us(17600);
                    }
                }
            goto mulai;
            }
        }
    }
}
```

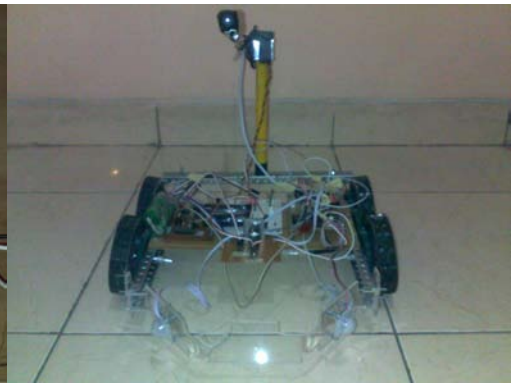
LAMPIRAN C

FOTO DAN GAMBAR

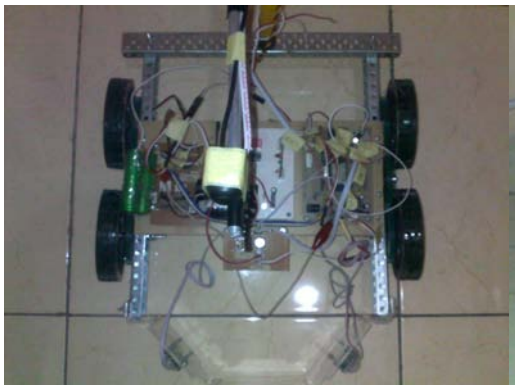
Foto Simulasi Robot



Pengontrol

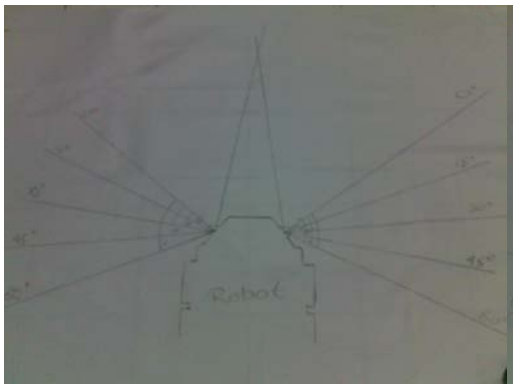


Robot tampak depan

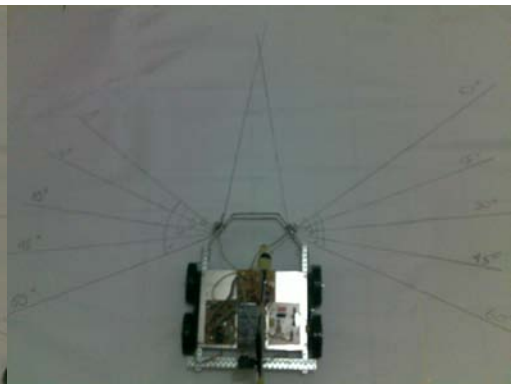


Robot tampak atas

Robot tampak samping



Pengujian sensor terhadap sudut



Robot diatas area pengujian sensor

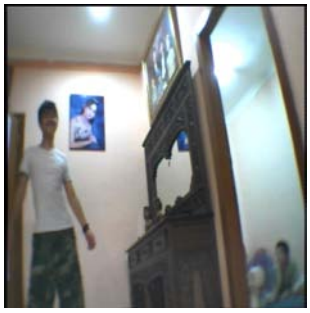
Foto Hasil *Wireless Camera*

5

kiri

3

atas



4

kanan

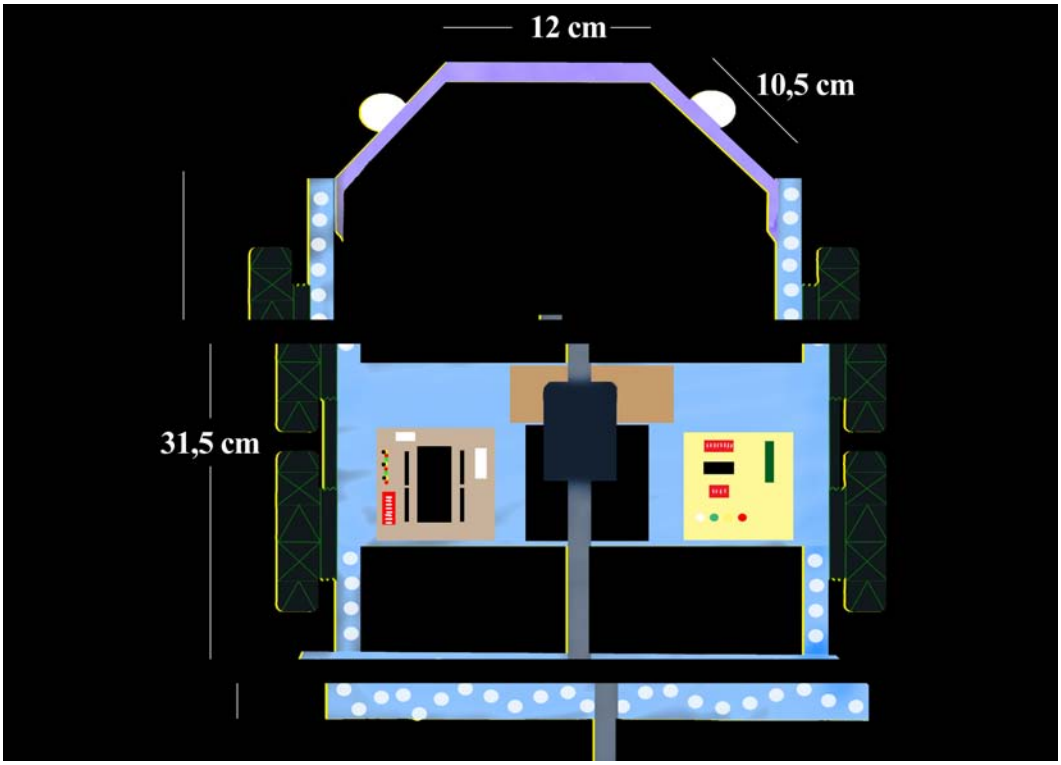
2

atas

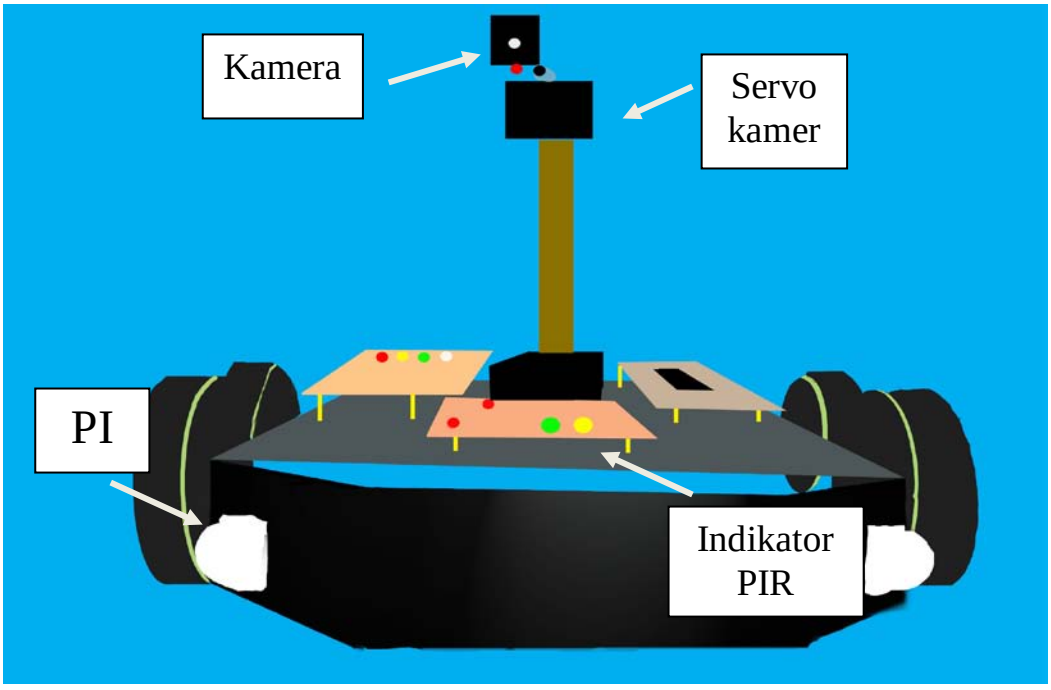
1

start

Gamabar Robot

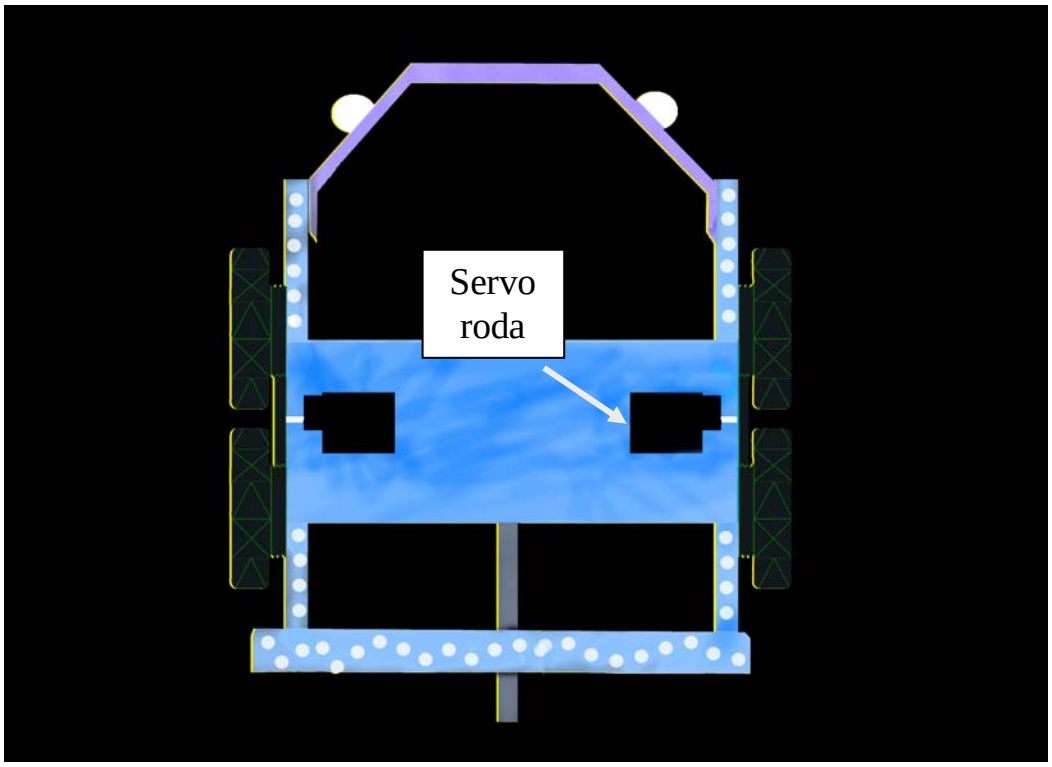


Ukuran robot

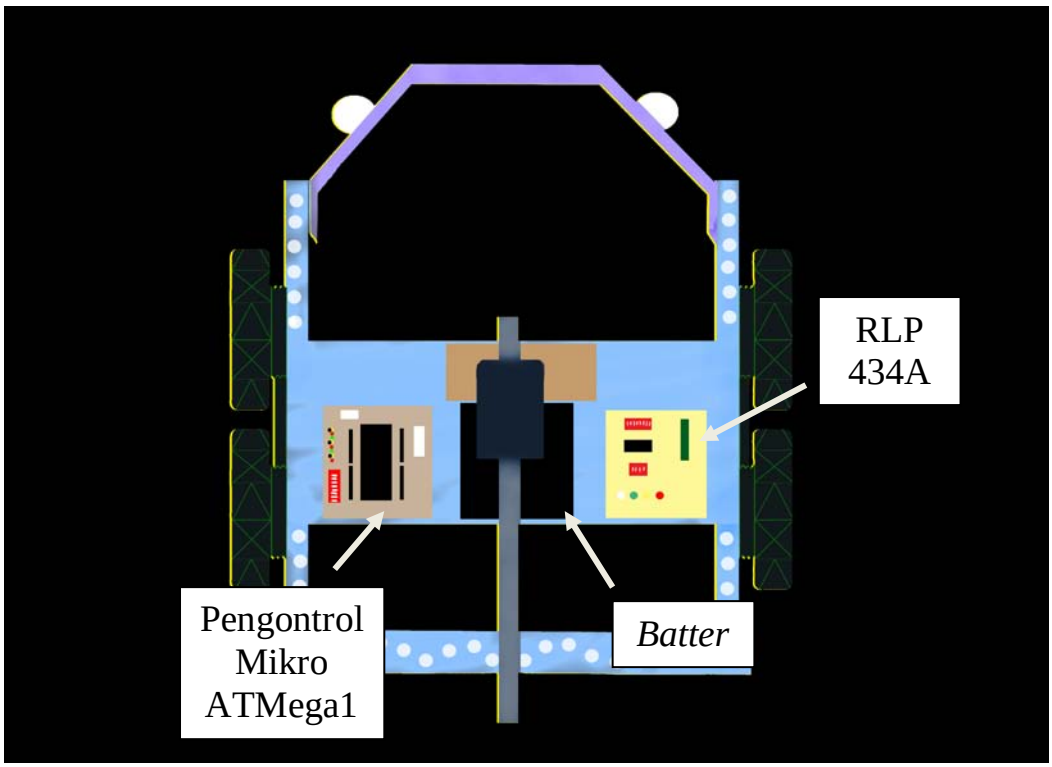


Tampak depan

Gambar Robot

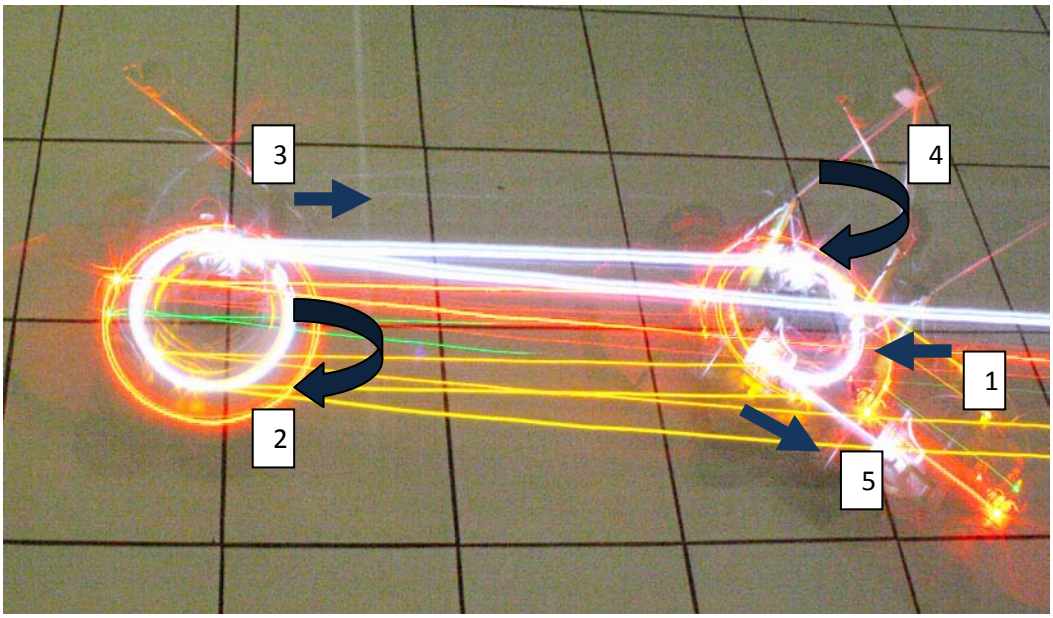


Tampak bawah

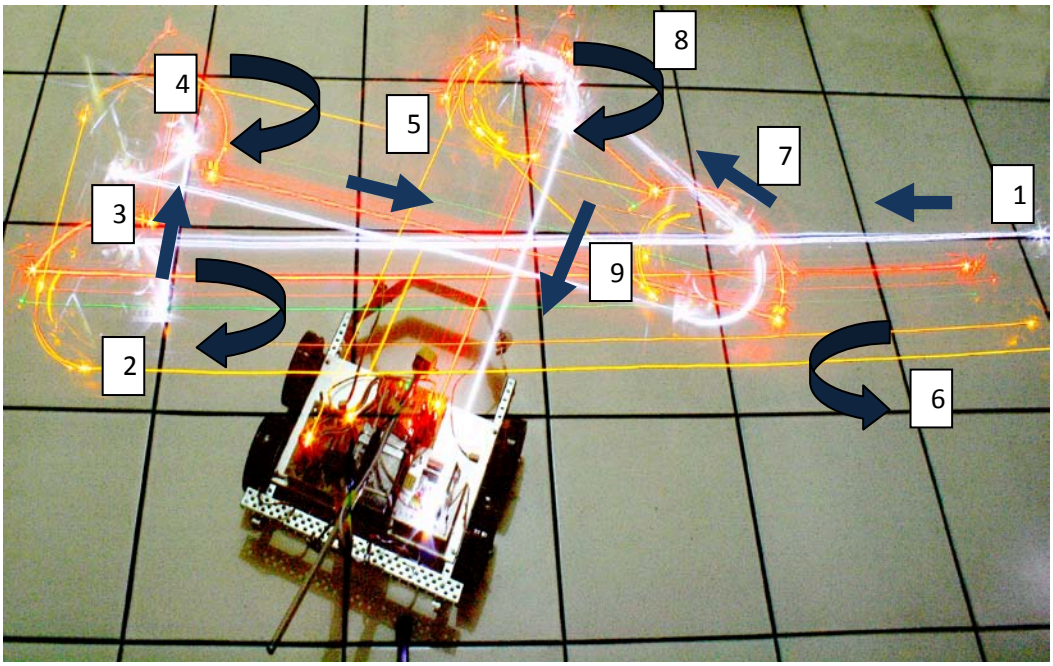


Tampak atas

Foto Pergerakan Robot

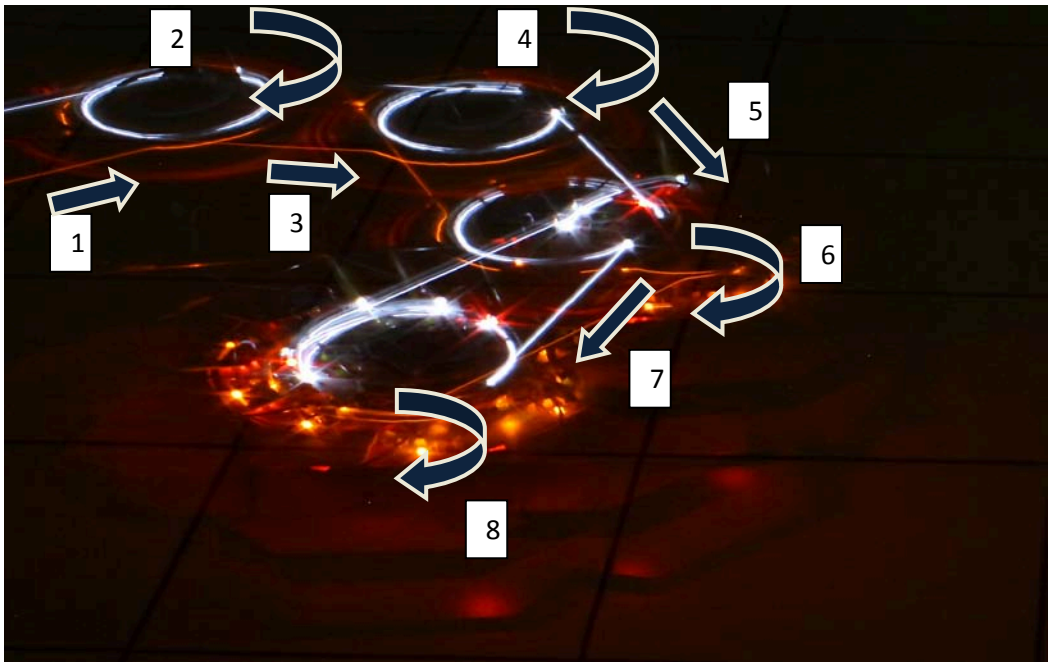


Pengendalian dengan PC, *error* karena kesalahan pengontrol

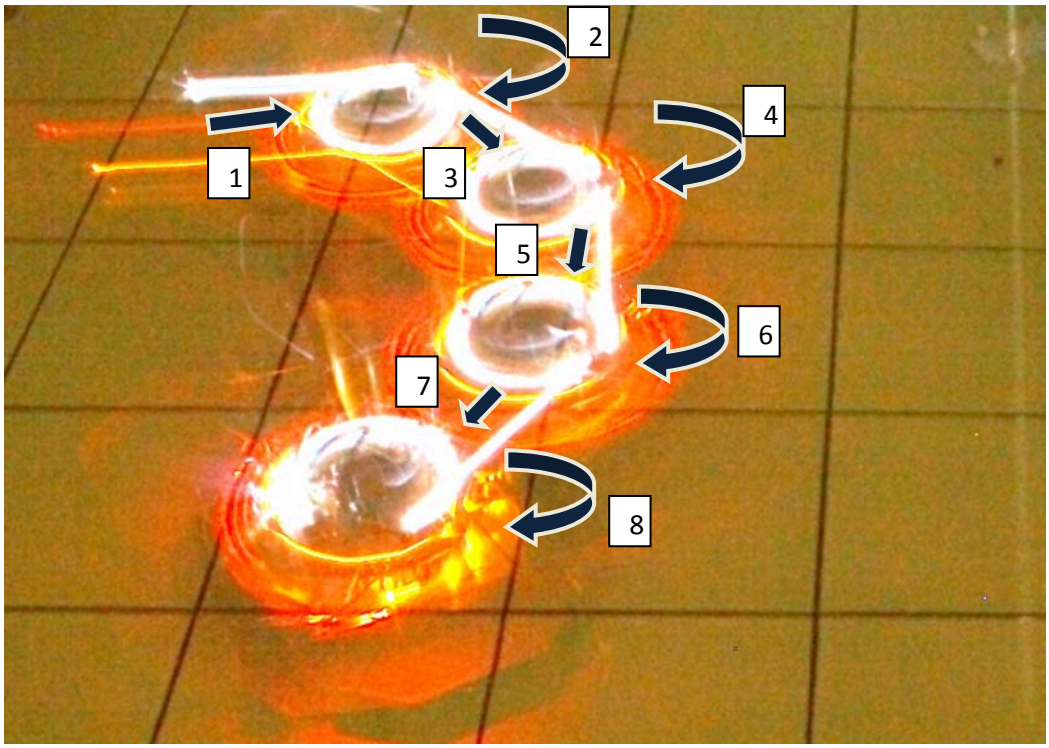


Pengendalian dengan PC, berhasil robot bergerak maju, kanan, maju, kanan, maju, kiri, maju, kanan, mundur

Foto Pergerakan Robot

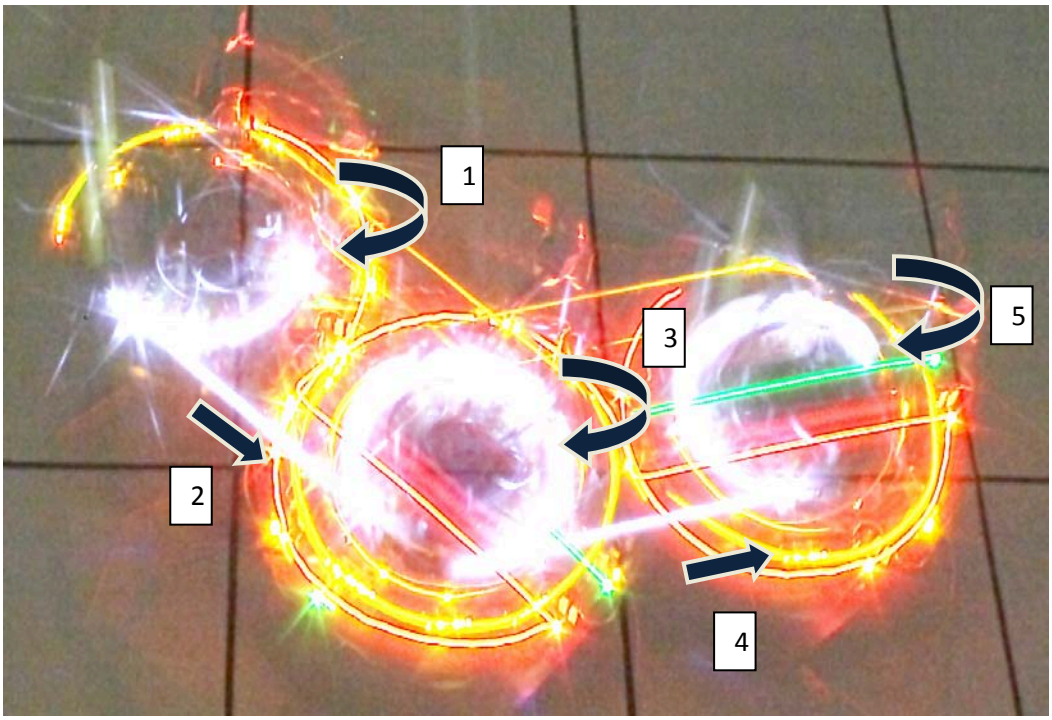


Tidak ada orang di sekitar robot, *error* karena robot sempat berhenti seperti ada orang

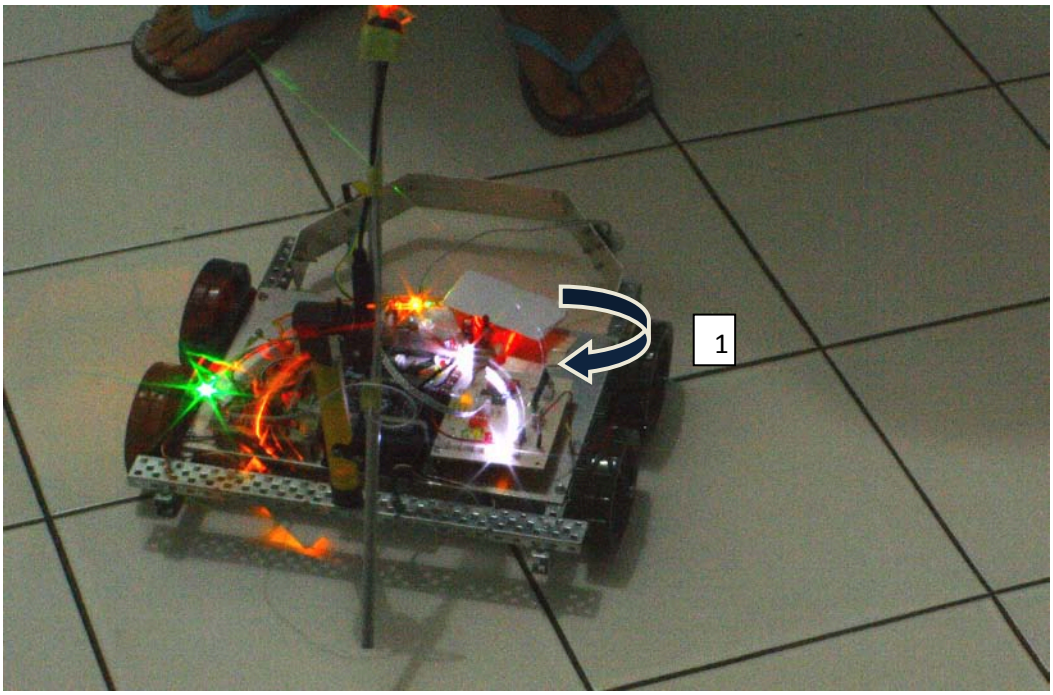


Tidak ada orang di sekitar robot, berhasil putar kanan kemudian maju

Foto Pergerakan Robot

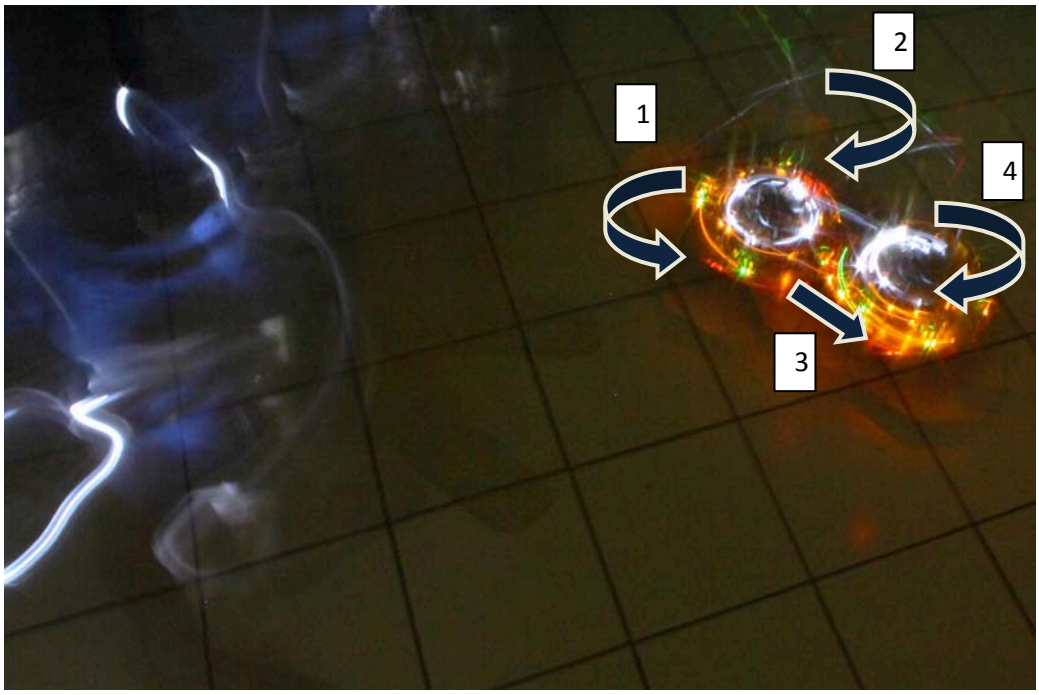


Orang di sebelah kanan robot, *error* karena sebelum sensor selesai menyensing sensor sudah tidak mendeteksi adanya orang sehingga robot berputar seperti tidak ada orang



Orang di sebelah kanan robot, berhasil robot putar kanan dan led on

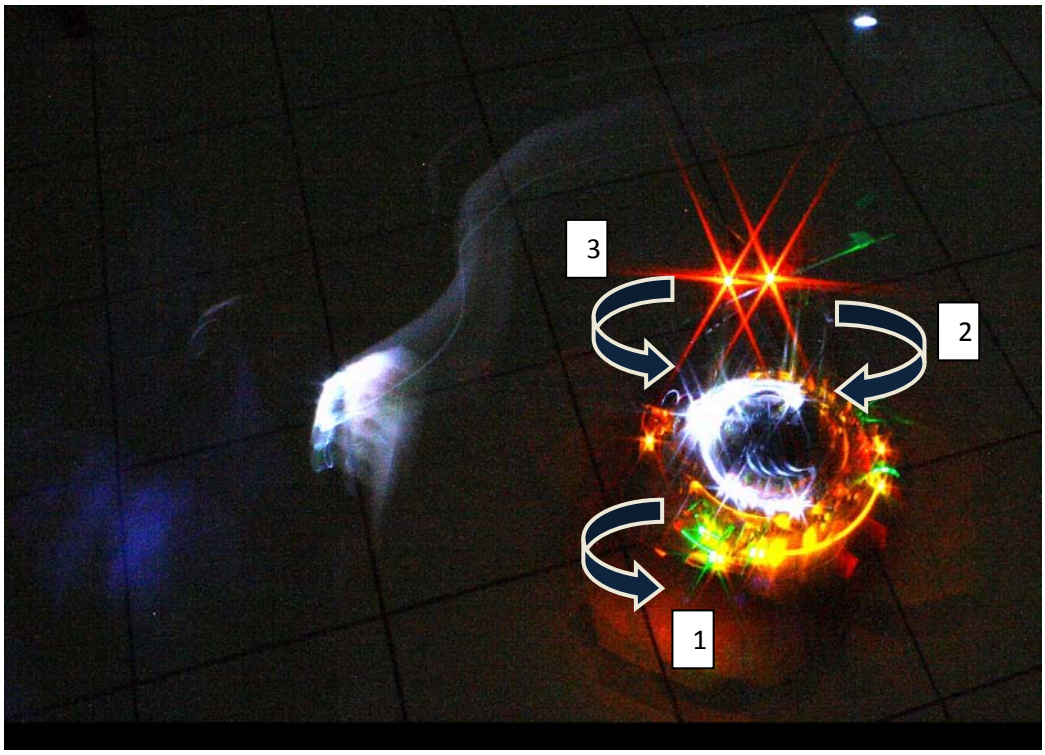
Foto Pergerakan Robot



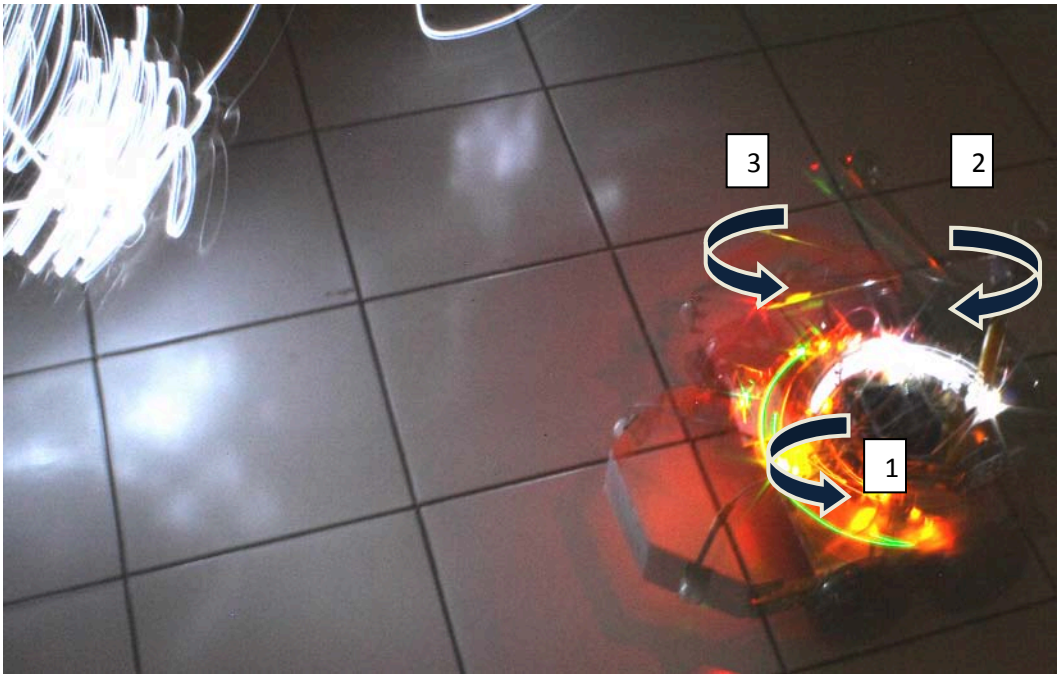
Orang di sebelah kiri robot, *error* karena sebelum sensor selesai menyensing sensor sudah tidak mendeteksi adanya orang sehingga robot berputar seperti tidak ada orang

Orang di sebelah kiri robot, berhasil robot putar kiri dan led on

Foto Pergerakan Robot



Orang di depan robot, *error* karena robot mengira orang ada di depan dan di kanan robot sehingga led on dua kali



Orang di depan robot, berhasil robot mencari ke sebelah kiri, ke sebelah kanan, dan kemudian ke depan, karena di depan ada orang led on