

Tallinna Tehnikaülikool

Virumaa Kolledž

Eriala: Tootmise Automatiseerimine

«Автоматическая кормилка для животных»

Lektor : S.Pavlov

Üliõpilane: Aleksei Hristoforov,

Andres Kozuhhovski,

Sergei Kukonenko

Rühm: RDDR71

- **Введение (основная идея проекта)**

Главная идея проекта заключается в создании автоматической системы кормления животных. Система освобождает человека от ежедневного кормления животного а животному гарантирует регулярную выдачу корма.

- **Цель и задачи проекта**

Цель - сделать самодостаточную систему, которая может обходиться без человека ни один день и исправно (и вовремя) кормить животного.

- **Требования к проекту (габариты, вес, масса, автономность)**

размеры: 500x300x700

ограничения по массе: Нет

Источник питания: Сеть

Дозированные порции: 3 раза в день

- **Ограничение в проектировании (использование KoduLabor комплектов или альтернативных плат Arduino)**

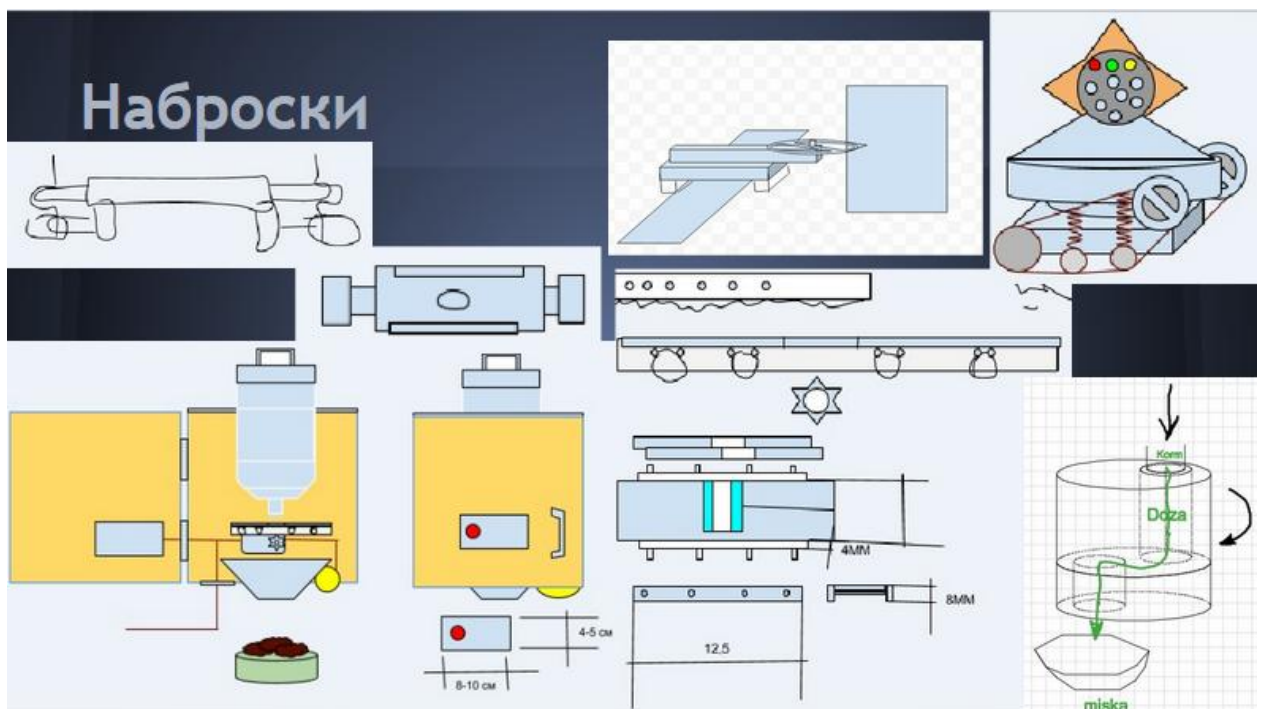
Использование KoduLabor

Стоимость остального оборудования, не должна превышать 150€

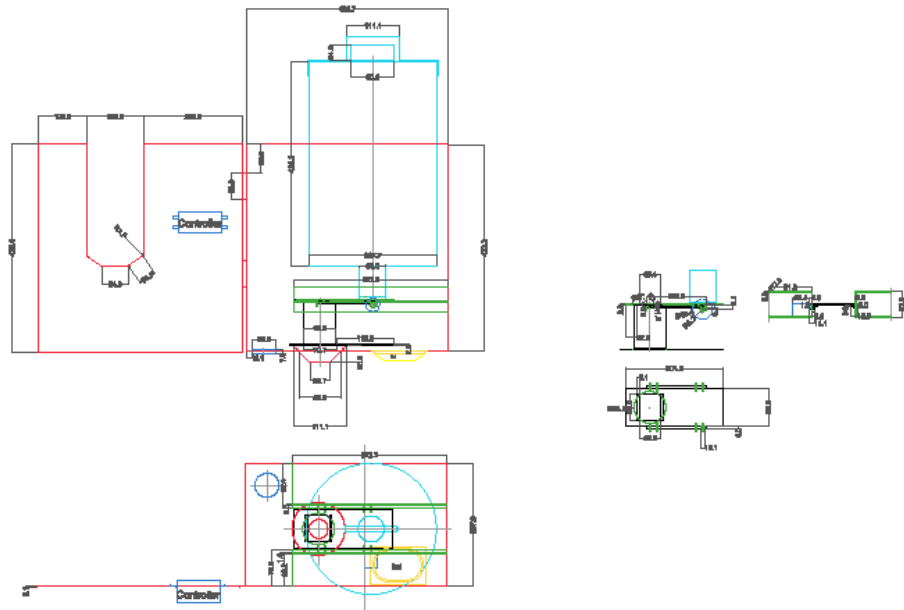
- Общая модель системы (структура системы и модели функциональности в виде блок- схем



Варианты реализации идеи проекта (3 варианта решения, в виде рисунков, схем, эскизов)



- Механическая схема проекта (3D модель, основные узлы, механизмы передачи т. д.- чертежи)



[.PDF](#)

[.DXF](#)

- **Алгоритм**

1. Начало

- Подача питания системе
- Библиотеки (или программа) часов, двигателей
- 1.1. Программа 1 (объем пищи)

- По умолчанию стоит "состояние" дозы - 1 (количество пищи)
- Если нажата **кнопка1** то "состояние" переводится в "текущее + 1" ($1 + 1 = 2$ состояние)
- Если стоит "состояние" 2 и опять нажата **кнопка1**, то состояние становится 3 ($2 + 1 = 3$ дозы)
- Если стоит "состояние" 3 и опять нажата **кнопка1**, то состояние становится снова 1 (1 доза)

- 1.2. Программа 2 (Настройка времени)

- По умолчанию таймер идет с 00.00 (время)
- Если нажата **кнопка2** то к таймеру прибавляется 1 час

- Если нажата **кнопка3** то к таймеру прибавляется 5 минут
1.1.1. ПодПрограмма 1 (питание)
- Когда на часах 06.00 или 12.00 или 18.00 (постоянный цикл) то
-> дозатор передвигается из положения 0 в положение 1, стоит там “N” секунд и возвращается в положение 0

-> цикл делается “*состояние*” раз (например, если “*состояние*” = 2, то 2 раза), соблюдая “N” секунд в положении 0 для наполнения дозатора кормом

1.3. Программа 3 (освещение)

- Если фотодатчик регистрирует освещение в 0.1 до 1 lx, то **работает** лампа освещения

Програмный код

Старый код

```
#include <homelab/module/motors.h>
#include <homelab/pin.h>
#include <homelab/delay.h>
#include <time.h>
#include <homelab/adc.h>

// Концевики положения дозатора
pin left  = PIN(B, 0); //лев. концевик
pin right = PIN(B, 1); //прав. концевик
pin button1      = PIN(C, 0); //кнопка порций
pin button2      = PIN(C, 1); //+ время
pin button3      = PIN(C, 2); //+ время
pin led1  = PIN(C, 3); //1 порция
pin led2  = PIN(C, 4); //2 порция
pin led3  = PIN(C, 5); //3 порция
pin lampa = PIN(B, 2); //лампа освещения

//
// Основная программа
//
int main(void)
{
    // Инициализация двигателя 0
    dcmotor_init(0);
    // Выводы концевиков
    pin_setup_input_with_pullup(left);
    pin_setup_input_with_pullup(right);
    //выводы кнопок и ламп
    pin_setup_output(led1);
    pin_setup_output(led2);
    pin_setup_output(led3);
    pin_setup_output(lampa);
    pin_setup_input(button1);
    pin_setup_input(button2); // + час
    pin_setup_input(button3); // + 5 мин
    //ADC преобрз.
    adc_init(ADC_REF_AVCC, ADC_PRESCALE_8);
    //Настройка LCD экрана
    lcd_alpha_init(LCD_ALPHA_DISP_ON);
    //Переменная
```

```

int N = 1; //Число нажатий
int seconds = 0; // секунды для часов

// Бесконечный цикл
while (true)
{
    //-----Время-----
    pin_set_to(seconds + 3600, pin_get_value(buttons2)); //+час при нажатии
    pin_set_to(seconds + 300, pin_get_value(buttons3)); //+ 5 мин
    char text[16];
    lcd_alpha_write_string("Время"); // Название программы
    sprintf(text, "%02d:%02d:%02d", (seconds / 3600) % 24, (seconds / 60) % 60, seconds % 60);
    lcd_alpha_goto_xy(0, 1);
    lcd_alpha_write_string(text);
    seconds++; // Увеличение секунд на 1
    hw_delay_ms(1000); // Аппаратная пауза в 1000 миллисекунд
    if (seconds = 86400) seconds = 1;

    //-----Освещение миски-----
    // Считывание осредненного значения фоторезистора
    adc_value = adc_get_average_value(1, 10);
    // Расчет напряжения во входе ADC
    voltage = 5.0 * ((double)adc_value / 1024.0);
    // Измерение сопротивления фоторезистора в делителе напряжения
    resistance = (10.0 * 5.0) / voltage - 10.0;
    // Вычисление освещенности в люксах
    illuminance = 255.84 * pow(resistance, -10/9);
    if (illuminance < 1) //при освещении меньше 1 люмен (темнота)
    {
        pin_clear(lamp); //включается лампа
    }
    else
    {
        pin_set(led_yellow); //отключается
    }

    //-----Количество порций-----
    if (pin_get_value(button1) && N<3)
    {
        N= N+1;
    }
    else if (pin_get_value(button1) == N<3)
    {
        N = 1
    }

    //-----Индикация порций-----
    if (N = 1)
    {
        pin_clear(led1);
    }
    else if (N = 2)
    {
        pin_clear(led1);
        pin_clear(led2);
    }
    else if (N = 3)
    {
        pin_clear(led1);
        pin_clear(led2);
        pin_clear(led3);
    }
}

```

```

}

//-----кормежка-----
//21600 - 06.00; 43200 - 12.00; 64800 - 18.00
if (seconds == 21600 || seconds == 43200 || seconds == 64800)
{
    if (N = 1)
    {
        dcmotor_drive(0, 1);
        if (pin_get_value(left))
        {
            sw_delay_ms(3000);
            seconds = seconds + 3;
            dcmotor_drive(0, -1);
            if (pin_get_value(right)) dcmotor_drive(0, 0);
        }
    }
    else if (N = 2)
    {
        dcmotor_drive(0, 1);
        if (pin_get_value(left))
        {
            sw_delay_ms(3000);
            seconds = seconds + 3;
            dcmotor_drive(0, -1);
            if (pin_get_value(right))
            {
                sw_delay_ms(3000);
                seconds = seconds + 3;
                dcmotor_drive(0, 1);
                if (pin_get_value(left))
                {
                    sw_delay_ms(3000);
                    seconds = seconds + 3;
                    dcmotor_drive(0, -1);
                    if (pin_get_value(right)) dcmotor_drive(0, 0);
                }
            }
        }
    }
    else
    {
        dcmotor_drive(0, 1);
        if (pin_get_value(left))
        {
            sw_delay_ms(3000);
            seconds = seconds + 3;
            dcmotor_drive(0, -1);
            if (pin_get_value(right))
            {
                sw_delay_ms(3000);
                seconds = seconds + 3;
                dcmotor_drive(0, 1);
                if (pin_get_value(left))
                {
                    sw_delay_ms(3000);
                    seconds = seconds + 3;
                    dcmotor_drive(0, -1);
                    if (pin_get_value(right))
                    {
                        sw_delay_ms(3000);
                        seconds = seconds + 3;
                        dcmotor_drive(0, 1);
                        if (pin_get_value(left))
                        {
                            sw_delay_ms(3000);
                            seconds = seconds + 3;
                        }
                    }
                }
            }
        }
    }
}

```

```

    }
}
}

```

```

dcmotor_drive(0, -1);
if (pin_get_value(right)) dcmotor_drive(0, 0);

```

Программа для макета

```

// Kodulabori mootori juhtimise n̄disprogramm

#include <homelab/delay.h>

#include <homelab/module/motors.h>

#include <stdio.h>

#include <homelab/module/lcd_gfx.h>

#include <homelab/adc.h>

/////////[--Podprogramma v̄voda na display--]////////

void display_time(int seconds, int delay)

{

    char text[16];

    char text1[16];

    char text2[16];

    lcd_gfx_init();

    lcd_gfx_clear();

    lcd_gfx_backlight(true);

    lcd_gfx_goto_char_xy(1, 1);

    lcd_gfx_write_string("Timer");

    //lcd_gfx_goto_char_xy(1, 5);

    //lcd_gfx_write_string("Button1 = every 10s");

    while(1)

    {

        sprintf(text, "%02d:%02d:%02d",

                (seconds / 3600) % 24,

                (seconds / 60) % 60,

                seconds % 60);

        lcd_gfx_write_string(text);

        delay_time(delay);

    }

}

```



```

        lcd_gfx_goto_char_xy(3, 3);

        lcd_gfx_write_string(text);

        hw_delay_ms(delay);

    }

    ////////////

int main(void)
{
    int seconds = 0;

    pin_setup_output(led_red);          //|
    pin_setup_output(led_yellow); //|Podkluchenie diodov
    pin_setup_output(led_green);       //|

    pin_set(LED1);                      //|
    pin_set(LED2);                      //|pogashenie diodov
    pin_set(LED3);                      //|

    adc_init(ADC_REF_AVCC, ADC_PRESCALE_8); //inic
    dcmotor_drive_pwm_init(1, TIMER_NO_PRESCALE);

    pin_setup_input(S1);                //|Podkluchenie knopok
    pin_setup_input(S2);                //|

    while(true)
    {

        display_time(seconds++,1000);
    }
}

```

```
if (seconds%10/10 == seconds%10%10) //
{
    dcmotor_drive_pwm(1, 1, 255);

    hw_delay_ms(750);

    dcmotor_drive_pwm(1, 1, 0);

                                display_time(seconds++,250);

    dcmotor_drive_pwm(1, 0, 255);

                                hw_delay_ms(750);

    dcmotor_drive_pwm(1, 0, 0);

                                display_time(seconds++250);

}

}

}
```

- **Сборка системы с полной инструкцией - документация**

Пользователю нужно установить “шкаф” в подходящее место (где так же присутствует миска), в котором уже установлена электрическая часть (для простоты установки пользователям), выбрать и поставить нужный контейнер для корма (величина зависит от предпочтений пользователя) и засыпать туда корм. Остальное что требуется - это подключить устройство в сеть и настроить время с количеством дозировки пищи.

- **Финансовые затраты на проект**

Стоимость сборки конструкции

Компонент	Марка	Кол-во	Стоимость	Цена без покупки комплектов
Бутылка	-	1	2	
Kodulabor дополнительный комплект(есть в коледже)	robolabor	1	425	
Базовый комплект(есть в коледже)	robolabor	1	300	
Двигатель	Gearmotor 6Vdc 25RPM	1		
Микроконтроллер	ATxmega128A1U v1.x	1		
Адаптер на 220В	robolabor	1		
Моторный модуль	robolabor	1		
Интерфейсный модуль	robolabor	1		
Реечная зубчатая передача	-	2	2,30	
Лампа светодиодная	LEEK LE010501-0018	1	1,40	
Проводники, крепления	-	-	5	
Стальная пластина (корпус, крышка)	-	1	20	
Изготовление остальных деталей	-	-	30	
Все остальные изготовительные работы	-	-	40	
			825.70	100,7

- **Проектная деятельность(временные рамки проекта, распределение обязанностей в группе, вклад каждого в проект)**

Временные рамки: 3 недели

Основное распределение обязанностей:1 человек - писал код и составлял алгоритм, 2-ой делал наброски и чертежи, 3-ий делал оценочную стоимость и остальную организационную деятельность. (Большая часть работы делалась коллективным трудом).

- **Выводы и заключения(необходимость проекта, трудности, результаты)**

Необходимость - “кормушку” можно ставить в зоопарках\приютах и на собственных территориях, что бы избавить себя от необходимости постоянно следить за питомцем.

Трудности - основные трудности были в компоновании различных частей одной машины для эффективной работы.

В итоге получилась стабильная и защищенная система.

- **Ссылки на использованный материал**

Бутылка <http://allforwater.ru/cat.php?r=2&t=16>

Мотор http://home.roboticlab.eu/media/kit/datasheets/actuators/dc_gearmotor_6v_25rpm.pdf

Моторный модуль http://home.roboticlab.eu/en/hardware/homelab/motor/v51?s=translation_true

Микроконтроллер <http://home.roboticlab.eu/et/hardware/homelab/controller/xmega128>

Интерфейсная плата http://home.roboticlab.eu/et/hardware/homelab/digi/v3_1

Kodulabor komplektid

http://www.innovatsioonikeskus.ee/sites/default/files/Jaak&robot/Rain_Ellermaa.pdf

Реечная передача http://arduino-ua.com/prod609-Reechnaya_zybchataya_peredacha