

УДК: 631.171.519.685

**ИНТЕЛЕКТУАЛЬНАЯ СИСТЕМА УПРАВЛЕНИЯ И ОБРАБОТКА  
ИНФОРМАЦИИ ТЕМПЕРАТУРЫ ОБЪЕКТА****Раджабова Махфуза Азимовна - доцент***Ташкентский государственный технический университет*

**Аннотация.** В данной статье были рассмотрены проблемные вопросы сельского хозяйства и были изучены методы сушки фруктов. В работе изучено недостатки технологического процесса о сушке фрукт. А так же было разработана автоматизированная система управления для сушки фрукт. В работе было составлена функционально-технологическая схема управления объектом и составлена программное обеспечения технологического процесса для функционирования объекта.

**Ключевые слова:** датчик, контроллер, сервомотор, инфракрасное излучения, обратная связь, температура, влажность, реле модуль, программное обеспечение.

**Введение.** В производстве многих пищевых продуктов сушка, как правило, является обязательной операцией и представляет собой достаточно энергоемкую технологическую стадию процесса. От аппаратурно-технологического оформления и режима сушки зависит в большой степени качество продукта. Сушке может предшествовать удаление влаги из материалов другими методами, например отжимом на прессах, центрифугированием. Однако механическим способом может быть удалена только часть свободной влаги. Сушкой называют процесс удаления влаги из твердых влажных, пастообразных или жидких материалов (суспензий) путем ее испарения и отвода образовавшихся паров. Это сложный тепломассообменный процесс [1]. Скорость его во многих случаях определяется скоростью внутридиффузионного переноса влаги в твердом теле. Сушке подвергают пищевые материалы, находящиеся в различном агрегатном состоянии, а именно: гранулированные, формованные и зернистые материалы; пастообразные материалы; растворы и суспензии. Выбор метода сушки и типа сушилки осуществляется на основе комплексного анализа свойств пищевых материалов как объектов сушки [2].

**Постановка задач.** Сушка дает возможность довести содержание воды в продукте до такого количества, при котором микроорганизмы уже не могут развиваться. Например, для развития бактерий требуется не менее 30, а плесеней -1.5% влаги. Если удалить в таком количестве воду, то питание микробов станет невозможным, несмотря на обильное количество пищевых веществ, микробы в процессе обмена всасывают через свою поверхность растворенные в воде пищевые вещества [3]. Поэтому при высушивании овощей в них оставляют до 14% влаги, а плодов - до 18% и даже до 25%. около 40°C. **Инфракрасная сушка.** Наиболее актуальной и перспективной в данный момент является сушка продуктов питания с применением инфракрасного излучения. Инфракрасное излучение твердых тел

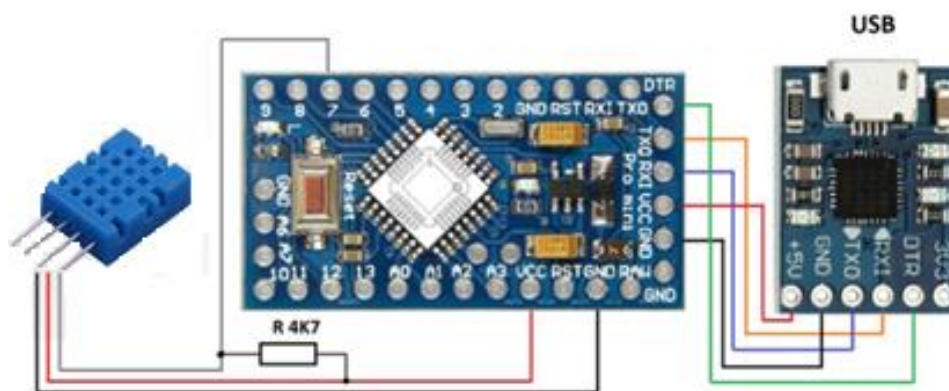
обусловлено возбуждением молекул и атомов тела вследствие их теплового движения. При поглощении инфракрасного излучения облучаемым телом в нем увеличивается тепловое движение атомов и молекул, что вызывает его нагревание. Перенос энергии происходит от тела с большим потенциалом переноса тепла к телу с меньшим потенциалом [3]. Оборудование для сушки фруктов как и все сушильное оборудование, применяемое при этом виде сушки овощей и фруктов и других продуктов, обладает следующими достоинствами: самое низкое удельное энергопотребление на 1 кг испаренной влаги; менее 1 кВт.ч/кг (в два раза меньше любых сушильных установок); сушка продуктов производится при низкой температуре – 50 - 60 градусов цельсия; сушка продуктов производится с высокой скоростью – 30 - 200 мин; простота и надежность, низкая цена и высокая окупаемость [4]. **Микроволновая сушка.** Микроволновая технология - серьезное достижение науки и техники, продукт десятилетних исследований ученых-аграриев и военно-промышленного комплекса, не имеющая аналогов в мировой практике [5]. Микроволновая технология и созданное на ее основе микроволновое оборудование для сушки фруктов, оборудование для сушки овощей позволяет не только высушивать продукцию, но и получать пищевые красители, размораживать рыбу, мясо, овощи, ягоды и другие продукты питания, проводить бестемпературное консервирование и многое другое [6]. **Сублимационная сушка.** Сублимационная сушка продуктов (сублимационная вакуумная сушка, также известная как лиофилизация или возгонка) - это удаление влаги из свежемороженых продуктов в условиях вакуума. В настоящее время этот метод сушки продуктов является наиболее совершенным, но в то же время и наиболее дорогостоящим.

**Акустическая сушка.** Акустический метод сушки продуктов основан на воздействии на обезвоживаемый продукт интенсивных ультразвуковых волн. Данный процесс сушки носит циклический характер, волна выбивает влагу, находящуюся на поверхности продукта, затем оставшаяся влага равномерно распределяется по капиллярам и процесс повторяется снова [7]. Сушка продуктов акустическим способом отличается от обычных методов и по скорости выработки сухопродукта. Например, при сушке ферментов (разрушающихся при температуре в 40 градусов Цельсия) в акустическом поле скорость сушки продуктов в сравнении с вакуумным методом повышается в 3-4 раза [8]. **Кондуктивная сушка.** Кондуктивный способ сушки пищевых продуктов основывается на передаче тепла высушиваемому продукту путем непосредственного контакта с нагреваемой поверхностью сушильного оборудования. Для сушки продуктов питания этот способ используется не часто.

**Решение задач.** Для осуществления и решения выше указанных проблем была создана функционально-технологическая схема и программное обеспечение. Объект состоит из стеклянной теплицы и технических средств автоматики. При естественной нагреве стеклянный корпус увеличивает радиацию солнца в 2 раза и повышает эффективность нагрева и сушки фрукта. Датчики температуры и влажности

контролируют за состоянием объекта верхнего и нижнего уровня. Сервомотор за определённое время вращается под углом  $30^\circ$ . Объект оснащён с двух сторон обдувными устройствами тёплого воздуха. Микроконтроллеру прошиваем программу и обеспечиваем без перебойную и устойчивую работу в технологическом процессе.

Контур измерения температуры состоит из первичного цифрового измерительного преобразователя - DS18B20 (Maxim Integrated), управление которым, по интерфейсу 1-wire, осуществляет микропроцессорный контроллер - Atmega328 (5V 16MHz (Microchip) на платформе Arduino Pro Mini. Выходной сигнал измерительной информации с DS18B20 по интерфейсу 1-wire поступает на дискретный вход микроконтроллера, обрабатывается, преобразуется в строку ASCII символов измеренных значений температуры в диапазоне от  $-55$  до  $+125$  °C и, по стандартному последовательному интерфейсу, через TTL-USB преобразователь, поступает в компьютер для дальнейшей программной обработки:



**Рис. 1. Схема подключения датчика температуры DS18B20**

Программное обеспечение составлено с использованием библиотеки OneWire, которую можно подключить через менеджера библиотек в интегрированной среде разработки Arduino Software:

```
#include <OneWire.h>
// OneWire DS18B20 Temperature Example
// http://www.pjrc.com/teensy/td_libs_OneWire.html
// The DallasTemperature library can do all this work for you!
// http://milesburton.com/Dallas_Temperature_Control_Library
//Вывод Data преобразователя DS18B20 подключаем к 7-му выводу //платформы
Arduino
OneWire ds(7);
void setup(void) {
  Serial.begin(9600);
}
void loop(void) {
  byte i;
```



```

byte present = 0;
byte type_s = 0;
byte data[12];
byte addr[8];
float celsius;
ds.search(addr);
ds.reset();
ds.select(addr);
ds.write(0x44, 1);
delay(800);
present = ds.reset();
ds.select(addr);
ds.write(0xBE); // Read Scratchpad
for ( i = 0; i < 9; i++) {
  data[i] = ds.read();
}
int16_t raw = (data[1] << 8) | data[0];
celsius = (float)raw / 16.0;
Serial.println(celsius);
}

```

Python интерфейс для обработки “кривой разгона” и её графической реализации. Основные задачи, решаемые подсистемой обработки и представления измерительной информации оператору:

```

1. import numpy as np
2. import
matplotlib.pyplot as plt
3. import serial
4. from drawnow import
drawnow
5. import datetime, time
6. #Вывод выборки в
графическое окно
7. def cur_graf():
8. plt.title("DS18B20")
9. plt.ylim( 20, 40 )
10. plt.plot(nw, lw1, "r.-")
11. plt.ylabel(r'$температу
ра, \degree C$')
12. plt.xlabel(r'$номер \

```

```

56. count = in_file.read()
57. count_v = eval(count) + 1
58. in_file.close()
59. in_file = open(filename,"w")
60. count = str(count_v)
61. in_file.write(count)
62. in_file.close()
63. filename = count + '_' + filename
64. out_file = open(filename,"w")
65. #Вывод информации для оператора на консоль
66. print("\n параметры:\n")
67. print("n - номер измерения;")
68. print("T - температура, град. С;")
69. print("\n измеряемые значения величины
температуры\n")
70. print("{0}{1}\n".format('n'.rjust(4),'T'.rjust(10)))

```

<pre> измерения\$) 13. plt.grid(True) 14. #вывод всех списков в графическое окно 15. def all_graf(): 16. plt.close() 17. plt.figure() 18. plt.title("DS18B20\n" + 19. str(count_v) + "-й эксперимент " + 20. "(" + now.strftime("%d- %m-%Y %H:%M") + ")") 21. plt.plot( n, l1, "r-") 22. plt.ylabel(r'\$температу ра, \degree C\$') 23. plt.xlabel(r'\$номер \ измерения\$' + 24. ';' (период опроса датчика: {:.6f}, c').format(Ts)) 25. plt.grid(True) 26. plt.show() 27. #определяем количество измерений 28. # общее количество измерений 29. str_m = input("введите количество измерений: ") 30. m = eval(str_m) 31. # количество элементов выборки 32. mw = 16 33. #настроить параметры последовательного порта 34. ser = serial.Serial() 35. ser.baudrate = 9600 36. port_num = input("введите номер последовательного порта: ") 37. ser.port = 'COM' + port_num 38. ser 39. #открыть </pre>	<pre> 71. #считывание данных из последовательного порта 72. #накопление списков 73. #формирование текущей выборки 74. #вывод значений текущей выборки в графическое окно 75. i = 0 76. while i &lt; m: 77. n.append(i) 78. nw.append(n[i]) 79. if i &gt;= mw: 80. nw.pop(0) 81. line1 = ser.readline().decode('utf-8')[:-2] 82. t1.append(time.time()) 83. if line1: 84. l1.append(eval(line1)) 85. lw1.append(l1[i]) 86. if i &gt;= mw: 87. lw1.pop(0) 88. print('{0:4d} {1:10.2f}'.format(n[i],l1[i])) 89. drawnow(cur_graf) 90. i += 1 91. #закрывать последовательный порт 92. ser.close() 93. ser.is_open 94. #time_tm -= time_t0 95. time_tm = t1[m - 1] - t1[0] 96. print("\n продолжительность времени измерений: {0:.3f}, c".format(time_tm)) 97. Ts = time_tm / (m - 1) 98. print("\n период опроса датчика: {0:.6f}, c".format(Ts)) 99. #запись таблицы в файл 100. print("\n таблица находится в файле {}\n".format(filename)) 101. for i in np.arange(0,len(n),1): 102. count = str(n[i]) + "\t" + str(l1[i]) + "\n" 103. out_file.write(count) 104. #закрывать файл с таблицей 105. out_file.close() 106. out_file.closed 107. #получить дату и время </pre>
---	--

<p>последовательный порт</p> <p>40. <b>try:</b></p> <p>41. ser.open()</p> <p>42. ser.is_open</p> <p>43. print("соединились с: " + ser.portstr)</p> <p>44. <b>except</b></p> <p>serial.SerialException:</p> <p>45. print("нет соединения с портом: " + ser.portstr)</p> <p>46. <b>raise</b> SystemExit(1)</p> <p>47. #определяем списки</p> <p>48. l1 = [] # для значений температуры</p> <p>49. t1 = []</p> <p>50. lw1 = [] # для значений выборки температуры</p> <p>51. n = [] # для значений моментов времени</p> <p>52. nw = [] # для значений выборки моментов времени</p> <p>53. #подготовить файлы на диске для записи</p> <p>54. filename = 'count.txt'</p> <p>55. in_file = open(filename, "r")</p>	<p>108. now = datetime.datetime.now()</p> <p>109. #вывести график в графическое окно</p> <p>110. all_graf()</p> <p>111. end = input("\n нажмите Ctrl-C, чтобы выйти ")</p>
--	--

n – номер измерения;

T – температура, град. С;

измеряемые значения величины температуры

n T

0 24.75

1 24.75

2 24.75

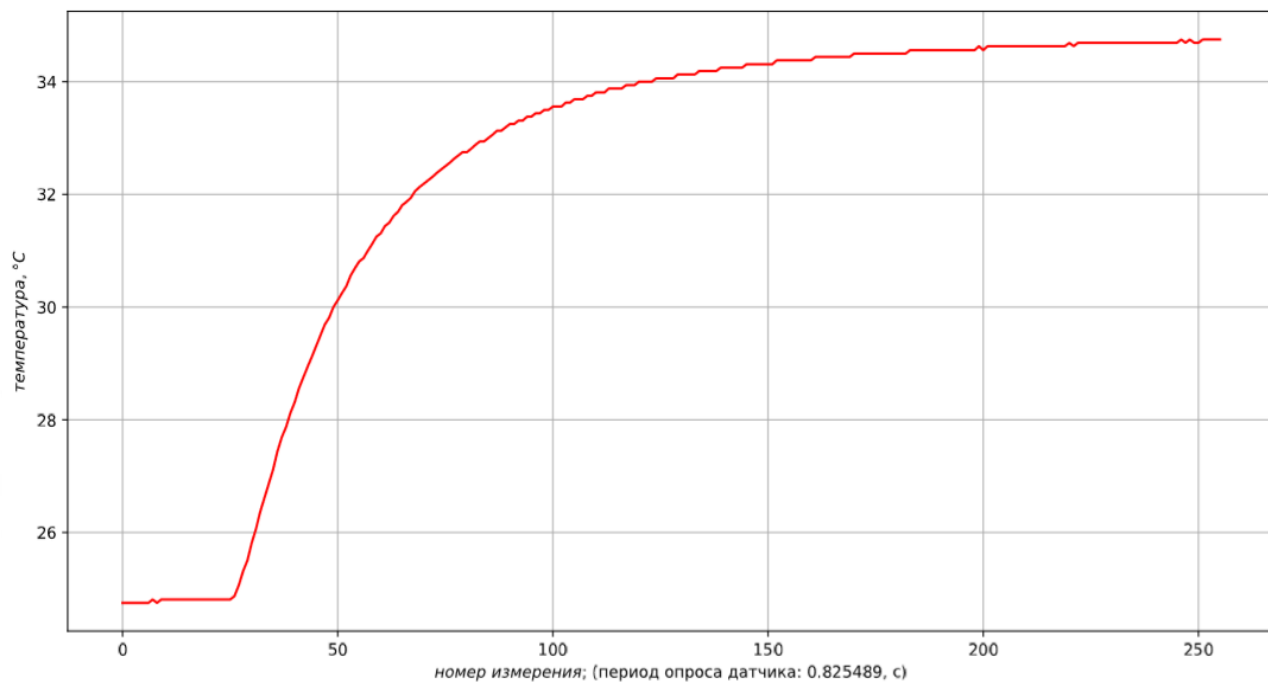
3 24.75

4 24.75

5 24.75

6 24.75

.....



**Рис.2. Python интерфейс для обработки “кривой разгона” и её графической реализации.**

Python интерфейс для получения передаточной функции и оценки адекватности модели. Для решения этой задачи можно использовать численные методы, поскольку модель не содержит дифференцирования, а предложенный метод решения согласно соотношениям не предполагает смены координаты времени. Кроме этого применим интерполяцию кубическим сплайном в соответствии со следующим листингом определяем передаточную функцию объекта по каналу регулирования температуры.



<pre># -*- coding: utf8 -*- import matplotlib.pyplot as plt import time start = time.time() from scipy.interpolate import splev, splrep import scipy.integrate as spint import numpy as np from scipy.integrate import quad xx =np.array(np.arang e(0,230,1)) yy1 =np.array([ 24.87, 25.06, 25.31, 25.5, 25.81, 26.06, 26.37, 26.62, 26.87, 27.12, 27.44, 27.69, 27.87, 28.12, 28.31, 28.56, 28.75, 28.94, 29.12, 29.31, 29.5, 29.69, 29.81, 30.0, 30.12, 30.25, 30.37, 30.56, 30.69, 30.81, 30.87, 31.0, 31.12, 31.25, 31.31, 31.44, 31.5, 31.62, 31.69, 31.81, 31.87, 31.94, 32.06,])</pre>	<pre>yy2=yy1-24.87#компенсация смещения нуля yy=yy2/max(yy2)#нормирование """ Интерполяция переходной характеристики при помощи сплайнов """ def h(x): spl = splrep(xx , yy ) return splev(x, spl) """ Численное интегрирование без смены координаты времени в соответствии с (6) """ S1=(spint.quad(lambda x:1-h(x),xx[0],xx[len(xx)-1])[0]) S2=(spint.quad(lambda x:(1-h(x)) * (S1-x),xx[0],xx[len(xx)-1])[0]) S3=(spint.quad(lambda x:(1-h(x)) * (S2- S1 * x+(1/2) * x ** 2),xx[0],xx[len(xx)-1])[0]) S4=(spint.quad(lambda x:(1-h(x)) * (S3-S2 * x+S1 * (1/2) * x ** 2- (1/6) * x ** 3),xx[0],xx[len(xx)-1])[0]) """ Определение коэффициентов передаточной функции """ b1=-S4/S3 a1=b1+S1 a2=b1 * S1+S2 a3=b1 * S2+S3 """ Возврат во временную область """ def ff(x,t): j=(-1) ** 0.5 return (2/np.pi) * ( (b1 * x ** j+1) * np.e ** (- 25 * x)/(a3 * (x * j) ** 3+a2 * (x * j) ** 2+a1 * x * j+1)).real) * (np.sin(x * t)/x) y=np.array([round(quad(lambda x: ff(x,t),0, 0.6)[0],2) for t in xx]) """ Определение критерия адекватности модели """ k=round(1-sum([(yy[i]-y[i]) ** 2 for i in np.arange(0,len(yy)- 1,1)])/sum([(yy[i]) ** 2 for i in np.arange(0,len(yy)-1,1)]),5) stop = time.time() print ("Время работы программы :",round(stop-start,3)) plt.title('Идентификация модифицированным методом площадей.\n Адекватность модели: %s'%k) plt.plot(xx, yy,label='W=(%s * p+1)/(%s * p ** 3+%s * p ** 2+%s * p+1)'%(round(b1,1),ro und(a3,1),round(a2,1),round(a1,1))) plt.legend(loc='best') plt.grid(True) plt.show()</pre>
---	--

Время работы программы: 1.144 Получена передаточная функция объекта (ёмкость с нагревателем) при адекватности к “кривой разгона”- 0,97.



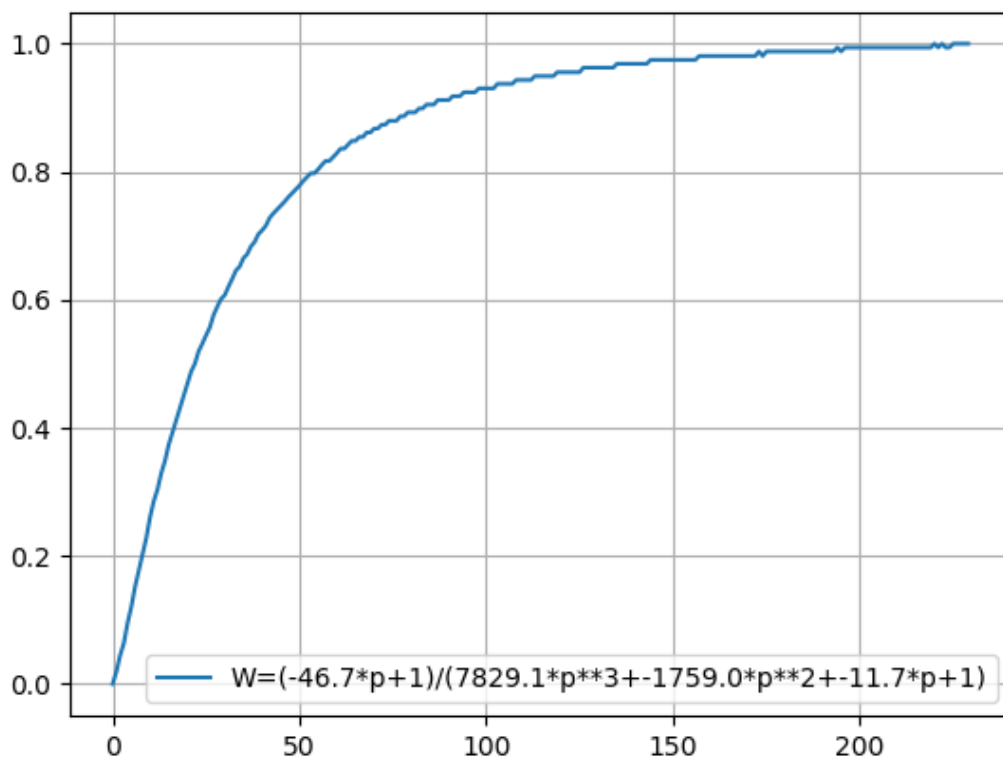


Рис. 3. Передаточной функции объекта по каналу регулирования температуры

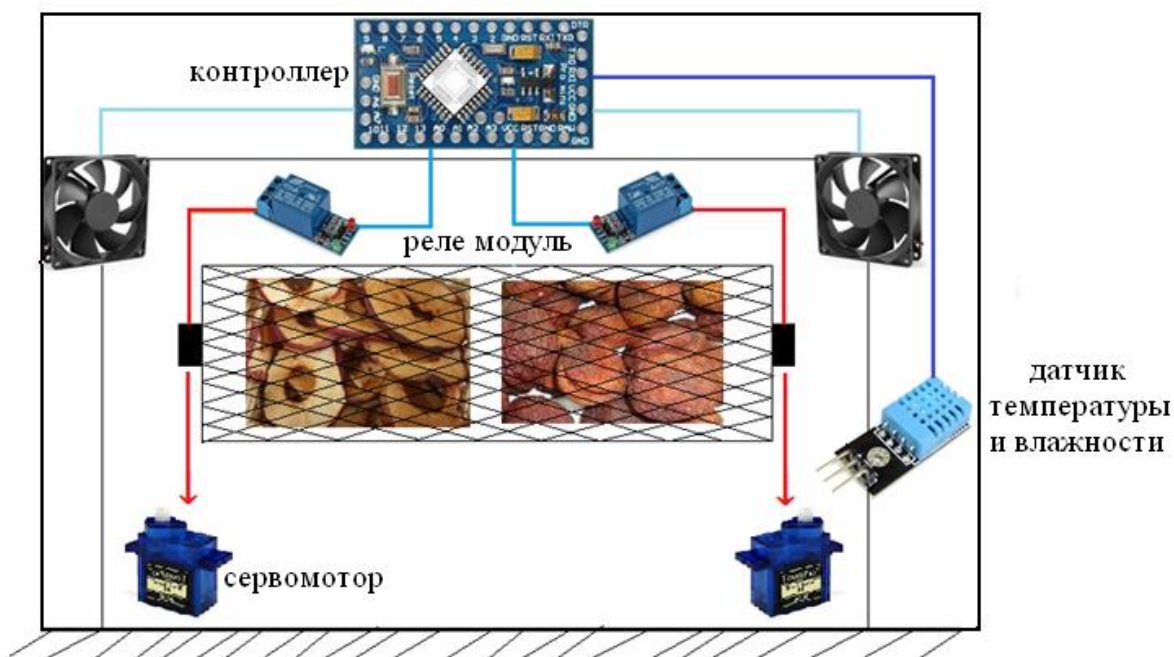


Рис. 4. Функционально-технологическая схема сушки фруктов

**Вывод.** Разработан и испытан контроллер Arduino с датчиком температуры и Python интерфейсом для динамической идентификации объектов управления по каналу регулирования температуры. Для решение некоторых вопросов сельской местности, а именно решение сушки фруктов были изучены. Данная функционально-технологическая схема и программное обеспечения повышает устойчивость и

надёжность системы. Повышает рентабельность экономической эффективности и за короткие сроки оправдывает себестоимость системы управления.

**ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА:**

1. М.З.Ганкин, Комплексная автоматизация и АСУТП водохозяйственных систем. 1991г.
2. Автоматизация технологических процессов., И.Ф.Бородин., Ю.А.Судник., Москва 2004г.
3. Котюк А.Ф. Датчики в современных измерениях.2006г.
4. Мир электроники.. , Джексон Р.Г. Москва 2007г.337с.
5. А.Д.Чудаков, Б.В.Шандров., Технические средства автоматизации. Москва 2007г. 102 с.
6. Водовозов А.М. Цифровые элементы в системе автоматики. ВГТУ-Вологда-2005г. 290с.
7. Крухмалев В.В., Гордиенко В.Н., Моченов А.Д. Цифровые системы передачи. Наука-М- 2007г. 277с.
8. ЛаврентьевБ.Ф., Схематехника электронных средств Москва 2010г. 278с.