DBpia

논문 19-45-01-11 The Journal of Korean Institute of Communications and Information Sciences '20-01 Vol.45 No.01
https://doi.org/10.7840/kics.2020.45.1.89

계사 내 이산화탄소 기반의 자동 환기 시스템에 관한 설계

이 병 선◆, 조 주 필°

A Design on CO_2 Based Automatic Ventilation System in Poultry Farm

Byeongseon Lee◆, Juphil Cho°

요 약

본 논문은 국내 계사에서의 효율적인 대기 환경 관리를 위하여 이산화탄소량에 따라 작동하는 자동 환기 시스템을 제안하였다. 국내 양계산업은 육계와 산란계 농장으로 구분되어 있으며, 환기제어 시스템은 미국의 Chore-time, 글로벌회사인 Munters 제품이 90%이상 설치되어있다. 이 회사들의 환기제어 방식은 정해진 절차, 정해진 시간에 따라 Timer 방식으로 온도 자료에 근간하여 환기제어를 한다. 즉 계사 내부의 온도가 높은 경우 이를 낮추기 위해 환기팬이 구동이 되는 방식이다. 하지만 본 논문에서는 온도 외의 습도, 이산화탄소량도 성장률에 크게 영향을 미치는 점을 착안하여 이산화탄소량에 따라 환기제어가 자동으로 이뤄지는 시스템을 Lab-scale 기반에서 설계하였다. 이 설계가 상용화 될 경우 기존 온도 기반의 환기제어에서 이산화탄소 농도가 추가적으로 제어 가능하기 때문에 닭의 육성률을 크게 신장시킬 수 있을 것이다.

키워드 : 이산화탄소, 환기 시스템, 계사, 육성률, 자동화
Key Words : CO_2, ventilation system, poultry farm, incubation ratem, automation

ABSTRACT

This paper proposed an automatic ventilation system that operates according to the volume of carbon dioxide for efficient management of the atmosphere environment in domestic poultry farms. More than 90 percent of the U. S. company Chore-time and global company Munters products are installed in the ventilation and control systems of domestic companies. The ventilation control methods of these companies control ventilation based on temperature data based on the Timer method according to the prescribed procedure and time. In other words, the ventilation fan is operated to reduce the temperature inside the relay. However, in this paper, the system in which ventilation control is automatically performed according to the volume of carbon dioxide was designed based on Lab-scale, based on the fact that humidity outside the temperature and CO2 volume also greatly affect the growth rate. If this design is commercialized, it can greatly increase the incubation rate of chickens because the concentration of carbon dioxide in the existing temperature-based ventilation control is additionally be controlled.

◆ First Author : Department of Integrated IT & Communication Engineering, Kunsan National University, bslee@demotion.co.kr, 학생회원
° Corresponding Author : Department of Integrated IT & Communication Engineering, Kunsan National University, stefano@kunsan.ac.kr, 정회원
논문번호 : 201911-320-C-RU, Received November 22, 2019; Revised December 3, 2019; Accepted December 3, 2019

www.dbpia.co.kr

Ⅰ. 서 론

대규모의 계사는 사람의 힘만으로는 운영되기 힘들기 때문에 자동 사육 시스템의 필요성이 크게 증가한다. 그래서 이미 대규모 계사에는 급수기와 급이기는 물론이고, 조명 조절이나 환기 또한 자동으로 할 수 있도록 제어해주는 시스템이 모두 설치되어 있다. 현재 우리나라 육계는 표 1의 데이터에서 나타나듯이 90%이상이 3만수 이상의 사육규모를 가진 농장에서 사육된다.

하지만 현재 이러한 자동화 시스템들이 존재함에도 불구하고, 제한적인 조건에서만 자동적으로 제어가 되고, 각각의 시스템들이 분리되어 있기 때문에 여전히 계사에서는 사람이 신경 써야 할 부분이 많이 존재한다. 때문에 우리는 계사 내 사육 전자동 시스템 구축을 위해 우선적으로 필요한 것이 무엇인지 고민했다.

닭은 온도, 습도, 이산화탄소 및 암모니아 등 계사 내 대기환경에 굉장히 민감한 동물이다. 계사 내 사육 전자동 시스템의 구성을 위해서는 이러한 대기환경을 항상 적절하게 맞춰줘야 한다. 하지만 현재의 자동 환기 시스템은 시간에 따라 주기적으로 시행하거나, 온도, 습도가 상승했을 때에만 시행되도록 시스템 되어 있다. 이러한 자동 환기 시스템으로는 계사 내 대기환경을 적절하게 맞추기가 어렵다.

이러한 문제점을 해결하기 위해서 본 논문에서는 실험을 통해 간이 계사를 구성하여 온·습도뿐만 아니라 이산화탄소까지 자동적으로 제어할 수 있는 자동 환기 시스템을 구현했으며 모의실험 결과 육계 계사 내에서 작동하는 대기 내 이산화탄소 농도를 제어할 수 있는 자동 환기 시스템을 실험실 단계에서 설계하고 그 성능을 확인하였다. 경제적인 이유로 기존의 온도 및 시간 기반의 자동 환기 시스템에 이산화탄소 제어만을 추가하였지만, 추가적으로 암모니아 등 닭의 생장과 질병에 영향을 주는 대기 요소를 제어할 수 있도록 발전이 가능하다.

Ⅱ. 본 론

2.1 설계 과정

표 1과 같이 국내에서 90%이상이 3만수 이상의 사육규모를 가진 농장에서 사육되고 있음을 알 수 있다.

성능 검증 시스템 구축을 위해 먼저 아크릴로 구성된 실험박스를 제작하였다. 실험박스의 크기는 가능한 실제 계사와 비슷한 결과 값을 얻기 위하여, 실제 계사의 크기에 비율에 맞춰서 (150cm * 18cm * 30cm)

표 1. 국내 육계 계사 수.
Table 1. Number of domestic chicken farms.

Breeding Scale	As of April 4, 2018	
	Farm	Number of poultry
Sum	1,507	85,914,577
Less than 10000	48	248,051
10,000~30,000	219	4,700,095
30,000~50,000	484	19,053,936
More than 50,000	756	61,912,495

의 크기로 제작했으며 설계도는 그림 1과 같다.

실제 계사 크기에 근거하여 축소시킨 모델에 환기를 위한 팬을 각각 실험박스의 좌, 우측에 하나씩 그리고 정면과 후면에 네 개씩 설치하였다. 완성된 설계 박스의 사진은 그림 2와 같다.

두 번째로는 온도, 습도, 이산화탄소를 측정하기 위해 각각의 센서와 아두이노 보드를 이용해 그림 3과

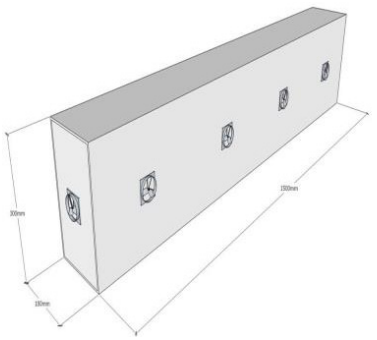

그림 1. 실험 박스 설계도.
Fig. 1. Design of simulation box.

그림 2. 실제 실험 박스.
Fig. 2. Real box for implementation.

www.dbpia.co.kr

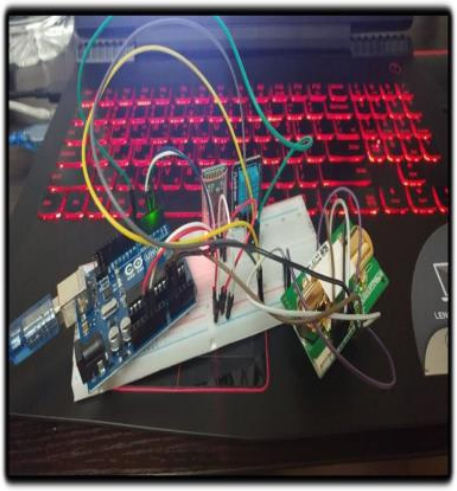

그림 3. 센서 기구부.
Fig. 3. Sensor parts.

같은 간이 센서 기구부를 제작하였다. 실제 계사는 약 (150m * 18m) 정도로 굉장히 커서 각각 구역의 대기 값이 다르게 측정되기 때문에 센서 기구부를 총 3개 제작하여, 실험박스에 설치해 각 구역마다 다른 값을 얻을 수 있도록 했다.

세 번째로는 Android Studio를 이용하여 전체 시스템을 조작할 수 있는 App을 그림 4와 같이 제작하였다. 이 App에는 온도와 습도, 이산화탄소 농도의 임계값을 설정하고, 각각의 환기팬을 수동조작 할 수 있는 기능을 설계하였다.

App은 각각 센서 기구부의 블루투스 모듈에 페어링을 한 후에 시간에 따라 측정데이터를 주기적으로 받을 수 있도록 하였다. 또한 받아온 데이터를 기반으로 자동으로 환기를 시행여부를 결정하는 알고리즘을

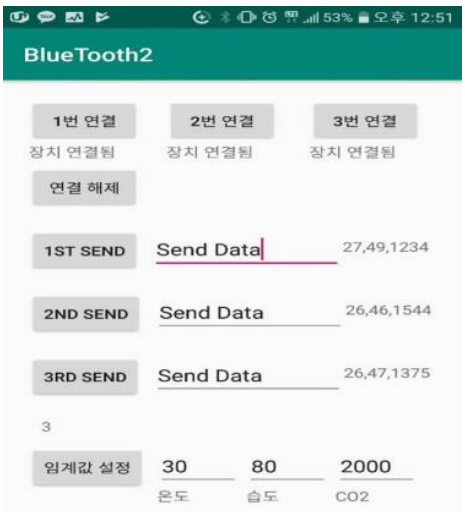

그림 4. 통합제어 App 인터페이스.
Fig. 4. Interface of total control app.

```
mBluetoothHandler = new Handler(){
   public void handleMessage(android.os.Message
msg){
      if(msg.what == BT_MESSAGE_READ){
         String readMessage = null;
         try {
            readMessage = new String((byte[]) msg.obj,
"UTF-8");
         } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
         }
         StringTokenizer st = new
StringTokenizer(readMessage, ",");
         int i = 0;
         while (st.hasMoreTokens()){
            bt_data[msg.arg2-1][i] =
Float.parseFloat(st.nextToken());
            i++;
            if(i >= 3) break;
         }

mTvReceiveData[msg.arg2-1].setText(readMessage);
      }
   }
};
```

그림 5. 데이터 수신과 알고리즘 소스 1.
Fig. 5. Data reception and Algorithm source 1.

```
public void run() {
   byte[] buffer = new byte[1024];
   int bytes;
   while (true) {
      try {
         bytes = this.mmInStream.available();
         if (bytes != 0) {
            SystemClock.sleep(100);
            bytes = mmInStream.available();
            for(int i = 0 ; i < buffer.length ; i++) buffer[i]
= 0x01;
            bytes = mmInStream.read(buffer, 0, bytes);
            mBluetoothHandler.obtainMessage
               (BT_MESSAGE_READ, bytes, myNum,
buffer)
               .sendToTarget();
         }
      } catch (IOException e) {break;}
   }
}
```

그림 6. 데이터 수신과 알고리즘 소스 2.
Fig. 6. Data reception and Algorithm source 2.

www.dbpia.co.kr

설계하였고, 알고리즘의 구체적인 내용은 그림 5, 6과 같다.

네 번째로는 C#으로 측정데이터를 MongoDB에 적재하는 프로그램을 제작하였다. 측정데이터를 지속적으로 데이터베이스에 적재하면 차후에 빅데이터 분석을 통한 질병예찰 또는 환경분석에 사용이 가능하기 때문에 관계형 데이터 베이스가 아닌 NoSql인 MongoDB에 적재할 수 있도록 구성했다. 추가적으로 실시간으로 데이터를 확인할 수 있는 프로그램을 제작하였다. PHP를 통해 제작하여 웹으로 확인할 수 있으며, 각 기구부의 데이터의 실시간 데이터와 1분간의 평균데이터를 각각 그림 7과 그림 8로 동시에 표시한다. Ajax를 통해 MongoDB 데이터를 주기적으로 호출하여 가져오며, 이 데이터를 jQuery를 통해 표와 그래프에 표시하게 되는 방식으로 실시간으로 계속 데이터를 확인할 수 있도록 하였다. jQuery와 데이터 호출 코드는 각각 그림 9, 10과 같다.

2.2 알고리즘 설계

알고리즘은 Android Studio를 이용하여 설계되었으며, 그림 11은 데이터의 측정부터 환기팬이 작동되는 과정을 그린 데이터 전송 블록도이다.

그림 7. 웹페이지 기구부 데이터.
Fig. 7. Data of Web page for sensing.

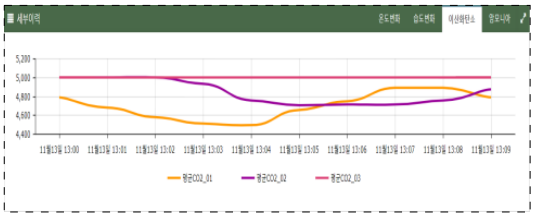

그림 8. 웹페이지 평균값 그래프.
Fig. 8. Web page graph of average value.

```
function getData(){
   var dataArr={}; dataArr['oper']="getData";
                        중간 생략
                    case '1':
                            $('#data_temp_1').text(rv_temp
+ "℃");
$('#data_humi_1').text(rv_humi + "%");
                    $('#data_co2_1').text(rv_co2 + "ppm");
                    $('#data_nh3_1').text(rv_nh3 + "ppm");
                            break;
                        case '2':

$('#data_temp_2').text(rv_temp + "℃");

$('#data_humi_2').text(rv_humi + "%");
                                $('#data_co2_2').text(rv_co2
+ "ppm");
                                $('#data_nh3_2').text(rv_nh3
+ "ppm");
                                break;
                            case '3':
                $('#data_temp_3').text(rv_temp + "℃");

$('#data_humi_3').text(rv_humi + "%");
                                $('#data_co2_3').text(rv_co2
+ "ppm");
                                $('#data_nh3_3').text(rv_nh3
+ "ppm");
                                break;
                default:
                                alert("오류");
                                break;
                }
              //임계값 초과시 색 변경 및 아이콘 변경
                if(rv_temp 〉 25) set_warning(rv_num, 0,
true);
              else set_warning(rv_num, 0, false);
                if(rv_humi 〉 75) set_warning(rv_num, 1,
true);
              else set_warning(rv_num, 1, false);
                if(rv_co2 〉 2000) set_warning(rv_num, 2,
true);
              else set_warning(rv_num, 2, false);
                if(rv_nh3 〉 5) set_warning(rv_num, 3,
true);
              else set_warning(rv_num, 3, false);
            });
        }
    });
}
```

그림 9. jQuery 코드.
Fig. 9. jQuery code.

먼저 각각의 기구부는 아두이노를 통해 주기적으로 온도, 습도, 이산화탄소 3개의 센서의 데이터를 읽어

www.dbpia.co.kr

```
$mongoResult=$mongoCollection->aggregate([
  [ '$match' => [
     'num' => [ '$gte' => '1', '$lte' => '3' ],
     'time' => [ '$gte' => $sDate, '$lte' => $eDate ]
     ]
  ],
  중간 생략
     'avgHumid_2' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '2' ]] ,'$humid', NULL ] ] ],
     'avgHumid_3' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '3' ]] ,'$humid', NULL ] ] ],
     'avgCO2_1' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '1' ]] ,'$co2', NULL ] ] ],
     'avgCO2_2' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '2' ]] ,'$co2', NULL ] ] ],
     'avgCO2_3' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '3' ]] ,'$co2', NULL ] ] ],
     'avgNH3_1' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '1' ]] ,'$nh3', NULL ] ] ],
     'avgNH3_2' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '2' ]] ,'$nh3', NULL ] ] ],
     'avgNH3_3' => ['$avg' => ['$cond' => [ ['$eq'
=> [ '$num', '3' ]] ,'$nh3', NULL ] ] ]
     ]
  ],
  [ '$sort' => ['_id' => 1] ],
  [ '$limit' => $chartLen ]
]);

$rvData = $mongoResult["result"];
```

그림 10. Ajax를 이용한 데이터 호출 코드.
Fig. 10. Data call code using Ajax.

와서 int값으로 기구부에 저장한다. 그 후에 아두이노는 저장된 값을 string형식으로 변환하고, 반점기호(,)를 구분자로 하여 하나의 데이터로 통합시킨다. 최종적으로 이 데이터를 char[]의 형태로 변환한 다음에 블루투스 모듈을 통해서 통합제어 App데이터를 전송한다. 통합제어 App에서는 그림 11과 같이 블루투스를 통해 받은 데이터를 블루투스 핸들러에서 반점기호를 구분자로 각각의 센서 값으로 토크나이징하여 저장하게 된다. 그 후 이 저장된 데이터를 기반으로 그림 11과 같은 알고리즘을 시행하고, 이 과정을 통해 각각 환기팬의 가동여부를 결정하게 된다. 이렇게 결정된 환기팬 가동 여부는 다시 byte[]로 통합되고, 블루투스를 통하여 각각의 기구부로 전송된다. 마지막으로 기구부에서는 받은 데이터를 토크나이징한 후 그 데이터에 따라 각각의 환기팬을 가동하게 된다.

개발 알고리즘의 주요 동작내용은 다음과 같다. 첫째로 통합제어 App에 저장된 온도 값을 설정된 임계값과 비교한다. 만약 온도 임계값보다 큰 측정값이 있

그림 11. 통합제어 App 데이터 전송 블록도.
Fig. 11. Data transmission block for total control app.

을 경우 해당 값이 측정된 기구부 주변의 환기팬의 가동 여부를 True로 한다. 이 경우 가동되는 환기팬의 개수는 3개 이상이며, 해당 측정값과 임계값의 차이가 클 경우 더 늘어난다. 온도는 표준온도보다 2°C 정도만 높아져도 닭은 쉽게 폐사하기 때문에 많은 환기팬을 가동하여 빠르게 낮출 수 있게 설계하였다.

둘째로 통합제어 App에 저장된 습도 값을 설정된 임계값과 비교한다. 만약 습도 임계값보다 큰 측정값이 있을 경우 해당 값이 측정된 기구부 주변의 환기팬의 가동 여부를 True로 한다. 이 경우 가동되는 환기팬의 개수는 2개 이상이며, 해당 측정값과 임계값의 차이가 클 경우 더 늘어난다. 습도는 적정 습도보다 높아질 경우 닭이 질병에 걸릴 확률을 크게 높이게 된다. 습도가 높은 경우 각종 병원성 미생물의 증식이 가속화되며, 또한 그로 인해서 계사 내 암모니아가 발생되기 때문에 빠르게 낮춰줄 필요가 있다. 하지만 습도가 너무 낮아질 경우 계사 내에서 먼지가 많이 발생하기 때문에 적정량의 환기를 할 수 있도록 설계하였다.

'마지막으로 통합제어 App에 저장된 이산화탄소 값을 설정된 임계값과 비교한다. 만약 이산화탄소 임계값보다 큰 측정값이 있을 경우 해당 값이 측정된 기구부 주변의 환기팬의 가동 여부를 True로 한다. 이 경우 가동되는 환기팬의 개수는 1개 이상이며, 해당 측정값과 임계값의 차이가 클 경우 더 늘어난다. 이산화탄소가 증가할 경우 닭의 생장률에 큰 영향을 준다. 닭의 생장률이 떨어질 경우 추가적인 사료 값이 증가하게 되므로, 환기를 시켜 낮출 수 있도록 하였다. 하

www.dbpia.co.kr

```
int r = 0;
float a,b = 0; int c = 0;
a = random(20);
a = a/100;
b = random(10);
b = b/10;
c = random(10);
for(int i=0; i<3; i++)
{
  r = random(2);
  switch(i)
  {
    case 0:
      if(r == 1) a = -a;
      break;
    case 1:
      if(r == 1) b = -b;
      break;
    case 2:
      if(r == 1) c = -c;
      break;
  }
}
이하 생략
```

그림 12. 데이터 생성 알고리즘
Fig. 12. Data generation algorithm

지만 온도와 습도 측정값이 정상일 경우에는 환기를 많이 시행하게 되면 계사 내 온도와 습도가 낮아지게 된다. 이산화탄소와 달리 온도와 습도는 적정 값을 유지해야 하므로, 여기서는 적은양의 환기를 시행하도록 설계하였다. 특히 겨울철의 경우에는 환기시 난방비가 급격히 증가하게 되므로 적절한 환기 량을 맞춰 줄 필요가 있다.

추가적으로 한 기구부에서 2개 이상의 측정값이 임계값을 넘어선 경우에는 긴급 상황으로 간주하여 다량의 환기를 시행하고, 통합제어 App에 경고표시를 나타내도록 설계하였다.

2.3 실험결과

성능 실험을 위해서 3개의 간이 센서 기구부를 통합제어 App에 페어링하고, 각각 온도, 습도, 이산화탄소를 측정하였다. 여기서 실험을 위해 측정된 한 주기의 데이터는 표 2와 같다.

표 2의 데이터가 통합제어 App에 저장되면, 이 데이터를 기반으로 알고리즘이 수행되고, 그 결과로 환기팬의 동작여부를 결정해 기구부로 전송한다. 최초 실행 시 설정되는 임계값은 그림 3의 하단 부분과 같으며, 온도는 30°C, 습도는 80%, 이산화탄소는 2000ppm이다.

표 2. 실제 측정 데이터.
Table 2. Real sensing data.

측정값	Temp.	Humi.	CO_2
inst. 1	27	49	1234
inst. 2	26	46	1544
inst. 3	26	47	1375
unit	°C	%	ppm

먼저 실험의 결과를 예상한 후 실험을 진행하였다. 여기서 표 2의 측정 데이터는 모두 설정된 임계값 보다 작다. 따라서 통합제어App에서는 알고리즘을 거쳐 환기팬의 가동여부를 모두 False로 설정할 것이고, 환기팬은 모두 가동되지 않을 것으로 예상하였다. 실험결과 환기팬이 모두 작동하지 않는 것을 확인하였고, 예상과 일치하였다.

두 번째로 데이터에 따른 동작 확인을 위하여 표 3의 임의 생성된 데이터로 실험하기로 하였다. 데이터 생성은 아두이노에서 이루어졌으며, 동일하게 블루투스를 통해 통합제어 App으로 전송되었다. 데이터 생성 알고리즘은 그림 12와 같다.

설정된 임계값은 첫 번째 실험과 같으며, 그에 따라 실험의 결과를 예상한 후 실험을 진행하였다. 표 3의 데이터는 기구부1의 온도값이 31°C로 설정된 임계값 30°C보다 크고, 기구부2의 CO2값이 3521ppm으로 임계값 2000ppm보다 크다. 따라서 통합제어App에서는 알고리즘을 거쳐 기구부1 주변의 환기팬 3개, 기구부2 주변의 환기팬 1개의 가동여부를 True로 설정할 것으로 예상하였다. 실험 결과 기구부1 주변의 환기팬 3개, 기구부2 주변의 환기팬 1개가 가동되는 것을 확인하였고, 예상과 일치하였다.

마지막으로 임계값 변경에 따른 동작 확인을 위하여 표 2와 동일한 데이터에서 이산화탄소의 임계값을 변경한 후 실험하기로 하였다.

먼저 통합제어 App을 통하여 이산화탄소 임계값을 1300ppm으로 변경하고, 결과를 예상한 후 실험을 진행하였다. 이 경우에는 기구부2의 이산화탄소 측정값이 1544ppm, 기구부3의 이산화탄소 측정값이

표 3. 임의 생성 데이터.
Table 3. Arbitrary formation data.

임의 값	Temp.	Humi.	CO_2
inst. 1	31	54	834
inst. 2	27	49	3521
inst. 3	24	72	1785
unit	°C	%	ppm

www.dbpia.co.kr

1375ppm으로 변경된 이산화탄소 임계값 1300ppm보다 크게 된다. 따라서 통합제어App에서는 알고리즘을 거쳐 기구부2 주변의 환기팬 1개, 기구부3 주변의 환기팬 1개의 가동여부를 True로 설정할 것으로 예상하였다. 실험 결과 기구부2 주변의 환기팬 1개, 기구부2 주변의 환기팬 1개가 가동되는 것을 확인하였고, 예상과 일치하였다.

Ⅲ. 결 론

본 논문은 육계 계사 내에서 작동하는 대기 내 이산화탄소 농도를 제어할 수 있는 자동 환기 시스템을 실험실 단계에서 설계하고 그 성능을 확인하였다. 경제적인 이유로 기존의 온도 및 시간 기반의 자동 환기 시스템에 이산화탄소 제어만을 추가하였지만, 추가적으로 암모니아 등 닭의 생장과 질병에 영향을 주는 대기 요소를 제어할 수 있도록 발전이 가능하다.

본 논문은 전체 시스템을 목표로 했을 때 일부 결과를 도출했지만, 궁극적으로 계사 내 사육 전자동 시스템을 구성하는데 목표를 두고 있다. 향후 추가 개발을 통해 계사 내의 다양한 대기 요소를 자동적으로 제어할 수 있게 되고, 급이·급수 등의 다른 제어 시스템과의 통합을 이루어낸다면 사람이 전혀 필요하지 않은 전자동화 사육 시스템을 완성할 수 있을 것이다.

사육 과정이 전부 자동화된다면 최근 가장 문제가 되는 인건비를 크게 줄일 수 있을 것이다. 또한 사람이 직접 관여하면서 범할 수 있는 실수나 오류를 줄일 수 있기 때문에 경제적, 시간적으로도 기존의 방법보다 효율적인 결과를 얻을 수 있다. 추가적으로 이 시스템이 모두 완성될 경우에 차후 약간의 설계 변경만으로도 계사뿐만 아니라 돼지, 소 등의 가축 사육장에서도 적용하여 활용도를 높일 수 있을 것으로 예상된다.

References

[1] Y.-C. Kwon, “A study on the characteristics of heating and cooling loads of standard chicken houses in South Korea,” *Journal of the Architectural Institute of Korea Structure & Construction*, vol. 35, no. 10, pp. 235-243, Oct. 2019.

[2] Nongsaro, “*Specification Management of Chicken - Effects of Temperature and Humidity Variation on Chicken*,” Apr. 2019, https://www.nongsaro.go.kr

[3] Bugwhale World, “Creating an android bluetooth application,” May 2019, https://bugwhale.com/android-bluetooth

[4] J. S. Eum, “Transitional Season Specifications Management Point,” *A Monthly Poultry*, vol. 43 no. 9, pp. 118-121, Sep. 2011.

[5] KOSIS, “*Number of households by size of chicken breeding*,” Apr. 2019, http://kosis.kr/statHtml/statHtml.do?orgId=101&tblId=DT_1EO071&conn_path=I2

이 병 선 (Byeongseon Lee)

2014년~현재 : 군산대학교 IT융합통신공학전공 재학

<관심분야> App, Web, Smart Farm

조 주 필 (Juphil Cho)

2001년 2월 : 전북대학교 전자공학과 공학박사

2000년~2005년 : ETRI 이동통신연구단 선임연구원

2006년~2007년 : ETRI 이동통신연구단 초빙연구원

2011년~2012년 : 미국 USF, 교환교수

2005년~현재 : 군산대학교 IT융합통신공학전공 교수

<관심분야> LTE-A, 5세대 이동통신, Cognitive Radio, LED-ID, 방송통신융합기술

www.dbpia.co.kr