International Journal of Electronics Communication and Computer Engineering
Volume 10, Issue 5, ISSN (Online): 2249–071X

Programming the Code to Control the Calibration and the Analog to Digital Signals Conversion in the Nano Sensor System to Monitor the Quality of Aquaculture Water

Truong Van Nguyen, Tung Thanh Bui, Dung My Thi Dang, Tin Chanh Duc Doan* and Chien Mau Dang
Institute for Nanotechnology, Vietnam National University, Ho Chi Minh City, Vietnam.
*Corresponding author email id: ddctin@vnuhcm.edu.vn

Date of publication (dd/mm/yyyy): 30/10/2019

Abstract – **In this study, a nano sensor system was developed to control the quality of the aquaculture water. The analog signals received from the sensor electrodes were converted to the binary digital signals by an Analog to Digitals Converter (ADC). The signals adjustment was concurrently processed with the signals conversion to ensure the most accurate results were obtained. The controlled firmware for these processes was programmed on the Arduino IDE environment with C or C++ languages for the ADC MCP3208 connected to the ATMega 2560 processor. The output signals from the sensors could be converted from analog to digital form, which could then be processed by the microcontrollers or microprocessors. The two-point calibration method was employed to enhance the accuracy of the sensor output signal and minimized the deviation during the conversion from analog to digital signals.**

Keywords – **Sensor System, Programing Code, Analog to Digital Signal Conversion, Arduino, Aquaculture Water.**

I. INTRODUCTION

The microcontrollers or microprocessors were only detecting digital signals in binary [1]. This meant when an input voltage of 5 V was applied, the processor understood that zero volts (0 V) as a binary 0 and a five volts (5 V) as a binary 1. However, almost every environmental measurable parameter was in analog form like temperature, sound, pressure, light, etc. [2]. A 5 V analog testing system might output 0.01 V or 4.99 V or anything in between. The question was how to convert these analog signals into digital format which could then be processed, manipulated, computed, transmitted or stored by the microcontrollers or microprocessors. An Analog to Digital Converter (ADC) was a very useful device that converted an analog voltage on a pin to a digital number [3-5]. It was often integrated on the processor board. An ADC picked an analog signal at uniform time intervals and assigned a digital value to each sample [5, 6]. The digital value appeared on the converter's output in a binary coded format. The value was obtained by dividing the sampled analog input voltage by the reference voltage and then multiplying by the number of digital codes [5, 6]. The resolution of converter was set by the number of binary bits in the output code [5].

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that was written in the programming language Java. It was used to write and upload programs to Arduino compatible boards, with the help of third party cores, other vendor development boards [7]. The source code for the IDE was released under the GNU General Public License (Version 2) [8]. The Arduino IDE supported the languages C and C++ using special rules of code structuring [9]. The Arduino IDE supplied a software library from the Wiring project, which provides many common input and output procedures [10]. The Arduino IDE converted the executable code into a text file in hexadecimal encoding that was loaded into the Arduino board by a loader program in the board's firmware [11].

The calibration process should be carried out to ensure the accuracy of the sensor output signal, thereby minimizing the deviation to the lowest level during the conversion from analog to digital signals [12]. It was critical to remember that no sensor is perfect. The operation of sensors was easy affected by manufacturing variation, working, shipping and storing environmental conditions as well as lifetime [13]. This means that the response of sensors would naturally change over time, thus requiring periodic re-calibration for two factors affecting a sensor's performance and accuracy as noise and hysteresis [13]. There were four adjustments that were used to get a good calibration: offset, gain, linearization and unit scaling. One point calibration, two point calibration and multi-point curve fitting were three of the most popular methods to calibrate sensor systems [14-16].

With the nano sensor system fabricated by INT, the output signal of the sensors was 4-20 mA corresponding to the measured parameters ranges. For example, the output of the pH sensor would be 4-20 mA corresponding to the pH range of 0-14. The controlled PCB used ATMega 2560 processor with a built-in 10 bits ADC. However, this built-in did not meet the resolution requirements, so the 12 bits external ADC MCP3208 was connected to the ATMega 2560 processor by SPI communication. The sensor output signal 4-20 mA would be converted to voltage signal in range of 0 - 4.096 V due to requirement of 12 bits ADC MCP3208 input signal. The sensors in this system were calibrated by two point calibration methods.

II. Experiments

The IC MCP1541 was used to convert sensors output signals 4-20 mA to the reference voltage 0-4.096 V for ADC MCP3208 (Fig. 1).

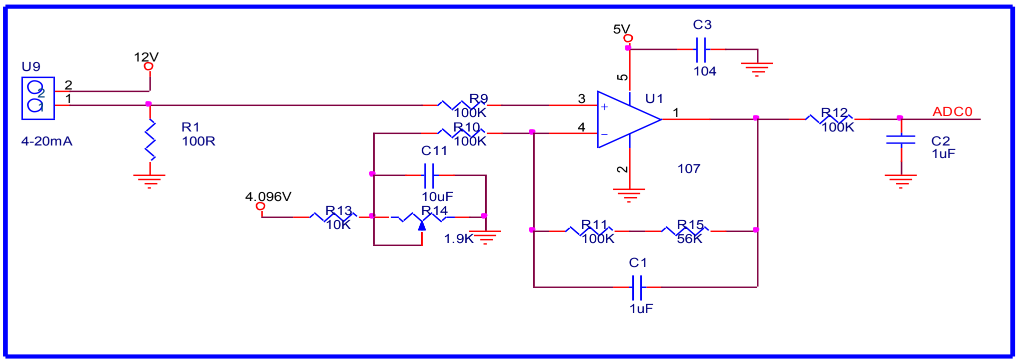

Fig. 1. The electrical diagram of IC MCP1541 in the converter system.

The ADC MCP3208 connected to ATMega 2560 processor via SPI protocol as shown in Figure 2 and Table 1.

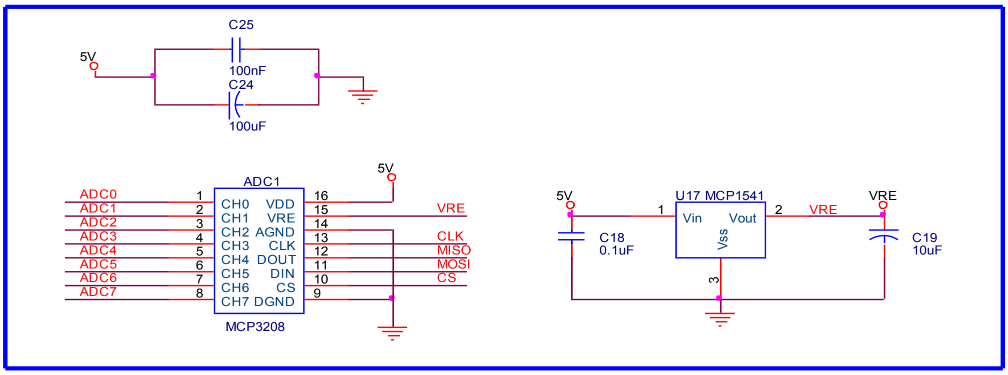

Fig. 2. The ADC MCP3208 connected to ATMega 2560 processor via SPI protocol.

Table 1. Corresponding pins of ADC MCP3208 and ATMega 2560.

ADC MCP3208	ATMega2560
CLK (13)	CLK (52)
DOUT (12)	MISO (50)
DIN (11)	MOSI (D51)
CS (10)	D53

There were two steps for MCP3208 to communicate with ATMega2560 via SPI protocol: write control byte on MCP3208 (the controlled byte was 5 bits included one START bit, one MODE bit and three select channel bits) and read data from MCP3208 (after the transmission end, 8 bits of the conversion data was stored in the memory control unit -MCU register).

III. Results and Discussions

3.1 *Programming the Code for Reading Voltage Signals from ADC MCP3208 Via SPI Protocol*

The algorithm block diagram of programming code used to read measured voltage signals from ADC MCP3208 via SPI protocol was shown in Fig. 3 below.

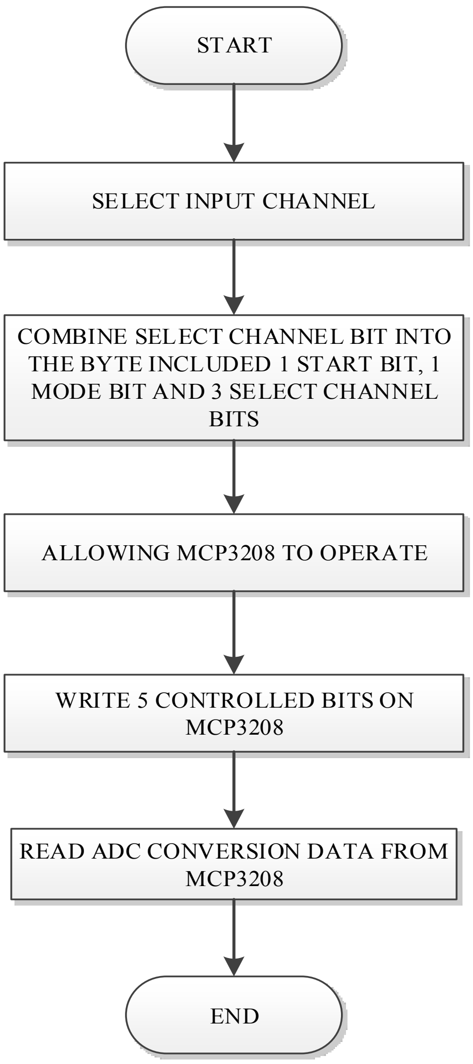

Fig. 3. The algorithm block diagram of programming code used to read measured voltage signals from ADC MCP3208 via SPI protocol.

The detailed programming code was shown as below.

```
int read_adc(int channel){
  float adcvalue = 0;
  byte commandbits = B11000000; //command bits - start, mode
```

Combine the select channel bits into the writing byte on MCP3208:

```
  commandbits|=((channel-1)<<3);
```

Allowing MCP3208 to operate:

```
  digitalWrite(SELPIN,LOW);
```

Write the controlled byte on MCP3208:

```
  for (int i=7; i>=3; i--){
    digitalWrite(DATAOUT,commandbits&1<<i);
    digitalWrite(SPICLOCK,HIGH);
    digitalWrite(SPICLOCK,LOW);
  }
```

Skip 2 null bits:

```
  digitalWrite(SPICLOCK,HIGH);
  digitalWrite(SPICLOCK,LOW);
  digitalWrite(SPICLOCK,HIGH);
  digitalWrite(SPICLOCK,LOW);
```

Read the value of 12 bits on MCP3208:

```
  for (int i=11; i>=0; i--){
    adcvalue  +=digitalRead(DATAIN)<<i;
    digitalWrite(SPICLOCK,HIGH);
    digitalWrite(SPICLOCK,LOW);
  }
  digitalWrite(SELPIN, HIGH);
  return adcvalue;
}
```

3.2 *Programming the Code to Convert Voltage Signals to Corresponding Measured Results and Send them to PC Via Serial Ports*

The current output signals of sensors had a linearly relationship with the values of measured parameters. Therefore:

$M = A_1X + B_1$ (1)

With: M was the values of measured parameters (pH, DO, SA, ORP, TEMP, NH_4^+, NO_3^-, Cl^-…)

X is the value of the current output signals in range of 4 – 20 mA.

A, B, A_1, B_1, A_2, B_2 were the correlation factors. They were calculated from the calibration process.

The current output signals were converted to V_{ADC} voltage input signals by a linear amplifier circuit. Therefore:

$X = A_2V_{ADC} + B_2$ (2)

Replace equation (2) into equation (1):

$M = A_1A_2V_{ADC} + A_1B_1 + B_2$ (3)

The relationship between the values of measured parameters and ADC MCP3208 input voltage signals was linear with the linear coefficient of A_1A_2. Therefore, the measured parameters values could be deduced from the voltage signals of ADC MCP3208 according to equation (3). The following Table 2 described the correlation between measured parameters values, the current output signals of sensors and the voltage signals of ADC MCP3208.

Table 2. The correlation between measured parameters values, the current output signals of sensors and the voltage signals of ADC MCP3208.

Parameter	**Range**	**Current output signals (mA)**	**Voltage input signals (V)**	**Correlation function**
pH	0 ÷ 14 pH	4 ÷ 20	0 ÷ 4.096	$pH = 3.418 \times V_{ADC}$
DO	0 ÷ 19.99mg/l	4 ÷ 20	0 ÷ 4.096	$DO = 4.88 \times V_{ADC}$
SA	0 ÷ 30 ppt	4 ÷ 20	0 ÷ 4.096	$EC = 12.207 \times V_{ADC}$ $SA = A \times EC + B$
TEMP	0 ÷ 100◦C	4 ÷ 20	0 ÷ 4.096	$TEMP = 24.414 \times V_{ADC}$
ORP	-500 ÷ 500 mV	4 ÷ 20	0 ÷ 4.096	$ORP = 244.141 \times V_{ADC} – 500$
NH_4^+	0 ÷ 500 ppm	4 ÷ 20	0 ÷ 4.096	$mV_{NH4} = 244.141 \times V_{ADC} – 500$ $ppm_{NH4} = f(mV_{NH4})$
NO_3^-	0 ÷ 500 ppm	4 ÷ 20	0 ÷ 4.096	$mV_{NO3} = 244.141 \times V_{ADC} – 500$ $ppm_{NO3} = f(mV_{NO3})$
Cl^-	0 ÷ 1000 ppm	4 ÷ 20	0 ÷ 4.096	$mV_{Cl} = 244.141 \times V_{ADC} – 500$ $ppm_{CL} = f(mV_{CL})$

The algorithm block diagram of the code to convert voltage signals to corresponding measured results and send them to PC via serial ports was shown in Fig. 4.

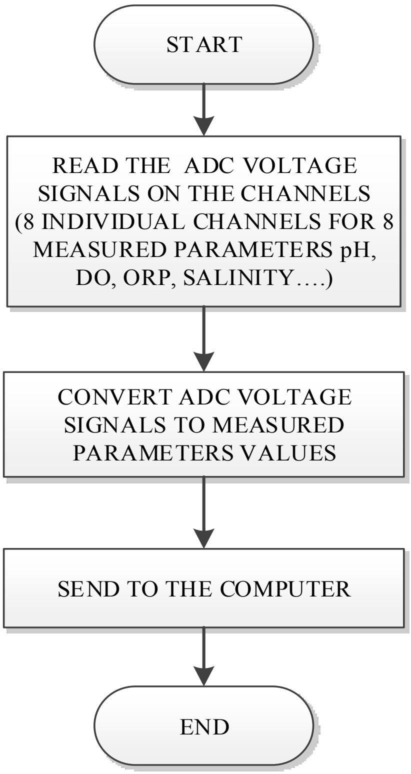

Fig. 4 The algorithm block diagram of the code to convert voltage signals to corresponding measured results and send them to PC via serial ports.

The detailed programming code was shown as below.

```
VADC = read_adc(channel number);
 VADC = VADC/1000 ;
 DO = "correlation function"; //
Serial.println(F("--------------------------------"));
Serial.print("2) parameter (measurement unit) : ")   ;
Serial.println (parameter,2)  ;
```

3.3 *Programming the Code for Sensors Calibration (the Calculation Process of the Correlation Factors)*

3.3.1 *Calibration Method of pH Parameters*

For pH calibration, the 2-point calibration was performed with two standard solutions of $pH_{dd1} = 7$ and $pH_{dd2} = 4$ (or $pH_{dd1} = 7$ and $pH_{dd2} = 10$).

The correlation equation between pH and V_{ADC} was:

$$pH = AV_{ADC} + B \quad (4\text{-}1)$$

Two coefficients A and B in the equation (4-1) were determined by the process as follow: put the sensor in the pH_{dd1} standard solution and read the value of V_{ADC1}, then did the same with pH_{dd2} solution and read the V_{ADC2} value. Replace V_{ADC1} and V_{ADC2} in equation (4-1).

$pH_{dd1} = AV_{ADC1} + B$ (4-2)

$pH_{dd2} = AV_{ADC2} + B$ (4-3)

Then:

$pH_{dd2} - pH_{dd1} = A (V_{ADC2} - V_{ADC1})$

Inferred: $A = (pH_{dd2} - pH_{dd1})/(V_{ADC2} - V_{ADC1})$ (4-4)

$B = pH_{dd1} - AV_{ADC1}$ (4-5)

The same calibration methods were used for DO, ORP, salinity, temperature parameters.

3.3.2 *Calibration Method of Ion Concentration Parameters*

The correlation equation of ion concentration (ppm) and voltage signals (mV) values from the ADC as follows:

$mV = A\times\log [ppm] + B$ (4-6)

$=> [ppm] = 10^{(mV-B)/A}$ (4-7)

Two coefficients A and B in the equation (4-6) were determined by the process as follow: put the sensor in the ppm1 standard solution and read the value mV_1 from ADC, then did the same with ppm2 solution and read the mV_2 value From ADC. Replace mV_1 and mV_2 in equation (4-6):

$mV_1 = A\times\log[ppm1] + B$ (4-7)

$mV_2 = A\times\log [ppm2] + B$ (4-8)

$=> mV_2 - mV_1 = A\times\log ([ppm2] / ppm1])$ (4-9)

If the value of [ppm2] was selected such that [ppm2] = 10 [ppm1] so:

$A = mV_2 - mV_1$ (4-10)

$B = mV_1 - A\times\log [ppm1]$ (4-11)

Therefore, in the ions concentrations calibration process of the sensors, the standard solutions were chosen with 10 times different concentrations, and were the multiple of 10, such as 1 ppm and 10 ppm, 10 ppm and 100 ppm as well as 100 ppm and 1,000 ppm. The algorithm block diagram of the code to calculate A and B correlation factors was shown in Fig. 5.

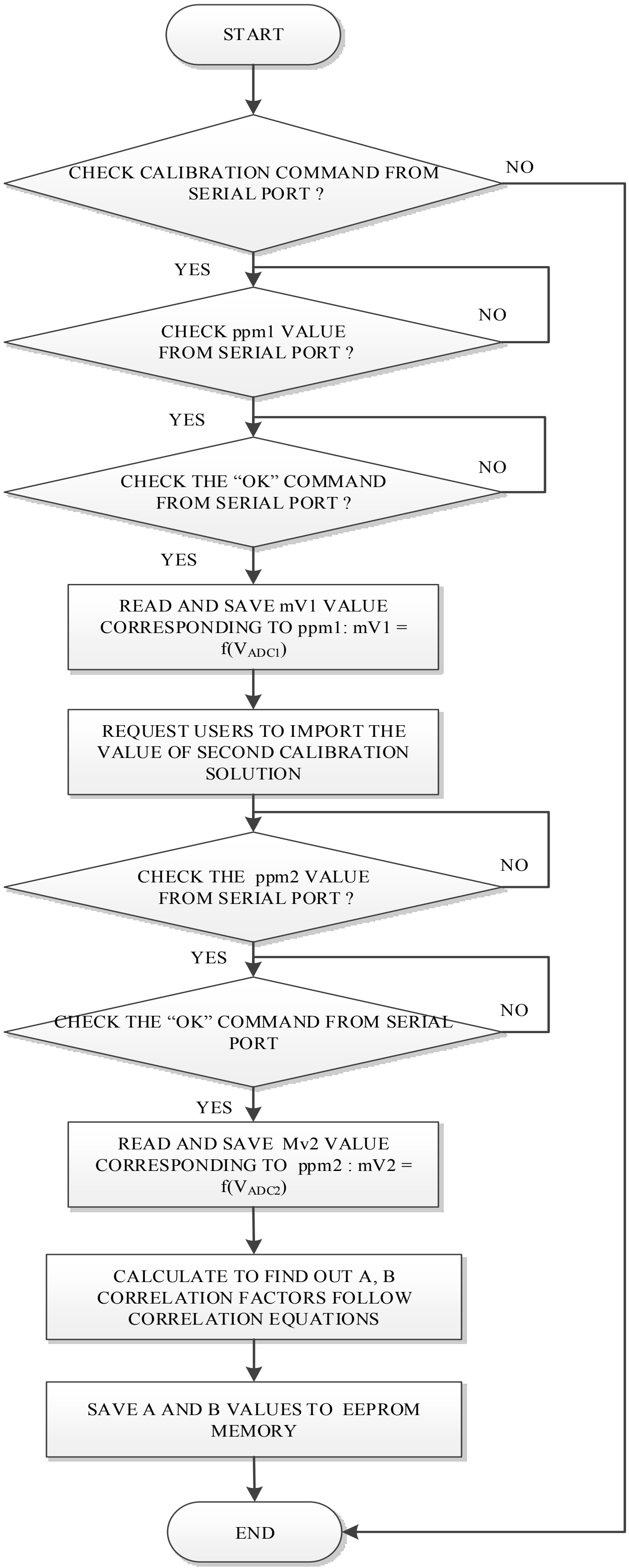

Fig. 5. The algorithm block diagram of the code to calculate A and B correlation factors.

The detailed programming code was shown as below.

```
void set_AB_ION () {
 char func_array[6] ;
 float value_ = 0 ;
 String buffer_string = "" ;
 while (Serial.available() > 0) {
  char c1   ;
  delay (5) ;
  c1 = Serial.read()  ;
  if (c1 == '*') {
  buffer_string = Serial.readStringUntil('_');
  buffer_string.toCharArray(func_array, 6)    ;
  buffer_string = ""   ;
  buffer_string = Serial.readStringUntil('#');
  value_       = buffer_string.toFloat()     ;
  buffer_string = ""   ;
  if (func_array[0]=='S'&& func_array[1]=='?') { //  ?
   Serial.print ("A_VARIABLES:") ;
   Serial.print (A_ VARIABLES,1)  ;
   Serial.print ("B_ VARIABLES:") ;
   Serial.println (B_ VARIABLES,1)  ;
   }
  if (func_array[0]=='S'&& func_array[1]=='C'&& func_array[2]=='1') {
   flag_mode_countinouns = 1 ;
   }
if (func_array[0]=='S'&& func_array[1]=='A'&& func_array[2]=='N'&& func_array[3]=='H'&& func_array[4]
=='4') { //  A_ VARIABLES
   A_ VARIABLES = value_ ;
   EEPROM.put(500, A_ VARIABLES) ;
   Serial.print ("Saved A_ VARIABLES = ") ;
   Serial.println (A_ VARIABLES,1) ;
   }
```

```
if (func_array[0]=='S'&& func_array[1]=='B'&& func_array[2]=='N'&& func_array[3]=='H'&& func_array[4]
=='4') { //  A_ VARIABLES

  B_ VARIABLES = value_ ;

  EEPROM.put(510, B_ VARIABLES) ;

  Serial.print ("Saved B_ VARIABLES = ") ;

  Serial.println (B_ VARIABLES,1) ;

  }
```

IV. Conclusion

In this study, the firmware to control the operation of ADC MCP3208 module in the nano sensor system for monitoring of water in aquacultural ponds was built up and optimized. With the ADC MCP3208 module, the output signals from the sensors could be converted from analog to digital form, which could then be processed, manipulated, computed, transmitted or stored by the microcontrollers or microprocessors. The two-point calibration was carried out to ensure the accuracy of the sensor output signal, thereby minimized the deviation to the lowest level during the conversion from analog to digital signals.

Acknowledgment

The authors would like to thank the South West Program of the Ministry of Science and Technology for their financial support with the contract number 02/2017/HD-KHCN-TNB.DT/14-19/C04.

References

[1] Walden R.H., 1999. “Analog-to-digital converter survey and analysis”, *IEEE Journal on Selected Areas in Communications* 17 (4), p. 539–550. doi:10.1109/49.761034.
[2] Rudy J. van de Plassche, *CMOS integrated analog-to-digital and digital-to-analog converters*. 2nd edition. Kluwer Academic, Boston 2003, ISBN 1-4020-7500-6.
[3] https://www.elprocus.com/analog-to-digital-adc-converter/ Retrieved 25 April 2019.
[4] https://learn.sparkfun.com/tutorials/analog-to-digital-conversion/all Retrieved 25 April 2019.
[5] https://wiki.analog.com/university/courses/electronics/text/chapter-20 Retrieved 25 April 2019.
[6] Kosonocky, S. & Xiao, P. *Analog-to-Digital Conversion Architectures*, Digital Signal Processing Handbook, Boca Raton: CRC Press LLC, 1999.
[7] "Updated: Arduino announces FPGA board, ATmega4809 in Uno Wi-Fi mk2, cloud-based IDE and IoT hardware". *Electronics Weekly*. 2018-05-18. Retrieved 2018-06-14.
[8] "The arduino source code". The Arduino source code.
[9] Purdum, Jack J. *Beginning C for Arduino : learn C programming for the Arduino* (Second ed.). [New York]. ISBN 9781484209400. OCLC 912875060.
[10] Castro, Jorge R. *Building a home security system with Arduino : design, build, and maintain a home security system with Arduino* Uno. Birmingham, UK. p. 15. ISBN 9781785283802. OCLC 922588951.
[11] Banzi, Massimo; Shiloh, Michael. *Getting started with Arduino* (Third ed.). Sebastopol, CA. ISBN 9781449363314. OCLC 898290173.
[12] Reza Langari, *Calibration of Measuring Sensors and Instruments, Measurement and Instrumentation: Theory and Application*, (c) 2012 Elsevier Inc.
[13] Quan Quan, *Introduction to Multicopter Design and Control*, Springer Nature Singapore Pte Ltd. 2017, eBook ISBN 978-981-10-3382-7, DOI 10.1007/978-981-10-3382-7.
[14] M. Kouider, M. Nadi and D. Kourtiche, “Sensors Auto-calibration Method - Using Programmable Interface Circuit Front-end”, *Sensors* 2003, 3, 490-497.
[15] S. Boripatkosol, S. Leleu, T. Coorevits, O. Gibaru, Calibration of capacitive sensors and electronic levels for the straightness measurements using multiprobe method, Conference: Congrès de MétrologieAt (Congress of Metrology and): Paris, October 2011.
[16] Pedro M. Ramos, N Nuno B Bras, A. Cruz Serra, A New Calibration Method for Current and Voltage Sensors Used in Power Quality Measurements, Conference Record-IEEE Instrumentation and Measurement Technology Conference DOI: 10.1109/IMTC.2006. 328579, May 2006.

AUTHOR'S PROFILE

Truong Van Nguyen is a Mechatronic engineering. He has worked at Institute for Nanotechnology (INT), Vietnam National University - Ho Chi Minh City (VNUHCM) from 2015 to now. Currently, he is Vice Head of Laboratory Product Testing and Development of Institute for Nanotechnology. His strength is making electronic circuits for collecting signal from sensors such as pH sensor, DO sensor, Conductivity sensor, ion sensors and auto-sensor systems. He has had 2 patents, two practical application products are "System for automatic monitoring & alarm of salinity invasion" and "Equipment for monitoring of water quality at aquacultural farming ponds". email id: nvtruong@vnuhcm.edu.vn

Tung Thanh Bui obtained his Material Engineer degree in Advanced Material in 2011 from Bach Khoa University, Vietnam National University - Ho Chi Minh City, Vietnam. He obtained his Master degree in Mechanical Engineering in the same time 2011 from the PFIEV Program (Engineer Training Program of Excellence in Vietnam), cooperation training program between Bach Khoa University, VNUHCM and Ecole Centrale (Central School) Paris, France. He obtained his Ph.D. diploma from University of Science, VNUHCM in 2017. His Ph.D. research focused on development of renewable energy, especially in solar cell. Currently, he is Head of Laboratory for Optoelectronic Materials and Devices at Institute for Nanotechnology, Vietnam National University -Ho Chi Minh City, Vietnam. He is author and co-author of 1 book, 8 articles in peer-reviewed journals, 8 proceedings papers of international conferences. His research interests include solar cell, renewable energy, materials and microsystems for energy applications. email id: bttung@vnuhcm.edu.vn

Dung My Thi Dang received her PhD and MSc in Physics from the University of Science, Vietnam National University - Ho Chi Minh City, in 2015 and 2008, respectively. From then she is a Researcher at the Institute for Nanotechnology (INT), Vietnam National University - Ho Chi Minh City, Vietnam. She is currently Vice Director of Institute for Nanotechnology (INT). She has studied the synthesis of copper and silver nanoparticle and inkjet ink formulation. She has authored or co-authored more than 60 publications in journals and conferences. email id: ddmdung@vnuhcm.edu.vn

Tin Chanh Duc Doan obtained his Mechanical Engineer degree in Materials for Mechanical Engineering in 2003 and Master degree in Materials Technology in 2005 from Ho Chi Minh City University of Technology, Vietnam National University - Ho Chi Minh City, Vietnam. He obtained his Ph.D. diploma from Wageningen University, the Netherlands in 2012. His Ph.D. research focused on development of polymer-based sensors for carbon dioxide detection in greenhouses. Currently, he is Vice Director of Institute for Nanotechnology, Vietnam National University - Ho Chi Minh City, Vietnam and concurrently Head of Laboratory for Materials and Environmental Sensors and Laboratory for Fuel Cell at Institute for Nanotechnology. He is author and co-author of 1 thesis book, 32 articles in peer-reviewed journals, 48 proceedings papers of international conferences, 3 patents and 6 applied patent applications. His research interests include sensors for environmental and biological applications, materials and microsystems for energy applications. email id: ddctin@vnuhcm.edu.vn

Chien Mau Dang received his M.Sc. and Ph.D. in Materials Science from the National Polytechnic Institute in Grenoble (Grenoble INP), France in 1991 and 1994. In 1996 and 2007, he received his Master in Management from the University Pierre Mendes France and the Diploma of Habilitation for Research Direction (DHDR) in Materials and Process Engineering from the Grenoble INP. From 1996 to 2004 he was Head of Department of Materials Science Fundamentals, Vice-Dean of Faculty of Material Technology, Ho Chi Minh City University of Technology. Since 2005 he has been Associate Professor. He is currently Director of Institute for Nanotechnology, Vietnam National University - Ho Chi Minh City, Vietnam. He is also a member of several national level research bodies and councils. He has authored or co-authored more than 100 publications in peer-reviewed international journals (among of them 41 ISI and 43 Scopus papers) and nine patents. email id: dmchien@vnuhcm.edu.vn