

Perbandingan Waktu Konversi antara ADC 8 bit dan 10 bit dalam Mikropengendali ATmega8535

Arief Hendra Saptadi
Pascasarjana Teknik Elektro
Universitas Gadjah Mada
Yogyakarta, Indonesia
ariefhendras@gmail.com

Risa Farrid Christianti
STT Telematika Telkom
Purwokerto, Indonesia
risachristianti@yahoo.com
risa@stttelematikatelkom.ac.id

Jaenal Arifin
Pascasarjana Teknik Elektro
Universitas Gadjah Mada
Yogyakarta, Indonesia
jaetoga@yahoo.com

Abstrak—Konversi analog ke digital adalah sebuah proses kuantifikasi nilai-nilai parameter fisik di alam (yang bersifat analog) ke dalam nilai-nilai digital yang setara, untuk keperluan komputasi. Salah satu parameter yang harus diatur dalam proses konversi tersebut adalah jumlah bit digital yang dihasilkan atau resolusi. Mikropengendali yang digunakan sebagai ADC (*Analog to Digital Converter*) pada umumnya menyediakan dua pilihan resolusi yaitu 8 bit dan 10 bit. Penelitian ini bertujuan untuk mengetahui apakah pemilihan ADC 8 bit atau 10 bit pada mikropengendali ATmega8535 berpengaruh terhadap waktu konversi atau tidak. Dari hasil penelitian telah dapat ditunjukkan bahwa resolusi bit yang berbeda akan tetap menghasilkan waktu konversi yang sama untuk frekuensi yang sama. ADC 8 bit dengan frekuensi 125 KHz, 250 KHz dan 500 KHz, masing-masing menghasilkan pulsa sebesar 5,2 div, 2,6 div, dan 1,3 div atau setara dengan waktu konversi sebesar 104 μ s, 52 μ s dan 26 μ s. Hasil yang sama juga diperoleh pada ADC 10 bit untuk frekuensi 125 KHz, 250 KHz dan 500 KHz. Dengan demikian, waktu konversi semata-mata hanya tergantung dari frekuensi ADC yang digunakan dan tidak terpengaruh oleh perbedaan resolusi bit. Meski demikian, hasil pengukuran yang diperoleh dalam penelitian ini masih dapat dioptimalkan menggunakan instrumen ukur yang lebih akurat.

Kata kunci—waktu konversi; resolusi bit; mikropengendali; ADC 8 bit; ADC 10 bit.

I. PENDAHULUAN

Pengubah analog ke digital (atau *analog-to-digital converters*—selanjutnya disebut ADC) memiliki peran penting dalam sistem digital karena terkait peranannya dalam mengubah sinyal masukan analog menjadi sinyal keluaran digital. ADC membentuk antarmuka yang penting untuk menganalisa data analog dengan sebuah komputer digital maupun mikropengendali dan merupakan bagian yang sangat diperlukan dalam sistem komunikasi digital untuk mentransmisikan sinyal analog dari sisi pengirim untuk kemudian didigitalisasi di sisi penerima [1]. ADC banyak digunakan dalam sistem pengujian maupun pengukuran, dari voltmeter digital hingga sistem pengukuran suhu dan kelembaban.

Pengubahan data ke dalam bentuk digital membutuhkan proses konversi sinyal analog yang bersifat kontinu ke dalam bit-bit biner diskrit. Karena terdapat perbedaan karakteristik, maka sinyal analog tersebut akan dipecah ke dalam beberapa rentang diskrit yang mendefinisikan istilah resolusi. Semakin tinggi resolusi maka rentang diskrit akan semakin kecil, sehingga karakteristik sinyal keluaran akan semakin kontinu,

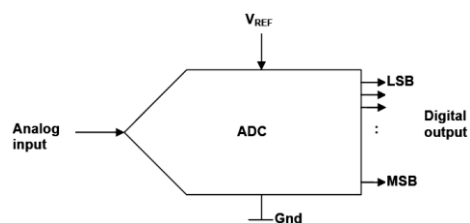
mendekati bentuk sinyal masukan. Di sisi lain, proses konversi tersebut juga memerlukan waktu yang bergantung pada frekuensi ADC dan jumlah siklus mesin yang diperlukan.

Tujuan dari penelitian ini adalah untuk mengetahui apakah perbedaan resolusi bit dalam ADC akan berpengaruh terhadap waktu konversi yang diperlukan. Sinyal masukan analog diwakili oleh hasil pembacaan sensor suhu LM35 (berupa nilai tegangan) sedangkan sinyal keluaran digital diwakili oleh hasil pemrosesan ADC dari mikropengendali ATmega8535 yang diamati menggunakan osiloskop.

A. Analog-to-Digital Converter (ADC)

Sebuah ADC sebenarnya adalah sistem elektronik sederhana yang memiliki input analog (V_{in}), input tegangan referensi (V_{REF}) dan keluaran digital. ADC menerjemahkan sinyal input analog ke nilai keluaran digital yang mewakili ukuran dari input analog yang bersifat relatif terhadap tegangan referensi [2]. Pada saat ini terdapat tiga metode konversi yang banyak digunakan, yaitu Successive Approximation (SAR), Sigma-Delta (Σ - Δ) dan Pipelined [3].

Berdasarkan Gambar 1, rentang tegangan masukan dari ADC ditentukan oleh V_{REF} . Keluaran dari ADC adalah berupa bit biner berjajar dari bit dengan bobot terkecil (*Least Significant Bit* atau LSB) hingga terbesar (*Most Significant Bit* atau MSB). LSB juga merupakan unit tegangan yang setara dengan resolusi terkecil dari ADC [4].



Gambar 1. Diagram Dasar dari Sebuah ADC [2].

Resolusi adalah jumlah bit biner dalam keluaran konversi yang menyatakan banyaknya kemungkinan kode yang muncul (BIT) [1], melalui persamaan:

$$BIT = 2^n \quad (1)$$

dengan n menyatakan jumlah bit. Sehingga dalam ADC 8 bit akan terdapat 256 kemungkinan nilai biner yang muncul (dari 0 hingga 255) dan 1024 kemungkinan (0 – 1023) untuk 10 bit.

Nilai ADC yang dihasilkan akan tergantung dari tegangan masukan analog, tegangan referensi dan jumlah kemungkinan kode yang dinyatakan dalam persamaan [5]:

$$ADC = \frac{V_{in}}{V_{REF}} \cdot BIT \quad (2)$$

Adapun waktu yang dibutuhkan untuk melakukan satu kali konversi akan tergantung dari frekuensi ADC (f_{ADC}) yang digunakan dan jumlah siklus mesin per konversi (MC):

$$T_{conv} = \frac{1}{f_{ADC}} \cdot MC \quad (3)$$

B. ADC dalam Mikropengendali ATmega8535

Mikropengendali ATmega8535 mendukung sistem ADC 8 bit dan 10 bit. Metode yang digunakan untuk melakukan konversi adalah *Successive Approximation* dengan dua mode masukan ADC:

- *Single ended*. Pada mode ini hanya satu input analog yang dimasukkan ke dalam ADC dan konversi dilakukan sebatas pada masukan tersebut.
- *Differential*. Dalam jenis ini, dua input analog diambil lewat masukan *inverting* dan *non-inverting*, baik secara langsung atau melalui penguatan dengan nilai tertentu.

Mikropengendali mengaktifkan ADC dengan salah satu dari tiga mode operasi berikut:

- *Single Conversion*. Jika mode ini dipilih, maka mikropengendali akan menerima perintah untuk melakukan konversi, menjalankannya sekali kemudian berhenti dan menunggu perintah berikutnya.
- *Auto Trigger*. Konversi akan langsung berjalan, ketika terdapat suatu kejadian yang mengaktifkannya (*event-driven*), seperti adanya interupsi.
- *Free Running*. Pada mode ini konversi akan berjalan secara terus-menerus. Konversi berikutnya akan langsung dimulai, setelah konversi sebelumnya selesai dilaksanakan.

Seperti diperlihatkan dalam Tabel 1 dari lembar data, waktu keseluruhan yang diperlukan (termasuk pencuplikan atau sample & hold) untuk melakukan satu kali konversi adalah 25 siklus mesin untuk konversi pertama (*first conversion*) dan 13 siklus mesin untuk konversi berikutnya (*normal conversions*).

TABEL I. WAKTU KONVERSI ADC [5]

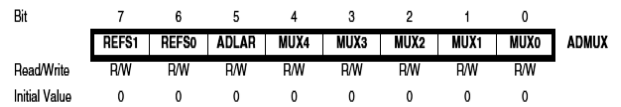
Conditions	Sample & Hold (Cycles from Start of Conversion)	Conversion Time (Cycles)
First Conversion	14.5	25
Normal Conversions, single ended	1.5	13
Auto Triggered Conversions	2	13.5
Normal Conversions, differential	1.5/2.5 ⁽¹⁾	13/14 ⁽¹⁾

¹ Depending on the state of CK_{ADC2} .

Mengingat frekuensi yang diperlukan dalam ADC adalah lebih rendah dari frekuensi detak pada mikropengendali (CK),

maka ADC menggunakan pra penyekalaan (*prescaling*). Pada AVR, pra penyekalaan yang tersedia adalah CK/2, CK/4, CK/8, CK/16, CK/32, CK/64 dan CK/128. Register AVR yang diperlukan untuk ADC adalah:

- ADMUX (ADC Multiplexer Selection Register). Register ini memiliki susunan bit seperti berikut.



Gambar 2. Susunan Bit Register ADMUX [5]

Bit 7 dan 6 (REF1:0) digunakan untuk memilih referensi tegangan (V_{REF}) untuk ADC (Tabel 2). Bit 5 (ADLAR: ADC Left Adjust Result) berfungsi untuk mengatur susunan bit hasil konversi. Bit 4 hingga 0 (MUX4:0) merupakan bit pengatur mode masukan ADC (Tabel 3).

TABEL II. PEMILIHAN REFERENSI TEGANGAN UNTUK ADC [5]

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal V_{REF} turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56 V Voltage Reference with external capacitor at AREF pin

TABEL III. KANAL INPUT DAN PEMILIHAN PENGUATAN

MUX4..0	Single Ended Input	Pos Differential Input	Neg Differential Input	Gain
0000	ADC0	N/A		
0001	ADC1			
0010	ADC2			
0011	ADC3			
0100	ADC4			
0101	ADC5			
0110	ADC6			
0111	ADC7			
0100	N/A	ADC0	ADC0	10x
0101		ADC1	ADC0	10x
0110		ADC0	ADC0	200x
0101		ADC1	ADC0	200x
01100		ADC2	ADC2	10x
01101		ADC3	ADC2	10x
01110		ADC2	ADC2	200x
01111		ADC3	ADC2	200x
10000		ADC0	ADC1	1x
10001		ADC1	ADC1	1x
10010		ADC2	ADC1	1x
10011		ADC3	ADC1	1x
10100		ADC4	ADC1	1x
10101		ADC5	ADC1	1x
10110		ADC6	ADC1	1x
10111		ADC7	ADC1	1x
11000	ADC0	ADC2	1x	
11001	ADC1	ADC2	1x	
11010	ADC2	ADC2	1x	
11011	ADC3	ADC2	1x	
11100	ADC4	ADC2	1x	
11101	ADC5	ADC2	1x	
11110	1.22V (V_{BG})	N/A		
11111	0V (GND)	N/A		

- ADCSRA (ADC Control and Status Register A). Susunan bit pada register ini seperti dalam Gambar 3.

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 3. Susunan Bit Register ADCSRA [5]

Bit 7 (ADEN: ADC Enable), digunakan untuk mengaktifkan ADC. Bit 6 (ADSC: ADC Start Conversion) untuk memulai konversi. Bit 5 (ADATE: ADC Auto Trigger Enable) untuk mengaktifkan mode Auto Trigger. Bit 4 (ADIF: ADC Interrupt Flag) merupakan penanda adanya interupsi. Bit 3 (ADIE: ADC Interrupt Enable) digunakan untuk mengaktifkan interupsi pada ADC. Bit 2 hingga 0 (ADPS2:0) merupakan pengatur nilai pra penyekalaan (Tabel 4).

TABEL IV. PEMILIHAN PRA PENYEKALAN ADC [5]

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

- SFIOR (Special Function IO Register). Bit dalam register ini tersusun sebagaimana dalam Gambar 4 berikut:

Bit	7	6	5	4	3	2	1	0	
	ADTS2	ADTS1	ADTS0	-	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 4. Susunan Bit Register SFIOR [5]

Bit 7 hingga 5 (ADTS2:0: ADC Auto Trigger Source) merupakan bit pemilih sumber auto trigger seperti diperlihatkan dalam Tabel 5. Bit 4 (RES: Reserved) merupakan bit tercadang dan selalu bernilai nol. Empat bit terakhir (3 hingga 0) tidak terkait dengan proses ADC.

TABEL V. PEMILIHAN SUMBER AUTO TRIGGER ADC [5]

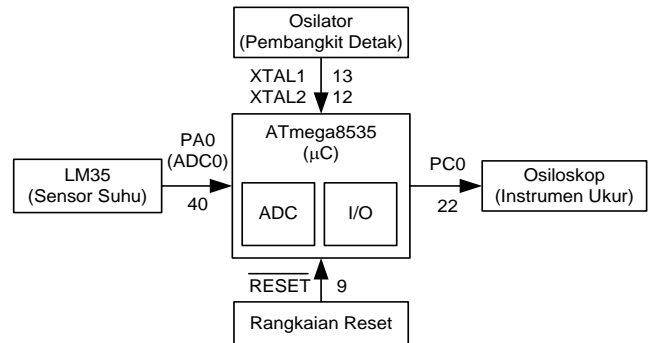
ADTS2	ADTS1	ADTS0	Trigger Source
0	0	0	Free Running Mode
0	0	1	Analog Comparator
0	1	0	External Interrupt Request 0
0	1	1	Timer/Counter0 Compare Match
1	0	0	Timer/Counter0 Overflow
1	0	1	Timer/Counter1 Compare Match B
1	1	0	Timer/Counter1 Overflow
1	1	1	Timer/Counter1 Capture Event

II. METODOLOGI PENELITIAN

Dalam pelaksanaan penelitian ini tahapan yang ditempuh meliputi:

A. Perancangan Sistem dan Rangkaian

Sistem secara keseluruhan diwujudkan sebagaimana blok diagram dalam Gambar 5 berikut ini:

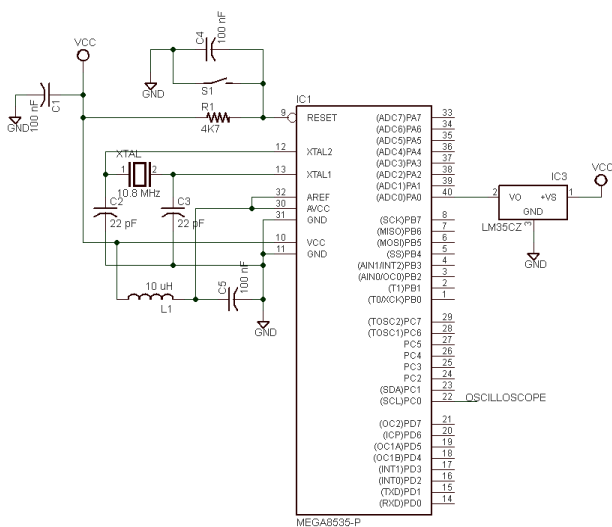


Gambar 5. Blok Diagram Sistem

Bagian-bagian dari sistem tersebut meliputi:

- Mikropengendali ATmega8535. Bagian ini berfungsi untuk mengatur seluruh pemrosesan yang terjadi dalam sistem, antara lain proses konversi dalam ADC dan penanganan port untuk keluaran maupun masukan.
- Rangkaian Reset. Bagian ini digunakan untuk mengembalikan program ke kondisi awal dan dihubungkan ke ATmega8535 pada port RESET atau pin 9.
- Sensor suhu LM35. Sensor ini berfungsi untuk mengubah suhu menjadi tegangan masukan analog (V_{in}) ke ATmega8535 melalui port masukan PA0 (pin 40). Tegangan tersebut selanjutnya dibandingkan dengan tegangan referensi (V_{REF}) dan dikalikan dengan jumlah kemungkinan kode, seperti dinyatakan dalam persamaan (2). Sensor ini dipilih karena sudah terkalibrasi dalam Celcius, rentang pengukuran dari -55°C hingga 150°C dan dapat beroperasi dari tegangan sumber 4 – 30 V [6].
- Osilator. Bagian ini berfungsi untuk membangkitkan detak ke mikropengendali, yang selanjutnya digunakan sebagai acuan untuk memperoleh nilai frekuensi ADC (f_{ADC}) melalui pra penyekalaan. Osilator terhubung ke mikropengendali melalui dua port, yaitu XTAL1 (pin 13) dan XTAL2 (pin 12). Osilator ini memiliki frekuensi 16 MHz.
- Osiloskop. Instrumen ukur ini digunakan untuk mengukur waktu konversi yang dihasilkan oleh mikropengendali pada port keluaran PC0 (pin 22).

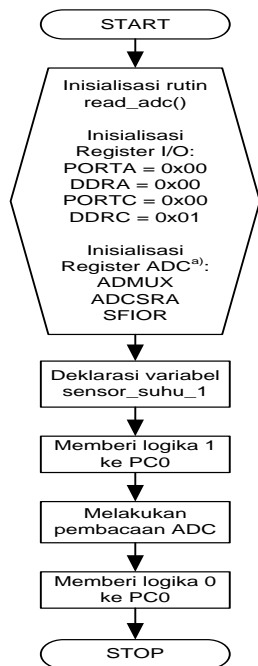
Rangkaian lengkap yang dibuat berdasarkan sistem tersebut adalah seperti dalam Gambar 6 berikut ini.



Gambar 6. Rangkaian Lengkap

B. Perancangan Program

Alur pemrosesan yang dijalankan oleh program utama mengikuti diagram alir berikut:



^a. Nilai dari register diatur sebagaimana dalam Tabel 6

Gambar 7. Diagram Alir Aplikasi.

Pada saat pertama kali dijalankan, program akan melakukan inisialisasi untuk rutin read_adc(). Selanjutnya port A (PA) diatur sebagai masukan dengan nilai awal 0x00 dan port C pin 1 (PC0) sebagai keluaran dengan nilai awal 0x00.

Berikutnya program melakukan inisialisasi register ADC, meliputi ADMUX, ADCSRA dan SFIOR. Ke dalam ketiga register tersebut dimasukkan nilai-nilai untuk mengatur hal-hal sebagaimana berikut ini:

- Mode masukan *Single Ended Input* pada ADC0

- Mode operasi *Free Running*
- Tegangan Referensi 2,56 V internal
- ADC 8 bit dan 10 bit
- Frekuensi ADC 125 KHz, 250 KHz dan 500 KHz

Keseluruhan nilai yang digunakan pada ketiga register ADC tersebut adalah seperti dalam tabel 6 berikut ini:

TABEL VI. NILAI YANG DITETAPKAN UNTUK REGISTER ADC

Register ADC	Nilai					
	ADC 8bit			ADC 10bit		
	125 KHz	250 KHz	500 KHz	125 KHz	250 KHz	500 KHz
SFIOR	0x0F	0x0F	0x0F	0x0F	0x0F	0x0F
ADMUX	0xE0	0xE0	0xE0	0xC0	0xC0	0xC0
ADCSRA	0xA7	0xA6	0xA5	0xA7	0xA6	0xA5

Setelah inisialisasi register, program mendeklarasikan variabel sensor_suhu_1 yang berfungsi untuk menampung hasil konversi ADC. Sebelum proses konversi dilakukan, port C0 diberikan logika 1 dan tepat setelah pembacaan ADC selesai dijalankan, port tersebut diberikan logika 0. Proses perubahan logika dari 1 ke 0 dan sebaliknya inilah yang kemudian ditampilkan oleh osiloskop dan menandai waktu konversi.

III. PENGUJIAN DAN ANALISIS HASIL

Untuk dapat membandingkan waktu konversi antara ADC 8 bit dan 10 bit, maka setelah dilakukan perancangan kemudian dilanjutkan dengan pengujian, pencatatan hasil dan analisa.

A. Prosedur Pengujian

Pengujian dilakukan dengan menempuh prosedur sebagaimana berikut ini:

- Kode sumber diatur untuk menghasilkan ADC 8 bit dengan frekuensi 125 KHz.
- Kode sumber dikompilasi dan hasilnya diunduh ke mikropengendali.
- Rangkaian dioperasikan dan tampilan dari bentuk gelombang keluaran di osiloskop didokumentasikan.
- Setelah selesai rangkaian dimatikan, kode sumber kemudian diubah untuk ADC 8 bit dan frekuensi 250 KHz. Demikian seterusnya dengan langkah yang sama untuk ADC 8 bit 500 KHz, ADC 10 bit 125 KHz, ADC 10 bit 250 KHz dan ADC 10 bit 500 KHz.

B. Hasil

Periode sinyal keluaran kemudian dihitung menggunakan rumus berikut:

$$T = \text{time} / \text{div} \times \text{division} \quad (4)$$

Dari Gambar 8 tersebut, osiloskop diatur dengan time/div sebesar 20 μs/div dan volt/div sebesar 5 V/div. Satu periode

sinyal diketahui selebar 5,2 kotak (5,2 div). Sehingga dengan demikian dapat diketahui bahwa periode sinyal adalah 104 μ s.



Gambar 8. Sinyal Keluaran ADC 8 bit 125 KHz.



Gambar 9. Sinyal Keluaran ADC 8 bit 250 KHz.



Gambar 12. Sinyal Keluaran ADC 10 bit 250 KHz.



Gambar 13. Sinyal Keluaran ADC 10 bit 500 KHz.

Menggunakan cara yang sama untuk sinyal keluaran sistem ADC dengan frekuensi yang berbeda-beda (Fig. 9 – 13) diperoleh hasil periode sinyal keluaran dalam Tabel 7 berikut:

TABEL VII. NILAI PARAMETER SINYAL KELUARAN

Parameter	Nilai					
	ADC 8bit			ADC 10bit		
	125 KHz	250 KHz	500 KHz	125 KHz	250 KHz	500 KHz
Time/div (μ s)	20					
Division	5.2	2.6	1.3	5.2	2.6	1.3
Periode (μ s)	104	52	26	104	52	26



Gambar 10. Sinyal Keluaran ADC 8 bit 500 KHz.



Gambar 11. Sinyal Keluaran ADC 10 bit 125 KHz.

C. Analisa

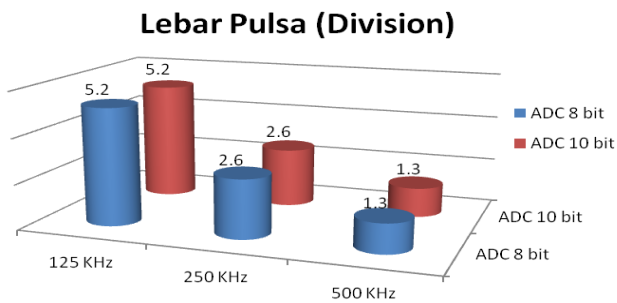
Berdasarkan Tabel 1, dengan mengambil *Normal Conversions* sebagai acuan, maka satu kali konversi membutuhkan 13 siklus mesin. Jika frekuensi ADC yang digunakan adalah 125 KHz, maka dengan menggunakan persamaan (3) diperoleh periode 104 μ s, yang merupakan waktu konversi. Adapun hasil perhitungan waktu konversi untuk frekuensi lainnya adalah seperti dalam Tabel 8 berikut:

TABEL VIII. HASIL PERHITUNGAN WAKTU KONVERSI ADC

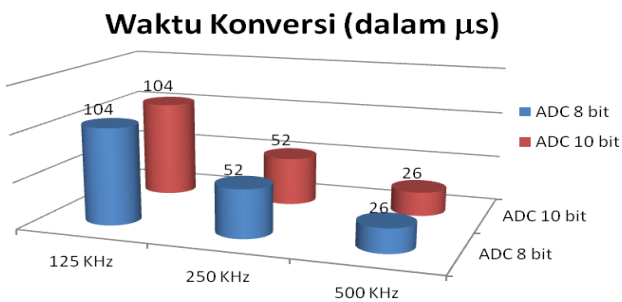
Frekuensi (KHz)	Waktu Konversi (μ s)
125	104
250	52
500	26

Dengan membandingkan frekuensi dan waktu konversi antara Tabel 7 dan 8, baik untuk ADC 8 bit maupun 10 bit, maka dapat disimpulkan bahwa hasil perhitungan adalah

identik dengan hasil pengujian. Fakta terpenting yang diperoleh adalah bahwa resolusi bit yang berbeda akan tetap menghasilkan waktu konversi yang sama untuk frekuensi yang sama. Dengan kata lain bahwa waktu konversi semata-mata tergantung dari frekuensi ADC yang digunakan dan tidak terpengaruh perbedaan resolusi bit. Gambar 14 dan 15 berikut ini menunjukkan perbandingan antara kedua resolusi tersebut.



Gambar 14. Lebar Pulsa (Division) dari Hasil Pengujian



Gambar 15. Waktu Konversi dari Hasil Pengujian

Dengan demikian pemilihan resolusi bit untuk ADC lebih didasarkan kepada seberapa akurat nilai konversi yang diinginkan, bukan berdasarkan waktu konversi yang diperlukan. Di samping itu, pengukuran semestinya dapat dilakukan menggunakan instrumen ukur dengan rentang nilai yang lebih lebar, resolusi yang lebih tinggi, untuk memperoleh hasil yang lebih akurat.

IV. PENUTUP

Dari pengujian yang telah dilaksanakan dan dengan menganalisa hasil-hasil yang diperoleh maka dapat disusun beberapa kesimpulan dan saran sebagaimana berikut.

A. Kesimpulan

Berdasarkan hasil-hasil yang diperoleh dari proses perhitungan maupun penggunaan alat ukur, dapat disimpulkan beberapa hal berikut ini:

- Hasil perhitungan yang diperoleh sama dengan hasil pengujian untuk ADC yang berbeda resolusinya namun dengan frekuensi yang sama. ADC 8 bit dengan frekuensi 125 KHz, 250 KHz dan 500 KHz, masing-masing menghasilkan pulsa selebar 5,2 div, 2,6 div, dan 1,3 div atau setara dengan waktu konversi sebesar 104 μs, 52 μs dan 26 μs. Hasil yang sama juga diperoleh pada ADC 10 bit untuk frekuensi 125 KHz, 250 KHz dan 500 KHz.
- Perbedaan resolusi dalam ADC tidak berpengaruh pada waktu konversi yang diperlukan sepanjang frekuensi ADC yang digunakan sama.

B. Saran

Adapun hal-hal yang dapat disarankan dari penelitian:

- Pemilihan resolusi bit adalah lebih terkait kepada seberapa akurat hasil konversi yang diinginkan. Pada sebagian besar aplikasi, penggunaan ADC 8 bit sudah mencukupi.
- Hasil pengukuran waktu konversi dapat lebih ditingkatkan dengan menggunakan instrumen ukur yang lebih akurat.

REFERENSI

- [1] A.K. Maini, Digital Electronics Principles, Devices and Applications, John Wiley & Sons: Chicester, 2007, pp. 473 – 477.
- [2] “Atmel AVR127: Understanding ADC Parameters application note,” ATMEL Corporation, San Jose, USA.
- [3] S. Rapuano, P. Daponte, E. Balestrieri, L. De Vito, S. J. Tilden, S. Max, dan J. Blair, “ADC Parameters and Characteristics,” IEEE Instrumentation & Measurement Magazine, pp. 44 – 47, December 2005.
- [4] J. Feddele, B. Lucas, ADC Definitions and Specifications, Freescale Semiconductor, 2004, pp. 18 – 19.
- [5] “ATmega8535 ATmega8535L data sheet,” ATMEL Corporation, San Jose, USA.
- [6] “LM35 data sheet,” National Semiconductor.