

4. PENGUJIAN SISTEM

Pada bab ini akan dilakukan pengujian sistem secara keseluruhan yang dibagi menjadi beberapa tahapan yaitu:

- Pengujian program Arduino yang telah dibuat sehingga dapat mengetahui ketepatan pengukuran *cos phi*. Pengujian ini dilakukan untuk mengetahui ketepatan pengukuran *cos phi* berdasar rumus setelah obyek pengukuran ditransformasi dan diolah oleh Arduino.
- Pengujian program Arduino yang telah dibuat sehingga dapat mengetahui perbandingan perubahan daya yang dihasilkan oleh kapasitor dengan *duty cycle*. Pengujian ini dilakukan untuk mengetahui kemampuan *driver* MOSFET dalam menghasilkan daya reaktif beserta efek yang ditimbulkan pada jaringan listrik.
- Pengujian program Arduino yang telah dibuat sehingga dapat mengetahui ketepatan penggunaan kapasitor. Pengujian ini dilakukan untuk mengetahui ketepatan Arduino dalam mengatur kapasitor yang digunakan untuk melakukan kompensasi ke jaringan listrik.
- Pengujian arus maksimum yang dibutuhkan oleh kapasitor sehingga dapat mengetahui proteksi arus yang dibutuhkan. Pengujian ini dilakukan untuk menentukan kapasitas proteksi yang paling optimal sehingga kapasitor dan sistem dapat terlindungi dengan baik.

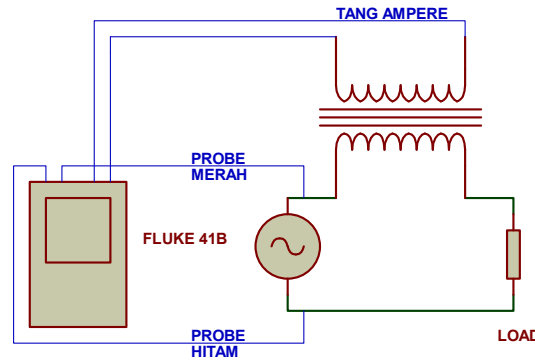
Untuk menghitung persentase kesalahan yang terjadi, maka digunakan rumus sebagai berikut.

$$\% \text{ error} = \frac{\text{nilai pengukuran} - \text{nilai acuan}}{\text{nilai acuan}} \quad (4.1)$$

4.1. Pengujian Program Arduino untuk Ketepatan Pengukuran *Cos Phi*

Pengujian ini dilakukan untuk mengetahui ketepatan pengukuran *cos phi* yang dilakukan oleh Arduino. Pengukuran difokuskan pada angka 0.5 hingga 0.99 *lagging* karena disesuaikan dengan rentang kerja sistem. Pengujian dilakukan

setelah program pada Gambar 3.14 dijalankan dan melakukan pengukuran. Cara pengukuran dapat dilihat pada Gambar 4.1.



Gambar 4.1. Pengukuran Menggunakan Fluke 41B

Pengujian program Arduino untuk ketepatan pengukuran $\cos \phi$ dilakukan dengan mengambil data sebanyak 70 data $\cos \phi$ yang dibandingkan dengan alat ukur *Power Harmonic Analyzer* Fluke 41B. Hasil pengujian dapat dilihat pada Tabel 4.1.

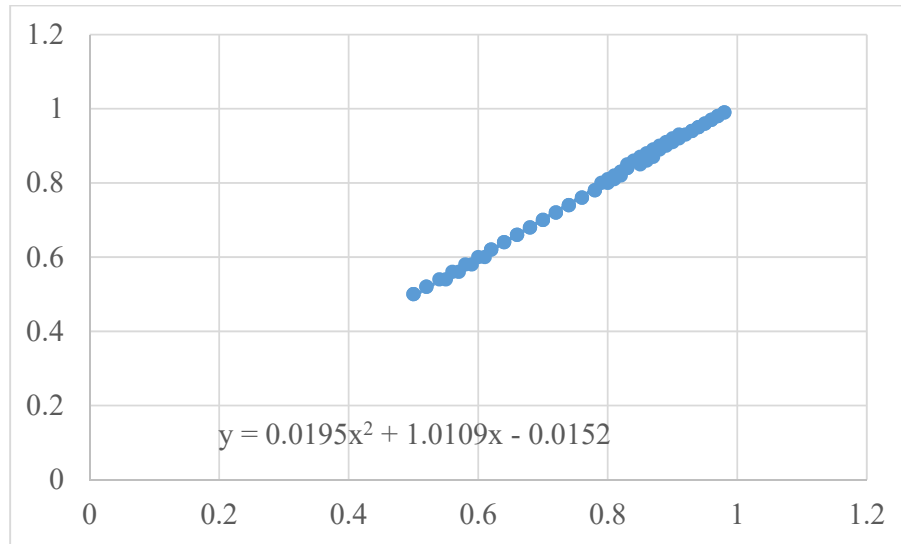
Tabel 4.1. Tabel Perbandingan Pengukuran Fluke 41B dan Arduino

Fluke 41B	Arduino	Fluke 41B	Arduino
0.99	0.98	0.5	0.5
0.98	0.97	0.52	0.52
0.97	0.96	0.54	0.54
0.96	0.95	0.56	0.56
0.95	0.94	0.58	0.58
0.94	0.93	0.6	0.6
0.93	0.92	0.62	0.62
0.92	0.91	0.64	0.64
0.91	0.9	0.66	0.66
0.9	0.89	0.68	0.68
0.89	0.88	0.7	0.7
0.88	0.87	0.72	0.72
0.87	0.87	0.74	0.74
0.86	0.86	0.76	0.76
0.85	0.85	0.78	0.78
0.84	0.83	0.8	0.79
0.83	0.82	0.81	0.8

Tabel 4.1. Tabel Perbandingan Pengukuran Fluke 41B dan Arduino (sambungan)

Fluke 41B	Arduino	Fluke 41B	Arduino
0.82	0.82	0.82	0.81
0.81	0.81	0.83	0.82
0.8	0.8	0.84	0.83
0.78	0.78	0.85	0.83
0.76	0.76	0.86	0.84
0.74	0.74	0.87	0.85
0.72	0.72	0.88	0.86
0.7	0.7	0.89	0.87
0.68	0.68	0.9	0.88
0.66	0.66	0.91	0.89
0.64	0.64	0.92	0.9
0.62	0.62	0.93	0.91
0.6	0.61	0.94	0.93
0.58	0.59	0.95	0.94
0.56	0.57	0.96	0.95
0.54	0.55	0.97	0.96
0.52	0.52	0.98	0.97
0.5	0.5	0.99	0.98

Pada Tabel 4.1 dapat dilihat ada sedikit perbedaan hasil pengukuran Fluke 41B dan Arduino dengan kesalahan rata-rata sebesar 0.78%. Hasil pengukuran yang telah dilakukan kemudian dikalibrasi sehingga didapatkan hasil yang lebih maksimal. Pengalibrasian dilakukan dengan menggunakan *Microsoft Excel* dan menggabungkan kedua data tersebut dalam grafik *scatter* dengan pengukuran Arduino sebagai nilai x dan pengukuran Fluke 41B sebagai nilai y. Setelah grafik terbentuk, ditambahkan *trendline* dengan jenis persamaan polinomial orde 2. Dapat dilihat hasil grafik yang terbentuk beserta persamaan kalibrasi pada Gambar 4.2.



Gambar 4.2. Grafik Perbandingan Pengukuran Arduino dengan Fluke 41B

Hasil dari pengalibrasian perhitungan $\cos \phi$ oleh Arduino didapatkan persamaan sebagai berikut:

$$y = 0.0195x^2 + 1.0109x - 0.0152 \quad (4.1)$$

Nilai $\cos \phi$ yang terkalibrasi dihitung ulang oleh Arduino dengan menggunakan persamaan 4.1 seperti program pada Gambar 4.3.

```

cosphiKalibrasi = (0.0195*cosphi1*cosphi1)
                  +(1.0109*cosphi1)-0.0152;

```

Gambar 4.3. Program Perhitungan $\cos \phi$ Terkalibrasi

Hasil program perhitungan Arduino terkalibrasi diuji kembali dengan cara dibandingkan dengan hasil pengukuran Fluke 41B seperti pada Tabel 4.2.

Tabel 4.2. Tabel Perbandingan Pengukuran Fluke 41B dan Arduino Terkalibrasi

Fluke 41B	Arduino	Fluke 41B	Arduino
0.99	0.99	0.5	0.5
0.98	0.98	0.52	0.52
0.97	0.97	0.54	0.54
0.96	0.96	0.56	0.56
0.95	0.95	0.58	0.58
0.94	0.94	0.6	0.6
0.93	0.93	0.62	0.62
0.92	0.92	0.64	0.64
0.91	0.91	0.66	0.66
0.9	0.9	0.68	0.68
0.89	0.89	0.7	0.7
0.88	0.88	0.72	0.72
0.87	0.87	0.74	0.74
0.86	0.86	0.76	0.76
0.85	0.85	0.78	0.78
0.84	0.84	0.8	0.8
0.83	0.83	0.81	0.81
0.82	0.82	0.82	0.82
0.81	0.81	0.83	0.83
0.8	0.8	0.84	0.84
0.78	0.78	0.85	0.85
0.76	0.76	0.86	0.86
0.74	0.74	0.87	0.87
0.72	0.72	0.88	0.88
0.7	0.7	0.89	0.89
0.68	0.68	0.9	0.9
0.66	0.66	0.91	0.91
0.64	0.64	0.92	0.92
0.62	0.62	0.93	0.93
0.6	0.6	0.94	0.94
0.58	0.59	0.95	0.95
0.56	0.57	0.96	0.96
0.54	0.55	0.97	0.97
0.52	0.52	0.98	0.98
0.5	0.5	0.99	0.99

Hasil pengujian program perhitungan Arduino terkalibrasi memiliki ketepatan baca dengan persentase kesalahan sebesar 0.02%. Kesalahan pengukuran masih terjadi karena adanya ketidaklinearan transformasi pada yang dilakukan oleh

trafo arus. Trafo arus akan menghasilkan tegangan yang berbeda sudut ketika arus yang lewat berubah sehingga mengganggu proses pengukuran faktor daya. Akan tetapi, angka ini lebih baik khususnya pada nilai kritis seperti 0.85-0.95 yang digunakan sebagai nilai acuan PF sehingga pengukuran Arduino layak digunakan.

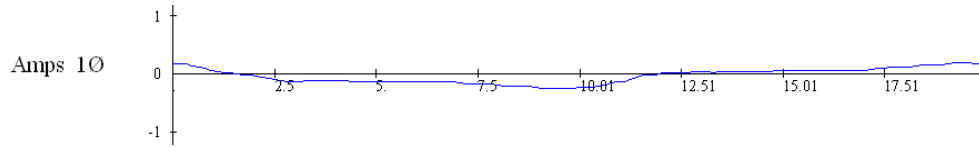
4.2. Pengujian Program Arduino untuk *Driver* MOSFET

Pengujian ini dilakukan untuk mengetahui daya reaktif yang dihasilkan oleh sebuah kapasitor $2\mu\text{F}$ 450VAC dengan *duty cycle* yang berbeda serta efek yang ditimbulkan akibat frekuensi *sampling* untuk membentuk *duty cycle*. Pengujian dilakukan dengan melakukan pengukuran seperti pada Gambar 4.1 dengan beban kapasitor yang menggunakan *driver* MOSFET dan diambil 11 buah data seperti pada Tabel 4.3.

Tabel 4.3. Tabel Perbandingan *Duty Cycle* dan Daya Reaktif

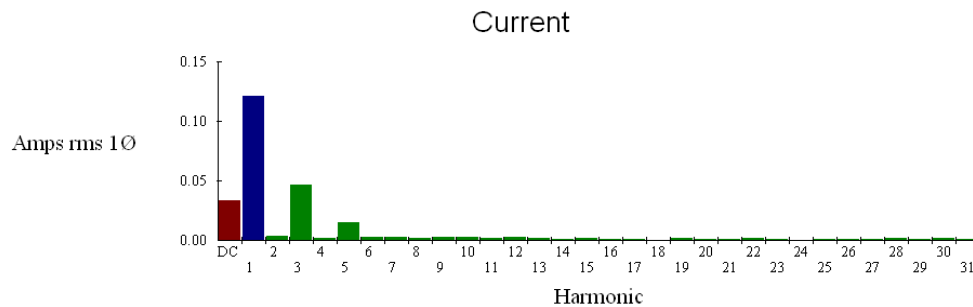
<i>Duty Cycle</i>	Daya Reaktif
0	0
10	8
20	12
30	15
40	15
50	22
60	34
70	34
80	37
90	37
100	38

Hasil pengukuran daya reaktif adalah nilai rata-rata yang dapat diambil karena nilai daya reaktif beresilasi ketika dilakukan pengukuran. Hal ini disebabkan adanya pergeseran fasa pada trafo tegangan yang digunakan sebagai patokan dalam menjalankan *driver* MOSFET.



Gambar 4.4. Grafik Arus Penggunaan *Driver* MOSFET pada *Duty Cycle* 80%

Pada Gambar 4.4 dapat dilihat bahwa penggunaan *driver* MOSFET dapat melakukan rekayasa tegangan sehingga ketika *driver* dijalankan dengan *duty cycle* sebesar 80% maka arus yang dihasilkan membentuk sinyal sinus yang tidak sempurna yang membuat daya reaktif yang dihasilkan berkurang dari daya reaktif yang seharusnya dihasilkan oleh kapasitor.



Gambar 4.5. Harmonisa Arus Penggunaan *Driver* MOSFET pada *Duty Cycle* 80%

Penggunaan metode TSC juga memiliki kelemahan karena merekayasa tegangan yang seharusnya memiliki sinyal sinus murni sehingga menimbulkan harmonisa pada urutan ke-3 dan ke-5. Pada dasarnya, penggunaan MOSFET untuk metode TSC dapat dilakukan dan sistem bekerja dengan normal.

4.3. Pengujian Program Arduino untuk Ketepatan Penggunaan Kapasitor

Pengujian ini dilakukan untuk mengetahui ketepatan penggunaan kapasitor yang dilakukan oleh Arduino. Proses pengukuran dilakukan sesuai dengan Gambar 4.1. Ada dua jenis pengujian ketepatan penggunaan kapasitor yang pengukurannya lebih difokuskan pada angka 0.85 hingga 0.95 *lagging* karena disesuaikan dengan nilai faktor daya acuan yang menjadi patokan pengukuran dari sistem ini. Pengujian dilakukan setelah program pada Gambar 3.15 dijalankan dan

dilakukan pengukuran menggunakan Fluke 41B. Pertama, dilakukan pengambilan data sebanyak 22 buah dengan beban linear yaitu beban universal milik laboratorium sistem tenaga listrik yang sebesar 700VA dengan faktor daya 0.75 dapat dilihat pada Tabel 4.4.

Tabel 4.4. Tabel Perbandingan Faktor Daya Acuan dengan *Cos Phi* Terkalibrasi dan Fluke 41B pada Pengukuran Beban Linear

SetPoint	Arduino	Fluke 41B	Waktu Respon (s)
0.85	0.85	0.84	17
0.86	0.86	0.84	19
0.87	0.88	0.87	21
0.88	0.87	0.87	22
0.89	0.88	0.87	25
0.9	0.89	0.88	29
0.91	0.91	0.88	33
0.92	0.93	0.89	36
0.93	0.93	0.89	38
0.94	0.93	0.89	43
0.95	0.95	0.92	43
0.95	0.95	0.92	42
0.94	0.93	0.89	43
0.93	0.93	0.89	39
0.92	0.92	0.89	36
0.91	0.91	0.88	33
0.9	0.89	0.88	30
0.89	0.89	0.88	25
0.88	0.88	0.87	22
0.87	0.88	0.87	21
0.86	0.86	0.84	20
0.85	0.85	0.84	16

Dari hasil pengukuran di atas, kompensasi yang dilakukan sudah memenuhi syarat yaitu nilai faktor daya terukur (Arduino) memiliki kesalahan sebesar 0.14% dari nilai faktor daya acuan atau *SetPoint*. Selain itu, nilai pengukuran Arduino menunjukkan nilai yang berbeda dari nilai pengukuran Fluke 41B dengan persentase kesalahan sebesar 2.4%. Kesalahan pengukuran yang terjadi disebabkan oleh adanya ketidaklinearan transformasi pada arus yang dilakukan oleh

trafo arus dan berdasarkan hasil pengujian pada Tabel 4.5, pengaruh dari harmonisa yang ditimbulkan oleh *driver* MOSFET seperti pada Gambar 4.5 juga menyebabkan kesalahan pengukuran. Selain itu, waktu respon berbanding lurus dengan nilai *SetPoint* yang digunakan. Hal ini disebabkan penggunaan metode pengkompensasian secara bertahap sehingga semakin besar beban atau *SetPoint* maka semakin lama regulator mencapai target.

Kedua, dilakukan pengujian terhadap beban tidak linear yaitu beban pada Gambar 3.8. Pengujian dilakukan dengan mengambil 22 buah data yang dapat dilihat pada Tabel 4.5.

Tabel 4.5. Tabel Perbandingan Faktor Daya Acuan dengan *Cos Phi* Terkalibrasi dan Fluke 41B pada Pengukuran Beban Tidak Linear

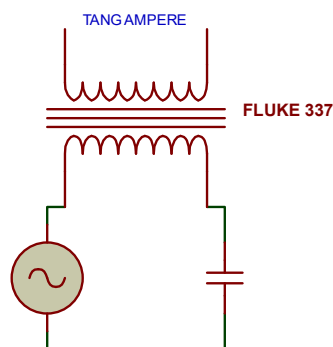
SetPoint	Arduino	Fluke 41B	Waktu Respon (s)
0.85	-	-	∞
0.86	-	-	∞
0.87	-	-	∞
0.88	-	-	∞
0.89	-	-	∞
0.9	0.91	0.67	32
0.91	0.91	0.67	32
0.92	0.91	0.67	32
0.93	0.92	0.68	33
0.94	0.95	0.73	36
0.95	0.95	0.73	36
0.95	0.95	0.73	36
0.94	0.95	0.73	37
0.93	0.93	0.68	33
0.92	0.91	0.67	32
0.91	0.91	0.67	31
0.9	0.91	0.67	31
0.89	-	-	∞
0.88	-	-	∞
0.87	-	-	∞
0.86	-	-	∞
0.85	-	-	∞

Hasil pengujian menunjukkan perbedaan pengukuran yang cukup jauh antara Arduino dan pengukuran Fluke 41B dengan persentase kesalahan sebesar 33.96%. Ketika nilai faktor daya acuan diatur pada angka 0.85-0.89, pengompensasian mengalami osilasi lebih dari 1% sehingga tidak dapat dilakukan pengukuran. Dari pengujian di atas, dapat disimpulkan bahwa pengukuran beban tidak linear atau memiliki harmonisa dapat menyebabkan kesalahan yang cukup besar pada Arduino.

Dibandingkan dengan persentase kesalahan *Varlogic* milik Schneider sebesar 2% (Schneider, 2009), sistem ini memiliki nilai kesalahan yang hampir sama pada beban linear yaitu sebesar 2.4%. Sedangkan untuk beban tidak linear dengan kesalahan sebesar 33.96%, sistem masih belum mampu mencapai hasil yang sama dengan *Varlogic*.

4.4. Pengujian Arus Maksimum Kapasitor untuk Menentukan Proteksi Arus

Pengujian ini dilakukan untuk mengetahui arus maksimum dari masing-masing kapasitor yang digunakan sehingga dapat ditentukan penggunaan proteksi arus yang dibutuhkan. Pengujian dilakukan dengan melakukan pengukuran arus menggunakan Fluke 337 *True RMS Clamp Meter* pada masing-masing kapasitor dengan nilai kapasitif yang berbeda seperti pada Gambar 4.6.



Gambar 4.6. Pengukuran Menggunakan Fluke 337

Pengujian dilakukan dengan cara mengambil data pengukuran sebanyak 10 kali pada masing-masing kapasitor. Hasil pengukuran dapat dilihat pada

Tabel 4.6.

Tabel 4.6. Tabel Pengukuran Arus Maksimum Kapasitor

Percobaan ke- \ Kapasitansi	2	3	5	10	20	35	60
1	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
2	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
3	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
4	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
5	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
6	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
7	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
8	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
9	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A
10	0.2A	0.3A	0.4A	0.8A	1.5A	2.7A	4.7A

Dari hasil pengukuran di atas, maka ditentukan proteksi arus yang digunakan terdiri dari 3 buah MCB 6A dengan rincian MCB pertama untuk kapasitor 60 μ F dengan total arus nominal sebesar 4.7A, MCB kedua untuk kapasitor 35 μ F dan 20 μ F dengan total arus nominal sebesar 4.2A, serta MCB ketiga untuk kapasitor 10 μ F, 5 μ F, 5 μ F, 3 μ F, dan 2 μ F dengan total arus nominal sebesar 2.1A.