

# SISTEM PEMANTAUAN METERAN AIR BERBASIS OPTICAL CHARACTER RECOGNITION

Ivan Sanjaya<sup>1</sup>, Indar Sugiarto<sup>1</sup>

<sup>1</sup>Program Studi Teknik Elektro, Universitas Kristen Petra  
Jl. Siwalankerto No. 121-131, Surabaya, 60236, Indonesia  
E-Mail: sanjaya.ivan15@gmail.com

**Abstrak** – Petugas pencatat meteran air perlu berkeliling ke rumah pelanggan setiap bulannya untuk mencatat angka meteran air secara manual. Sebagian pelanggan memiliki meteran air yang terletak di bagian dalam rumah. Oleh karena itu pelanggan perlu mencatat angka meteran dan menempelkannya di depan rumah agar petugas dapat melihat angka tersebut. Terkadang pelanggan dapat lupa untuk mencatat meteran mereka secara manual.

Teknologi *Optical Character Recognition* dapat digunakan untuk mengenali angka yang terdapat pada meteran air. Dengan memanfaatkan teknologi *Internet of Things*, hasil pengenalan angka meteran air dapat disimpan dalam *database* melalui koneksi nirkabel. Kedua teknologi tersebut dapat digunakan untuk melakukan pencatatan dan penyimpanan angka meteran air secara otomatis. Kemudian angka meteran tersebut dapat ditampilkan agar petugas pencatat dapat melihat dengan menggunakan *7-segment display*. Setelah itu dengan menggunakan *dashboard*, pelanggan dapat memantau penggunaan air mereka dan dapat mendeteksi kemungkinan kebocoran pipa.

Dari hasil pengujian, pembacaan angka meteran air memiliki akurasi 100% untuk pembacaan bagian m<sup>3</sup>. Sistem tidak dapat mengkalifikasikan angka meteran melakukan pembacaan bagian liter. Pengisian baterai 3000mAh dengan arus 1A memerlukan waktu 5 jam.

**Kata Kunci**— ESP32-CAM, *Optical Character Recognition*, *Raspberry Pi Zero*, *Dashboard*, *7-Segment*, Meteran Air

## I. PENDAHULUAN

Setiap rumah dipertanian mendapat pemasukan air dari perusahaan air daerah. Perusahaan air daerah meminjamkan meteran air kepada pelanggan untuk mengukur penggunaan air pelanggan. Setiap bulannya, pelanggan perlu membayar tagihan air yang telah dikonsumsi. Di negara berkembang umumnya perusahaan memiliki pegawai yang bertugas untuk mencatat meteran dengan berkeliling ke setiap rumah pelanggan. Terkadang meteran air pelanggan terletak dalam pagar bahkan dalam rumah pelanggan sehingga petugas kesulitan menjangkau meteran tersebut. Oleh karena itu, pelanggan perlu mencatat angka meteran dan menempel catatan tersebut di depan rumah. Terkadang tersebut lupa untuk mencatat angka meteran mereka sehingga mempersulit pekerjaan petugas pencatat.

Pelanggan tidak mungkin untuk memperhatikan meteran air terus-menerus setiap harinya. Oleh karena itu, pelanggan hanya akan mengetahui konsumsi air mereka pada saat membayar tagihan. Kurangnya perhatian terhadap penggunaan air dapat mengakibatkan pelanggan mengkonsumsi air lebih dari yang diperlukan. Hal itu mengakibatkan setiap bulannya

pelanggan perlu membayar tagihan air yang banyak. Selain itu dengan memperhatikan meteran air, pelanggan juga dapat memprediksi kemungkinan terjadinya kebocoran pipa. Oleh karena itu dibutuhkan sistem untuk memantau meteran air otomatis secara berkala. Terdapat beberapa sistem yang menginspirasi pembuatan sistem ini. Berikut beberapa referensi untuk pembuatan sistem pemantauan meteran air:

### A. *Monitoring Meteran Air Pdam Menggunakan Database Web Server*

Referensi ini merupakan penelitian sistem monitoring meteran air dengan membaca jumlah debit air yang digunakan. Sistem tersebut menggunakan *hall effect sensor* untuk membaca laju air. Kemudian sinyal dari sensor diproses menggunakan Arduino Uno. Setelah itu data yang telah diproses ditampilkan pada LCD dan *database web server* menggunakan komunikasi SIM 900 [1]. Akan tetapi, penggunaan *hall effect sensor* tidak dapat digunakan, karena diperlukan pemasangan magnet dalam meteran air. Pembongkaran meteran air tidak diperbolehkan oleh pihak perusahaan air.

### B. *Pencatatan Meter Air dengan Long ShortTerm Memory Recurrent Neural Network*

Referensi ini merupakan penelitian sistem pembacaan meter air yang dapat bekerja pada gawai tanpa menggunakan Internet. Sistem bekerja dengan memanfaatkan *Optical Character Recognition* (OCR) dengan metode *Long Short Term Memory Recurrent Neural Network* (LSTM-RNN). LSTM-RNN adalah sebuah metode klasifikasi yang termasuk jaringan saraf tiruan dengan umpan balik. Penelitian tersebut menunjukkan bahwa sistem pembacaan meter air dapat bekerja tanpa menggunakan koneksi Internet [2]. Akan tetapi pada sistem ini pencatatan angka meteran tidak dilakukan oleh sistem secara otomatis. Pelanggan perlu menggunakan gawai mereka untuk mengambil gambar meteran.

Sistem pemantauan pada penelitian ini merupakan kombinasi dan penambahan sistem dari kedua referensi di atas. Terdapat berbagai macam metode OCR, salah satunya adalah metode *Deep Learning Convolutional Neural Network* (CNN). Metode CNN memiliki hasil paling signifikan dalam pengenalan gambar/citra dan dapat melakukan pengenalan citra dengan akurasi yang menyaingi manusia pada *dataset* tertentu [3]. Oleh karena itu, metode CNN dapat dimanfaatkan untuk mendeteksi angka meteran air. Kemudian dengan memanfaatkan teknologi *Internet of Things* (IoT), angka meteran air yang terdeteksi dapat disimpan ke dalam *database* dari jarak jauh. Berikutnya dalam sistem pemantauan

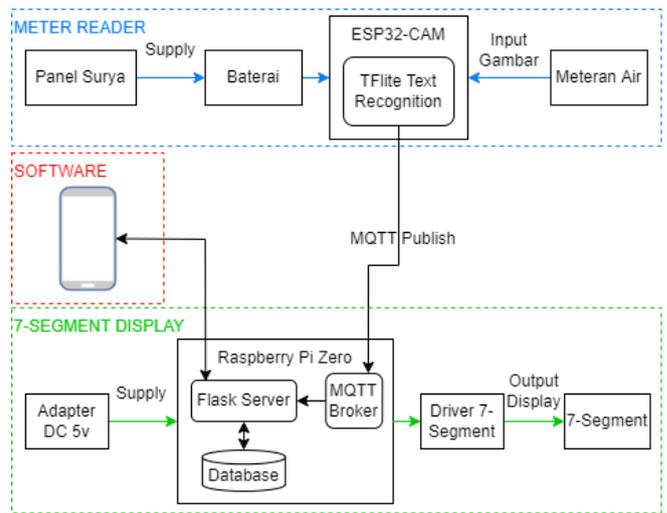
membutuhkan *dashboard* untuk melihat data-data yang telah tersimpan dalam *database*. Selain itu dibutuhkan *display* agar petugas pencatat dapat melihat angka meteran tanpa perlu mendekati ke meteran air.

Berdasarkan kebutuhan-kebutuhan dalam sistem pemantauan, maka dapat memanfaatkan mikrokontroler yang dapat terhubung dengan jaringan nirkabel. ESP32-CAM merupakan mikrokontroler yang dapat disambungkan dengan kamera dan memiliki kemampuan untuk masuk dalam jaringan nirkabel. Oleh karena itu ESP32-CAM dapat digunakan untuk mengambil gambar meteran air dan mengirim data melalui koneksi nirkabel. Selain itu, dengan menggunakan model dari Tensor Flow Lite (Tflite), ESP32-CAM mampu untuk melakukan proses OCR pada gambar meteran air. Namun ESP32-CAM membutuhkan sumber daya tersendiri seperti baterai, karena letak meteran air yang biasanya jauh dari jangkauan sumber listrik. Selain itu dibutuhkan panel surya untuk mengisi baterai secara otomatis.

Raspberry Pi Zero merupakan komputer berukuran kecil yang dilengkapi dengan koneksi nirkabel dan dilengkapi dengan banyak *pin General Purpose Input/Output (GPIO)*. GPIO adalah media komunikasi atau sebagai akses untuk mengirim atau menerima data. Dengan demikian Raspberry Pi Zero dapat digunakan untuk menerima data dari ESP32-CAM dan sebagai tempat penyimpanan data/*database*. Selain itu Raspberry Pi Zero juga dapat digunakan sebagai *server* untuk *dashboard* untuk pemantauan meteran air. Kemudian Raspberry Pi Zero yang memiliki banyak *pin GPIO* mampu diprogram untuk menampilkan data pada *display*. Untuk *display* dari angka meteran air, *7-segment* dapat digunakan untuk menampilkan angka-angka meteran air.

## II. DESKRIPSI SISTEM

Sistem ini memiliki *input* berupa kamera yang terpasang pada mikrokontroler ESP32-CAM. Kamera tersebut berfungsi untuk mengambil gambar dari angka meteran air. Gambar tersebut akan diolah oleh ESP32-CAM menjadi data berupa teks menggunakan metode OCR. ESP32-CAM tersebut menggunakan baterai sebagai sumber daya dan panel surya yang berfungsi untuk mengisi daya baterai. Kemudian hasil OCR tersebut dikirim ke *server* yang berjalan dalam Raspberry Pi Zero dengan menggunakan komunikasi MQTT. Berikutnya data yang diterima oleh *server* dimasukkan ke dalam *database*. Selanjutnya data yang disimpan dalam *database* ditampilkan ke *7-segment display* melalui rangkaian *driver* dan ditampilkan ke *dashboard* pada gawai. Penjelasan digambarkan pada diagram perancangan melalui Gambar 1.

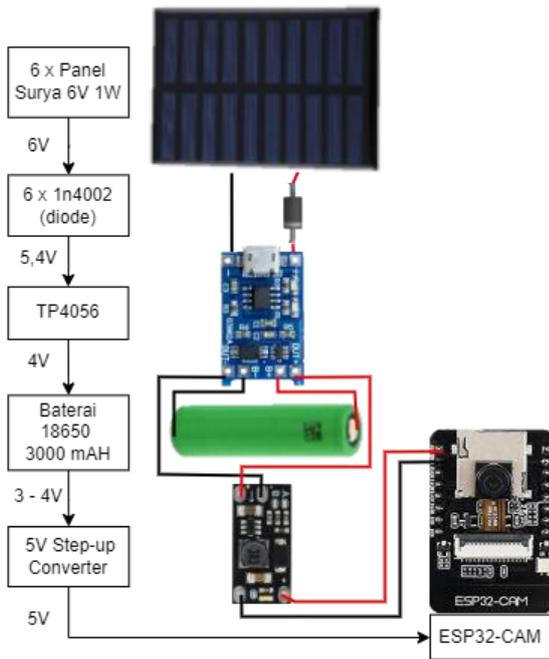


Gambar 1. Garis besar desain sistem

### A. Rangkaian Hardware Pada ESP32-CAM

Rangkaian *hardware* yang terhubung pada ESP32-CAM terdiri dari panel surya, dioda, *charge controller*, dan *step-up converter*. Panel surya yang digunakan adalah enam panel yang memiliki kapasitas sebesar enam volt satu watt puncak. Oleh karena itu panel surya tersebut menghasilkan enam watt puncak atau satu ampere puncak ketika panel surya mendapat sinar yang cukup. Kemudian keluaran dari panel surya masuk ke dioda sebagai pengaman agar tidak ada arus yang masuk kembali ke panel surya. Berikutnya keluaran dari dioda masuk ke modul TP4056 yang terhubung dengan baterai.

Modul TP4056 digunakan sebagai *charge controller* untuk baterai. Modul tersebut memiliki kemampuan untuk memotong arus yang masuk ke baterai ketika baterai yang diisi telah mencapai tegangan empat volt. Selain itu modul tersebut dapat memotong arus yang keluar dari baterai ketika tegangan baterai telah turun hingga tiga volt. Selanjutnya modul TP4056 juga terhubung dengan modul *step-up converter* lima volt untuk menaikkan tegangan dari TP4056 maupun baterai menjadi lima volt. Hal tersebut diperlukan karena input dari ESP32-CAM adalah lima volt. Penjelasan rangkaian *hardware* digambarkan melalui Gambar 2.

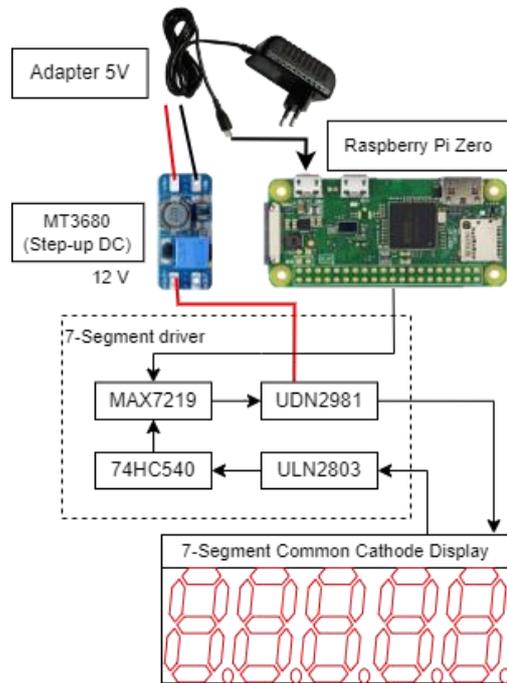


Gambar 2. Hardware meteran air

### B. Rangkaian Hardware Pada Raspberry Pi Zero

Rangkaian hardware yang terhubung pada Raspberry Pi Zero terdiri dari Adapter 5V, step-up converter, 7-segment driver, dan 7-segment common cathode. Adapter 5V digunakan sebagai sumber daya untuk Raspberry Pi Zero. Raspberry Pi Zero digunakan untuk memprogram rangkaian 7-segment. Sebelum terhubung dengan 7-segment, keluaran dari Raspberry Pi Zero masuk ke rangkaian 7-segment driver. Rangkaian driver tersebut digunakan untuk mengontrol 7-segment yang membutuhkan tegangan yang lebih besar dari lima volt. Raspberry Pi Zero akan mengirimkan sinyal ke rangkaian driver menggunakan protokol komunikasi Serial Peripheral Interface (SPI).

Rangkaian detail driver dalam penelitian ini diambil dari <https://embedded-lab.com/blog/high-voltage-seven-segment-led-display-driver-with-spi-interface/>. Rangkaian ini menggunakan IC MAX7219 yang hanya dapat mengontrol 7-segment common cathode yang membutuhkan tegangan lima volt. Kemudian terdapat IC UDN2981 yang berfungsi untuk meningkatkan tegangan keluaran dari IC MAX7219. IC UDN2981 membutuhkan tegangan sesuai tegangan komponen yang akan disuplainya. Oleh karena itu dibutuhkan MT3680 untuk meningkatkan tegangan adapter menjadi dua belas volt sesuai kebutuhan 7-segment yang dipakai. Berikutnya terdapat juga IC ULN2803 yang berfungsi untuk current sink/pengaman IC MAX7219. Selain itu juga terdapat IC 74HC540 yang digunakan untuk mengubah keluaran ULN2803 yang active high menjadi active low sebelum masuk ke IC MAX7219. Penjelasan rangkaian yang terhubung dengan Raspberry Pi Zero digambarkan melalui Gambar 3.



Gambar 3. Hardware Raspberry Pi Zero dan 7-segment

### C. Software OCR Pada ESP32-CAM

Software/program OCR yang berjalan pada ESP32-CAM adalah program dari proyek publik yang diunduh dari <https://github.com/jomjol/AI-on-the-edge-device>. Proyek OCR tersebut berisikan tentang program pembacaan meteran air menggunakan ESP32-CAM dengan metode klasifikasi gambar CNN. Proyek tersebut juga menyediakan program webserver yang berjalan pada ESP32-CAM untuk mempermudah melakukan konfigurasi program. Konsep kerja program pada ESP32-CAM adalah mula-mula program menjalin koneksi dengan WiFi. Setelah ESP32-CAM telah terkoneksi dengan WiFi, ESP32-CAM akan menyalakan lampu LED untuk penerangan agar gambar/citra yang diambil memiliki tingkat kontras yang tinggi. Kemudian ESP32-CAM mengambil citra meteran air. Selanjutnya citra dirapikan dan diubah posisinya untuk mempermudah proses object detection, setelah itu lampu LED dimatikan.

Cara kerja dari object detection adalah program akan mendeteksi pola tanda dari citra gambar yang telah diubah posisinya. Tanda yang dideteksi oleh program memiliki tujuan agar letak dari Region of Interest (ROI) angka meteran tidak berubah. ROI dari angka meteran dan penandaan yang digunakan, sebelumnya telah dikonfigurasi melalui webserver ESP32-CAM. Kemudian ROI angka meteran digunakan untuk menentukan wilayah angka meteran yang akan dipotong/diambil. Berikutnya angka meteran yang telah diambil akan diklasifikasikan oleh model training CNN. Model tersebut sudah termasuk dalam proyek publik yang telah disebutkan, dengan menggunakan dataset sebanyak 1000 gambar. Setelah itu pada proses post processing, hasil klasifikasi disatukan untuk dikirim ke Raspberry Pi Zero melalui protokol komunikasi MQTT. Kemudian program diberi delay, yang dapat diatur melalui webserver ESP32-CAM. Penjelasan konsep kerja program ESP32-CAM digambarkan pada diagram perancangan melalui Gambar 4.



Gambar 4. Diagram alir program ESP32-CAM

**D. Software Pada Raspberry Pi Zero Bagian Dashboard**

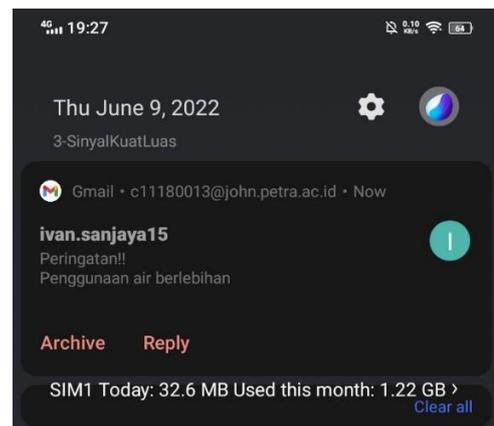
Software/program yang digunakan untuk menyimpan data dalam database dan menampilkan dashboard adalah Flask server. Program menyimpan data yang dikirim oleh ESP32-CAM melalui MQTT broker akan disimpan di dalam database SQL pada Raspberry Pi Zero. Kemudian data dari database tersebut akan ditampilkan pada dashboard yang dapat diakses melalui gawai. Isi dari dashboard adalah grafik kenaikan angka meteran air setiap harinya, grafik penggunaan air harian, dan grafik penggunaan air bulanan. Pemilihan/penggeseran waktu pada garfik dapat dilakukan dengan memasukan tanggal atau bulan atau tahun pada kolom yang ada di tombol settings. Selain itu dashboard juga memiliki fitur untuk memunculkan notifikasi peringatan menggunakan e-mail ketika air yang dibaca telah melebihi dari angka yang telah ditetapkan pengguna. Tampilan dashboard pada gawai digambarkan melalui Gambar 5, sedangkan tampilan kolom pada settings dapat dilihat pada Gambar 6. Kemudian notifikasi dari e-mail ditampilkan pada Gambar 7.



Gambar 5. Tampilan dashboard grafik meteran air



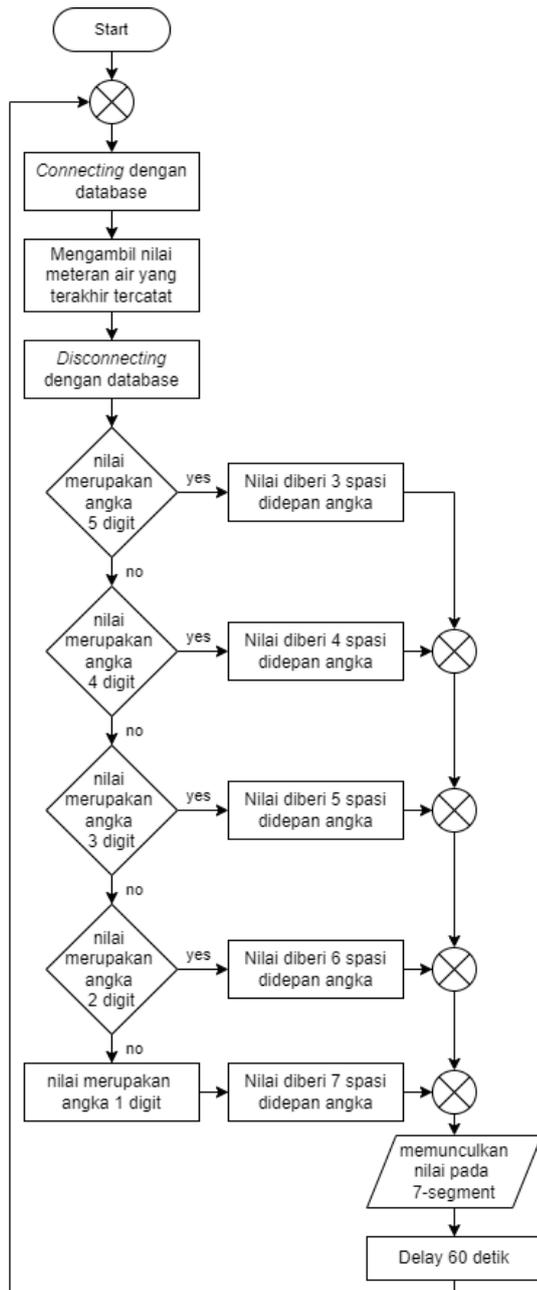
Gambar 6. Tempat pengisian waktu dan penggunaan air maksimal



Gambar 7. E-mail peringatan ketika air melebihi target

E. Software Pada Raspberry Pi Zero Bagian 7-Segment Display

Program kontrol 7-segment mula-mula bekerja dengan menjalin koneksi dengan database. Kemudian program mengambil nilai angka meteran air yang terakhir tercatat pada database, lalu program memutuskan koneksi dengan database. Berikutnya angka meteran akan diubah dengan memberi spasi/jarak di depan angka. Pengubahan tersebut adalah jumlah spasi dengan digit angka adalah 8, contoh bila digit angka adalah 5 jumlah spasi di depan angka sebesar 3 spasi. Setelah penambahan jumlah digit, hasil tersebut dikirimkan ke rangkaian driver untuk dimunculkan ke 7-segment display. Setelah itu, program akan dibuat delay selama 60 detik, lalu kembali ke awal yaitu pada koneksi dengan database. Penjelasan digambarkan pada diagram alir melalui Gambar 8.



Gambar 8. Diagram alir program 7-segment

III. PENGUJIAN SISTEM DAN ANALISA

Pengujian ini bertujuan untuk mengetahui performa dari sistem yang telah dibuat. Pengujian pertama adalah pengujian akurasi pembacaan angka meteran menggunakan ESP32-

CAM. Pengujian kedua adalah pengujian waktu pembacaan angka meteran menggunakan ESP32-CAM. Pengujian ketiga adalah pengujian keefektifan solar panel dalam pengisian daya baterai. Pengujian keempat adalah pengujian waktu pengosongan daya baterai saat digunakan oleh ESP32-CAM.

A. Pengujian Akurasi Pembacaan Meteran Menggunakan ESP32-CAM

Pengujian ini bertujuan untuk mengetahui tingkat akurasi pembacaan. Pada pengujian akurasi, pengujian dilakukan dengan dua cara. Pengujian akurasi pertama adalah pembacaan meteran air secara berulang-ulang setiap tiga menit sekali sebanyak sepuluh kali. Angka meteran air yang diuji adalah empat angka awal dari enam angka meteran air. Hasil dari pengujian tersebut ditampilkan pada Tabel 1. Untuk pengujian kedua adalah pembacaan meteran air sekali setiap hari selama 30 hari. Angka meteran air yang diuji juga empat angka awal dari enam angka meteran air. Hasil dari pengujian tersebut ditampilkan pada Tabel 2.

Tabel 1. Hasil Pembacaan berulang-ulang

	Jumlah pembacaan									
	1	2	3	4	5	6	7	8	9	10
Angka meteran sesungguhnya	36	36	36	36	36	36	36	36	36	36
Angka meteran pembacaan	36	36	36	36	36	36	36	36	36	36

Tabel 2. Hasil pembacaan setiap hari

Tanggal pembacaan	Angka meteran sesungguhnya	Angka meteran pembacaan
Mei 11	30	30
Mei 12	30	30
Mei 13	30	30
Mei 14	30	30
Mei 15	31	31
Mei 16	31	31
Mei 17	31	31
Mei 18	31	31
Mei 19	32	32
Mei 20	32	32
Mei 21	32	32
Mei 22	32	32
Mei 23	33	33
Mei 24	33	33
Mei 25	33	33
Mei 26	33	33
Mei 27	34	34
Mei 28	34	34
Mei 29	34	34
Mei 30	34	34
Mei 31	34	34
Juni 1	35	35
Juni 2	35	35
Juni 3	35	35
Juni 4	35	35
Juni 5	36	36
Juni 6	36	36
Juni 7	36	36
Juni 8	36	36
Juni 9	36	36

Dari hasil pengujian pada Tabel 1 dapat dilihat bahwa hasil dari pembacaan angka meteran air adalah konstan. Hasil pembacaan selalu sesuai dengan angka pada meteran air sesungguhnya. Dari hasil pengujian pada Tabel 2 dapat dilihat bahwa hasil pembacaan air selama 30 hari selalu berhasil. Prosentase keberhasilan pembacaan adalah 100% untuk pembacaan empat angka awal. Empat angka tersebut adalah angka yang melambangkan volume air dalam satuan m<sup>3</sup>.

**B. Pengujian Waktu Pembacaan Meteran Menggunakan ESP32-CAM**

Pengujian waktu pembacaan meteran bertujuan untuk mengetahui waktu dari proses yang bekerja dalam ESP32-CAM. Proses tersebut adalah mulai dari mengambil gambar hingga mengirim data hasil proses melalui MQTT. Pengujian dilakukan dengan melihat *timestamp* proses pengambilan gambar hingga pengiriman data dari ESP32-CAM. Pengujian pengambilan dilakukan sebanyak sepuluh kali. Dari pengujian tersebut didapatkan bahwa waktu yang diperlukan mulai dari pengambilan gambar hingga mengirim data melalui MQTT adalah 55 detik hingga 57 detik.

**C. Pengujian Keefektifan Solar Panel Dalam Pengisian Daya Baterai**

Pengujian ini bertujuan untuk mengetahui waktu yang dibutuhkan solar panel untuk mengisi baterai. Pengujian pengisian baterai dilakukan dengan mengukur arus yang keluar dari panel surya. Pengukuran dilakukan di dalam ruangan menggunakan lampu dengan daya besar sebagai sumber cahaya. Kemudian untuk mengukur cahaya, digunakan gawai yang memiliki sensor untuk mengukur daya pancar cahaya. Semakin dekat gawai dengan sumber cahaya, maka angka yang ditunjukkan akan semakin besar. Pengukuran daya pancar cahaya juga dilakukan dengan mengukur pada jam-jam tertentu. Sekitar Pk 10.00 hingga Pk 14.00, ketika gawai menghadap ke atas atau sedikit berawan, angka menunjukkan angka sekitar 30000 LUX. Akan tetapi pada waktu yang sama, ketika gawai dimiringkan persis menghadap matahari, angka dapat menunjukkan angka 60000 LUX. Ketika matahari terhalang awan, angka pada gawai menunjukkan angka antara 8000 LUX hingga 3000 LUX. Setelah dilakukan beberapa kali pengukuran dengan lampu dan pengukuran dengan matahari, maka perbandingan yang didapatkan dapat dilihat pada Tabel 3.

Tabel 3. Pengaruh cahaya terhadap arus dari panel surya

Daya pancar cahaya	Arus dari panel surya
56000 LUX	1000 mA
30000 LUX	900 mA
20000 LUX	450 mA
10000 LUX	270 mA
5000 LUX	130 mA

Untuk mengetahui waktu pengisian baterai, perlu mencari tahu arus yang masuk ke dalam baterai. Setelah diukur, maka didapatkan arus yang masuk ke ESP32-CAM berkisar 200mA. Maka arus yang masuk ke baterai adalah arus dari Tabel 3 dikurangi 200mA. Kemudian jika arus dari panel surya sebesar 950mA, maka arus yang masuk ke baterai adalah 750mA. Selanjutnya jika kapasitas baterai adalah 3000mAh, maka waktu yang dibutuhkan untuk mengisi daya baterai adalah 3000mAh dibagi dengan 750 mA, dan hasilnya adalah 4 jam.

Akan tetapi setelah dilakukan percobaan secara langsung, waktu yang dibutuhkan berkisar 3 jam 40 menit.

**D. Pengujian Waktu Pengosongan Daya Baterai Saat Digunakan Oleh ESP32-CAM**

Pengujian ini bertujuan untuk mengetahui waktu yang dibutuhkan ESP32-CAM untuk menghabiskan daya baterai. Pengujian pengosongan dilakukan dengan memperhatikan arus yang keluar dari baterai. Pada saat ESP32-CAM menarik arus 162mA, baterai mengeluarkan arus 288mA. Akan tetapi ketika ESP32-CAM menarik arus 92mA, arus yang keluar adalah 162mA. Jika diambil rata-rata pengeluaran arus baterai tersebut, maka arus yang keluar adalah 225mA. Oleh karena itu jika kapasitas baterai adalah 3000mA, maka waktu yang dibutuhkan dalam pengosongan daya baterai adalah 3000mAh dibagi 225mA, dan hasilnya adalah 13 jam 20 menit. Akan tetapi setelah diuji secara langsung, waktu yang didapat berkisar 10 jam 40 menit hingga 11 jam.

**IV. KESIMPULAN**

Kesimpulan yang dapat diambil dari pengujian adalah sebagai berikut:

1. Sistem pembacaan meteran air memiliki tingkat keberhasilan/akurasi yang tinggi bila angka yang diukur adalah empat digit meteran awal. Apabila angka meteran yang dibaca adalah dua digit akhir, pembacaan memiliki tingkat kegagalan yang tinggi. Tingkat keberhasilan dari hasil percobaan pembacaan empat digit awal meteran adalah 100%. Hasil pembacaan sesuai dengan angka meteran sesungguhnya.
2. Waktu yang diperlukan untuk pembacaan meteran air hingga mengirim data ke Raspberry Pi Zero adalah 55 sampai 57 detik.
3. Pengisian daya baterai membutuhkan waktu yang lebih lama dibandingkan dengan waktu perhitungan. Perbedaan waktu tersebut berkisar 30 hingga 40 menit. Selain itu, kecepatan pengisian baterai yang paling tinggi adalah pada Pk 09.30 hingga 14.30 ketika langit cerah tidak berawan.
4. Pengosongan daya baterai membutuhkan waktu yang lebih cepat dibandingkan dengan waktu dari perhitungan. Perbedaan waktu tersebut berkisar 2 jam 20 menit hingga 3 jam 40 menit.

**DAFTAR PUSTAKA**

[1] A. H. Raditya and W. S. Prayogo, "MONITORING METERAN AIR PDAM MENGGUNAKAN DATABASE WEB SERVER DI INDUSTRI," 2016. Accessed: Jul. 06, 2022. [Online]. Available: <https://repository.its.ac.id/id/eprint/621>

[2] V. G. Utomo, A. P. R. Pinem, and B. V. Christoko, "Pengenalan Karakter Optis untuk Pencatatan Meter Air dengan Long Short Term Memory Recurrent Neural Network," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 132–138, Feb. 2021, doi: 10.29207/RESTI.V5I1.2807.

[3] Wayan Suartika E. P, A. Y. Wijaya, and R. Soelaiman, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) pada Caltech 101," *JURNAL TEKNIK ITS*, vol. 5, no. 1, pp. A65–A69, 2016.