



## PERANCANGAN BEL SEKOLAH OTOMATIS MENGGUNAKAN MIKROKONTROLER AVR ATMEGA 328 DAN REAL TIME CLOCK DS3231

Dewi Kusumawati<sup>1)</sup>, Bayu Angga Wiryanto

STMIK Bina Mulia Palu  
Website: [stmik-binamulia.ac.id](http://stmik-binamulia.ac.id)

### ABSTRAK

Kemajuan bangsa sangat didukung oleh kemajuan dalam bidang pendidikan yang membina sumber daya penerus pembangunan bangsa. Pendidikan formal dari SD hingga SMA berlangsung menurut jadwal yang telah ditentukan dimana awal dan akhir setiap pelajaran ditandai dengan bunyi bel sekolah. Umumnya pengoperasian bel sekolah dilakukan manual oleh guru piket atau petugas sesuai jadwal yang ditentukan dengan cara melihat jam dinding dan menekan tombol bel. Namun terkadang mereka lupa atau terlambat membunyikan bel sekolah sehingga terjadi ketidak efisienan waktu dan tenaga dalam proses belajar mengajar. Karenanya penelitian ini akan merancang dan merealisasikan bel sekolah otomatis yang dapat berbunyi secara otomatis sesuai waktu yang sudah ditentukan sehingga dapat mengefisienkan waktu dan tenaga dalam menunjang kegiatan belajar mengajar. Penelitian ini menggunakan pendekatan eksperimen. Pengumpulan data melalui pengamatan, wawancara dan studi pustaka. Dengan metode *extreme programing*, penelitian ini membuat perancangan bel sekolah otomatis menggunakan RTC DS3231 dan mikrokontroler AVR ATMEGA 328 sebagai pengontrolnya. Hasil penelitian ini menunjukkan bel sekolah otomatis yang rancang dapat bekerja dengan baik karena bel berdering sesuai dengan dengan jadwal pembelajaran yang ditentukan. Hasil penelitian ini dapat dikembangkan antara lain dengan membuat desain *box* yang lebih minimalis, menambahkan *fitur-fitur* pada alat sehingga mempermudah penggunaan, menambahkan *suplay* daya cadangan atau batere sebagai cadangan listrik, dan gunakan nada dering yang lebih menarik.

**Kata Kunci:** *Bel Otomatis, Mikrokontroler ATMEGA 328, Real Time Clock DS3231.*

### 1. Pendahuluan

Kemajuan suatu bangsa sangat didukung oleh kemajuan dalam bidang pendidikan yang membina generasi muda sebagai sumber daya penerus pembangunan bangsa. Hal ini telah menjadi pokok perhatian Pemerintah Indonesia yang nampak saat Pemerintah mengalokasikan minimal 20% dana APBN/APBD untuk bidang pendidikan dasar sembilan tahun.

Pendidikan formal yang dilaksanakan sekolah mulai dari tingkat SD hingga SMA berlangsung menurut jadwal atau daftar pelajaran yang telah ditentukan masing-masing sekolah selama enam hari dalam seminggu. Pada awal dan akhir setiap pelajaran biasanya ditandai dengan berbunyinya bel sekolah, begitu juga saat waktu istirahat maupun pulang sekolah.

Umumnya pada sebagian besar sekolah pengoperasian bel sekolah ini dilakukan secara manual, yaitu ada guru piket atau petugas sekolah yang harus selalu siap untuk membunyikan bel sekolah sesuai dengan jam-jam yang telah ditentukan dengan cara melihat waktu pada jam dinding dan menekan tombol bel.

Namun pada prakteknya, guru piket atau petugas sekolah terkadang lupa atau terlambat

untuk membunyikan bel sekolah dengan alasan satu dan lain hal sehingga akan terjadi ketidak efisienan waktu maupun tenaga dalam proses belajar mengajar.

Kondisi diatas menunjukkan bahwa sekolah membutuhkan suatu bel sekolah otomatis yang akan berbunyi secara otomatis sesuai jadwal waktu yang telah ditentukan sehingga dapat mengefisienkan waktu maupun tenaga dalam menunjang kegiatan belajar mengajar.

Hal ini menyebabkan Peneliti tertarik untuk merancang dan merealisasikan sebuah contoh bel sekolah otomatis berbasis mikrokontroler *Atmega 328* dan *Real Time Clock (RTC) DS3231* yang dihubungkan ke *port mikrokontroler* dengan memakai sistem *bus Inter Integrated Circuit (I2C)* yang hanya menggunakan dua konduktor/kabel saja.

Bel sekolah otomatis ini juga akan dilengkapi dengan layar *Liquid Crystal Display (LCD)* sehingga *input* data dapat nampak untuk memudahkan operator pada saat memasukkan jadwal sekolah maupun perubahannya.

## 2. Kerangka Teoritis

### 2.1 Perancangan

Perancangan merupakan suatu upaya untuk menemukan komponen-komponen fisik yang tepat dari sebuah struktur fisik [1].

Pengertian lain menyatakan perancangan adalah penggambaran, perencanaan dan pembuatan atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi [2].

Dengan demikian dapat dikatakan bahwa perancangan terdiri dari tiga tahap, mulai penggambaran, perencanaan, dan pembuatan atau pengaturan elemen-elemen sehingga menjadi struktur fisik yang memiliki fungsi tertentu.

### 2.2 Bel

Bel atau lonceng adalah suatu peralatan sederhana yang digunakan untuk menciptakan bunyi. Bentuk umumnya adalah sebuah tabung dengan salah satu sisi terbuka sehingga menimbulkan akan suara saat dipukul. Alat yang digunakan untuk memukul dapat berupa pemukul panjang yang digantung di dalam lonceng tersebut atau pemukul yang terpisah.

Bel atau lonceng dapat menghasilkan bunyi dikarenakan adanya gelombang yang dihasilkan dari benturan kedua logam (bandul dan badan lonceng) ketika digoyangkan. Gelombang tersebut akan menggetarkan udara disekitarnya dan perambatan getaran membentuk pola rapatan dan regangan. Pola rapatan dan regangan inilah yang menggetarkan udara disekitarnya dan menjalar ke segala arah.

Pada zaman dahulu lonceng digunakan sebagai penanda waktu ibadah, sebagai bagian dari perangkat ritual keagamaan, atau untuk mengabarkan suatu berita kepada masyarakat. Sedangkan di masa modern lonceng sudah jarang digunakan, namun beberapa tempat masih menggunakannya seperti pada sekolah untuk memberikan tanda pergantian jam pelajaran/ujian, waktu istirahat, masuk kelas, pulang sekolah, atau pengumuman [3].

### 2.3 Otomatis

Arti kata otomatis adalah secara otomat; bekerja sendiri; dengan sendirinya. Sedangkan kata mengotomatiskan berarti menjadikan sesuatu otomatis [4].

Jadi, kata otomatis mengandung pengertian sebagai sesuatu yang bekerja dengan sendirinya, sedangkan mengotomatiskan berarti membuat sesuatu dapat bekerja dengan sendirinya. Maksudnya adalah suatu perangkat atau peralatan yang dapat bekerja secara otomat atau bekerja sendiri sesuai dengan fungsinya, tanpa harus menunggu perintah/instruksi dari luar.

Dengan demikian dapat dikatakan bahwa sistem otomatis merupakan susunan beberapa perangkat yang masing-masing memiliki fungsi yang berbeda namun saling berkaitan sehingga membentuk satu kesatuan untuk melaksanakan pekerjaan sesuai dengan fungsinya secara otomat atau dengan sendirinya.

Jadi, beberapa sistem kecil dapat digabung menjadi sebuah sistem yang besar dan kompleks. Contohnya adalah tubuh manusia merupakan sebuah sistem besar yang terdiri dari gabungan beberapa sistem kecil seperti sistem pernafasan, sistem pencernaan, sistem saraf, sistem peredaran darah, dan lain-lain yang otomat sehingga tubuh yang dapat berfungsi dengan baik.

### 2.4 Arduino IDE

Arduino adalah mikrokontroler *board* atau papan rangkaian elektronik *open source* yang berasal dari Italia. Mikrokontroler itu sendiri adalah *chip* atau *Integrated Circuit* (IC) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca *input*, memprosesnya dan menghasilkan *output* sesuai yang diinginkan. Jadi mikrokontroler bertugas sebagai 'otak' yang mengendalikan *input*, proses dan *output* rangkaian elektronik. Karena komponen utama Arduino adalah mikrokontroler, maka Arduino dapat diprogram menggunakan komputer sesuai kebutuhan [5].

Komponen lain dalam Arduino adalah *software Integrated Development Environment* (IDE) yang merupakan lingkungan terintegrasi untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui *software* ini Arduino melakukan pemrograman untuk fungsi-fungsi yang ada melalui sintaks pemrograman.

Arduino IDE dibuat dari bahasa pemrograman JAVA dan dilengkapi *library C/C++* (biasa disebut *Wiring*) yang membuat operasi *input* dan *output* menjadi lebih mudah. Arduino IDE menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual, IC mikrokontroler Arduino telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler* Arduino dengan mikrokontroler.

Program yang ditulis dengan menggunakan Arduino IDE disebut *Sketch* yang ditulis dalam suatu editor teks dan disimpan dalam *file* dengan ekstensi *.ino*. Teks editor pada Arduino IDE memiliki fitur-fitur seperti *cutting/paste* dan *searching/replacing* sehingga memudahkan pengguna dalam menulis program.

Pada Arduino IDE terdapat semacam *message box* berwarna hitam yang berfungsi menampilkan status seperti pesan *error*, *compile*, dan *upload* program. Di bagian bawah paling kanan menunjukkan *board* yang terkonfigurasi beserta *COM Ports* yang digunakan.

Tampilan *software* Arduino IDE sebagai berikut [6]:



Gambar 1 *Software* Arduino IDE

Karena itu saat membahas Arduino maka ada dua hal yang terlintas, yaitu *hardware* dan *software* karena didalam Arduino kedua bagian ini seakan-akan menjadi satu kesatuan yang utuh dan tidak dapat dipisahkan.

*Hardware* Arduino sudah terintegrasi untuk memuat kode baru ke *hardware* hanya menggunakan kabel USB yang dihubungkan dari modul Arduino ke *Personal Computer* (PC) atau *notebook* [7].

## 2.5 Mikrokontroler AVR ATmega 328

Mikrokontroler adalah perkembangan dari mikroprosesor. Mikrokontroler adalah *Single Chip Mikrokomputer* (SCM), yaitu sebuah komputer yang di paket dengan sebuah *chip* (IC). Didalamnya sudah terdapat RAM, ROM atau EPROM, *timer*, asilator, ADC, *buffer I/O port*, saluran alamat, serta saluran data sehingga dapat bekerja dengan tepat dan mampu melakukan pekerjaan yang rumit walaupun hanya dengan rangkaian sangat sederhana [8].

Hal ini membuat mikrokontroler dapat langsung dibuat menjadi suatu sistem hanya dengan menambahkan sebuah *peripheral* lain. Sifat mikrokontroler yang mampu diprogram menyebabkan mikrokontroler yang mempunyai kemampuan aplikasi yang sangat luas.

Mikrokontroler dengan arsitektur *Reduced Instructions Set Computer* (RISC) kini semakin berkembang pesat dan semakin banyak diminati dalam aplikasi sistem kendali. Salahnya satunya adalah jenis mikrokontroler *Alf and Vegard's Risc processor* (AVR) dari ATMel. Mikrokontroler AVR memiliki konsep yang hampir sama dengan PIC *micro* dari Microchip Inc. yang memiliki arsitektur RISC 8-bit [9].

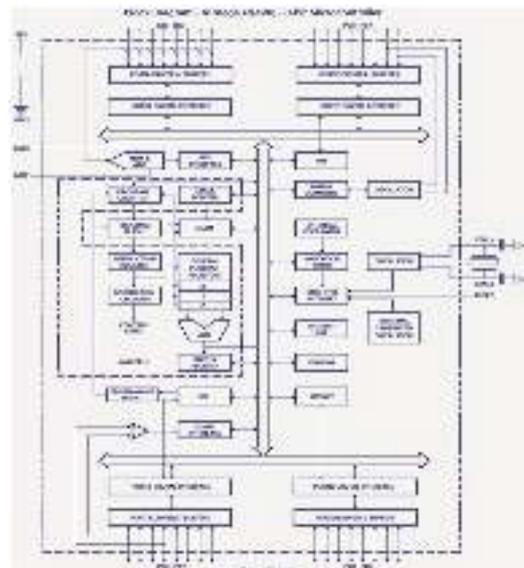
Mikrokontroler AVR dengan arsitektur RISC berarti prosesor ini memiliki set intruksi

program yang lebih sedikit jika dibandingkan dengan MCS-51 yang menerapkan arsitektur *Complex Instruction Set Computer* (CISC).

Hampir semua instruksi dalam prosesor RISC merupakan instruksi dasar (belum tentu sederhana), sehingga intruksi-intruksi ini umumnya hanya memerlukan satu siklus mesin untuk menjalankannya, kecuali untuk intruksi percabangan yang membutuhkan dua siklus mesin. RISC biasanya dibuat dengan arsitektur Harvard karena arsitektur ini yang memungkinkan untuk membuat suatu eksekusi intruksi dapat selesai dikerjakan dalam satu atau dua siklus mesin, sehingga makin cepat dan handal. Proses *downloading* programnya juga relatif lebih mudah karena dapat dilakukan secara langsung pada sistemnya [8].

Saat ini mikrokontroler AVR dapat dikelompokkan menjadi enam kelas, yaitu ATtiny, AT90Sxx, ATmega, AT90CAN, AT90PWM, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas tersebut adalah memori, *peripheral* dan fungsinya, sedangkan dari segi arsitektur maupun instruksi yang digunakan pada setiap kelas hampir sama. Arsitektur AVR menggabungkan perintah-perintah secara efektif dengan 32 register umum. Semua register tersebut secara langsung terhubung dengan *Arithmetic Logic Unit* (ALU) yang memungkinkan dua register terpisah diproses dengan satu perintah tunggal dalam satu *clock cycle*. Hal ini menghasilkan suatu kode yang efektif dan kecepatan prosesnya 10 kali lebih cepat dari pada mikrokontroler CISC biasa.

Adapun blok diagram mikrokontroler AVR ATmega 328 digambarkan sebagai berikut [10]:



Gambar 2 Blok Diagram Mikrokontroler AVR ATmega 328

## 2.6 Real Time Clock DS3231

*Real Time Clock* (RTC) adalah jam elektronik berupa *chip* yang dapat menghitung waktu (mulai detik hingga tahun) dengan akurat dan menjaga/menyimpan data waktu tersebut secara *real time*. Karena jam tersebut bekerja *real time*, maka setelah proses hitung waktu dilakukan, *output* datanya langsung tersimpan atau dikirim ke *device* lain melalui sistem antarmuka.

*Chip* RTC seringkali dijumpai pada *motherboard* PC (biasanya terletak dekat *chip* BIOS). Semua komputer menggunakan RTC karena berfungsi menyimpan informasi jam terkini dari komputer yang bersangkutan. RTC dilengkapi dengan baterai sebagai pen-*supply* daya pada *chip* sehingga jam akan tetap *up-to-date* walaupun komputer dimatikan. RTC dinilai cukup akurat sebagai pewaktu (*timer*) karena menggunakan osilator kristal [11].

Serial RTC DS3231 merupakan *low-power* dan *full binary-coded-decimal* (BCD). Data dan alamat ditransfer berurutan secara serial melalui dua kabel dan *bidirectional bus*. *Clock/Calendar* menyediakan detik, menit, jam, hari, tanggal, bulan dan informasi tahun. Akhir dari tanggal dan bulan secara otomatis akan disesuaikan selama sebulan paling sedikit 31 hari yang mencakup koreksi untuk tahun kabisat. Jam beroperasi dalam format 12 jam atau 24 jam dengan AM/PM indikator. RTC DS3231 mempunyai suatu pendeteksi gangguan daya dan secara otomatis men-*supply* tegangan dari baterai apabila VCC lebih kecil dari VBAT [12].

Blok diagram RTC DS3231 dapat dilihat pada gambar sebagai berikut [13]:



Gambar 3 Blok Diagram RTC DS3231

RTC DS3231 mendukung *bus* dua kawat *bidirectional* dan protokol transmisi data. Piranti pengiriman data disebut *transmitter* sedangkan piranti penerima data disebut *receiver*. Adapun piranti pengontrol pesan disebut *master* sedangkan piranti yang dikontrol disebut *slave*. *Bus* harus dikontrol oleh *master* yang membangkitkan *Clock Serial* (SCL). Mengontrol akses *bus* dan membangkitkan kondisi *start* dan *stop*. RTC DS3231 beroperasi sebagai *slave* pada bus 2+ *wire*.

## 2.7 Display Seven Segment dan LCD

Peralatan keluaran yang seringkali digunakan dalam menampilkan bilangan adalah penampil *seven segment*. Segmen-segmen (*seven segment*) dilabelkan dengan huruf a sampai dengan huruf g. Adapun peraga dari ketujuh segmen tersebut dapat dibuat dalam berbagai bentuk dan ukuran yang umumnya dapat menampilkan angka.

Peraga jenis lainnya yang biasa digunakan adalah *Liquid Crystal Display* (LCD) yang merupakan suatu alat peraga dari cairan yang dapat menghasilkan angka-angka berwarna kelabu atau putih perak. Modul LCD *character* dapat dengan mudah dihubungkan dengan sebuah mikrokontroler [14].

## 2.8 Waktu Belajar Sekolah

Pada setiap sekolah, pengaturan alokasi waktu untuk setiap mata pelajaran yang terdapat pada sistem semester ganjil dan semester genap dalam satu tahun ajaran dapat dilakukan secara fleksibel sesuai dengan jumlah beban belajar yang ditetapkan. Setiap satuan pendidikan pada jenjang SD, SMP hingga SMA dimungkinkan untuk menambah maksimum empat jam pembelajaran per minggu secara keseluruhan. Pemanfaatan dari jam pembelajaran tambahan tersebut dilakukan dengan mempertimbangkan kebutuhan dari para peserta didik dalam mencapai kompetensi.

Disamping itu, jam pembelajaran tambahan dapat dimanfaatkan untuk mata pelajaran lain yang dianggap penting atau untuk mata pelajaran bermuatan lokal yang tidak terdapat didalam struktur kurikulum. Perbedaan antara sistem paket dengan sistem satuan kredit semester (sks) adalah rata-rata 1 sks sama dengan 2 jam waktu pembelajaran yang ada didalam sistem paket.

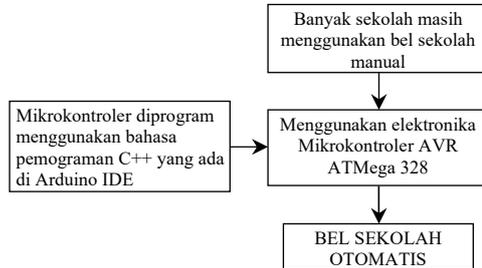
Alokasi waktu untuk tatap muka, penugasan terstruktur dan kegiatan mandiri tidak terstruktur untuk jenjang pendidikan SMP/MTs dan SMA/MA/SMK/MAK yang menggunakan sistem satuan kredit semester (sks) diwajibkan untuk mengikuti aturan sebagai berikut [15]:

- a. Satu sks pada SMP/MTs terdiri atas 40 menit tatap muka, 40 menit kegiatan terstruktur dan 40 menit kegiatan mandiri tidak terstruktur. Total waktu keseluruhan adalah 120 menit.
- b. Satu sks pada SMA/MA/SMK/MAK terdiri atas 45 menit tatap muka, 45 menit kegiatan terstruktur dan 45 menit kegiatan mandiri tidak terstruktur. Total waktu keseluruhan adalah 135 menit.

Pada prakteknya di sekolah-sekolah, seluruh pelaksanaan waktu belajar tersebut selalu ditandai dengan berbunyinya bel sekolah, mulai dari waktu pelajaran pertama, waktu pelajaran kedua, dan seterusnya hingga waktu pulang sekolah.

## 2.9 Kerangka Pemikiran

Kerangka pemikiran merupakan tahap-tahap proses untuk mencapai tujuan [16]. Dengan demikian, secara terstruktur proses pelaksanaan penelitian ini dapat digambarkan sebagai berikut:



Gambar 4 Kerangka Pikir Penelitian

## 3. Metode Penelitian

Jenis penelitian ini adalah kualitatif, yaitu penelitian menghasilkan penemuan-penemuan yang tidak dapat dicapai dengan prosedur-prosedur statistik atau cara-cara lain dari kuantifikasi (pengukuran) [17].

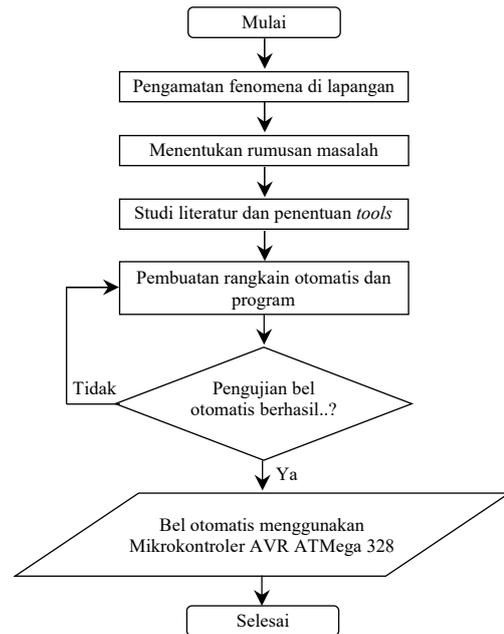
Penelitian ini dilakukan dengan membuat manipulasi atas objek yang diteliti dengan cara membuat suatu kondisi tertentu yang akan diuji seberapa pengaruhnya terhadap variabel lain sebagai pengontrolnya. Jadi, berdasarkan sifatnya maka penelitian ini termasuk dalam tipe penelitian eksperimen, yaitu bentuk khusus investigasi yang digunakan untuk menentukan variabel-variabel apa saja serta bagaimana bentuk hubungan antara satu variabel dengan variabel lainnya [18].

Teknik pengumpulan data yang dibutuhkan penelitian ini adalah:

- Pengamatan atau observasi, yaitu melakukan pengamatan langsung terhadap fenomena yang diamati dan melakukan pencatatan sistematis terhadap gejala-gejala yang ditemukan.
- Wawancara, yaitu melakukan pembicaraan langsung (tanya jawab) terhadap informan yang dianggap mengetahui pokok permasalahan penelitian ini.
- Studi kepustakaan, yaitu mengumpulkan data-data dan keterangan dengan cara membaca berbagai buku literatur maupun artikel yang berhubungan dengan permasalahan penelitian untuk mendapatkan landasan serta pengertian secara teoritis yang mendalam.

Metode pengembangan sistem yang digunakan pada penelitian ini adalah *extreme programing*, yaitu suatu metode ringan yang menekankan pada komunikasi yang intens antara klien dan tim, pengembangan yang efisien melalui model pengujian yang intens hingga model pengerjaan yang iteratif dan incremental [19].

Adapun tahap-tahap dalam penelitian ini didesain sebagai berikut:



Gambar 5 Desain Penelitian

## 4. Hasil Penelitian

### 4.1 Analisa Sistem

#### 4.1.1 Analisis Sistem Bel Sekolah Yang Ada

Saat ini masih banyak sekolah yang menandai mulainya pelajaran, pergantian jam mata pelajaran, jam istirahat, atau jam pelajaran usai dengan membunyikan bel sekolah secara manual, yaitu dengan cara menugaskan seorang petugas sekolah atau guru piket untuk memencet bel yang menggunakan bel listrik atau memukul bel jika masih menggunakan loncong.

Hal ini sangat tidak efisien karena terkadang petugas sekolah atau guru piket terlambat atau lupa membunyikan bel pada waktu yang telah ditentukan. Bahkan saat listrik padam bel sekolah sama sekali tidak dapat difungsikan.

Hasil pengamatan sistem bel sekolah manual yang berjalan pada banyak sekolah dapat digambarkan sebagai berikut:



Gambar 6 Sistem Bel Sekolah Manual

#### 4.1.2 Analisis Sistem Bel Otomatis Yang Diusulkan

Dengan kemajuan teknologi digital saat ini maka sistem bel sekolah manual dapat digantikan dengan bel sekolah otomatis yang sangat mudah untuk digunakan dan efisien untuk mengatur waktu pembelajaran. Untuk itu, perancangan bel sekolah otomatis yang dibangun dalam penelitian ini akan memanfaatkan teknologi *Real Time*

*Clock* (RTC) sebagai data jam yang sebenarnya/nyata (*real*), mikrokontroler [20] sebagai pengendali bel, serta *buzzer* sebagai *output* suara sehingga petugas sekolah atau guru piket dapat mengerjakan pekerjaan lain tanpa khawatir akan terlambat/lupa membunyikan bel sekolah.

Bagan sistem bel sekolah otomatis yang usulkan adalah sebagai berikut:

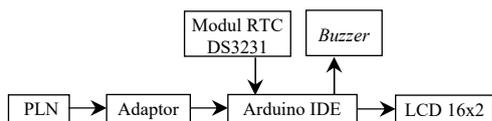


Gambar 7 Sistem Bel Sekolah Otomatis Yang Diusulkan

## 4.2 Desain Sistem

### 4.2.1 Perancangan Sistem

Untuk memudahkan pemahaman sistem kerja bel sekolah otomatis yang menggunakan mikrokontroler AVR ATmega 328, Peneliti menampilkan blok diagram sebagai berikut:



Gambar 8 Blok Diagram Bel Sekolah Otomatis Yang Diusulkan

Dalam 8 nampak bahwa setiap bagian yang ada saling terhubung menjadi satu bagian agar dapat berjalan dengan baik. Bel sekolah otomatis yang menggunakan mikrokontroler AVR ATmega 328 terdiri dari enam bagian blok diagram. Bagian *input* terdiri dari listrik PLN, adaptor, modul RTC sebagai data jam, dan Arduino sebagai *input* bel. Sedangkan bagian *output* terdiri dari LCD 16x2 sebagai penampil waktu dan *buzzer* sebagai *output* suara.

Listrik PLN memberikan daya listrik ke adaptor yang akan menurunkan tegangan listrik agar bisa diterima Arduino, yaitu sebesar 5 volt dan Arduino juga memberikan daya pada modul RTC dan LCD. Arduino sebagai pengendali utama dari semua perangkat, RTC sebagai data *input* waktu ke Arduino dan Arduino memberi perintah untuk menampilkannya di LCD, ketika waktu menunjukkan jam yang sesuai dengan yang diatur maka Arduino akan memberi perintah untuk membunyikan *buzzer*.

### 4.2.2 Perancangan Kebutuhan Alat

Untuk membuat sebuah bel sekolah otomatis dibutuhkan beberapa peralatan sebagai berikut:

- Mika sebagai *box* atau wadah alat.

- Mikrokontroler AVR ATmega 328.
- Papan *board* Arduino atau *module* Arduino.
- Modul *Real Time Clock* (RTC).
- Buzzer*.
- Liquid Crystal Display* (LCD).
- Kabel *jumper male to female*.
- Papan PCB atau papan *project*.
- Pin penghubung *male to female* dan *male to male*.
- Potensiometer.
- Solder*.
- Gergaji besi.
- Bor mini.
- Cutter*.
- Lem.
- Baut.
- Laptop.

### 4.2.3 Perancangan *Box* Rangkaian

Pertama kali yang dikerjakan adalah perancangan untuk kotak (*box*) rangkaian sebagai tempat meletakkan *hardware* yang dibuat. *Box* rangkaian berfungsi untuk melindungi perangkat elektronik dari pengaruh luar seperti percikan air dan sentuhan tangan manusia yang dapat merusak komponen elektronik yang ada.

*Box* bel sekolah otomatis yang dihasilkan dapat dilihat dalam gambar sebagai berikut:



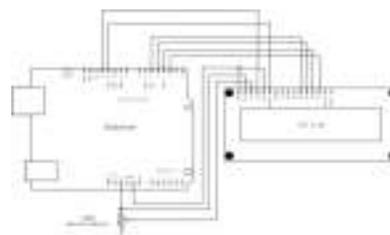
Gambar 9 *Box* Rangkaian Bel Sekolah Otomatis Yang Diusulkan

### 4.2.4 Perancangan *Hardware*

#### 4.2.4.1 Perancangan *Liquid Crystal Display* (LCD)

*Liquid Crystal Display* (LCD) yang digunakan adalah ukuran 16x2 yang akan berfungsi sebagai penampil jam digital atau *interface* dari bel sekolah otomatis yang diusulkan.

Rangkaian LCD 16x2 dengan Arduino dapat dilihat dalam gambar sebagai berikut:



Gambar 10 Rangkaian *Liquid Crystal Display* (LCD) 16x2

#### 4.2.4.2 Perancangan Modul *Real Time Clock* (RTC)

*Real Time Clock* (RTC) DS3231 yang digunakan dalam penelitian ini akan berfungsi sebagai penghitung waktu bel sekolah otomatis yang diusulkan.

Rangkaian RTC DS3231 dengan Arduino dapat dilihat dalam gambar sebagai berikut:

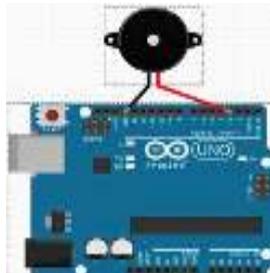


Gambar 11 Rangkaian Modul *Real Time Clock* (RTC)

#### 4.2.4.3 Perancangan *Buzzer*

*Buzzer* yang digunakan dalam penelitian ini akan berfungsi sebagai *output* dari bel sekolah otomatis yang diusulkan.

Rangkaian *buzzer* dengan Arduino dapat dilihat dalam gambar sebagai berikut:

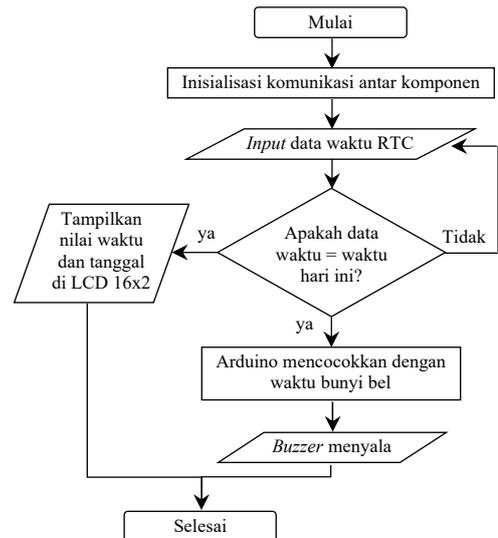


Gambar 12 Rangkaian *Buzzer*

#### 4.2.5 Perancangan *Software*

Perancangan perangkat lunak (*software*) berfungsi untuk menentukan setiap alur eksekusi dari pengendali utama yang diatur oleh mikrokontroler AVR ATmega 328. Setiap alur *input* atau masukan yang diterima akan diatur pada perangkat lunak dan akan diproses untuk menentukan eksekusi pada bagian keluaran atau *output*, agar peralatan dan juga sistem yang telah dibuat dapat bekerja dengan baik.

*Flowchart* dari perancangan perangkat lunak (*software*) bel sekolah otomatis yang diusulkan dengan menggunakan mikrokontroler AVR ATmega 328 dapat dilihat dalam gambar sebagai berikut:



Gambar 13 *Flowchart* Perancangan *Software*

### 4.3 Implementasi Sistem

Tahap implementasi adalah proses penginstalan perangkat Arduino dan komponen lainnya yang meliputi penginstalan Arduino IDE ke *laptop* atau CPU yang akan digunakan untuk pembuatan *codec* sistem, penentuan *pin input* untuk RTC, serta penentuan *pin output* untuk *buzzer* dan LCD 16x2 sebagai berikut:

#### 4.3.1 Penginstalan OS Arduino IDE

Instalasi *operating system* Arduino IDE adalah proses *em-booting-an file-file driver* Arduino pada *laptop* atau CPU yang akan digunakan. Fungsi Arduino IDE adalah tempat pembuatan *codec/coding* sistem bel otomatis yang diusulkan.

Ketika *codec* telah dibuat maka *codec* tersebut akan di *upload* ke papan *board* Arduino sehingga muncul tampilan sebagai berikut:



Gambar 14 Tampilan Penginstalan OS Arduino IDE

#### 4.3.2 Penentuan *Pin Input* dan *Pin Output* Arduino

*Pin input* dan *pin output* Arduino adalah serangkaian pin pada Arduino yang digunakan untuk berinteraksi dengan komponen-komponen lainnya seperti RTC, LCD 16x2 dan *buzzer*.

Dalam penelitian ini digunakan sembilan pin *input* dan pin *output* sebagai interaksi dengan komponen-komponen lainnya sebagai berikut:



Gambar 15 Pin *Input* dan Pin *Output* Arduino IDE

Penggunaan kesembilan pin *input* dan pin *output* dalam gambar 15 adalah sebagai berikut:

- Nomor 1 adalah pin 5 volt untuk menyuplai daya ke modul RTC, LCD 16x2 dan potensiometer.
- Nomor 2 adalah pin *ground* yang akan terhubung dengan *ground* kabel modul RTC, LCD 16x2 dan *buzzer*.
- Nomor 3 adalah pin analog 4 dan 5 sebagai *input* data jam yang berasal dari modul RTC.
- Nomor 4 adalah pin digital 2 sebagai *output* yang akan diteruskan ke *buzzer* untuk mengeluarkan suara.
- Nomor 5 adalah pin digital 8, 9, 10 dan 11 sebagai *output* yang akan terhubung dengan LCD 16x2.

#### 4.3.3 Pembuatan *Codec* Program

Pembuatan program bel sekolah otomatis yang diusulkan akan menggunakan bahasa pemrograman Arduino IDE yang merupakan penyederhanaan dari bahasa .C yang telah terinstal atau bawaan dari Arduino.

Cuplikan *Codec* program yang dibuat untuk memenuhi tujuan penelitian sebagai berikut:

```
#include <DS3231.h>
#include <LiquidCrystal.h> // includes the LiquidCrystal Library
DS3231 rtc(SDA, SCL);
LiquidCrystal lcd(8,9,10,11,12,13); // Creates an LC object. Parameters: (rs,
enable, d4, d5, d6, d7)
Time t;
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(2, OUTPUT);
  Serial.begin(115200);
  // Uncomment the next line if you are using an Arduino Leonardo
  //while (!Serial) {}
  // Initialize the rtc object
  rtc.begin();
  // The following lines can be uncommented to set the date and time
  rtc.setDOW(MONDAY); // Set Day-of-Week to SUNDAY
  rtc.setTime(21, 3, 0);
  lcd.begin(16,2); // Initializes the interface to the LCD screen, and specifies the
dimensions (width and height) of the display }
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" BEL OTOMATIS ");
  lcd.setCursor(0,1);
  lcd.print(" ARDUINO UNO R3 ");
  delay(3000);
  lcd.clear();
}
void loop() {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Time: ");
```

```
lcd.print(rtc.getTimeStr());
lcd.setCursor(0,1);
lcd.print("Date: ");
lcd.print(rtc.getDateStr());
t = rtc.getTime();
int time1=t.hour;
int time2=t.min;
int time3=t.sec;
// int lastread=0;
// int time1,time2,time3;
// rtc.updateTime();
// delay(1500);
// time1=t.hour-lastread;
// time2=t.min-lastread;
// time3=t.sec-lastread;
// masuk jam 1
if ((time1==7)&&(time2==1)&&(time3==0)) // bunyi sekali saja
```

## 4.4 Pengujian Sistem

### 4.4.1 Pengujian Modul RTC dan LCD 16x2

Pengujian pertama yang dilakukan adalah pengujian modul *Real Time Clock* (RTC) yang digunakan sebagai pewaktu jam digital dan kalender yang merupakan tampilan utama sistem yang digunakan. Jam digital ini akan menunjukkan data waktu untuk jam, menit dan detiknya, terdapat pula tambahan waktu untuk hari, tanggal, bulan dan tahun. Fungsi utama dari rangkaian ini sebagai pewaktu untuk penanda bel sekolah otomatis yang diusulkan.

Blok diagram pengujian modul RTC dapat dilihat pada gambar sebagai berikut:



Gambar 16 Blok Diagram Pengujian Modul RTC

Pengujian modul RTC dan LCD 16x2 dilakukan dengan cara membandingkan data waktu pada RTC dengan data waktu yang sudah terkalibrasi. Pada pengujian ini perbandingan dilakukan dengan menggunakan *stopwatch* sebagai acuan waktu dan hasil pengujian adalah sebagai berikut:

Tabel 1 Hasil Pengujian Keakuratan Data RTC

Modul RTC	<i>Stopwatch</i>
00 : 10 : 00	00 : 10 : 00
00 : 45 : 00	00 : 45 : 00
01 : 00 : 00	01 : 00 : 00
02 : 00 : 00	02 : 00 : 00

Dari hasil pengujian keakuratan RTC menggunakan *stopwatch* pada tabel 1 dapat diketahui bahwa waktu yang ada pada RTC sudah tepat dan akurat.

Hasil pengujian menggunakan *stopwatch* terhadap jam hari ini sebagai berikut:



Gambar 17 Hasil Pengujian Modul RTC

#### 4.4.2 Pengujian Kesesuaian Bel Sekolah Otomatis dengan Jadwal Sekolah

Pengujian ini bertujuan untuk mengetahui apakah bel sekolah otomatis yang diusulkan telah berbunyi sesuai dengan *setting* jadwal sekolah yang telah di-*input*. Pengujian ini dilakukan dengan cara menyalakan sistem selama satu hari penuh tanpa dimatikan, mulai jam pelajaran pertama hingga jadwal sekolah berakhir

Hasil pengujian kesesuaian bel sekolah otomatis dengan jadwal sekolah yang diperoleh adalah sebagai berikut:

Tabel 2 Hasil Pengujian Kesesuaian Bel Sekolah Otomatis Dengan Jadwal Sekolah

Keterangan	Jadwal (waktu)	Hasil Pengujian	
		Waktu	Keterangan
Masuk sekolah	6.45	6.45	Bunyi panjang
Pelajaran ke-1	7.15	7.15	Bunyi 1x
Pelajaran ke-2	8.00	8.00	Bunyi 2x
Pelajaran ke-3	8.45	8.45	Bunyi 3x
Istirahat I	9.30	9.30	Bunyi panjang pendek
Masuk kelas	9.45	9.45	Bunyi panjang pendek
Pelajaran ke-4	10.30	10.30	Bunyi 4x
Pelajaran ke-5	11.15	11.15	Bunyi 5x
Istirahat II	12.00	12.00	Bunyi panjang pendek
Masuk kelas	12.15	12.15	Bunyi panjang pendek
Pulang sekolah	13.00	13.00	Bunyi panjang

Berdasarkan hasil pengujian dalam tabel 2 nampak bahwa bel sekolah otomatis telah berbunyi sesuai dengan waktu yang ditentukan dan suara bel sesuai dengan pengaturan yang diinginkan pihak sekolah.

#### 5. Penutup

Hasil berbagai percobaan dan analisis yang dilakukan menunjukkan bahwa bel sekolah otomatis yang dibangun telah bekerja dengan baik karena berdering sesuai dengan dengan jadwal pembelajaran yang ditentukan.

Selanjutnya bel otomatis yang dibangun dalam penelitian ini masih dapat dikembangkan antara lain dengan membuat desain *box* rangkaian yang lebih minimalis, menambahkan fitur-fitur lain agar dapat mempermudah penggunaannya, menambahkan *supplay* daya cadangan atau baterai sebagai cadangan listrik, dan menggunakan nada dering yang lebih menarik.

#### Daftar Pustaka

- [1] <https://affifmaulizar.blogspot.com/2013/04/apa-itu-perencanaan-perancangan-dan-.html>.
- [2] Syifaun, Nafisah. 2003. *Java for Mobile Programming*. Semarang: Wahana Komputer.
- [3] <https://id.wikipedia.org/wiki/Lonceng>.
- [4] <https://kbbi.web.id/otomatis>.
- [5] <https://rainboard.id/pengertian-arduino>.
- [6] <https://www.sinuarduino.com/artikel/mengenal-arduino-software-ide/>.
- [7] <https://www.idebebas.com/arduino-ide/>.
- [8] <http://elektronika-dasar.web.id/2012/pengertian-mikrokontroler>.
- [9] <http://www.nulis-ilmu.com/2015/09/mikrokontroler-avr-atmega32>.
- [10] <http://www.robotics-university.com/2015/port-input-output-mikrokontroler-avr-atmega32>.
- [11] <http://ferballcompany.blogspot.com/2012/04/apa-itu-satu-rtc.html>.
- [12] <http://www.proyekrumahan.com/2013/06/rtc-real-time-clock-ds1307-code-vision>.
- [13] [www.ablabsolution&source.com](http://www.ablabsolution&source.com).
- [14] <http://elektronika-dasar.web.id/display-7-segment>.
- [15] Peraturan Pemerintah No. 19 Tahun 2005 Tentang *Standar Pendidikan Dasar Menengah..*
- [16] Sugiyono. 2011. *Metode Penelitian Kuantitatif, Kualitatif dan R&D*. Jakarta: Alfabeta.
- [17] Straus, Anselm & Corbin, Juliet. 2003. *Dasar-Dasar Penelitian Kualitatif: Tata Langkah dan Teknik-Teknik Teoritis Data* (Terjemahan Muhammad Sodik dan Imam Muttaqin). Yogyakarta: Pustaka Pelajar..
- [18] <http://metode-penelitian/navelsblog.html>.
- [19] Ferdiana, Ridi. 2012. *Rekayasa Perangkat Lunak Yang Dinamis Dengan Global Extreme Programing*. Yogyakarta: Andi Offset.
- [20] Prayitno, A., & Nurdin, N. (2017). Analisa Dan Implementasi Kriptografi Pada Pesan Rahasia Menggunakan Algoritma Cipher Transposition. *Jurnal Elektronik Sistem Informasi dan Komputer*, 3(1), 1=10.

