Jurnal TIMES , Vol. IV No 2 : 45-48 , 2015
ISSN : 2337 - 3601

SIMULASI RUMAH PINTAR DENGAN ANDROID SEBAGAI PENGENDALI

Herman
Program Studi Teknik Informatika
STMIK TIME Medan
Jln. Merbabu No 32 AA-BB Medan
Telp. 061-4561932 , Email : hrman_ang@yahoo.com

ABSTRAK

Rumah pintar merupakan satu sistem pengendali rumah yang memberikan kemudahan kepada pemilik rumah untuk mengendalikannya di kehidupan kesehariannya dengan menggunakan komputer. Dengan perkembangan teknologi seperti ponsel pintar yang sudah banyak dimiliki orang serta perkembangan teknologi jaringan dapat digunakan untuk mengendalikan rumah. Sistem saklar lampu dan beberapa peralatan lainnya dapat diganti dengan menggunakan peralatan *relay* dan dikendalikan melalui peralatan mikrokontroler berbasis jaringan sehingga dapat terhubung ke ponsel pintar yang sudah terpasang program pengendali akan digunakan untuk mengatasi masalah tersebut. Dari hasil percobaan dengan menggunakan teknologi tersebut, ponsel pinter berbasis android yang terpasang program pengendali saklar listrik dapat mematikan atau menghidupkan peralatan listrik dari jarak yang cukup jauh.

Kata kunci : Rumah pintar, Android, Wifi, Simulasi, Mikrokontroler.

1. Pendahuluan

Rumah pintar adalah rumah yang dapat dikendalikan oleh pemilik rumah, seperti membuka atau menutup tirai jendela, membuka kipas pembuangan udara, menghidupkan mesin pemanas kopi, menginformasikan keadaan ruangan seperti suhu udara, kelembaban, kecepatan angin dan sensor gerak.

Semua itu dapat dikendalikan oleh telepon seluler yang memiliki sistem operasi android yang memberikan kemudahan untuk mengendalikan dan melihat keadaan rumah kita. Saat kita sedang berpergian sering kita teringat dengan keadaan rumah kita, apakah rumah kita aman? Apakah tidak ada peralatan elektronik kita yang lupa dimatikan? Kita hanya perlu membuka telepon seluler kita dan melihat keadaan rumah kita.

Simulasi rumah pintar dengan android adalah peniruan terhadap rumah yang akan dipasang sistem kendali rumah jarak jauh yang dikendalikan oleh perangkat dengan sistem operasi android. Dengan menggunakan simulasi maka dapat memperlihatkan peniruan terhadap keadaan sebenarnya bagaimana simulasi tersebut berkerja.

2. Landasan Teori

2.1. Android

Android adalah sistem operasi berbasis *Linux* yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. *Android* awalnya dikembangkan oleh A*n*droid, *Inc*., dengan dukungan finansial dari *Google*, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. *Ponsel Android* pertama mulai dijual pada bulan Oktober 2008.[1]

Aplikasi *Android* dikembangkan dalam bahasa pemrograman *Java* dengan menggunakan *kit* pengembangan perangkat lunak *Android* (SDK). SDK ini terdiri dari seperangkat perkakas pengembangan, termasuk *debugger*, perpustakaan perangkat lunak, *emulator handset* yang berbasis QEMU, dokumentasi, kode sampel, dan *tutorial*. Didukung secara resmi oleh lingkungan pengembangan terpadu (IDE) *Eclipse*, yang menggunakan *plugin Android Development Tools* (ADT). Perkakas pengembangan lain yang tersedia di antaranya adalah *Native Development Kit* untuk aplikasi atau ekstensi dalam *C* atau *C++*, *Google App Inventor*, lingkungan *visual* untuk pemrogram pemula, dan berbagai kerangka kerja aplikasi *web* seluler lintas *platform*.

2.2. Arduino

Arduino adalah sebuah *board* mikrokontroller yang berbasis ATmega328. *Arduino* memiliki 14 pin *input/output* yang mana 6 pin dapat digunakan sebagai *output* PWM, 6 analog input, *crystal osilator* 16 MHz, koneksi USB, *jack power*, kepala ICSP, dan tombol *reset*". *Arduino* mampu men-*support* mikrokontroller, dapat dikoneksikan dengan komputer menggunakan kabel USB.[2]

Arduino dapat diberikan *power* melalui koneksi USB atau *power supply*. dayanya pilih secara otomatis. *Power supply* dapat menggunakan *adaptor* DC atau baterai. *Adaptor* dapat dikoneksikan dengan mencolok *jack* adaptor pada koneksi port *input supply*. *Board arduino* dapat dioperasikan menggunakan *supply* dari luar sebesar 6 - 20 volt. Jika *supply* kurang dari 7V, kadang kala pin 5V akan menyuplai kurang dari 5 volt dan *board* bisa menjadi tidak stabil. Jika menggunakan lebih dari 12 V,

tegangan di *regulator* bisa menjadi sangat panas dan menyebabkan kerusakan pada papan pcb (*board*). Rekomendasi tegangan ada pada 7 sampai 12 volt.

2.3. Arduino Ehternet Shield

Arduino Ehernet Shield memungkinkan sebuah papan *Arduino* terhubung ke *internet"*. Hal ini karenakan penggunaan *Chip* W5100. *Chip* W5100 menyediakan jaringan (IP) *stack* berkemampuan TCP dan UDP. Mendukung hingga 4 koneksi secara simultan.[3]

2.4. Mikrokontroller

Mikrokontroller adalah suatu *chip* berupa IC (*Integrated Circuit*) yang dapat menerima sinyal *input*, mengolahnya dan memberikan sinyal *output* sesuai dengan program yang diisikan ke dalamnya". Sinyal *input* mikrokontroller berasal dari sensor yang merupakan informasi dari lingkungan sedangkan sinyal *output* ditujukan kepada aktuator yang dapat memberikan efek ke lingkungan.[4]

Beberapa penggunaan mikrokontroller antara lain terdapat pada bidang-bidang berikut ini:

1. Otomotif : *Engine Control Unit, Air Bag, fuel control, Antilock Braking System*, sistem pengaman alarm, transmisi automatik, hiburan, pengkondisi udara, *speedometer* dan *odometer*, navigasi, suspensi aktif
2. Perlengkapan rumah tangga dan perkantoran : sistem pengaman *alarm*, *remote control*, mesin cuci, *microwave*, pengkondisi udara, timbangan *digital*, mesin foto kopi, *printer*, *mouse*.
3. Pengendali peralatan di industri.
4. Robotika.

2.5. *Eclipse*

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*).

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya *populer* adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

3. Metode Penelitian

Pembangunan sistem dilakukan dengan menggunakan *Android* dan *Arduino* dengan peralatan tambahan seperti *modem wifi*, *Arduino Ethernet Shield*, *Arduino Expansion Shield* v5.0 dan beberapa komponen elektronik yang lainnya.

Arduino sebagai perangkat keras yang digunakan untuk mengontrol lampu LED dan motor atau dinamo yang akan dikoding menggunakan bahasa mesin atau bahasa C. *Router wifi* disambungkan dengan kabel *LAN* ke *Ethernet Shield W5100*. *Arduino*, *Ethernet shield* dan *shield expansion* dihubungkan menjadi satu. Kemudian akan dihubungkan ke lampu LED dan motor menggunakan *digital pin* yang terdapat pada *arduino*.

Pembuatan perangkat lunak pengendali di *Android* menggunakan *Eclipse* menggunakan bahasa *java*. Yang dirancang untuk berkomunikasi dengan *Arduino* menggunakan *modem Wifi* tanpa koneksi *internet*. Jadi sistem rumah pintar ini hanya sebatas jangkauan radius pancaran sinyal dari modem wifi untuk berkomunikasi.

Setelah dilakukan pemasangan perangkat lunak untuk pemograman *arduino* di sistem operasi *windows*, perlu dilakukan instalasi *Eclipse* disistem operasi *windows.* Setelah itu perancangan tampilan perangkat lunak dan penulisan coding untuk aplikasi dengan Algoritma *UDPSendReceiveString*

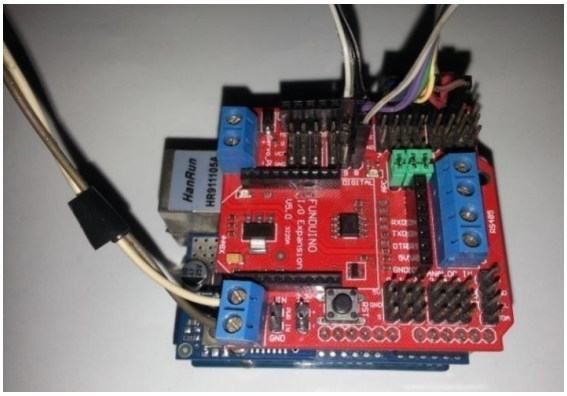
User Datagram Protocol (UDP) adalah jenis protokol *connectionless oriented*. UDP bergantung pada lapisan atas untuk mengontrol keutuhan data. Oleh karena penggunaan *bandwith* yang efektif, UDP banyak dipergunakan untuk aplikasi-aplikasi yang tidak peka terhadap gangguan jaringan seperti SNMP dan TFTP.

UDP *broadcast* menggunakan algoritma *spanning-tree* untuk meneruskan paket secara terkontrol. *Transparent bridging* diterapkan pada *router interface* untuk membina *spanning-tree*. UDP *broadcast* dapat juga diteruskan menggunakan IP *helper address*, setiap *interface* yang akan menerima UDP *broadcast* harus dikonfigurasikan dengan IP *helper address*.

Pengkodingan aplikasi pengendali Simulasi Rumah pintar ini penulis menggunakan Alogaritma *UDPSendReceiveString*. *Ethernet Shield* dibutuhkan untuk algoritma *UDPSendReceiveString*. Sistem kerja lgoritma ini adalah *Android* mengirimkan *string* melalui *Ethernet Shield* dan akan di terima oleh *Arduino* selanjutnya *string* akan diproses oleh *arduino* kemudian *arduino* akan mengaktifkan perangkat LED atau Motor.

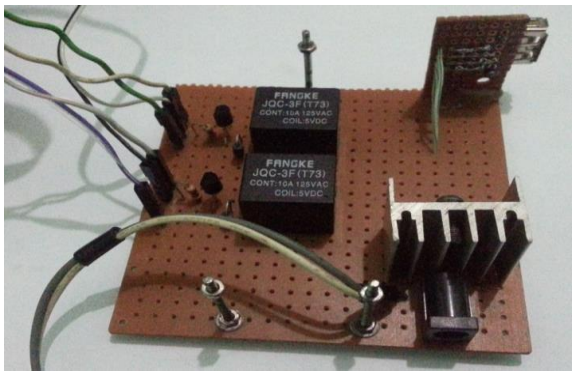
4. Hasil

Berikut Gambar 1 merupakan hasil yang didapatkan setelah seluruh proses dilakukan hingga akhir. *Arduino uno* digabungkan dengan *Ethernet Shield* W5100 dan *Shield Expansion* v5, *Ethernet Shield* digunakan untuk menghubungkan *arduino* ke *router wifi* menggunakan kabel UTP.

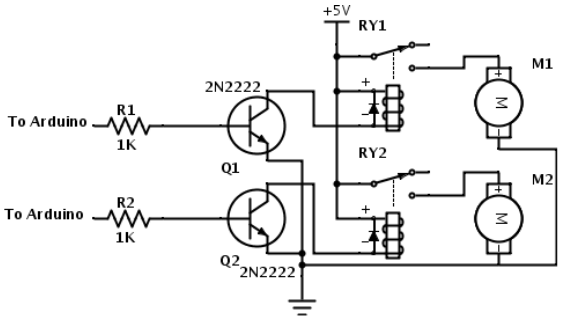


Gambar 1. *Arduino Uno, Ethernet Shield* dan *Shield Expansion*

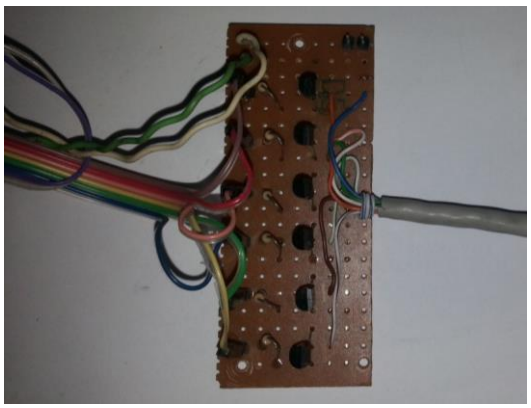
Relay, resistor, transistor semua disolder dipapan sirkuit. Berikut adalah gambar dari rangkaian pengendali LED dan motor.



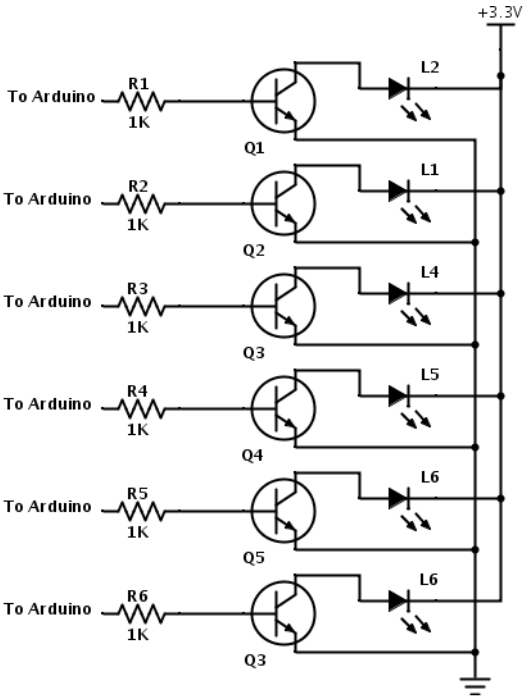
Gambar 2. Pengendali Motor



Gambar 3. Skema Rangkaian Pengendali Motor

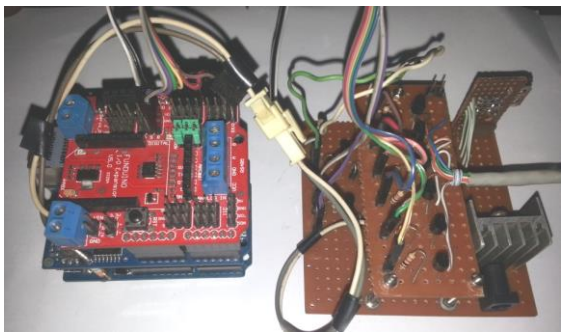


Gambar 4. Papan Pengendali Lampu LED



Gambar 5. Skema Rangkaian Pengendali Lampu LED

Setelah selesai semua kemudian disatukan menjadi satu bagian dan kemudian dihubungkan ke-*arduino* dan model rumah simulasi.



Gambar 6. *Arduino* dan Papan Pengendali Lampu LED dan Motor

5. Kesimpulan

Dari hasil simulasi rumah pintar dikendalikan oleh *android* ini dapat ditrarik kesimpulan dengan perkembangan teknologi memungkinkan orang lain dapat membuat rumah pintar sesuai dengan keinginan dan kebutuhan. Penggunaan android dengan andruino memudahkan pengendalian peralatan eletronik dan merupakan salah satu solusi membangun rumah pintar dengan harga yang relatif murah dan mudah dikerjakan setiap orang

6. Daftar Pustaka

[1] Guandotra Vic, 2013, *1,000,000,000 activation*, Google+,https://plus.google.com/+VicGundotra/posts/8CVJ79nPQwN, tanggal akses 25 febuari 2014

[2] Andreas Göransson dan David Cuartielles Ruiz, 2013, *Profesional Android Open Accessory Programming with Arduino*, John Wiley & Sons, Inc, Indianapolis

[3] Arduino, 2014, *Arduno Ethernet Shield*, arduino.cc, http://arduino.cc/en/Main/ArduinoEthernetShield, tanggal akses 25 Febuari 2014

[4] Wikipedia, 2014, *microcontroller*, http://en.wikipedia.org/wiki/Microcontroller, tanggal akses 9 mei 2014