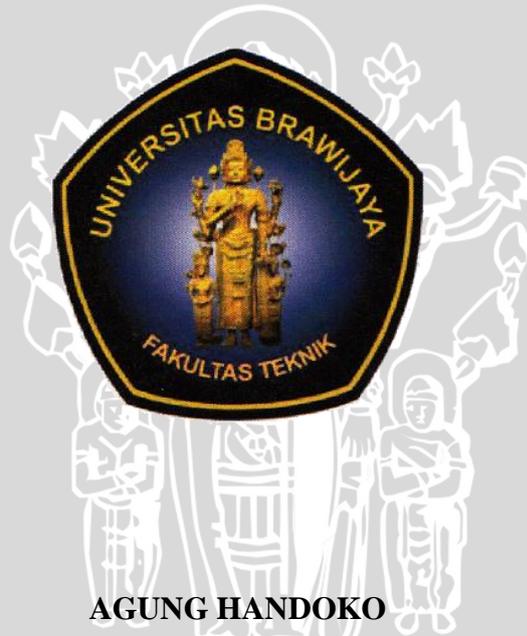


**RANCANG BANGUN SISTEM PEMANTAU KONDISI
SUNGAI DENGAN FASILITAS SMS SEBAGAI PERINGATAN
DINI BAHAYA BANJIR**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK ELEKTRONIKA

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



AGUNG HANDOKO

NIM. 115060301111019

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2015

LEMBAR PERSETUJUAN

**RANCANG BANGUN SISTEM PEMANTAU KONDISI
SUNGAI DENGAN FASILITAS SMS SEBAGAI PERINGATAN
DINI BAHAYA BANJIR**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK ELEKTRONIKA

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



AGUNG HANDOKO

NIM. 115060301111019

Skripsi ini telah direvisi dan disetujui oleh dosen pembimbing
pada tanggal 5 Agustus 2015

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Ponco Siwindarto, M.Eng.Sc.
NIP. 19590304 198903 1 001

Ir. Nanang Sulistiyanto, M.T.
NIP. 19700113 199403 1 002

LEMBAR PENGESAHAN

**RANCANG BANGUN SISTEM PEMANTAU KONDISI SUNGAI DENGAN
FASILITAS SMS SEBAGAI PERINGATAN DINI BAHAYA BANJIR**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK ELEKTRONIKA

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

AGUNG HANDOKO

NIM. 115060301111019

Skripsi ini telah diuji dan dinyatakan lulus sidang skripsi
pada tanggal 30 Juni 2015

Dosen Penguji I

Eka Maulana, S.T., M.T., M.Eng.
NIK. 841130 06 1 1 0280

Dosen Penguji II

Mochammad Rif'an, S.T., M.T.
NIP. 19710301 200012 1 001

Dosen Penguji III

Ir. M. Julius St, M.S.
NIP. 19540720 198203 1 002

**Mengetahui,
Ketua Jurusan Teknik Elektro**

M. Aziz Muslim, S.T., M.T., Ph.D.
NIP. 19741203 200012 1 001

PENGANTAR

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT, karena atas segala petunjuk dan nikmat-Nya lah skripsi ini dapat diselesaikan. Skripsi berjudul “Rancang Bangun Sistem Pemantau Kondisi Sungai dengan Fasilitas SMS sebagai Peringatan Dini Bahaya Banjir” ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Universitas Brawijaya.

Penulis menyadari bahwa dalam penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan ketulusan dan kerendahan hati penulis menyampaikan terima kasih kepada:

- Allah SWT atas rahmat dan hidayah yang telah diberikan,
- Rasulullah Muhammad SAW, semoga shalawat serta salam selalu tercurah kepada beliau,
- Bapak Moch. Toha Y, Ibu Mujiati selaku orang tua tercinta dan Ayu Jihanita selaku adik tersayang yang senantiasa memberikan semangat, doa, kasih sayang, perhatian, serta dukungan baik materi maupun non-materi kepada penulis hingga terselesaikannya skripsi ini..
- Bapak M. Aziz Muslim, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya, Bapak Hadi Suyono, S.T., M.T., Ph.D selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya, dan Bapak Ali Mustofa, S.T., M.T. selaku Ketua Prodi Strata Satu Jurusan Teknik Elektro Universitas Brawijaya.
- Bapak Ir. Purwanto, M.T. selaku dosen Penasehat Akademik,
- Ibu Ir. Nurussa'adah, M.T. selaku Ketua Kelompok Dosen Keahlian Elektronika Jurusan Teknik Elektro Universitas Brawijaya
- Bapak Dr. Ir. Ponco Siwindarto, M.Eng.Sc. sebagai Dosen Pembimbing I atas segala bimbingan, pengarahan, saran, dan kritik yang telah diberikan,
- Bapak Ir. Nanang Sulistiyanto, M.T. sebagai Dosen Pembimbing II atas segala bimbingan, pengarahan, saran, dan kritik yang telah diberikan,
- Seluruh dosen pengajar Teknik Elektro Universitas Brawijaya,
- Staff Recording Jurusan Teknik Elektro,

- Teman – teman Inverter angkatan 2011,
- Rekan-Rekan KKN-P, Andrian Rangga, Ardyanto, Wildan, terima kasih atas pengalaman, dukungan dan inspirasinya,
- Rekan seperjuangan dalam skripsi, Swaraka, Nurdin, Rizal, Naufal, Tegar, Bustanul, April, terima kasih atas segala bantuan yang telah diberikan,,
- Rekan-rekan PKM-KC “Madu” Universitas Brawijaya, Abdul Rozaq, M. Ainun, Suraduita, dan Arini, terima kasih atas pengalaman yang berharga serta inspirasinya,
- Patricia Palmaningtias dan teman-teman SMA yang hingga saat ini masih terus memberikan dukungan dan inspirasinya,
- Seluruh teman-teman serta semua pihak yang tidak mungkin bagi penulis untuk mencantumkan satu-persatu, terimakasih banyak atas bantuan dan dukungannya.

Pada akhirnya, penulis menyadari bahwa skripsi ini masih belum sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun. Penulis berharap semoga skripsi ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi serta bagi masyarakat.

Malang, Juli 2015

Penulis

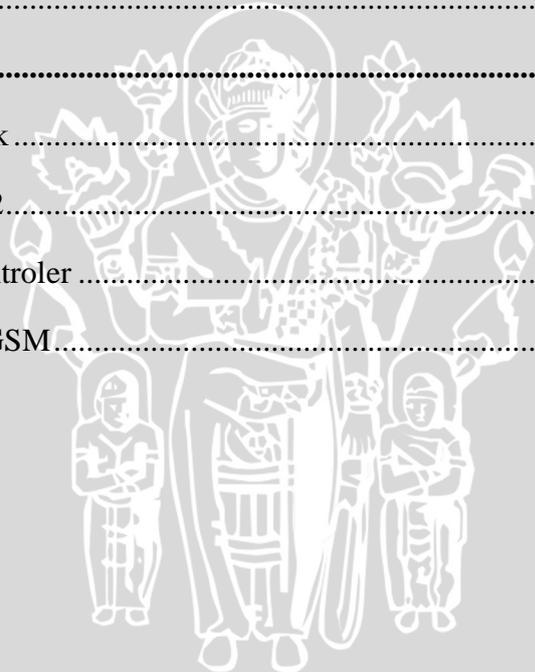
DAFTAR ISI

PENGANTAR	i
DAFTAR ISI	iii
DAFTAR TABEL	vi
DAFTAR GAMBAR.....	vii
RINGKASAN	viii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA.....	5
2.1 Hujan.....	5
2.2 Sungai	6
2.3 Banjir.....	6
2.4 Alat Ukur Curah Hujan.....	7
2.5 Sensor Ultrasonik.....	8
2.6 Mikrokontroler.....	9
2.7 Buzzer	11
2.8 Modul Liquid Crystal Display (LCD).....	12
2.9 RS-232	13
2.10 Modem GSM	13
2.11 AT-Command	14
BAB III METODE PENELITIAN	15



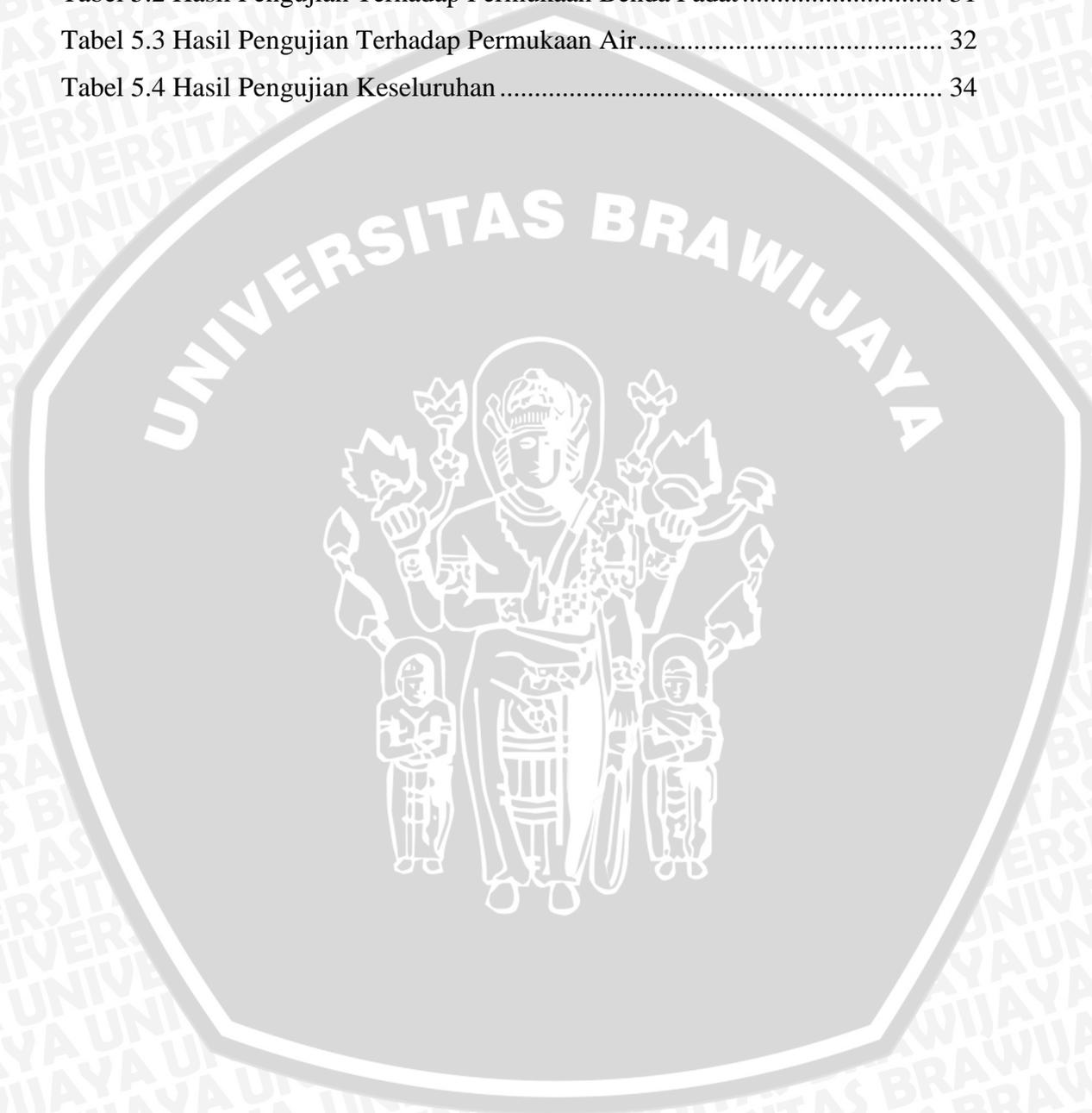
3.1	Perancangan Rangkaian Elektrik	15
3.2	Perancangan Sistem Mekanik	15
3.3	Perancangan Perangkat Lunak	15
3.4	Pengujian Alat.....	16
BAB IV PERANCANGAN ALAT.....		17
4.1	Perancangan Sistem	17
4.2	Perancangan Rangkaian Elektrik	18
4.1.1	Perancangan Rangkaian Pengukur Curah Hujan	18
4.1.2	Perancangan Rangkaian Sensor Ultrasonik	18
4.1.3	Perancangan Sistem Minimum Mikrokontroler.....	19
4.1.4	Perancangan Rangkaian Antarmuka Mikrokontroler dengan Modem GSM	21
4.1.5	Perancangan Rangkaian LCD	22
4.3	Perancangan Sistem Mekanik	23
4.1.6	Perancangan Alat Ukur Curah Hujan <i>Tipping Bucket</i>	23
4.4	Perancangan Perangkat Lunak.....	24
4.1.7	Perancangan Subrutin Pengukuran Curah Hujan.....	26
4.1.8	Perancangan Subrutin Pengukuran Jarak Air Sungai.....	27
4.1.9	Perancangan Subrutin Kirim SMS.....	28
BAB V HASIL DAN ANALISIS.....		29
5.1	Pengujian Rangkaian Alat Ukur Curah Hujan.....	29
5.2	Pengujian Rangkaian Sensor Ultrasonik.....	30
5.3	Pengujian Rangkaian Modem GSM	32
5.4	Pengujian Keseluruhan Sistem	34
BAB VI KESIMPULAN DAN SARAN.....		36
6.1	Kesimpulan	36

6.2	Saran	36
DAFTAR PUSTAKA		37
LAMPIRAN I		38
	Dokumentasi Alat.....	38
LAMPIRAN II.....		39
	Skematik Sistem Minimum Mikrokontroler	39
	Skematik Antarmuka Mikrokontroler dengan Modem GSM (RS-232)	40
	Skematik LCD.....	41
LAMPIRAN III.....		42
	Listing Program	42
LAMPIRAN IV		49
	Datasheet Ultrasonik	49
	Datasheet MAX 232.....	50
	Datasheet Mikrokontroler	52
	Datasheet Modem GSM.....	64



DAFTAR TABEL

Tabel 2.1 Tabel Perintah AT-Command.....	14
Tabel 5.1 Hasil pengujian alat ukur curah hujan.....	29
Tabel 5.2 Hasil Pengujian Terhadap Permukaan Benda Padat	31
Tabel 5.3 Hasil Pengujian Terhadap Permukaan Air.....	32
Tabel 5.4 Hasil Pengujian Keseluruhan	34



DAFTAR GAMBAR

Gambar 2.1 Alat ukur curah hujan tipe Tipping Bucket.....	8
Gambar 2.2 Sistem kerja Sensor Ultrasonik	9
Gambar 2.3 Konfigurasi Pin ATmega32	10
Gambar 2.4 LCD 16x2.....	12
Gambar 2.5 Modem GSM Wavecom Fastrack M1306B.....	13
Gambar 4.1 Diagram Blok Sistem	18
Gambar 4.2 Rangkaian Sensor Ultrasonik.....	19
Gambar 4.3 Konfigurasi Oscilator Eksternal.....	20
Gambar 4.4 Hasil pengamatan bentuk gelombang saat menghidupkan minimum sistem dengan osiloskop.....	20
Gambar 4.5 Rangkaian Reset Mikrokontroler	21
Gambar 4.6 Rangkaian antarmuka mikrokontroler dengan modem	22
Gambar 4.7 Rangkaian LCD.....	22
Gambar 4.8 Tipping Bucket.....	24
Gambar 4.9 Diagram Alir Program Utama.....	25
Gambar 4.10 Subrutin Pengukuran Curah Hujan	26
Gambar 4.11 Subrutin Pengukuran Jarak Air	27
Gambar 4.12 Subrutin Kirim SMS	28
Gambar 5.1 Blok Diagram Pengujian Alat Ukur Curah Hujan	29
Gambar 5.2 Blok Diagram Pengujian Sensor Ultrasonik.....	30
Gambar 5.3 Blok Diagram Pengujian Rangkaian Modem GSM.....	33
Gambar 5.4 Hasil Pengujian Rangkaian Modem GSM.....	33

RINGKASAN

Agung Handoko, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Juni 2015, *Rancang Bangun Sistem Pemantau Kondisi Sungai dengan Fasilitas SMS sebagai Peringatan Dini Bahaya Banjir*, Dosen Pembimbing: Ponco Siwindarto dan Nanang Sulistiyanto.

Banjir adalah salah satu bencana yang sering terjadi di Indonesia yang sangat merugikan. Kerugian besar akibat banjir terjadi karena ketidaksiapan masyarakat dalam menghadapi datangnya bencana banjir. Hal ini terjadi karena kita tidak dapat mengetahui secara cepat jika akan terjadi banjir, padahal hal itu dapat dilakukan apabila kita dapat memantau faktor yang menjadi penyebab banjir, salah satunya adalah meluapnya air sungai. Hingga saat ini pemantauan ketinggian air sungai masih dilakukan secara manual, masih sedikit alat yang digunakan untuk dapat melakukan pemantauan kondisi sungai secara otomatis demi mengurangi kerugian akibat bencana banjir. Oleh karena itu diperlukan alat yang dapat memantau kondisi sungai secara terus menerus dan otomatis serta dapat memberikan peringatan dini kepada masyarakat bila terjadi kondisi bahaya banjir, dengan begitu diharapkan masyarakat dapat mengantisipasi datangnya bencana banjir dan mengurangi kerugian akibat bencana banjir.

Agar alat ini dapat memberikan peringatan dini bahaya banjir, maka alat ini harus mampu memantau faktor penyebab terjadinya banjir, dalam hal ini adalah perubahan ketinggian air sungai. Alat ini akan mengetahui ketinggian permukaan sungai dengan memanfaatkan sensor ultrasonik. Selain itu, alat ini juga dapat mengetahui curah hujan dengan menggunakan alat ukur curah hujan tipe Tipping Bucket. Curah hujan juga perlu dipantau karena curah hujan juga mempengaruhi ketinggian air sungai. Di sinilah letak keunggulan alat ini karena tidak hanya memantau perubahan ketinggian air sungai secara langsung, tetapi juga memantau faktor yang mempengaruhi kecepatan perubahan ketinggian permukaan air sungai itu sendiri. Ketika terjadi kondisi bahaya banjir, maka alat ini akan mengeluarkan peringatan secara otomatis melalui buzzer dan mengirim SMS peringatan melalui modem GSM. Buzzer dan modem GSM inilah yang berfungsi sebagai peringatan dini bahaya banjir.

Berdasarkan pengujian yang dilakukan diketahui sensor ultrasonik dapat mengukur jarak permukaan air sungai hingga 200 cm dengan ketelitian 1 cm dan untuk mengatasi kesalahan pengukuran akibat adanya riak air maka digunakan program yang akan menghitung rata-rata jarak air pada selang waktu 1 menit. Sedangkan alat ukur curah hujan *Tipping Bucket* dapat mengukur curah hujan per menit dan curah hujan per jam dengan ketelitian 0,5 mm. Ketelitian pengukuran curah hujan tersebut sudah cukup dikarenakan perubahan ketinggian air yang drastis terjadi saat curah hujan tinggi. Sebagai peringatan dini bahaya banjir alat ini mampu mengirimkan SMS melalui modem GSM secara otomatis ketika kondisi bahaya banjir terjadi. Secara keseluruhan alat ini bekerja sesuai dengan perancangan dan dapat digunakan sesuai dengan tujuan sebagai pemantau kondisi sungai dan peringatan dini bahaya banjir.

Kata Kunci: banjir, ketinggian air sungai, curah hujan, sensor ultrasonik, *tipping bucket*, SMS.

SUMMARY

Agung Handoko, Department of Electrical Engineering, Faculty of Engineering, Brawijaya University, June 2015, *Design of River Condition Monitoring System with SMS facility as Flood Hazard Early Warning*, Academic Supervisor: Ponco Siwindarto dan Nanang Sulistiyanto.

Flood is one of the disasters that often occur in Indonesia, which is very detrimental. Huge losses due to floods occur due to unpreparedness of society facing of the coming flood. This happens because we are not able to know quickly if there will be a flood, but it can be done if we can monitor the factors that cause flooding, one of which is the overflow of river. Until now monitoring the water level of the river is still done manually, still a little tool to use to be able to monitor river conditions automatically in order to reduce losses due to floods. Therefore we need a tool that can monitor the condition of the river continuously and automatically and can provide early warning to the public when there is danger of flooding, so people can anticipate the arrival of floods and reduce losses due to floods.

So that these tools can provide an early warning flood, then this tool should be able to monitor factors causing flooding, in this case the river water level changes. This tool will know the height of the river surface by using ultrasonic sensors. In addition, this tool also can determine rainfall/precipitation using a measuring instrument precipitation, Tipping Bucket. Rainfall also need to be monitored because of rainfall also affected the water level of the river. Herein lies the advantage of this tool because it not only monitors changes river water level directly, but also monitor the factors that affect the speed of change in the water level of the river itself. When there is a danger of flooding, the tool will automatically issue a warning via buzzer and send SMS alerts through GSM modem. Buzzer and this GSM modem that serves as an early warning system.

Based on tests we made, ultrasonic sensor can measure the distance of the river water level up to 200 cm with a precision of 1 cm and to overcome the measurement error due to the ripple of water, then we use program that will calculate the average of the water level at an interval of 1 minute. While the rainfall measuring instrument, Tipping Bucket can measure rainfall per minute and rainfall per hour with an accuracy of 0.5 mm. Measurement accuracy is sufficient rainfall due to drastic changes in water levels that occur when rainfall is high. As an early warning system, the tool is able to send SMS via GSM modem automatically when the danger of flooding occur. Overall this tool works in accordance with the design and can be used in accordance with its purpose as condition monitoring and early warning system of the danger of the river flooding.

Keywords: flood, river water levels, precipitation, ultrasonic sensors, tipping bucket, SMS.

BAB I PENDAHULUAN

1.1 Latar Belakang

Bencana banjir merupakan salah satu bencana yang sering terjadi di Indonesia. Hampir setiap tahun bencana banjir terjadi dan menyebabkan banyak kerugian, tak jarang banjir juga menelan korban jiwa. Banjir yang terjadi pada tahun awal tahun 2014 di Jakarta saja diperkirakan BPPT (Badan Pengkajian dan Penerapan Teknologi) menyebabkan kerugian sekitar 12 triliun rupiah. Banjir sering mengakibatkan kerusakan rumah dan pertokoan yang dibangun di dataran banjir sungai alami. Meski kerusakan akibat banjir dapat dihindari dengan pindah menjauh dari sungai dan badan air yang lain, orang-orang menetap dan bekerja dekat air untuk mencari nafkah dan memanfaatkan biaya murah serta perjalanan dan perdagangan yang lancar dekat perairan. Manusia terus menetap di wilayah rawan banjir adalah bukti bahwa nilai menetap dekat air lebih besar daripada biaya kerusakan akibat banjir periodik. Salah satu contoh kerugian yang dapat dihasilkan dari banjir seperti yang diberitakan pada sains.kompas.com adalah banjir Manado pada bulan Januari tahun 2014 yang mengakibatkan 25 jiwa meninggal, ribuan rumah terendam dan 40 ribu warga harus mengungsi.

Pengertian banjir berdasar SK SNI M-18-1989-F adalah aliran air yang relatif tinggi, dan tidak tertampung oleh alur sungai atau saluran. Adapun penyebab banjir pada dasarnya adalah meluapnya air pada sungai atau saluran air. Banjir terjadi ketika aliran air melebihi kapasitas sungai atau saluran air, terutama di kelokan sungai. Selain itu, banjir juga dapat disebabkan oleh tingginya curah hujan yang terjadi. Tingginya curah hujan ini dapat mengakibatkan sungai atau saluran air tidak dapat menampung air sehingga sungai atau saluran air meluap dan menyebabkan banjir.

Berbagai upaya dilakukan untuk dapat menanggulangi bencana banjir. Mulai dari usaha pencegahan hingga usaha penanggulangan. Salah satu upaya penanggulangan yang dilakukan adalah membuat sistem peringatan dini. Berbagai sistem peringatan dini telah diciptakan untuk meminimalisir kerugian dari bencana

banjir. Mulai dari cara manual seperti penempatan petugas pemantau pintu air hingga pembuatan alat pemantau sungai otomatis.

Beberapa pernyataan di atas memberikan gambaran kerugian yang disebabkan oleh banjir dan juga menjelaskan bahwa faktor utama terjadinya banjir adalah meluapnya air sungai. Oleh karena itu dibutuhkan suatu alat atau sistem yang dapat mendeteksi tanda-tanda akan terjadinya luapan sungai yang dapat menyebabkan banjir. Adapun alat yang sudah ada saat ini masih memiliki beberapa kekurangan dan terus dikembangkan. Seperti yang terdapat dalam skripsi berjudul *Dual Supply RTU (Remote Telemetry Unit)* pada Sistem Pemantau Banjir Terpusat Menggunakan SMS sebagai Media Transmisi Data yang ditulis oleh Judis Priastono Utama. Pada penelitian sebelumnya pemantauan dilakukan pada ketinggian air dan juga kelembaban dimana kelembabann ini mengindikasikan adanya hujan. Berdasarkan Skripsi tersebut penulis mencoba mengembangkan lebih lanjut hasil dari penelitian tersebut dengan melakukan pemantauan pada ketinggian permukaan air dan juga curah hujan secara langsung. Berdasarkan uraian tersebut, maka penulis mengambil judul “Rancang Bangun Sistem Pemantau Kondisi Sungai dengan Fasilitas SMS sebagai Peringatan Dini Bahaya Banjir”.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, dapat disusun rumusan masalah sebagai berikut:

- 1) Bagaimanakah merancang dan membuat sistem yang dapat mendeteksi ketinggian permukaan air sungai dan curah hujan di daerah aliran sungai sebagai parameter bahaya banjir.
- 2) Bagaimanakah merancang dan membuat sistem pengirim peringatan bahaya banjir menggunakan modul GSM.
- 3) Bagaimanakah merancang dan membuat sistem monitoring sungai.

1.3 Batasan Masalah

Hal-hal yang berkaitan dengan penelitian ini akan diberi batas sebagai berikut:

- 1) Kondisi sungai yang diamati pada penelitian ini adalah ketinggian permukaan air sungai dan curah hujan di sekitar sungai.
- 2) Sensor curah hujan yang digunakan adalah tipe *Tipping Bucket* yang dirancang sendiri.
- 3) Penerima SMS dalam penelitian ini dibatasi hanya 1 orang.

1.4 Tujuan

Tujuan dari penelitian ini adalah membuat suatu sistem pemantau ketinggian permukaan air sungai dan curah hujan di daerah aliran sungai sebagai parameter bahaya banjir dengan fasilitas SMS sebagai peringatan dini bahaya banjir.

1.5 Sistematika Penulisan

Sistematika penulisan dalam penelitian ini sebagai berikut:

BAB I Pendahuluan

Memuat tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika pembahasan.

BAB II Tinjauan Pustaka

Membahas mengenai teori-teori tentang banjir, alat dan sensor yang digunakan, ATmega32 sebagai *microcontroller* dan modul GSM yang mendukung dalam perancangan dan pembuatan alat atau sistem.

BAB III Metodologi

Membahas mengenai metode-metode yang digunakan dalam studi literatur tentang banjir dan sensor-sensor yang digunakan dan perancangan, pembuatan, pengujian, serta analisis data terhadap alat pemantau kondisi sungai.

BAB IV Perancangan

Membahas mengenai penentuan spesifikasi, fungsi dan prinsip kerja dari sensor-sensor yang digunakan untuk memantau kondisi sungai, perancangan diagram blok kerja secara keseluruhan, perancangan perangkat keras alat pemantau kondisi sungai, dan perancangan perangkat lunak alat pemantau kondisi sungai.

BAB V Pengujian dan Analisis

Memuat tentang aspek-aspek pengujian yaitu penjelasan tentang cara pengujian dan hasil pengujian alat pemantau kondisi sungai. Aspek analisis meliputi hasil-hasil

pengujian. Pengujian dan analisis dilakukan pada seluruh sub sistem dan sistem secara keseluruhan.

BAB VI Kesimpulan dan Saran

Memuat tentang intisari hasil pengujian, menjawab rumusan masalah, serta saran untuk perbaikan kualitas penelitian alat pemantau kondisi sungai pada masa mendatang.



BAB II

TINJAUAN PUSTAKA

Teori penunjang yang digunakan dalam penelitian ini diperlukan untuk memahami objek yang kita teliti dan juga untuk memahami prinsip kerja dari masing-masing komponen yang digunakan dalam perancangan dan pembuatan alat sehingga alat dapat bekerja secara efektif. Teori penunjang yang akan dijelaskan dalam bab ini adalah:

- Hujan
- Sungai
- Banjir
- Alat ukur curah hujan
- Sensor Ultrasonik
- Mikrokontroler ATmega32
- *Buzzer*
- Modul *Liquid Crystal Display* (LCD) 4x20 karakter
- Modem GSM Wavecom Fastrack M1360B

2.1 Hujan

Hujan merupakan salah satu bentuk dari presipitasi. Presipitasi sendiri adalah segala bentuk uap air yang mencapai permukaan bumi dari atmosfer. Selain hujan bentuk lain dari presipitasi adalah salju. Di daerah tropis, seperti Indonesia, mayoritas presipitasi terjadi hanya dari hujan. Oleh karena itu, di negara tropis seperti Indonesia, presipitasi juga diartikan atau disamakan sebagai hujan.

Hujan biasanya diukur dalam satuan curah hujan yang merupakan tebal atau tingginya permukaan air hujan yang menutupi suatu area di permukaan bumi sebelum mengalami aliran permukaan, evaporasi, dan peresapan ke dalam tanah. Tebal atau tingginya air hujan ini diukur menggunakan satuan milimeter (mm). Curah hujan 1 mm didefinisikan sebagai tebal atau tinggi air yang tertampung dalam area datar dengan luas 1 m² adalah setinggi 1 mm atau tertampung air sebanyak 1 liter. Biasanya satuan curah hujan dihitung dalam kurun waktu tertentu, seperti per menit, per jam, per hari, per bulan atau per tahun.

Badan Meteorologi Klimatologi dan Geofisika (BMKG) biasanya menggunakan satuan milimeter per jam (mm/jam) dalam mengukur curah hujan. Hal ini berarti tebalnya hujan yang terkumpul dalam suatu area dalam kurun waktu satu jam. Jenis-jenis hujan berdasarkan besarnya curah hujan menurut BMKG dibagi menjadi:

- 1) Gerimis, yaitu hujan dengan tingkat presipitasi kurang dari 2,5 mm/jam.
- 2) Hujan Sedang, yaitu hujan dengan tingkat presipitasi antara 2,5 mm/jam sampai 10 mm/jam.
- 3) Hujan Deras, yaitu hujan dengan tingkat presipitasi antara 10 mm/jam sampai 50 mm/jam.
- 4) Hujan Badai, yaitu hujan dengan tingkat presipitasi lebih dari 50 mm/jam.

2.2 Sungai

Air hujan yang jatuh dari udara di suatu daerah di permukaan bumi untuk sebagian meresap ke dalam tanah, sebagian ditahan tanaman, sebagian lagi menguap kembali dan sisanya mengalir di atas permukaan bumi ke bagian-bagian yang rendah di daerah tersebut, yaitu sungai. Permulaan dari sungai disebut mata air sungai. Sungai mengalirkan air sampai ke laut atau danau ataupun sungai lainnya. Akhir sungai tersebut disebut muara sungai. Sungai memiliki dua variabel yang biasanya diukur untuk digunakan dalam berbagai hal. Kedua variabel tersebut adalah tinggi permukaan sungai dan debit sungai.

Dalam penelitian ini, variabel yang dikumpulkan dari sungai adalah tinggi permukaan sungai. Tinggi permukaan sungai (*river stage*) adalah elevasi muka air pada suatu stasiun di atas datum nol. Kadang-kadang datum diambil sama dengan elevasi air laut rata-rata, tetapi lebih sering diambil sedikit di bawah titik nol aliran sungai.

2.3 Banjir

Menurut Kamus Besar Bahasa Indonesia (KBBI), banjir adalah peristiwa terbenamnya daratan (yang biasanya kering) karena volume air yang meningkat. Definisi kedua dari kamus tersebut, banjir adalah berair banyak dan deras, kadang-kadang meluap. Pengertian kedua ini biasanya dipakai untuk menyebutkan sungai

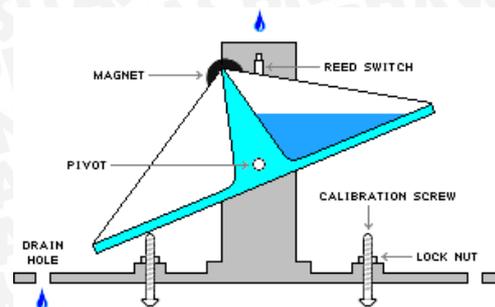
atau kali yang banjir. Banjir sebagai suatu keadaan air yang menenggelami atau menggenangi sesuatu kawasan atau tempat yang luas. Adapula yang mendefinisikan banjir sebagai luapan air yang melebihi dari standar kapasitas akibat hujan yang terus-menerus. Ada lagi yang menyatakan banjir sebagai hadirnya air di suatu kawasan luas sehingga menutupi permukaan bumi kawasan tersebut. Pengertian banjir berdasar SK SNI M-18-1989-F adalah aliran air yang relatif tinggi, dan tidak tertampung oleh alur sungai atau saluran.

Pada dasarnya banjir itu disebabkan oleh luapan aliran air yang terjadi pada saluran atau sungai. Bisa terjadi di mana saja, di tempat yang tinggi maupun tempat yang rendah. Pada saat air jatuh kepermukaan bumi dalam bentuk hujan (*presipitation*), maka air itu akan mengalir ke tempat yang lebih rendah melalui saluran-saluran atau sungai-sungai dalam bentuk aliran permukaan (*run off*) sebagian akan masuk/meresap ke dalam tanah (*infiltration*) dan sebagiannya lagi akan menguap ke udara (*evapotranspiration*). Namun ketika banjir terjadi, hal tersebut tidak terjadi dengan baik.

2.4 Alat Ukur Curah Hujan

Alat ukur curah hujan terbagi menjadi dua macam, yaitu *Non-recording Rainguge* dan *Recording Raingauge*. *Non-recording Rainguge* merupakan alat ukur curah hujan yang hanya dapat mengukur intensitas curah hujan saja. *Non-recording Raingauge* tidak dapat mengukur kapan hujan turun, kapan hujan berakhir, dan fluktuasi selama hujan turun. Sedangkan *Recording Raingauge* merupakan alat ukur curah hujan yang selain dapat mengukur besarnya curah hujan juga dapat mengukur waktu turunnya hujan, lama turunnya hujan serta fluktuasi yang terjadi selama hujan turun. Alat ukur curah hujan yang termasuk kedalam *Recording Raingauge* adalah *Tipping Bucket Type*, *Wheighing Bucket Type* dan *Float Type*.

Alat ukur curah hujan tipe *Tipping Bucket* seperti yang ditunjukkan pada Gambar 2.1 merupakan alat ukur curah hujan yang penulis pilih dalam penelitian ini. Hal ini dikarenakan alat ukur tipe ini paling mudah ditempatkan di berbagai lokasi dan dapat dikontrol dari jauh. Standar alat ukur curah hujan tipe *tipping bucket* dapat mengukur dengan resolusi terkecil 0,01 inchi atau sekitar 0,2 mm/jam (Reddy, 2011).

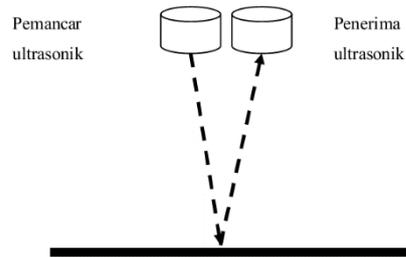


Gambar 2.1 Alat ukur curah hujan tipe
Sumber: Reddy (2011:94)

Prinsip kerja dari alat ukur curah hujan tipe *tipping bucket* ini adalah air hujan akan masuk melalui permukaan corong penakar, kemudian mengalir untuk mengisi salah satu *bucket*. Setiap jumlah air hujan yang masuk sebanyak 0.5 mm. atau sejumlah 20 ml maka *bucket* akan berjungkit, di mana *bucket* yang satunya akan terangkat dan siap untuk menerima air hujan yang akan masuk berikutnya. Pada saat *bucket* berjungkit maka pena akan menggores pias 0,5 skala (0,5 mm.), pena akan menggores pias dengan gerakan naik ataupun turun. Demikianlah seterusnya *bucket* akan bergantian berjungkit bila ada air hujan yang masuk, dari goresan pena pada skala pias dapat diketahui jumlah curah hujannya.

2.5 Sensor Ultrasonik

Gelombang Ultrasonik adalah gelombang yang mempunyai besaran frekuensi lebih dari 20 kHz dan bekerja berdasarkan pantulan gelombang suara. Gelombang ultrasonik bisa merambat pada medium padat, cair dan gas. Sensor ultrasonik terdiri dari dua rangkaian yang bekerja sebagai pemancar ultrasonik (Tx) dan rangkaian penerima (Rx). Rangkaian sensor yang berfungsi sebagai pemancar akan memancarkan gelombang ultrasonik dengan frekuensi tertentu, kemudian apabila terjadi benturan terhadap suatu benda atau objek maka gelombang ultrasonik akan dipantulkan kembali dan diterima oleh rangkaian sensor yang berfungsi sebagai penerima. Maksimum jarak yang dapat dibaca sensor ultrasonik adalah 0 meter sampai dengan 3 meter. Sistem kerja sensor ultrasonik secara sederhana ditunjukkan dalam Gambar 2.2 dimana sensor akan memancarkan gelombang ultrasonik dan apabila gelombang ultrasonik tersebut mengenai permukaan benda maka gelombang tersebut akan memantul dan ditangkap kembali oleh sensor ultrasonik.



Gambar 2.2 Sistem kerja Sensor Ultrasonik

Jarak sensor ke objek pantul dapat dihitung dengan Persamaan (2-1) (Prawiroredjo,2008). Persamaan ini adalah persamaan untuk menghitung jarak dengan menggunakan gelombang ultrasonik. Persamaan ini merupakan turunan dari persamaan untuk menghitung jarak pada umumnya dengan menggunakan kecepatan dan waktu yang diketahui.

$$S = \frac{(t_{IN} \times V)}{2} \quad (2-1)$$

Dengan:

S = Jarak sensor ke objek yang dideteksi

t_{IN} = Selisih waktu pemancaran dan penerimaan pantulan

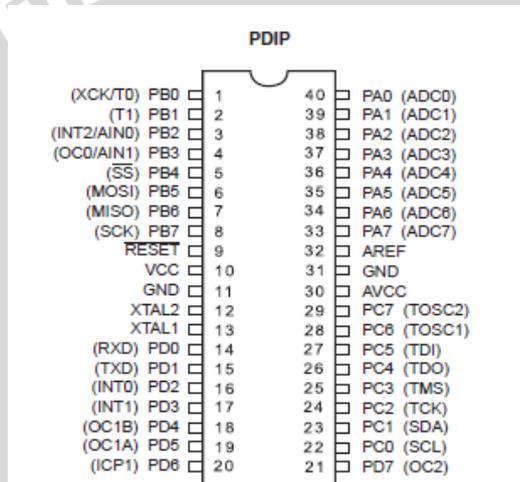
V = Cepat rambat gelombang ultrasonik di udara (344 m/s)

2.6 Mikrokontroler

ATMega 32 adalah IC mikrokontroler *low power* CMOS 8 bit berdasarkan arsitektur AVR RISC. Arsitektur AVR ini menggabungkan perintah secara efektif dengan 32 register umum. Semua register tersebut langsung terhubung dengan *Arithmetic Logic Unit* (ALU) yang memungkinkan 2 register terpisah diproses dengan satu perintah tunggal dalam satu *clock cycle*. Hal ini menghasilkan kode yang efektif dan kecepatan prosesnya 10 kali lebih cepat dari pada mikrokontroler CISC biasa. Mikrokontroler ini memiliki karakteristik sebagai berikut.

- Menggunakan arsitektur AVR RISC
 - 131 perintah dengan satu *clock cycle*
 - 32 x 8 register umum
- Data dan program memori
 - 32 Kb *In-System Programmable Flash*

- 2 Kb SRAM
- 1 Kb *In-System* EEPROM
- 8 Channel 10-bit ADC
- *Two Wire Interface*
- *USART Serial Communication*
- *Master/Slave SPI Serial Interface*
- *On-Chip Oscillator*
- *Watch-dog Timer*
- 32 *Bi-directional I/O*
- Tegangan operasi 2,7 – 5,5 V



Gambar 2.3 Konfigurasi Pin ATmega32

Sumber : Atmel, 2006

Secara fungsional konfigurasi pin ATmega32 adalah sebagai berikut:

- A. VCC merupakan *pin* yang berfungsi sebagai masukan catu daya
- B. GND (Ground) merupakan *pin Ground*.
- C. Port A (PA7 – PA0), port A adalah 8-bit port I/O yang bersifat *bi-directional* dan setiap pin memiliki internal pull-up resistor. *Output buffer port A* dapat mengalirkan arus sebesar 20 mA. Ketika *port A* digunakan sebagai *input* dan dilakukan *pull-up* secara langsung, maka *port A* akan mengeluarkan arus jika internal *pull-up* resistor diaktifkan. Pin-pin dari *port A* memiliki fungsi khusus yaitu dapat berfungsi sebagai channel ADC (*Analog to Digital Converter*) sebesar 10 bit.

- D. Port B (PB7 – PB0), port B adalah 8-bit port I/O yang bersifat *bi-directional* dan setiap pin mengandung internal *pull-up* resistor. *Output buffer port B* dapat mengalirkan arus sebesar 20 mA. Ketika *port B* digunakan sebagai *input* dan dilakukan *pull-down* secara eksternal, *port B* akan mengalirkan arus jika *internal pull-up* resistor diaktifkan.
- E. Port C (PC7 – PC0), *port C* adalah 8-bit port I/O yang berfungsi *bi-directional* dan setiap pin memiliki *internal pull-up* resistor. *Output buffer port C* dapat mengalirkan arus sebesar 20 mA. Ketika *port C* digunakan sebagai *input* dan dilakukan *pull-down* secara langsung, maka *port C* akan mengeluarkan arus jika *internal pull-up* resistor diaktifkan.
- F. Port D (PD7 – PD0), *port D* adalah 8-bit *port I/O* yang berfungsi *bi-directional* dan setiap pin memiliki *internal pull-up* resistor. *Output buffer port D* dapat mengalirkan arus sebesar 20 mA. Ketika *port D* digunakan sebagai *input* dan dilakukan *pull-down* secara langsung, maka *port D* akan mengeluarkan arus jika *internal pull-up* resistor diaktifkan.
- G. *Reset* merupakan *pin* yang digunakan untuk mereset mikrokontroler
- H. XTAL1 merupakan pin masukan *inverting Oscillator Amplifier* dan rangkaian operasional clock
- I. XTAL2 merupakan pin keluaran dari *inverting Oscillator Amplifier*
- J. AVCC merupakan sumber catu daya untuk Port A dan ADC
- K. AREF merupakan *pin* masukan tegangan referensi ADC

2.7 Buzzer

Buzzer adalah suatu alat yang dapat mengubah sinyal listrik menjadi sinyal suara. *Buzzer* terdiri dari alat penggetar yang berupa lempengan yang tipis dan lempengan logam tebal. Bila kedua lempengan diberi tegangan maka electron dan proton akan mengalir dari lempengan satu ke lempengan lain. Kejadian ini dapat menunjukkan bahwa gaya mekanik dan dimensi dapat digantikan oleh muatan listrik.

Bila *buzzer* mendapatkan tegangan maka lempengan 1 dan 2 bermuatan listrik. Dengan adanya muatan listrik maka terdapat beda potensial di kedua lempengan, beda potensial akan menyebabkan lempengan 1 bergerak saling

bersentuhan dengan lempengan 2. Diantara lempengan 1 dan 2 terdapat rongga udara, sehingga apabila terjadi proses getaran di rongga udara maka *buzzer* akan menghasilkan bunyi dengan frekuensi tinggi. *Buzzer* biasanya digunakan sebagai alarm. Frekuensi suara yang keluar dari *buzzer* mencapai 1-5 KHz.

2.8 Modul Liquid Crystal Display (LCD)

Liquid Crystal Display (LCD) merupakan perangkat elektronika termodul yang digunakan untuk menampilkan karakter, baik berupa karakter angka, huruf, atau karakter lainnya, sehingga tampilan tersebut dapat dilihat secara visual. Modul LCD karakter yang ada di pasaran sangatlah bermacam-macam. Jenis modul LCD karakter pada umumnya ditentukan menurut spesifikasi jumlah karakter yang dapat ditampilkan, warna karakter yang ditampilkan, dan juga warna *backlight* LCD. Meskipun memiliki spesifikasi berbeda-beda, jumlah serta fungsi pin LCD karakter tetap sama. Gambar 2.4 berikut menunjukkan konfigurasi pin dari LCD 16×2.



Gambar 2.4 LCD 16×2

Dalam penelitian ini digunakan modul LCD karakter 16×2. LCD ini nantinya akan menampilkan informasi berupa curah hujan dan jarak permukaan air. Adapun spesifikasi dari LCD yang digunakan adalah sebagai berikut:

- Memiliki 16 karakter dan 2 baris tampilan.
- Memerlukan tegangan 5 V DC.
- Otomatis *reset* saat catu daya dinyalakan.
- Memiliki EEPROM 95 *full* (80 karakter).
- Menggunakan 4 bit data.
- Menggunakan 3 bit kontrol.

2.9 RS-232

RS-232 adalah standar komunikasi serial yang didefinisikan sebagai antarmuka antara perangkat terminal data (*data terminal equipment* atau DTE) dan perangkat komunikasi data (*data communications equipment* atau DCE) menggunakan pertukaran data biner secara serial. Di dalam definisi tersebut, DTE adalah perangkat komputer dan DCE sebagai modem walaupun pada kenyataannya tidak semua produk antarmuka adalah DCE yang sesungguhnya. Komunikasi RS-232 diperkenalkan pada 1962. Standar RS-232 mendefinisikan kecepatan 256 kbps atau lebih rendah dengan jarak kurang dari 15 meter. Dengan susunan pin khusus yang disebut null modem cable, standar RS-232 dapat juga digunakan untuk komunikasi data antara dua komputer secara langsung.

2.10 Modem GSM

Modem Fastrack M1306B seperti pada Gambar 2.5 adalah modul komunikasi seluler GSM yang menggunakan prinsip *Plug and Play* sehingga tidak memerlukan instalasi yang rumit untuk dapat menggunakannya. Wavecom Fastrack M1306B juga menyediakan komunikasi data dengan perangkat luar melalui antarmuka serial serta yang dapat diprogram dengan menggunakan perintah – perintah *AT Command* (Wavecom, 2009).



Gambar 2.5 Modem GSM Wavecom Fastrack M1306B
Sumber: Wavecom, (2009:1)

Pada penelitian ini, modem GSM Wavecom Fastrack M1306B digunakan sebagai pengirim SMS. Modem GSM ini juga merupakan penghubung antara mikrokontroler dan *handphone user*. Perintah untuk mengirimkan peringatan dini bahaya banjir dengan SMS nantinya akan dilakukan oleh modem GSM berdasarkan

perintah mikrokontroler. Kemudian digunakan juga rangkaian serial RS-232 untuk menghubungkan modem GSM Wavecom Fastrack M1306B dengan minimum sistem dengan menggunakan konektor tipe DB-9.

2.11 AT-Command

AT-Command adalah perintah yang dapat diberikan kepada handphone atau GSM/CDMA modem untuk melakukan sesuatu hal, termasuk untuk mengirim dan menerima SMS. Dengan memprogram pemberian perintah ini di dalam komputer/mikrokontroler maka perangkat kita dapat melakukan pengiriman atau penerimaan SMS secara otomatis untuk mencapai tujuan tertentu. Komputer ataupun mikrokontroler dapat memberikan perintah AT-Command melalui hubungan kabel secara serial atau bluetooth. Beberapa contoh AT-Command ditunjukkan pada Tabel 2.1

Tabel 2.1 Tabel Perintah AT-Command

AT Command	Keterangan
AT	Mengecek apakah handpone telah tersambung
AT+CMGF	Menetapkan mode dari terminal
AT+CSCS	Menetapkan jenis encoding
AT+CNMI	Mendeteksi pesan SMS
AT+CMGL	Membuka daftar SMS pada SIM Card
AT+CMGS	Mengirim pesan SMS
AT+CMGR	Membaca pesan SMS
AT+CMGD	Menghapus pesan SMS
ATE1	Mengatur ECHO
AT+CSQ	Mengecek kualitas sinyal

Perintah tersebut nantinya akan digunakan untuk memberikan perintah kepada modem GSM melalui mikrokontroler. Hal tersebut dilakukan dengan cara memasukkan perintah AT-Command ke dalam subrutin program bahasa C yang akan ditanam ke dalam mikrokontroler. Melalui perintah AT-Command inilah modem dapat mengirimkan SMS secara otomatis. Hal ini merupakan bentuk dari sistem peringatan dini pada alat.

BAB III

METODE PENELITIAN

Penyusunan proposal ini didasarkan pada masalah yang bersifat aplikatif. Perencanaan dan perealisasiannya dilakukan agar dapat bekerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang dirancang adalah perancangan alat, pembuatan alat, serta pengujian .

3.1 Perancangan Rangkaian Elektrik

Perancangan rangkaian elektrik meliputi perancangan rangkaian sensor, perancangan rangkaian kontrol utama dan perancangan rangkaian output sesuai dengan blok diagram sistem. Rangkaian elektrik sensor sendiri terdiri dari rangkaian elektrik untuk antarmuka sensor ketinggian air dengan mikrokontroler dan rangkaian elektrik untuk antarmuka pengukur curah hujan dengan mikrokontroler. Rangkaian kontrol utama merupakan sistem minimum. Sedangkan rangkaian output terdiri dari rangkaian LCD dan rangkaian elektrik untuk antarmuka modul GSM dengan mikrokontroler.

3.2 Perancangan Sistem Mekanik

Perancangan sistem mekanik adalah perancangan mekanik pengukur curah hujan. Seperti yang telah dijelaskan pada BAB II subbab 2.4 alat ukur curah hujan yang digunakan adalah alat ukur curah hujan tipe *Tipping Bucket*. Perancangan *Tipping Bucket* ini merupakan penentuan ukuran dan bentuk *tipping bucket* yang disesuaikan dengan ketelitian yang dirancang yaitu 0,5 mm. Ukuran yang dirancang berdasarkan ketelitian yang diinginkan adalah corong pengumpul dan pias.

3.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak sistem merupakan perancangan program dalam bahasa C yang akan mengatur kerja dari sistem secara keseluruhan. Pengaturan tersebut meliputi pengaturan sensor, pengolahan data serta pengaturan *output* dari sistem.

Dalam perancangan perangkat lunak, setelah diagram alir dibuat maka digram alir tersebut harus dirubah ke dalam bahasa pemrograman C terlebih dahulu. Hal ini dapat dilakukan dengan menulis program pada *compiler* Atmel Studio. Kemudian progam dengan bahasa C ini akan dirubah menjadi bentuk hex juga menggunakan Atmel Studio. Perangkat lunak atau progam dalam bentuk hex inilah yang akan ditanam ke dalam mikrokontroler dengan menggunakan *downloader* USB. Setelah program berhasil ditanam ke dalam mikrokontroler, maka perancangan dan pembuatan perangkat lunak telah selesai dan sistem telah siap untuk diuji.

3.4 Pengujian Alat

Pengujian alat diperlukan untuk memastikan bahwa alat dapat bekerja sesuai dengan perancangan yang telah dilakukan. Selain itu pengujian juga diperlukan untuk mengetahui performa kerja secara keseluruhan dari alat itu sendiri. Tentunya untuk dapat mengetahui performa keseluruhan dari alat, maka performa dari setiap bagian tentu perlu diketahui. Oleh karena itu, pengujian dilakukan pada setiap bagian dari alat dan sesuai dengan fungsi bagian tersebut.

Pengujian alat dalam skripsi ini dibagi menjadi beberapa bagian sesuai dengan blok diagram kerja. Setelah dilakukan pengujian pada tiap bagian, maka dilakukan pengujian alat secara keseluruhan. Adapun pengujian-pengujian yang dilakukan adalah sebagai berikut.

- 1) Pengujian Rangkaian Sensor Curah Hujan
- 2) Pengujian Modul Sensor Ketinggian Air
- 3) Pengujian Rangkaian LCD
- 4) Pengujian Modul GSM dan *Buzzer*
- 5) Pengujian Alat Keseluruhan

Melalui pengujian ini, diharapkan kita dapat mengetahui secara riil apakah alat dapat berfungsi sesuai dengan perancangan yang telah dilakukan. Hasil dari pengujian inilah yang nantinya akan memberikan gambaran pada kita bagaimana performa kerja dari alat yang telah dirancang. Selain itu kita juga dapat menganalisis faktor-faktor apa saja yang dapat mempengaruhi dari performa kerja alat..Sehingga kedepannya alat dapat dikembangkan dan menjadi lebih baik lagi.

BAB IV

PERANCANGAN ALAT

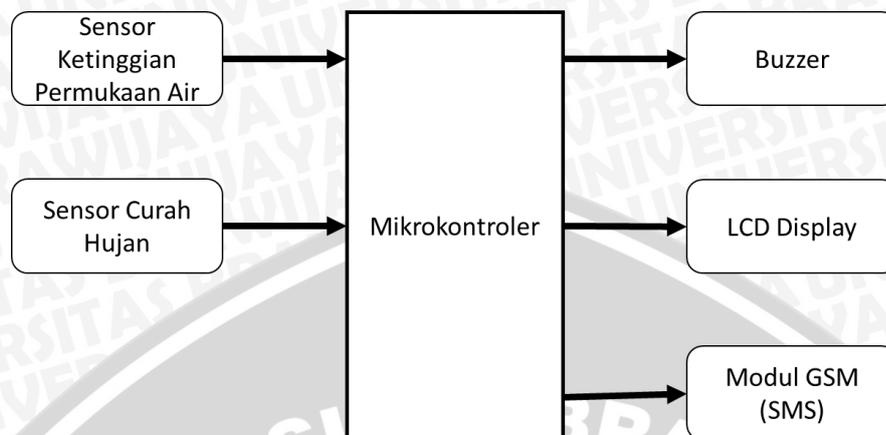
4.1 Perancangan Sistem

Sistem yang dibuat adalah sistem yang dapat memantau kondisi sungai dan memberikan peringatan dini jika terjadi potensi bencana banjir. Adapun fenomena yang akan dipantau oleh sistem adalah fenomena yang dapat menyebabkan banjir. Fenomena tersebut adalah ketinggian permukaan sungai dan intensitas curah hujan pada daerah aliran sungai. Kedua fenomena inilah yang akan menjadi parameter pada sistem ini untuk menentukan apakah ada potensi bahaya banjir atau tidak.

Sistem Pemantau Kondisi Sungai ini secara umum bekerja dengan cara membaca kondisi sungai yang kemudian hasilnya akan ditampilkan dalam LCD dan apabila kondisi sungai dianggap melebihi batas aman atau dengan kata lain berpotensi menyebabkan banjir, maka sistem secara otomatis akan mengirimkan peringatan kepada penduduk sekitar melalui suara dan SMS. Apabila curah hujan kurang dari 40 mm/jam maka batas aman level ketinggian permukaan air adalah 30 cm dari tepi sungai. Sedangkan apabila curah hujan melebihi 40 mm/jam maka batas aman level ketinggian permukaan air adalah 50 cm dari tepi sungai. Artinya sistem akan mengeluarkan peringatan dini bahaya banjir jika level ketinggian permukaan air kurang dari 30 cm dari tepi sungai saat curah hujan di bawah 40 mm/jam atau jika level ketinggian permukaan air kurang dari 50 cm dari tepi sungai saat curah hujan melebihi 40mm/jam.

Sistem dapat menjalankan fungsinya apabila memiliki komponen-komponen yang sesuai dengan cara kerja sistem. Sistem dapat mengetahui fenomena atau keadaan yang terjadi pada sungai tentu melalui sensor sebagai komponen input. Adapun sensor yang digunakan adalah sensor ultrasonik untuk mengetahui level ketinggian permukaan sungai dan sensor curah hujan tipe *Tiping Bucket* untuk mengetahui intensitas curah hujan pada daerah aliran sungai. Sedangkan agar alat ini dapat memberikan peringatan dini akan adanya potensi banjir, maka alat ini akan dilengkapi dengan komponen output berupa *buzzer* sebagai peringatan dini melalui audio dan modul GSM sebagai peringatan dini melalui SMS. Selain itu sistem akan dilengkapi dengan LCD untuk menampilkan kondisi sungai secara langsung. Satu lagi komponen yang sangat penting adalah mikrokontroler sebagai penghubung antara komponen input dan komponen output. Selain itu mikrokontroler yang telah diprogram juga berperan sebagai otak untuk mengolah data

dan menjalankan kerja sistem secara keseluruhan. Berdasarkan uraian tersebut maka diagram blok yang dibuat ditunjukkan pada Gambar 4.1



Gambar 4.1 Diagram Blok Sistem

Alat ini akan terdiri dari dua buah rangkaian sebagai input untuk mikrokontroler yaitu sensor ketinggian permukaan air yang menggunakan sensor ultrasonik dan sensor curah hujan yang menggunakan alat ukur curah hujan tipe *Tipping Bucket*. Kemudian sebagai keluaran dari pengolahan data yang dilakukan oleh mikrokontroler terdapat LCD *Display* yang menampilkan hasil pemantauan baik ketinggian air maupun curah hujan. Buzzer dan modul GSM merupakan bagian dari sistem peringatan dini ahaya banjir dimana keduanya akan aktif saat terjadi kondisi bahaya banjir.

4.2 Perancangan Rangkaian Elektrik

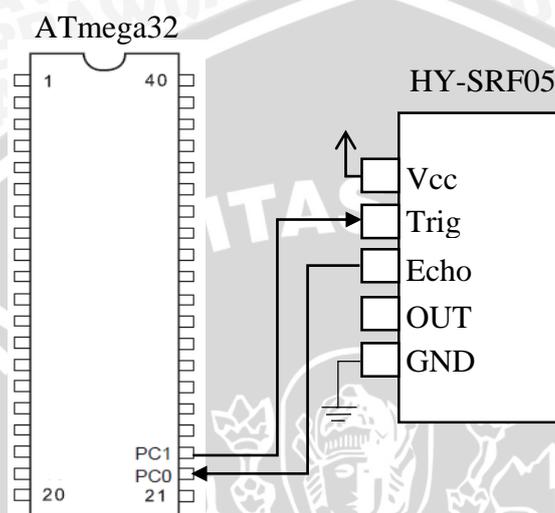
4.1.1 Perancangan Rangkaian Pengukur Curah Hujan

Berdasarkan cara kerja dari alat ukur curah hujan tipe *Tipping Bucket* yang telah dijelaskan dalam BAB II subbab 2.4 maka diperlukan suatu komponen yang dapat mengubah pergerakan mekanik pias menjadi sinyal elektrik. Komponen yang diperlukan untuk mengubah pergerakan mekanik pias menjadi sinyal elektrik adalah *Switch*. *Switch* akan diletakkan di bawah pias yang dapat tertekan ketika pias terjungkit. Melalui *switch* inilah mikrokontroler akan menghitung berapa kali pias terjungkit. *Switch* yang dipilih dalam perancangan ini adalah *limit switch*.

4.1.2 Perancangan Rangkaian Sensor Ultrasonik

Sensor Ultrasonik HY-SRF05 merupakan sebuah sensor yang akan digunakan sebagai perangkat yang akan mengumpulkan data *input* ketinggian air yang akan diolah oleh mikrokontroler. Data yang dihasilkan oleh sensor ini merupakan data digital.

Terdapat 5 buah pin pada sensor ultrasonik, yaitu vcc, Trig, Echo, Data, GND. Sensor ultrasonik ini bekerja pada level tegangan TTL, sehingga tidak dibutuhkan rangkaian pengkondisi sinyal untuk menghubungkan pin pada sensor dengan port pada mikrokontroler. Selain itu pada sensor juga sudah terdapat resistor *pull-up* untuk tiap pinnya sehingga sensor dapat dihubungkan secara langsung dengan mikrokontroler seperti pada Gambar 4.2



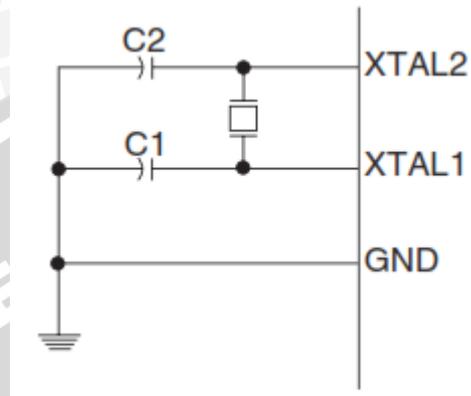
Gambar 4.2 Rangkaian Sensor Ultrasonik

Mode yang digunakan untuk menggunakan sensor ultrasonik ini adalah Mode 1 dimana trigger dan echo menggunakan pin yang terpisah, oleh karena itu, pin Out dibiarkan tidak terhubung. Pin trig akan dihungkan dengan port C1 pada mikrokontroler. Melalui port ini mikrokontroler akan mengirimkan perintah pada *transmitter* sensor ultrasonik untuk mengeluarkan gelombang ultrasonik selama selang waktu tertentu sesuai dengan program. Pin Echo akan dihubungkan dengan port C0 pada mikrokontroler. Melalui port ini mikrokontroler akan membaca apakah *recevier* dari sensor ultrasonik telah menerima pantulan gelombang ultrasonik yang telah dikirim oleh *transmitter*.

4.1.3 Perancangan Sistem Minimum Mikrokontroler

Sistem minimum mikrokontroler merupakan rangkaian yang dibutuhkan agar sebuah mikrokontroler dapat bekerja dengan baik. Dengan kata lain, sistem minimum mikrokontroler merupakan rangkaian paling sederhana agar mikrokontroler dapat digunakan sesuai dengan tujuan sistem setelah mikrokontroler diberikan program. Minimum sistem mikrokontroler terdiri dari 3 rangkaian. Ketiga rangkaian tersebut yaitu rangkaian oscilator eksternal, rangkaian reset dan rangkaian regulator tegangan.

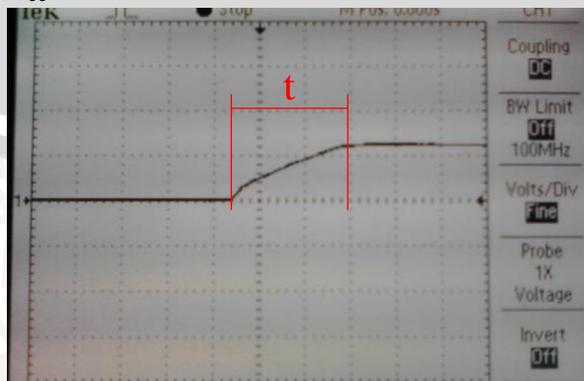
Untuk osilator eksternal digunakan kristal 16,000 MHz untuk memaksimalkan kemampuan mikrokontroler. Diperlukan 2 buah kapasitor dengan nilai kapasitansi 22 pF sesuai dengan spesifikasi yang diberikan oleh *datasheet* untuk menggunakan osilator eksternal. Kedua kapasitor dihubungkan secara paralel dengan kristal dengan port xtal1 dan port xtal2 seperti yang ditunjukkan pada Gambar 4.3



Gambar 4.3 Konfigurasi Oscilator Eksternal

Rangkaian reset terdiri dari resistor *pull-up*, *push button*, dan kapasitor. Hubungan antara waktu reset dengan resistor dan kapasitor ditunjukkan dalam Persamaan (4-1). Berdasarkan *datasheet* ATmega32 dapat diketahui bahwa waktu minimal untuk melakukan *reset* adalah sebesar 1,5 μ s, sedangkan tegangan logika rendah pada PIN Reset (V_{RST}) sebesar $0,1 \times V_{cc}$ dan dengan V_{cc} sebesar 5 V maka didapat V_{RST} sebesar 0,5 V. Nilai resistor yang digunakan menurut *datasheet* harus bernilai antara 30 k Ω sampai 60 k Ω , dengan menentukan nilai resistor sebesar 47 k Ω . Waktu yang dibutuhkan untuk menghidupkan minimum sistem diperoleh berdasarkan pengamatan menggunakan osiloskop (Gambar 4.4) yaitu selama 2,6 ms.

$$t = -R \times C \times \ln \left(1 - \frac{V_{RST}}{V_{cc}} \right) \quad (4-1)$$



Gambar 4.4 Hasil pengamatan bentuk gelombang saat menghidupkan minimum sistem dengan osiloskop

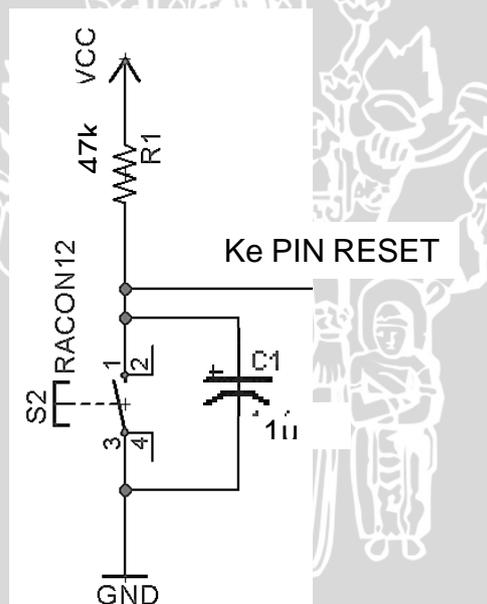
Berdasarkan beberapa hal tersebut, maka dapat kita tentukan besarnya nilai kapasitor dengan menggunakan Persamaan 4-1.

$$2,6 \times 10^{-3} = -(47 \times 10^3) \times C \times \ln\left(1 - \frac{0,5}{5}\right)$$

$$-5,53 \times 10^{-8} = C \times (-0,1054)$$

$$C = 5,2484 \times 10^{-7} F$$

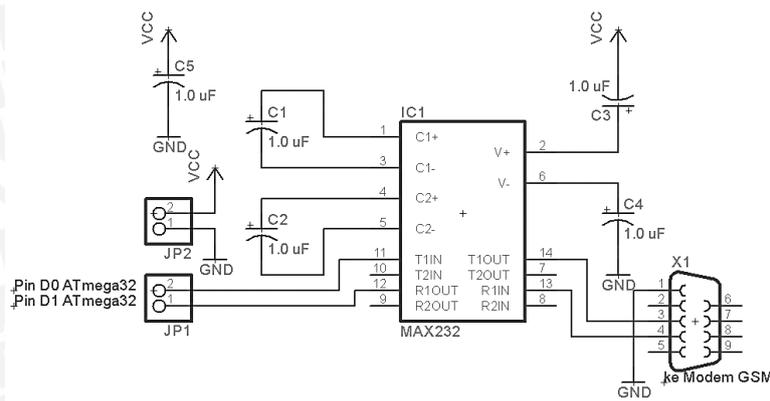
Nilai kapasitor sebesar $5,2484 \times 10^{-7} F$ atau 524,84 nF adalah nilai kapasitor agar rangkaian reset bekerja selama 2,6 ms atau sama dengan waktu menghidupkan minimum sistem, namun karena sulit untuk mendapatkan kapasitor dengan nilai tersebut di pasaran, maka nilai kapasitor akan dibulatkan ke atas menjadi 1 μF . Pembulatan ke atas dilakukan karena waktu untuk menghidupkan minimum sistem tidak boleh kurang dari 2,6 ms dan dengan membuat nilai kapasitor lebih besar maka hal ini akan mematiskan pada saat minimum sistem dihidupkan, power ON reset dapat bekerja. Gambar 4.5 menunjukkan hasil perancangan rangkaian reset minimum sistem mikrokontroler.



Gambar 4.5 Rangkaian Reset Mikrokontroler

4.1.4 Perancangan Rangkaian Antarmuka Mikrokontroler dengan Modem GSM

Modem GSM yang digunakan adalah modem Wavecom Fastrack M1306B dengan konektor RS-232. Oleh karena itu dibutuhkan antarmuka antara modem GSM dengan ATmega32 yang termasuk dalam IC TTL. Untuk menghubungkan keduanya maka digunakan sebuah konverter RS-232 menggunakan IC MAX 232 dengan skema seperti pada Gambar 4.6.

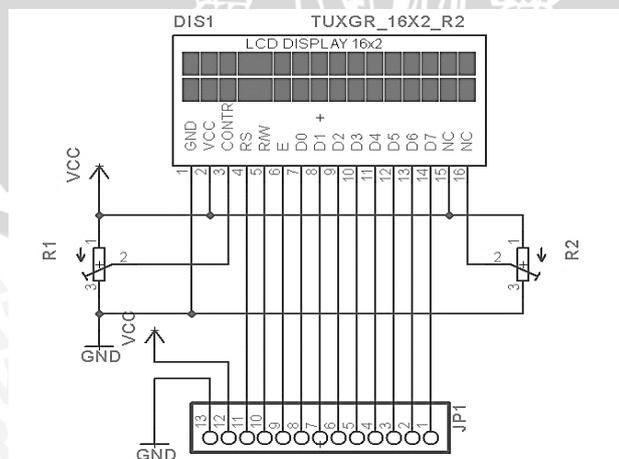


Gambar 4.6 Rangkaian antarmuka mikrokontroler dengan modem

Rangkaian antarmuka mikrokontroler dengan modem GSM ini digunakan sebagai media agar komunikasi kirim dan terima data antar dua perangkat tersebut dapat terlaksana dengan cara menyetarakan level tegangan. Rangkaian antarmuka ini terdiri atas IC MAX232 dan beberapa kapasitor. Besar kapasitor didapat dari datasheet MAX232 milik Texas Instrument yaitu sebesar $1\mu\text{F}$. Rangkaian tersebut disambungkan pada bagian Transmitter dan Receiver pada mikrokontroler dan modem GSM Wavecom Fastrack. Perancangan Perangkat Lunak

4.1.5 Perancangan Rangkaian LCD

Rangkaian LCD (Gambar 4.7) merupakan rangkaian yang terdiri dari antarmuka LCD dengan ATmega32 dan satu buah *variable resistor* yang digunakan untuk mengubah kontras dari karakter LCD. Satu buah *variable resistor* akan dihubungkan ke pin *contr/v_o* dari LCD yang digunakan untuk mengatur kontras dari karakter. Untuk mengatur tingkat kecerahan karakter digunakan trimpot dengan nilai $5\text{ k}\Omega$ yang dihubungkan dengan port CONTR pada LCD.



Gambar 4.7 Rangkaian LCD

Pada perancangan kali ini komunikasi LCD dengan mikrokontroler menggunakan komunikasi 4 bit, sehingga hanya 4 pin data yang digunakan pada LCD yaitu pin D4 sampai pin D7 yang akan dihubungkan dengan port D4 sampai port D7 pada mikrokontroler. Port D4 sampai D5 pada mikrokontroler untuk memudahkan karena memiliki nama yang sama dengan pin pada LCD. Selain itu pin RS akan dihubungkan dengan port B0, pin RW dihubungkan dengan port B1 dan pin E dihubungkan dengan port B2 pada mikrokontroler.

4.3 Perancangan Sistem Mekanik

4.1.6 Perancangan Alat Ukur Curah Hujan *Tipping Bucket*

Perancangan dan pembuatan mekanik dari sensor curah hujan akan mengacu sesuai dengan prinsip kerja Alat Ukur curah Hujan Tipe *Tipping Bucket*. Dimana prinsip kerja dari alat ukur ini adalah air hujan akan masuk melalui permukaan corong penakar, kemudian mengalir untuk mengisi salah satu *bucket*. Setiap jumlah air hujan yang masuk sebanyak 0.5 mm atau sejumlah 20 ml maka *bucket* akan berjungkit, di mana *bucket* yang satunya akan terangkat dan siap untuk menerima air hujan yang akan masuk berikutnya. Maka hal yang pertama harus dilakukan adalah menghitung ukuran dari *bucket/pias*.

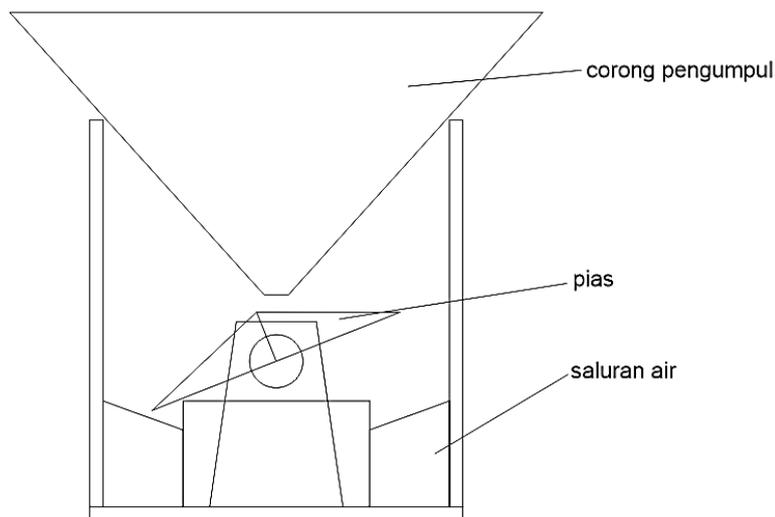
Luas permukaan corong yang digunakan adalah 400 cm² dengan bentuk persegi yang mempunyai panjang sisi 20 cm. Resolusi yang diinginkan adalah 0,5 mm. Berdasarkan dua variabel tersebut kita dapat menentukan volume *bucket/pias* yang akan dibuat dengan menggunakan rumus volume balok.

$$V=p \times l \times t \quad (4-2)$$

$$V=20 \text{ cm} \times 20 \text{ cm} \times 0,05 \text{ cm}$$

$$V=20 \text{ cm}^3=20 \text{ ml}$$

Berdasarkan hasil perhitungan dari Persamaan (4-2) diperoleh bahwa volume yang harus ditampung oleh tiap *pias* adalah 20 ml. *Pias* akan dibuat dengan bentuk prisma segitiga. Alas sisi segitiga dibuat dengan ukuran 10 cm dan tinggi sisi segitiga dengan ukuran 2 cm, sedangkan untuk lebar *pias* dibuat dengan ukuran 4 cm. Volume yang dihasilkan oleh *pias* tersebut adalah 40 cm³ = 40 ml yang terdiri dari dua buah bagian yang masing-masing dapat menampung 20 ml. Berdasarkan perhitungan tersebut maka dibuatlah *Tipping Bucket* dengan bentuk seperti pada Gambar 4.8

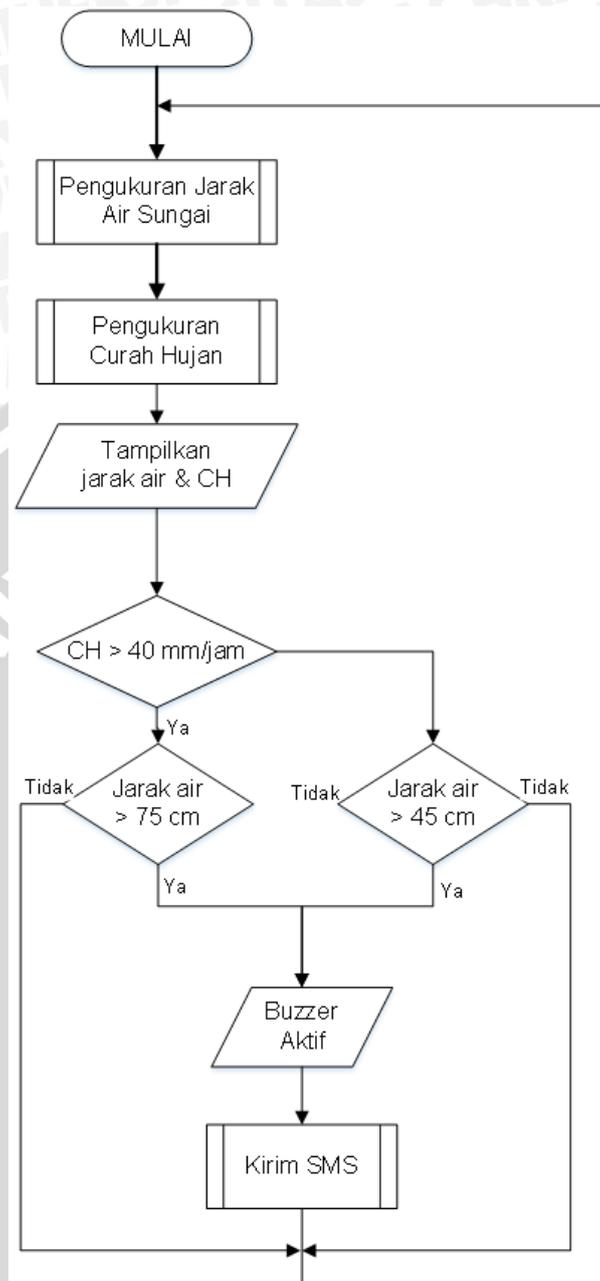


Gambar 4.8 Tipping Bucket

4.4 Perancangan Perangkat Lunak

Perangkat lunak merupakan sebuah program yang akan diberikan kepada mikrokontroler untuk mengolah data yang didapatkan oleh sensor dan kemudian memberikan data yang telah diolah untuk ditampilkan melalui komponen output. Tanpa perangkat lunak, sistem ini tidak akan mampu bekerja sebagaimana mestinya. Karena perangkat lunak menyangkut kerja sistem secara keseluruhan maka dalam perancangan dan pembuatannya tentu harus mengacu kepada subbab perancangan kerja sistem. Perancangan dan pembuatan perangkat lunak sistem ini melalui beberapa tahapan yaitu perancangan diagram alir, pembuatan program dan penanaman program ke dalam mikrokontroler.

Perancangan digram alir dilakukan dengan menerjemahkan isi dari subbab kerja sistem ke dalam bentuk diagram alir seperti pada Gambar 4.9. Pada perancangan diagram alir ini terdapat tiga buah subrutin. Penggunaan subrutin ini dilakukan untuk lebih memudahkan dalam perancangan dan juga untuk merapikan listing program. Ketiga subrutin tersebut adalah subrutin pengukuran jarak air sungai, subrutin pengukuran curah hujan dan subrutin kirim SMS. Subrutin pengukuran jarak air berisi program untuk mengaktifkan sensor ultrasonik agar dapat mengukur jarak permukaan air sungai. Subrutin pengukur curah hujan digunakan agar *tipping bucket* yang mengukur curah hujan secara mekanis dapat diubah menjadi sinyal elektrik hingga dapat ditampilkan pada layar LCD sebagai curah hujan dengan satuan mm. Subrutin ketiga, kirim SMS berisi perintah AT-Command agar modem GSM dapat mengirimkan SMS ketika terjadi kondisi bahaya banjir.

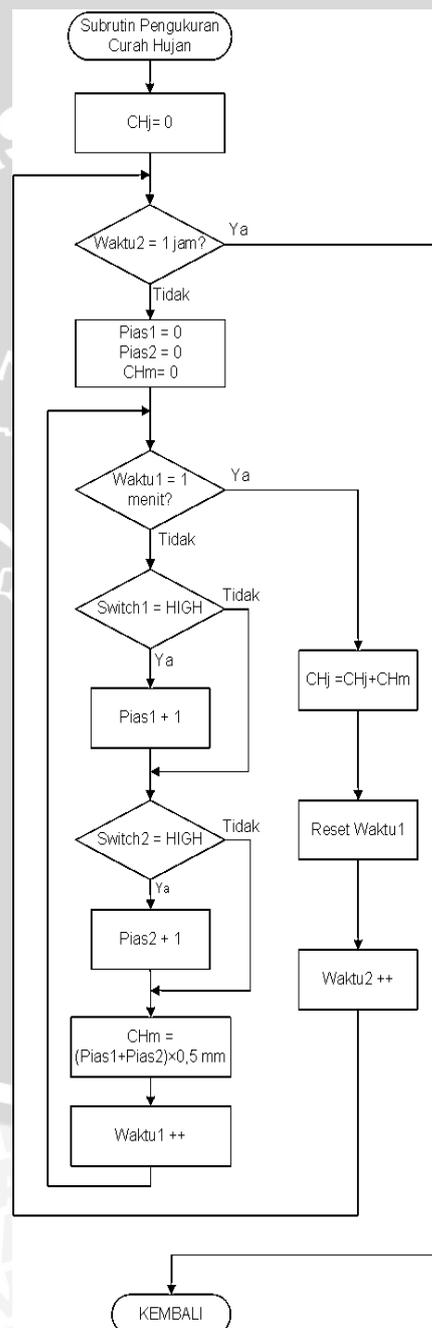


Gambar 4.9 Diagram Alir Program Utama

Pertama-tama, perangkat lunak akan membaca besaran-besaran yang dibaca oleh sensor. Kemudian data dari sensor diolah agar dapat ditampilkan dalam LCD. Sesuai dengan kerja sistem, maka sistem akan mengeluarkan peringatan dini bahaya banjir jika level ketinggian permukaan air kurang dari 50 cm dari tepi sungai saat curah hujan di bawah 40 mm/jam atau jika level ketinggian permukaan air kurang dari 30 cm dari tepi sungai saat curah hujan melebihi 40 mm/jam. Apabila kondisi bahaya banjir tersebut terjadi, maka *buzzer* akan menyala dan alat akan mengirimkan peringatan bahaya banjir secara otomatis.

4.1.7 Perancangan Subrutin Pengukuran Curah Hujan

Pengukuran curah hujan dilakukan secara mekanik dan kemudian dirubah menjadi sinyal elektrik dengan menggunakan *switch*. Sinyal dari *switch* yang berupa sinyal HIGH dan LOW ini merupakan penghitung berapa banyak pias pada *Tipping Bucket* terjungkit, oleh karena itu diperlukan program penghitung untuk mengkonversi hasil penghitungan *switch* menjadi curah hujan. Maka dirancanglah program sesuai dengan kebutuhan tersebut dengan diagram alir seperti pada Gambar 4.10. Secara umum digram alir ini mengubah *counter* (sinyal dari *switch*) menjadi curah hujan.

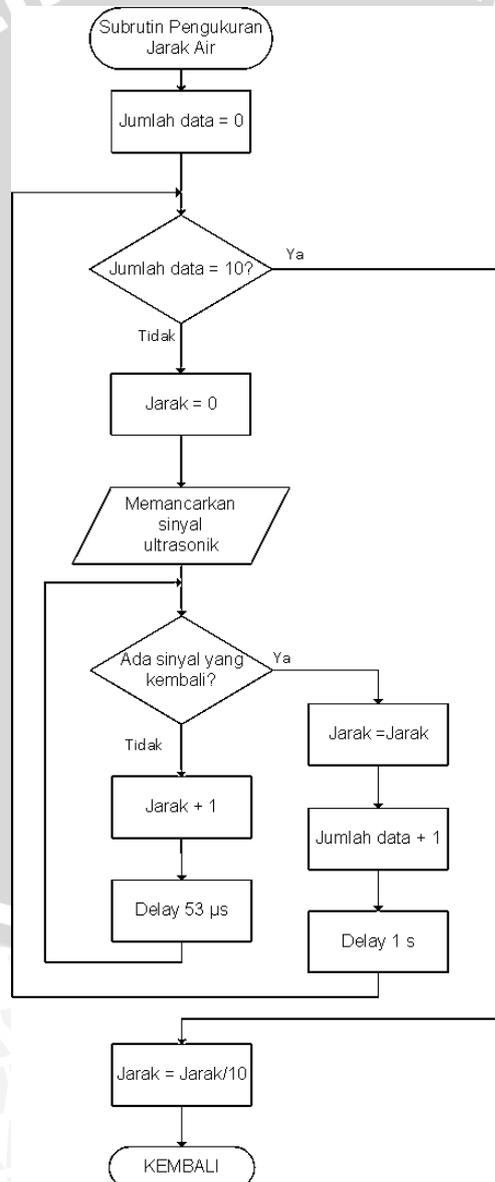


Gambar 4.10 Subrutin Pengukuran Curah Hujan

Switch yang tertekan oleh pias *Tipping Bucket* akan memberikan input berupa logika HIGH kepada mikokontroler. Saat input berlogika HIGH inilah program akan menghitung/*counter up*. Jumlah dari logika HIGH dari kedua *switch* selama 1 menit inilah yang akan dikalikan dengan ketelitian *Tipping Bucket* yaitu sebesar 0,5 mm. Hasil inilah yang merupakan curah hujan per menit (CHm). Kemudian CHm akan dijumlahkan selama 60 menit untuk menjadi curah hujan per jam (CHj).

4.1.8 Perancangan Subrutin Pengukuran Jarak Air Sungai

Subrutin pengukuran jarak air sungai (Gambar 4.11) ini digunakan agar sensor ultrasonik dapat mengukur jarak sesuai dengan perancangan. Program pengukuran jarak air sungai ini memanfaatkan delay yang diatur sesuai dengan Persamaan (2-1).

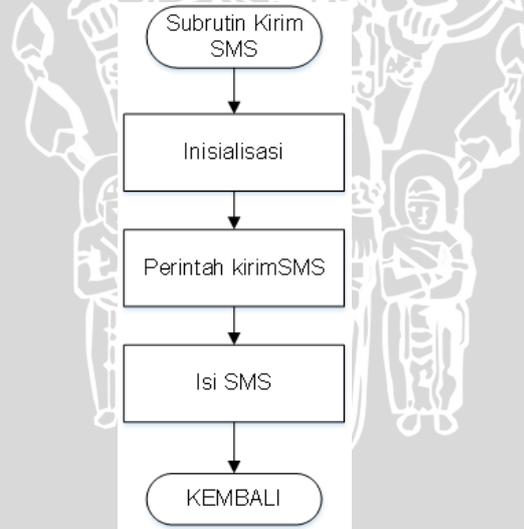


Gambar 4.11 Subrutin Pengukuran Jarak Air

Pada program ini rangkaian sensor ultrasonik akan diprogram agar memancarkan gelombang ultrasonik yang kemudian akan ditangkap kembali oleh sensor ultrasonik. Sensor diprogram untuk memeriksa apakah ada gelombang ultrasonik yang dipantulkan kembali setiap 53 μ s. Jeda ini setara dengan jarak 1 cm sesuai dengan Persamaan (2-1). Hal ini berarti rangkaian sensor ultrasonik diharapkan memiliki ketelitian sebesar 1 cm. Kemudian data dikumpulkan setiap 1 detik sekali selama 10 detik untuk kemudian diambil rata-ratanya.

4.1.9 Perancangan Subrutin Kirim SMS

Program untuk mengirimkan SMS menggunakan perintah AT-Command. Program ini dibutuhkan agar alat dapat mengirimkan SMS secara otomatis. Subrutin program kirim SMS ini diawali dengan inisialisasi seperti dapat dilihat pada Gambar 4.12 untuk mengetahui apakah modem sudah terhubung atau belum, kemudian program akan mengirim SMS dengan menggunakan perintah AT+CMGS yang diikuti dengan nomor telepon penerima. Isi dari SMS berada setelah perintah AT+CMGS. Subrutin ini akan berjalan ketika terjadi kondisi bahaya banjir.



Gambar 4.12 Subrutin Kirim SMS

BAB V
PENGUJIAN DAN ANALISIS

5.1 Pengujian Rangkaian Alat Ukur Curah Hujan

Pengujian Rangkaian Sensor Curah hujan dilakukan untuk mengetahui apakah alat ukur curah hujan dapat mendeteksi intensitas curah hujan sesuai dengan perancangan, di mana curah hujan selama satu jam akan dibandingkan dengan akumulasi curah hujan per menit selama satu jam. Melalui pengujian ini pula dapat diketahui besarnya *error* dari hasil pengukuran sensor.



Gambar 5.1 Blok Diagram Pengujian Alat Ukur Curah Hujan

Pengujian dilakukan dengan cara menuangkan sejumlah air setiap menit pada lima menit pertama dan lima menit terakhir selama kurun waktu satu jam. Volume air yang dituangkan merupakan simulasi air hujan yang diukur oleh *tipping bucket*. Dimana 20 ml air yang dituangkan mewakili 5 mm curah hujan. Dalam setiap penuangannya jumlah volume air yang dituangkan dibuat berbeda-beda. Setiap menit volume air yang dituangkan dicatat dan jumlahnya selama satu jam akan dibandingkan dengan hasil keluaran yang ditampilkan pada LCD.

Tabel 5.1 Hasil pengujian alat ukur curah hujan

Menit	Curah hujan per menit	Curah hujan per jam
0	5,5 mm	0,0 mm
1	22,5 mm	0,0 mm
2	0,5 mm	0,0 mm
3	1,0 mm	0,0 mm
4	10,0 mm	0,0 mm
55	7,0 mm	0,0 mm
56	11,5 mm	0,0 mm
57	2,0 mm	0,0 mm
58	13,0 mm	0,0 mm
59	4,0 mm	0,0 mm
60	0,0 mm	77,0 mm

Pengujian ini menunjukkan bahwa alat ukur curah hujan dapat mengukur curah hujan paling sedikit sebesar 0,5 mm sesuai dengan tingkat ketelitian alat ukur



curah hujan yang dirancang pada subbab 3.2. Selain itu penunjukan curah hujan per jam sudah sesuai dengan jumlah penunjukan curah hujan per menit selama 60 menit. Hasil pengukuran yang ditampilkan pada LCD ini berarti pengukuran curah hujan yang dilakukan secara mekanik oleh *Tipping Bucket* sudah dapat diubah ke dalam bentuk sinyal elektrik dengan memanfaatkan *switch* yang kemudian diolah oleh mikrokontroler hingga bisa ditampilkan hasil pengukuran curah hujan lewat LCD. Hal ini juga menunjukkan bahwa program sudah bekerja sesuai dengan perancangan dimana program mampu mengubah sinyal elektrik yang mewakili volume dari pias *Tipping Bucket* menjadi curah hujan yang ditampilkan LCD. Sinyal elektrik berupa *counter* dimana tiap *counter* mewakili volume 50 ml sudah dapat diubah menjadi curah hujan sebesar 0,5 mm.

5.2 Pengujian Rangkaian Sensor Ultrasonik

Pengujian Rangkaian Sensor Level Ketinggian Permukaan Air dilakukan untuk mengetahui apakah sensor dapat mendeteksi level ketinggian air sesuai dengan perancangan. Melalui pengujian ini pula dapat diketahui besarnya *error* dari hasil pengukuran sensor dengan membandingkan hasil pengukurannya dengan alat ukur standar. Alat ukur standar yang akan digunakan dalam pengujian ini adalah mistar.



Gambar 5.2 Blok Diagram Pengujian Sensor Ultrasonik

Terdapat dua macam pengujian yang dilakukan, yaitu pengujian sensor ultrasonik terhadap permukaan benda padat dan pengujian sensor ultrasonik terhadap permukaan air. Pengujian sensor ultrasonik terhadap permukaan benda padat dilakukan dengan cara menghadapkan sensor ultrasonik ke lantai keramik dengan jarak tertentu. Pengujian sensor ultrasonik terhadap permukaan air dilakukan dengan cara menghadapkan sensor ultrasonik ke permukaan air dalam wadah dengan jarak tertentu. Jarak antara sensor dan permukaan benda ditentukan

mulai dari 10 cm hingga 200 cm dengan perubahan setiap 10 cm. Hasil pembacaan dari sensor kemudian dibandingkan dengan ketinggian penempatan sensor yang telah diatur dengan menggunakan mistar.

Tabel 5.2 Hasil Pengujian Rangkaian Ultrasonik Terhadap Permukaan Benda Padat

Jarak Sebenarnya (cm)	Hasil Pengukuran Sensor (cm)
10	9
20	20
30	30
40	41
50	51
60	60
70	70
80	79
90	89
100	99
110	109
120	119
130	129
140	140
150	149
160	159
170	170
180	179
190	189
200	199

Hasil Pengujian sensor ultrasonik terhadap permukaan benda padat menunjukkan bahwa sensor ultrasonik memiliki *margin error* sebesar 1 cm. Dari 20 data yang diambil hanya terdapat 6 data yang pembacaan jarak oleh sensor akurat. Sementara 12 data pembacaan jarak oleh sensor 1 cm lebih kecil dari jarak sebenarnya dan 2 data pembacaan jarak oleh sensor 1 cm lebih besar dari jarak sebenarnya. Error 1 cm tidak terlalu berpengaruh terhadap performansi sistem, hal ini karena perubahan 1 cm ketinggian air cukup kecil dan terjadi dalam waktu yang relatif lama. Selain itu, pada program sudah terdapat perintah untuk menghitung rata-rata jarak air selama 1 menit, sehingga perubahan ketinggian air sebesar 1 cm tidak akan berpengaruh besar. Perubahan jarak air sebesar 1 cm tersebut secara alami juga terjadi akibat adanya riak air sungai. Jadi error 1 cm tidak berpengaruh.

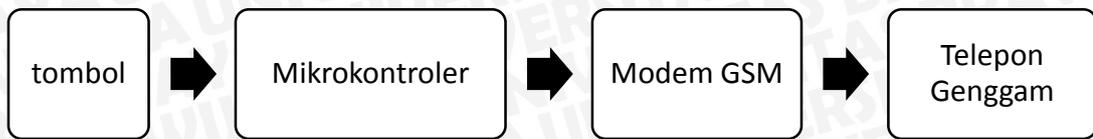
Tabel 5.3 Hasil Pengujian Rangkaian Ultrasonik Terhadap Permukaan Air

Jarak Sebenarnya (cm)	Hasil Pengukuran Sensor (cm)
10	10
20	19
30	29
40	39
50	49
60	60
70	69
80	79
90	89
100	99
110	110
120	120
130	130
140	139
150	149
160	160
170	169
180	179
190	189
200	199

Hasil yang didapat dari pengujian sensor ultrasonik terhadap permukaan air sama halnya dengan hasil pengujian sensor ultrasonik terhadap permukaan benda padat dimana *margin error* dari sensor sebesar 1 cm. Berdasarkan kedua pengujian tersebut dapat kita simpulkan bahwa permukaan benda yang diukur oleh sensor ultrasonik tidak terlalu berpengaruh terhadap hasil pengukuran.

5.3 Pengujian Rangkaian Modem GSM

Pengujian rangkaian modem GSM dilakukan dengan tujuan untuk mengetahui apakah modem GSM dapat mengirim SMS kepada handphone user dan menerima pesan SMS dari telepon genggam pengguna. Pengiriman SMS juga dilakukan untuk mengetahui apakah rangkaian serial RS-232 yang menggunakan IC MAX 232 dapat bekerja sesuai dengan perancangan. Hal ini dilakukan untuk mengetahui apakah data yang dikirim mikrokontroler dapat sampai hingga telepon genggam pengguna secara utuh.



Gambar 5.3 Blok Diagram Pengujian Rangkaian Modem GSM

Pengujian kirim SMS dilakukan dengan cara memprogram mikrokontroler untuk mengirimkan AT-Command kepada modul GSM menggunakan komunikasi serial dimana AT-Command ini digunakan untuk mengirimkan sebuah SMS kepada handphone user, kemudian isi SMS itu dicocokkan dengan SMS yang diterima oleh user. Pengujian terima SMS dilakukan dengan cara memprogram mikrokontroler untuk mengirimkan AT-Command kepada modul GSM menggunakan komunikasi serial dimana AT-Command ini digunakan untuk menerima pesan SMS dari handphone user, kemudian isi SMS dicocokkan dengan SMS yang dikirim oleh user.



Gambar 5.4 Hasil Pengujian Rangkaian Modem GSM

SMS yang diterima oleh handphone sama dengan data yang ada pada mikrokontroler yang ditunjukkan pada LCD. Hal ini menunjukkan bahwa rangkaian antarmuka mikrokontroler dengan modem GSM bekerja dengan baik karena perintah dari mikrokontroler dapat diterima oleh modem melalui rangkaian antarmuka untuk selanjutnya modem mengirim SMS sesuai dengan perintah mikrokontroler. Hal ini menunjukkan bahwa rangkaian antarmuka mikrokontroler dengan modem GSM dapat bekerja sesuai dengan perancangan dengan meneruskan perintah mengirim SMS dari mikrokontroler kepada modem GSM tanpa ada data yang hilang. Hal ini juga menunjukkan bahwa rangkaian RS-232 dapat bekerja

sesuai perancangan, dimana rangkaian ini dapat meneruskan perintah dari mikrokontroler kepada modem GSM Wavecom Fastrack M1306B.

5.4 Pengujian Keseluruhan Sistem

Pengujian alat secara keseluruhan dilakukan dengan cara membuat simulasi beberapa kondisi, mulai dari kondisi normal hingga kondisi bahaya banjir. Kondisi normal adalah saat level ketinggian air lebih dari 75 cm. Kondisi bahaya banjir terpenuhi dengan dua kondisi, yaitu saat curah hujan kurang dari 40 mm/jam dan level ketinggian air kurang dari 45 cm dan saat curah hujan lebih dari 40 mm/jam dan level ketinggian air kurang dari 75 cm. Melalui pengujian ini dapat diketahui apakah sistem dapat bekerja sesuai dengan perancangan atau tidak. Melalui pengujian ini dapat diketahui apakah LCD menampilkan data yang sesuai dan apakah Modem GSM dan *Buzzer* dapat memberikan peringatan dini saat terjadi bahaya banjir.

Tabel 5.4 Hasil Pengujian Keseluruhan

Curah Hujan (mm/jam)	Jarak Permukaan Air (cm)	Buzzer	SMS
30	100	OFF	-
30	90	OFF	-
30	80	OFF	-
30	70	OFF	-
30	60	OFF	-
30	50	OFF	-
30	40	ON	TERKIRIM
30	30	ON	-
40	100	OFF	-
40	90	OFF	-
40	80	OFF	-
40	70	ON	TERKIRIM
40	60	ON	-
40	50	ON	-
40	40	ON	-
40	30	ON	-

Dari hasil pengujian keseluruhan didapatkan hasil bahwa alat dapat bekerja sesuai dengan perancangan. Semua komponen dapat bekerja dengan baik. Sensor ultrasonik dapat menunjukkan jarak permukaan air dan alat ukur curah hujan dapat mengukur curah hujan sesuai perancangan. Program juga dapat bekerja sesuai

perancangan dimana hal ini terlihat saat curah hujan 30 mm/jam dan jarak permukaan air 40 cm maka *buzzer* menyala dan modem GSM mengirimkan SMS peringatan. Begitu pula pada kondisi curah hujan 40 mm/jam dan jarak permukaan air mencapai 70 cm *buzzer* akan menyala dan modem akan mengirimkan SMS. Hal ini sesuai dengan kondisi bahaya banjir yang ditentukan dalam perancangan dimana kondisi banjir adalah saat jarak permukaan air sama dengan 45 cm atau kurang ketika curah hujan di bawah 40 mm/jam atau jarak permukaan air sama dengan 75 cm atau kurang ketika curah hujan sama dengan 40 mm/jam atau lebih.



BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan dan pembahasan berikut adalah kesimpulan yang dapat diambil, yaitu:

1. Sistem pemantau kondisi ini terdiri atas alat ukur curah hujan tipe tipping bucket dengan ketelitian 0,5 mm dan sensor ultrasonik yang dapat mengukur jarak permukaan air dengan *margin error* 1 cm.
2. Sistem peringatan dini pada alat ini menggunakan modem GSM yang dapat mengirimkan SMS peringatan secara otomatis ketika terjadi kondisi bahaya banjir.
3. Dengan *margin error* 1 cm pada sensor ultrasonik dan tipping bucket dengan ketelitian 0,5 mm alat ini mampu memberikan peringatan dini bahaya banjir melalui buzzer dan SMS ketika ketinggian air kurang dari 45 cm saat curah hujan 20-39 mm atau ketika ketinggian air kurang dari 75 cm saat curah hujan sama dengan atau lebih dari 40 mm/jam.

6.2 Saran

Saran yang dapat peneliti berikan untuk penelitian selanjutnya yaitu pada penelitian berikutnya dapat ditambahkan sensor untuk mengetahui debit air dari sungai.

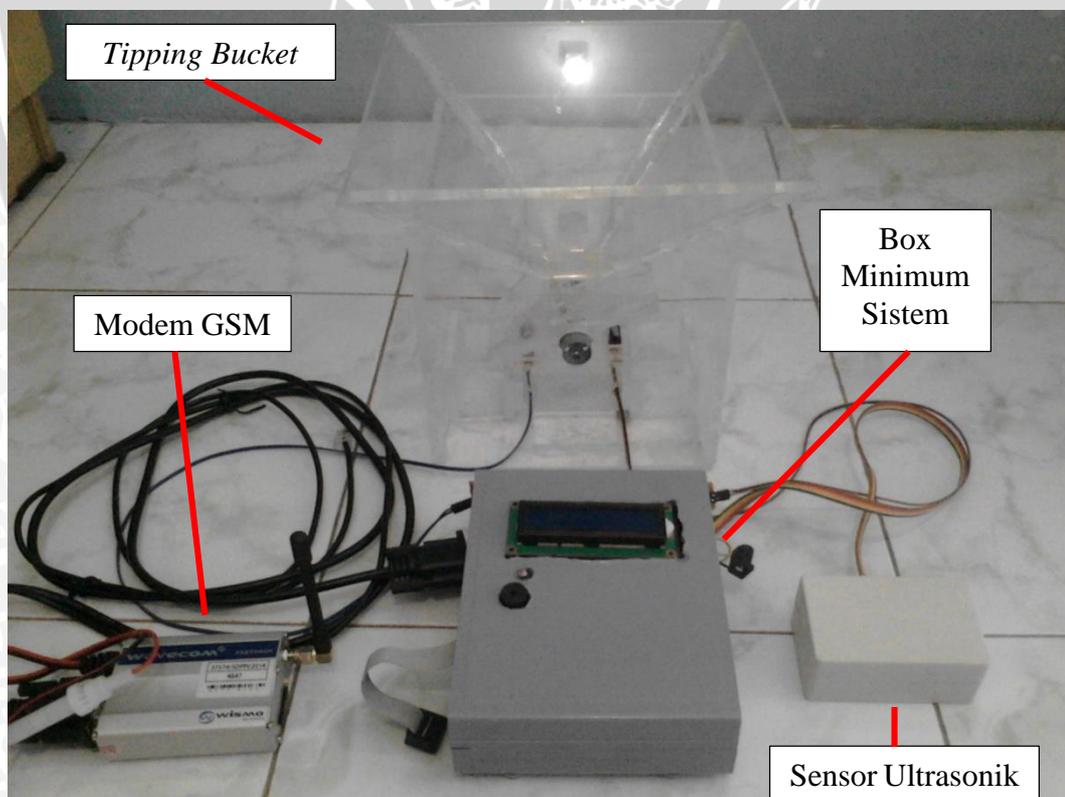
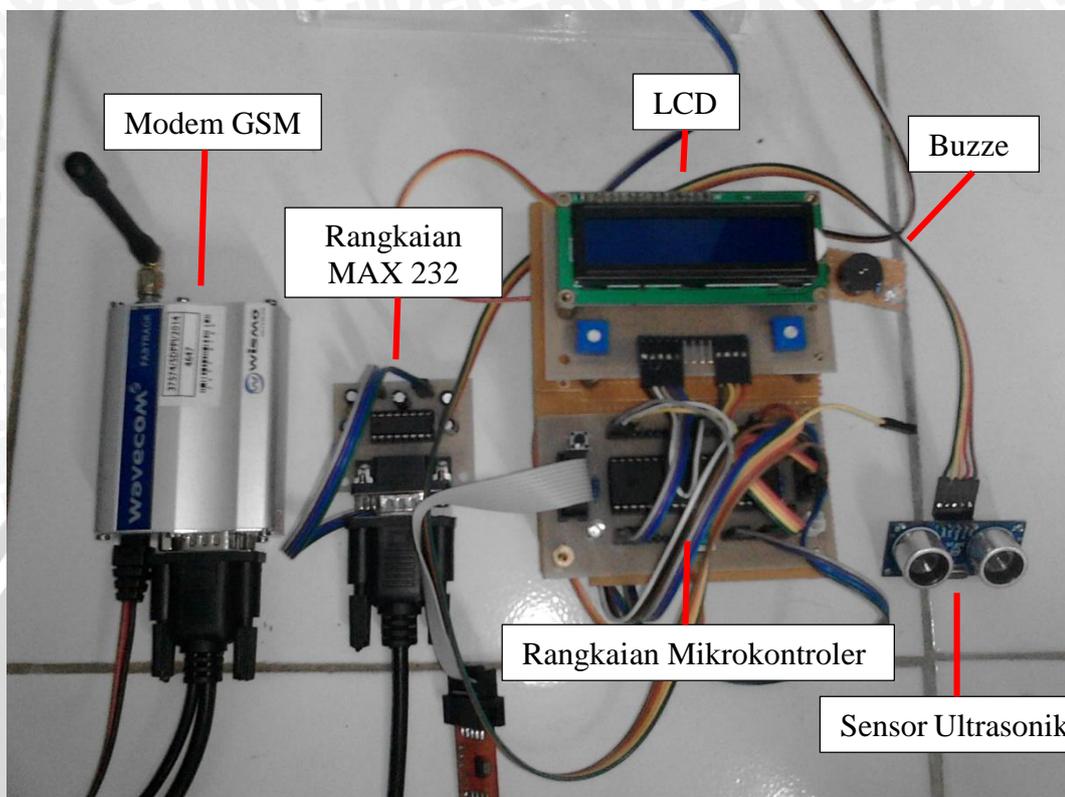
DAFTAR PUSTAKA

- Atmel. 2006. ATmega32 Datasheet. San Diego: Atmel.
- Boylestad, R., & Nashelsky, L. 2012. *Electronic Devices and Circuit Theory 11th Edition*. New Jersey: Prentice Hall.
- Maxim. 2015. MAX220-MAX249. California: Maxim
- Naryanto, H. S. 2009. *Indonesia Diantara Berkah dan Musibah*. Jakarta: Kementrian Negara Riset dan Teknologi.
- Reddy, P. J. 2011. *A Textbook of Hydrology*. New Delhi: University Science Press.
- Sosrodarsono, S. 1999. *Hidrologi untuk Pengairan*. Jakarta: Pradnya Paramita.
- Subarkah, I. 1978. *Hidrologi untuk Perancangan Bangunan Air*. Bandung: Idea Dharma.
- Vishay. 2002. 16 x 2 LCD Character. Pennsylvania: Vishay.
- Wavecom. 2006. Fastrack M1306B User Guide.
- Weathershack. 2009. *Tipping Bucket Rain Gauge*. Dipetik Februari 24, 2015, dari <https://www.weathershack.com/static/ed-tipping-bucket-rain-gauge.html>
- Wilson, J. S. 2005. *Sensor Technology Handbook*. Oxford: Elsevier.
- Winoto, A. 2010. *Mkrokontroler AVR ATmega8/16/32/8535 dan Pemrogramannya dengan Bahasa C WinAVR*. Bandung: Penerbit Informatika.



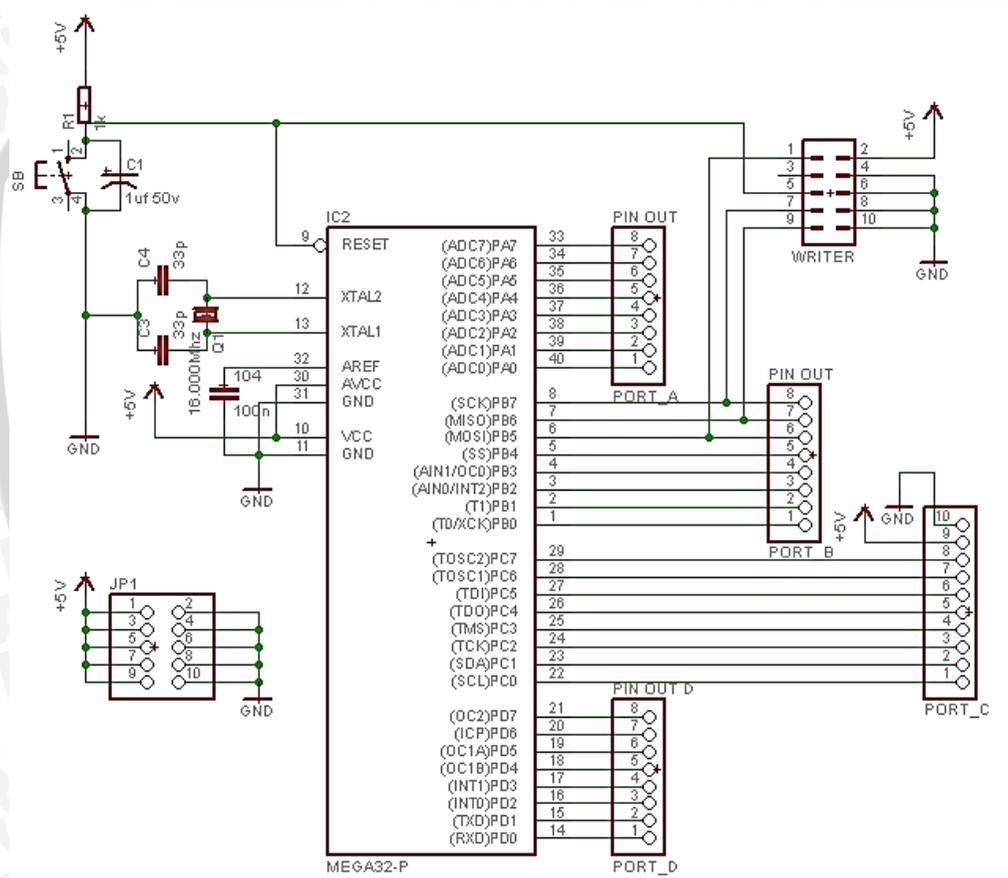
LAMPIRAN I

Dokumentasi Alat

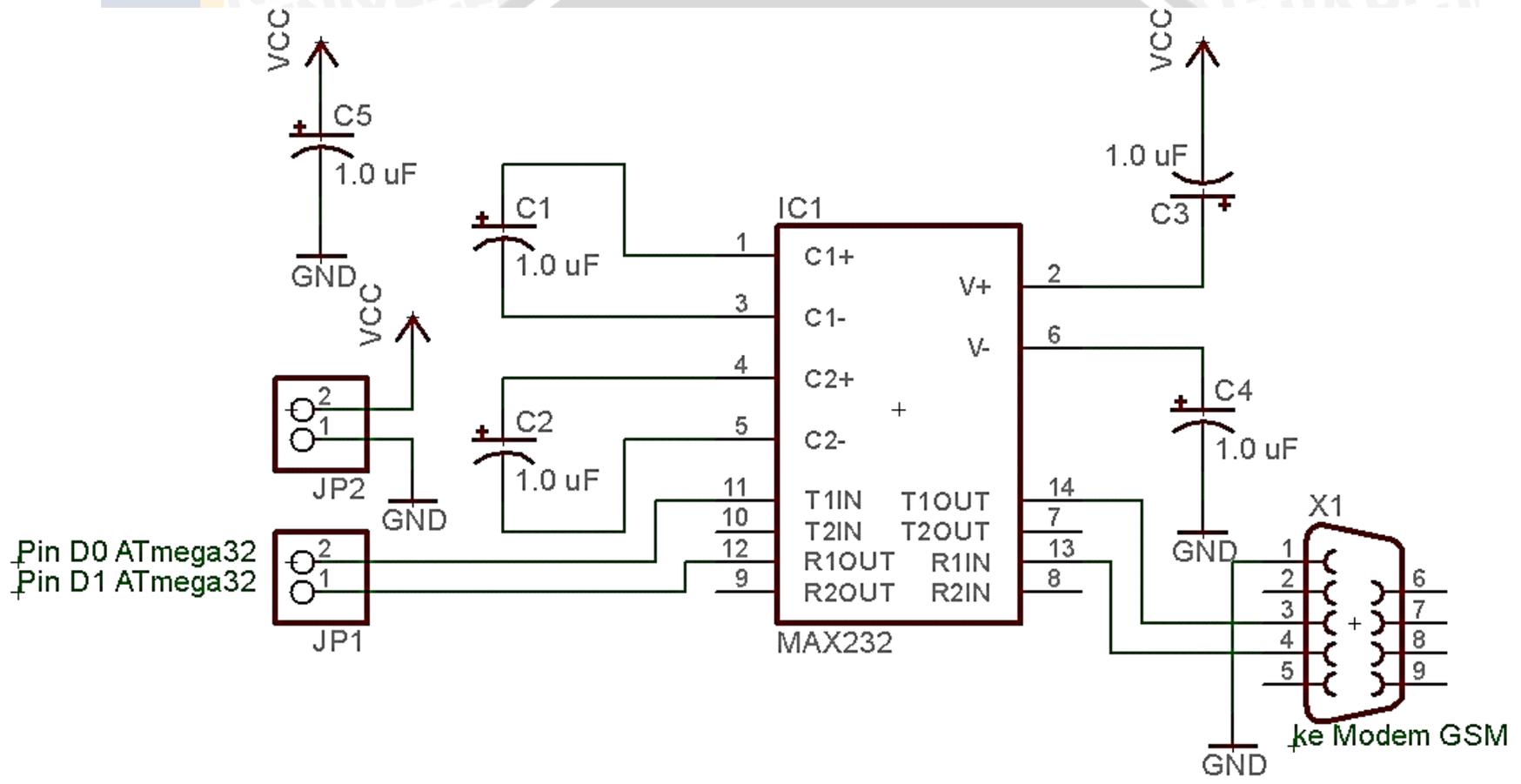


9. LAMPIRAN II

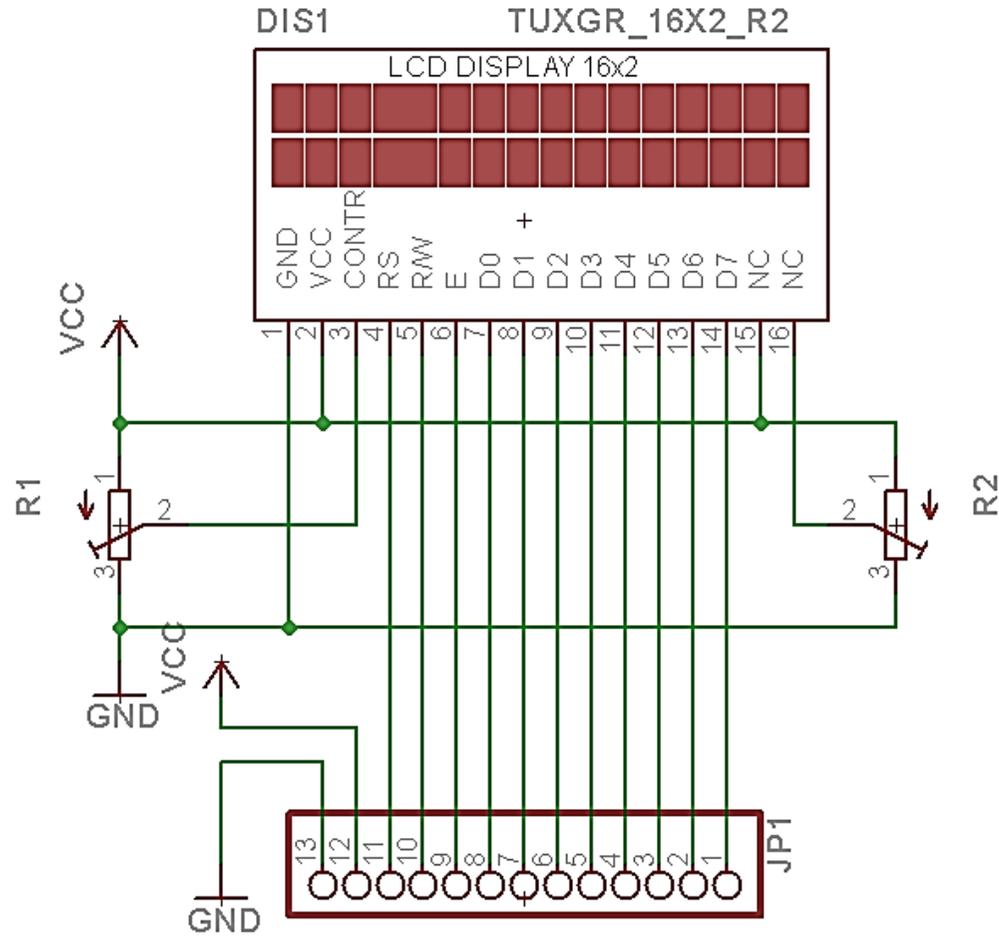
Skematik Sistem Minimum Mikrokontroler



Skematik Antarmuka Mikrokontroler dengan Modem GSM (RS-232)



Skematik LCD



LAMPIRAN III

Listing Program

```

#include <mega32.h>                                unsigned char bufch1[16];

// Alphanumeric LCD functions                      int jarak_t, jarak_a, jarak;
#include <alcd.h>                                    char bufj[16];

// Standard Input/Output functions                //curah hujan
#include <stdio.h>

#include <delay.h>                                  void tippingbucket()
#include <stdlib.h>                                  {
#define pias1 PINA.0                                if(pias1==off)
#define pias2 PINA.1                                status1=on;
#define on 0                                         if((pias1==on)&&(status1==on))
#define off 1                                       {
                                                       jungkit1++;
                                                       status1=off;
                                                       }
#define trigger PORTC.1                             if(pias2==off)
#define echo PINC.0                                 status2=on;
#define buz PORTC.2                                 if((pias2==on)&&(status2==on))
                                                       {
// Declare your global variables here                jungkit2++;
                                                       status2=off;
                                                       }
bit status1=0, status2=0;
int  jungkit,  jungkit1=0,  jungkit2=0,
dstat=0, mstat=0, jstat=0;                          jungkit=jungkit1+jungkit2;
float  ch_mt=0,  ch_m=0,  ch_jt=0,                  ch_mt=(float)jungkit/2;
ch_j=0;
unsigned char bufch[16];                             }

```

```

void ultrasonik()
{
    unsigned int i;
    jarak_t=0;
    delay_us(100);
    trigger=1;
    delay_us(15);
    trigger=0;
    delay_us(100);
    while(!echo);

    for (i=0;i<=500;i++)
    {
        if (echo)
        {
            jarak_t++;
        }
        delay_us(58);
    }
}

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
    // Reinitialize Timer1 value
    TCNT1H=0xC2F7 >> 8;
    TCNT1L=0xC2F7 & 0xff;
    // Place your code here

    //curah hujan
    dstat++;

    if(dstat>=60)
    {
        ch_m=ch_mt;
        jungkit1=0;
        jungkit2=0;
        dstat=0;
        ch_jt=ch_jt+ch_m;
        mstat++;
    }
    if(mstat>=60)
    {
        ch_j=ch_jt;
        ch_jt=0;
        mstat=0;
        jstat++;
    }
    if(jstat>=24)
    {
        jstat=0;
    }
}

//ultrasonik
jarak_a=jarak_a+jarak_t;
dstat++;
if(dstat==10)
{
    jarak=jarak_a/10;
    dstat=0;
    jarak_a=0;
}
}

void init_sms()
{

```

```

lcd_gotoxy(0,0);
lcd_putsf("intialization");

printf("ATE0");
putchar(0x0D);
delay_ms(1000);

printf("AT+CMGF=1");
putchar(0x0D);
delay_ms(1000);

printf("AT+CMGD=1");
putchar(0x0D);
delay_ms(1000);

printf("AT+CMGD=1");
putchar(0x0D);
delay_ms(1000);

lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("initialization success");
delay_ms(5000);

lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Modem GSM");
lcd_gotoxy(0,1);
lcd_putsf("Tidak Aktif");

}

void kirim_sms()
{
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("BAHAYA BANJIR!!!");
    delay_ms(3000);
}

printf("AT+CMGF=1");
putchar(0x0D);
delay_ms(500);

printf("AT+CMGD=1");
putchar(0x0D);
delay_ms(500);

lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Mengirim SMS");

printf("AT+CMGS=089672899234");
putchar(0x0D);
delay_ms(4000);

printf("BAYAHA NBANJIR!!!");
putchar(0x0D);
delay_ms(1000);

putchar(0x1A);
//while(getchar()!=0x0A){};
lcd_clear();
delay_ms(3000);

lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("selesai");

void main(void)
{

```

```

// Declare your local variables here           // Func7=In  Func6=In  Func5=In
                                                // Func4=In  Func3=In  Func2=In  Func1=In
                                                // Func0=In
// Input/Output Ports initialization
// Port A initialization                       // State7=T  State6=T  State5=T  State4=T
                                                // State3=T  State2=T  State1=T  State0=T
Func7=In  Func6=In  Func5=In
Func4=In  Func3=In  Func2=In  Func1=In
Func0=In
// State7=T  State6=T  State5=T  State4=T
State3=T  State2=T  State1=T  State0=T

PORTA=0x00;
DDRA=0x00;

// Port B initialization                       // Timer/Counter 0 initialization
                                                // Clock source: System Clock
                                                // Clock value: Timer 0 Stopped
                                                // Mode: Normal top=0xFF
                                                // OCO output: Disconnected
Func7=In  Func6=In  Func5=In
Func4=In  Func3=In  Func2=In  Func1=In
Func0=In
// State7=T  State6=T  State5=T  State4=T
State3=T  State2=T  State1=T  State0=T

PORTB=0x00;
DDRB=0x00;

// Port C initialization                       // Timer/Counter 1 initialization
                                                // Clock source: System Clock
                                                // Clock value: 16000,000 kHz
                                                // Mode: Normal top=0xFFFF
                                                // OC1A output: Discon.
                                                // OC1B output: Discon.
                                                // Noise Canceler: Off
                                                // Input Capture on Falling Edge
                                                // Timer1 Overflow Interrupt: On
                                                // Input Capture Interrupt: Off
                                                // Compare A Match Interrupt: Off
                                                // Compare B Match Interrupt: Off
Func7=In  Func6=In  Func5=In
Func4=In  Func3=In  Func2=Out
Func1=Out  Func0=In
// State7=T  State6=T  State5=T  State4=T
State3=T  State2=0  State1=0  State0=T

PORTC=0x00;
DDRC=0x06;

// Port D initialization                       TCCR1A=0x00;
                                                TCCR1B=0x05;

```

```

TCNT1H=0xC2; // Communication Parameters: 8 Data, 1
TCNT1L=0xF7; // Stop, No Parity
ICR1H=0x00; // USART Receiver: On
ICR1L=0x00; // USART Transmitter: On
OCR1AH=0x00; // USART Mode: Asynchronous
OCR1AL=0x00; // USART Baud Rate: 9600
OCR1BH=0x00;
OCR1BL=0x00; UCSRA=0x00;
UCSRB=0x18;
// Timer/Counter 2 initialization UCSRC=0x86;
// Clock source: System Clock UBRRH=0x00;
// Clock value: Timer2 Stopped UBRRL=0x67;
// Mode: Normal top=0xFF
// OC2 output: Disconnected // Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by
ASSR=0x00; // Timer/Counter 1: Off
TCCR2=0x00;
TCNT2=0x00; ACSR=0x80;
OCR2=0x00; SFIOR=0x00;
// External Interrupt(s) initialization
// INT0: Off // ADC initialization
// INT1: Off // ADC disabled
// INT2: Off
ADCSRA=0x00;
MCUCR=0x00;
MCUCSR=0x00; // SPI initialization
// SPI disabled
// Timer(s)/Counter(s) Interrupt(s)
initialization SPCR=0x00;
TIMSK=0x04; // TWI initialization
// TWI disabled
// USART initialization

```

```

TWCR=0x00;
// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C
Compiler|Libraries|Alphanumeric LCD
menu:
// RS - PORTB Bit 0
// RD - PORTB Bit 1
// EN - PORTB Bit 2
// D4 - PORTD Bit 4
// D5 - PORTD Bit 5
// D6 - PORTD Bit 6
// D7 - PORTD Bit 7
// Characters/line: 16

lcd_init(16);

lcd_clear ();
lcd_gotoxy (0,0);
lcd_putsf ("CHm=");
lcd_gotoxy(8,0);
lcd_putsf("CHj=");

lcd_gotoxy(0,1);
lcd_putsf("Jarak=");
lcd_gotoxy(14,1);
lcd_putsf("cm");

// Global enable interrupts

#asm("sei")

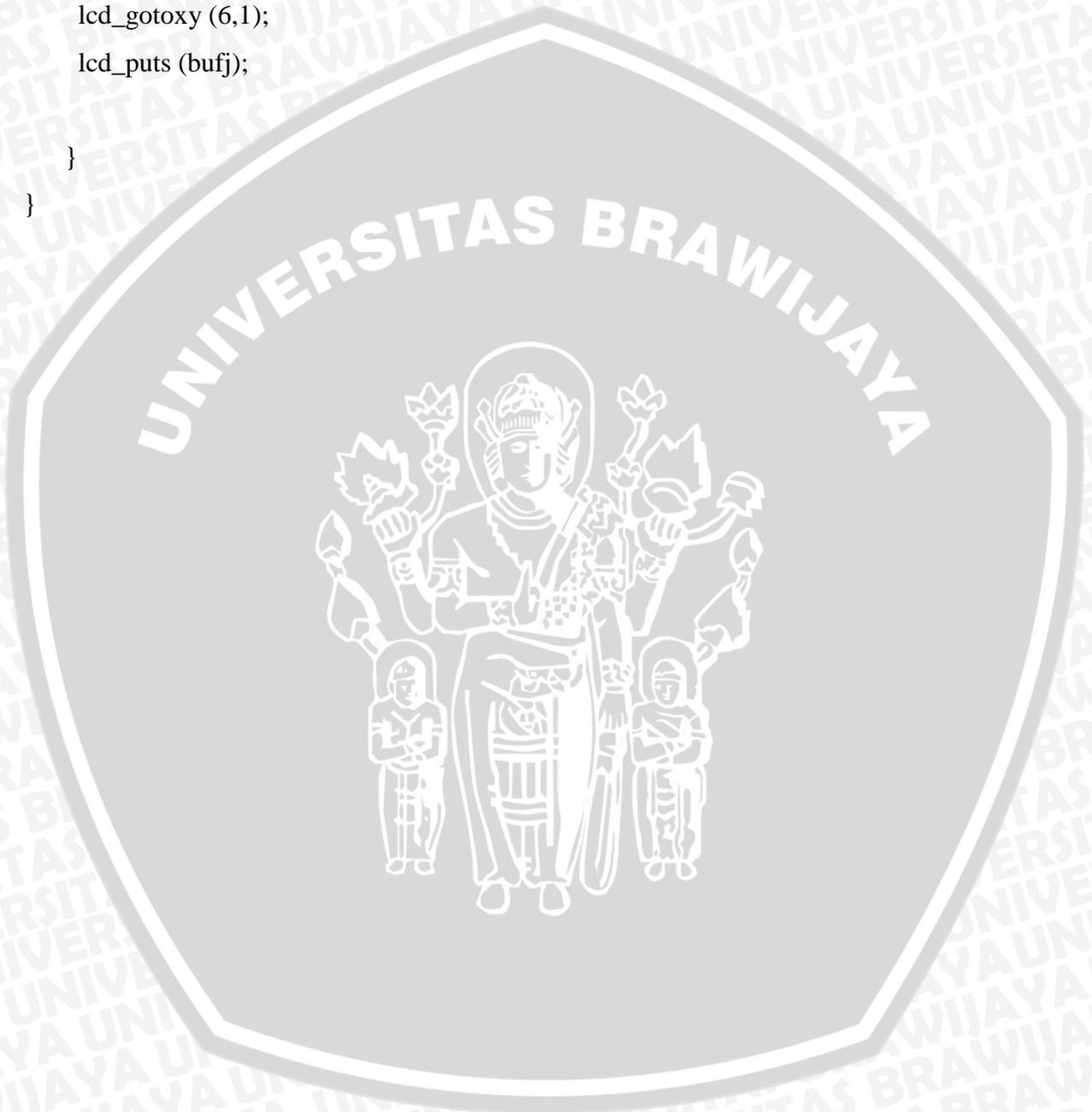
while (1)
{
// Place your code here
tippingbucket();
ultrasonik();

if (ch_j>=50)
{
if (jarak<=75)
{
buz=0xFF;
 kirim_sms();
}
else
{
buz=0x00;
}
}
else
{
if (jarak<=45)
{
buz=0xFF;
 kirim_sms();
}
else
{
buz=0x00;
}
}
}

ftoa(ch_j,1,bufch);
lcd_gotoxy(4,0);
lcd_puts(bufch);

```

```
ftoa(ch_m,1,bufch);  
lcd_gotoxy(12,0);  
lcd_puts(bufch);  
  
sprintf (bufj,"%d ",jarak);  
lcd_gotoxy (6,1);  
lcd_puts (bufj);  
  
}  
}
```

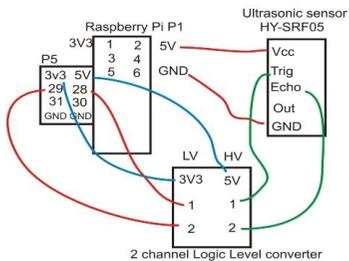
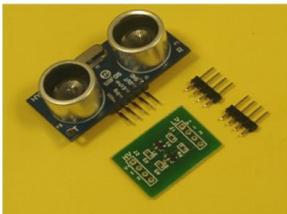


12. LAMPIRAN IV

Datasheet Ultrasonik

Ultrasonic distance sensor (HY-SRF05) with 2channel Logic Level converter

Use two GPIO Trig(T) Echo(E)
 Update --> check distance and update
 T28 --> Trig use GPIO28
 E29 --> Echo use GPIO29



Ultrasonic sensor module Description :

- Working Voltage : 5V(DC)
- Static current: Less than 2mA.
- Output signal: Electric frequency signal, high level 5V, low level 0V.
- Sensor angle: Not more than 15 degrees.

- Detection distance: 2cm-450cm.
 - High precision: Up to 0.2cm
 - Input trigger signal: 10us TTL impulse
 - Echo signal : output TTL PWL signal
-
- Mode of connection:
 - 1.VCC 2.trig(T) 3.echo(R) 4.OUT 5.GND
-
- Use method:
 - Supply module with 5V, the output will be 5V while obstacle in range, or 0V if not. The out pin of this module is used as a switching output when anti-theft module, and without the feet when ranging modules.

2 channel Logic Level Converter Description :

The logic level converter is a small PCB which safely allows you to interface devices which steps down 5V signals to 3.3V signals and step up 3.3V signals to 5V.

The converter has 2, inputs and outputs and can be used with I2C, SPI, UART, etc.

The board needs to be powered via two sources, 5V and 3.3V and your data lines are linked via pins 1 to 2 on each side so an input on pin 1 will be converted to the lower or higher output on the adjoining pin 1

Download GPIO library

<https://pypi.python.org/pypi/RPi.GPIO> GPIO library

GPIO library - RPi.GPIO-0.5.3a.tar.gz

Install python , library and run the test program

```
# sudo apt-get install python-dev
# wget http://www.pridopia.co.uk/pi-pgm/RPi.GPIO-0.5.3a.tar.gz
# gunzip RPi.GPIO-0.5.3a.tar.gz
# tar -xvf RPi.GPIO-0.5.3a.tar
# cd RPi.GPIO-0.5.3a
# sudo python setup.py install
```

Datasheet MAX 232

MAX220–MAX249

+5V-Powered, Multichannel RS-232 Drivers/Receivers

General Description

The MAX220–MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, particularly applications where $\pm 12V$ is not available.

The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

The MAX220–MAX249 are offered in 26 different packages with temperatures from 0 to +70°C up to -55°C to +125°C. See ordering information table at the end of the data sheet for all package and temperature options.

Applications

- Interface Translation
- Multidrop RS-232 Networks
- Portable Diagnostics Equipment

AutoShutdown and UCSP are trademarks of Maxim Integrated Products, Inc.

Benefits and Features

Saves Board Space

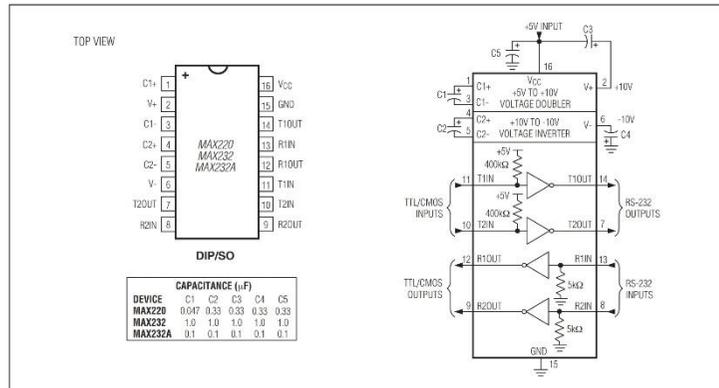
- Integrated Charge Pump Circuitry
 - Eliminates the Need for a Bipolar $\pm 12V$ Supply
 - Enables Single Supply Operation from +5V Supply
- Integrated Capacitors (MAX223, MAX233, MAX235, MAX245–MAX247)

Saves Power for Reduced Power Requirements

- 5 μW Shutdown Mode

Ordering Information and Selection Table appears at end of data sheet.

MAX220/MAX232/MAX232A Pin Configuration and Typical Operating Circuit



19-4323; Rev 17; 1/15



MAX220–MAX249

+5V-Powered, Multichannel RS-232 Drivers/Receivers

Absolute Maximum Ratings—MAX223/MAX230–MAX241

(Voltages referenced to GND.)

V _{CC}-0.3V to +6V
V ₊(V _{CC} - 0.3V) to +14V
V ₋+0.3V to -14V
Input Voltages	
TIN-0.3V to (V _{CC} + 0.3V)
RIN $\pm 30V$
Output Voltages	
TOUT(V ₊ + 0.3V) to (V ₋ - 0.3V)
ROUT-0.3V to (V _{CC} + 0.3V)
Short-Circuit Duration, TOUT to GNDContinuous
Continuous Power Dissipation (T _A = +70°C)	
14-Pin Plastic DIP (derate 10.00mW/°C above +70°C)800mW
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)842mW
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C)889mW
24-Pin Narrow Plastic DIP (derate 13.33mW/°C above +70°C)1.07W
24-Pin Plastic DIP (derate 9.09mW/°C above +70°C)500mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C)762mW
20-Pin Wide SO (derate 10.00mW/°C above +70°C)800mW
24-Pin Wide SO (derate 11.76mW/°C above +70°C)941mW

28-Pin Wide SO (derate 12.50mW/°C above +70°C)1W
44-Pin Plastic FP (derate 11.11mW/°C above +70°C)889mW
14-Pin CERDIP (derate 9.09mW/°C above +70°C)727mW
16-Pin CERDIP (derate 10.00mW/°C above +70°C)800mW
20-Pin CERDIP (derate 11.11mW/°C above +70°C)889mW
24-Pin Narrow CERDIP (derate 12.50mW/°C above +70°C)1W
24-Pin Sidebraze (derate 20.0mW/°C above +70°C)1.6W
28-Pin SSOP (derate 9.52mW/°C above +70°C)762mW
Operating Temperature Ranges	
MAX2__C_0°C to +70°C
MAX2__E_-40°C to +85°C
MAX2__M_-55°C to +125°C
Storage Temperature Range-65°C to +160°C
Lead Temperature (soldering, 10s)+300°C
Soldering Temperature (reflow)	
20 PDIP (P20M+1)+225°C
24 PDIP (P24M-1)+225°C
All other lead(Pb)-free packages+260°C
All other packages containing lead(Pb)+240°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Electrical Characteristics—MAX223/MAX230–MAX241

(MAX223/230/232/234/236/237/238/240/241, V_{CC} = +5V $\pm 10\%$; MAX233/MAX235, V_{CC} = +5V $\pm 5\%$, C1–C4 = 1.0 μF ; MAX231/MAX239, V_{CC} = +5V $\pm 10\%$; V₊ = +7.5V to +13.2V; T_A = T_{MIN} to T_{MAX}; unless otherwise noted.) (Note 5)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Output Voltage Swing	All transmitter outputs loaded with 3k Ω to ground	± 5.0	± 7.3		V
V _{CC} Supply Current	No load, T _A = +25°C		5	10	mA
	MAX223/230/234–238/240/241		7	15	
V ₊ Supply Current			0.4	1	mA
	MAX231/239		1.8	5	
Shutdown Supply Current	T _A = +25°C		15	50	μA
	MAX223		1	10	
Input Logic-Low Voltage	TIN, EN, SHDN (MAX233); EN, SHDN (MAX230/235–241)			0.8	V
Input Logic-High Voltage	TIN	2.0			V
	EN, SHDN (MAX223); EN, SHDN (MAX230/235/236/240/241)	2.4			
Logic Pullup Current	V _{TIN} = 0V		1.5	200	μA
Receiver Input Voltage Operating Range		-30		+30	V

MAX220–MAX249

+5V-Powered, Multichannel RS-232 Drivers/Receivers

Electrical Characteristics—MAX223/MAX230–MAX241 (continued)

(MAX223/230/232/234/236/237/238/240/241, V_{CC} = +5V ±10%; MAX233/MAX235, V_{CC} = +5V ±5%, C₁–C₄ = 1.0μF; MAX231/MAX239, V_{CC} = +5V ±10%; V₊ = +7.5V to +13.2V; T_A = T_{MIN} to T_{MAX}; unless otherwise noted.) (Note 5)

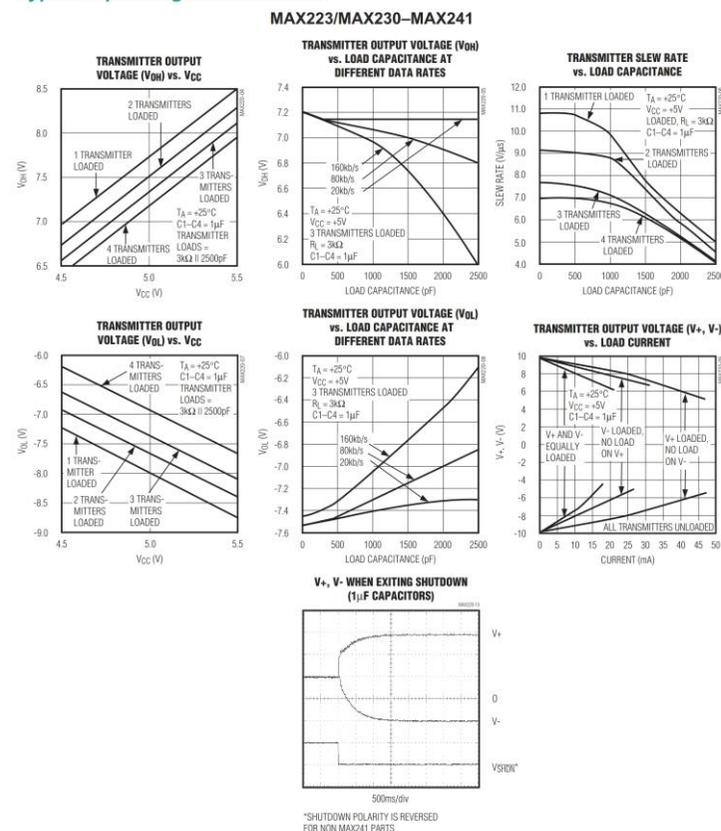
PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 Input Logic-Low Voltage	T _A = +25°C, V _{CC} = +5V	Normal operation V _{SHDN} = +5V (MAX223) V _{SHDN} = 0V (MAX235/236/240/241)	0.8	1.2		V
		Shutdown (MAX223) V _{SHDN} = 0V, V _{EN} = +5V (R4IN, R5IN)	0.6	1.5		
RS-232 Input Logic-High Voltage	T _A = +25°C, V _{CC} = +5V	Normal operation V _{SHDN} = 5V (MAX223) V _{SHDN} = 0V (MAX235/236/240/241)		1.7	2.4	V
		Shutdown (MAX223) V _{SHDN} = 0V, V _{EN} = +5V (R4IN, R5IN)		1.5	2.4	
RS-232 Input Hysteresis	V _{CC} = +5V, no hysteresis in shutdown		0.2	0.5	1.0	V
RS-232 Input Resistance	T _A = +25°C, V _{CC} = +5V		3	5	7	kΩ
TTL/CMOS Output Voltage Low	I _{OUT} = 1.6mA (MAX231/232/233, I _{OUT} = 3.2mA)				0.4	V
TTL/CMOS Output Voltage High	I _{OUT} = -1mA		3.5	V _{CC} - 0.4		V
TTL/CMOS Output Leakage Current	0V ≤ R _{OUT} ≤ V _{CC} ; V _{EN} = 0V (MAX223); V _{EN} = V _{CC} (MAX235–241)			±0.05	±10	μA
Receiver Output Enable Time	Normal operation	MAX223		600		ns
		MAX235/236/239/240/241		400		
Receiver Output Disable Time	Normal operation	MAX223		900		ns
		MAX235/236/239/240/241		250		
Propagation Delay	RS-232 IN to TTL/CMOS OUT, C _L = 150pF	Normal operation V _{SHDN} = 0V (MAX223)	t _{PHLS}	4	40	μs
			t _{PLHS}	6	40	
Transition Region Slew Rate	MAX223/MAX230/MAX234–241, T _A = +25°C, V _{CC} = +5V, R _L = 3kΩ to 7kΩ, C _L = 50pF to 2500pF, measured from +3V to -3V or -3V to +3V		3	5.1	30	V/μs
	MAX231/MAX232/MAX233, T _A = +25°C, V _{CC} = +5V, R _L = 3kΩ to 7kΩ, C _L = 50pF to 2500pF, measured from +3V to -3V or -3V to +3V			4	30	
Transmitter Output Resistance	V _{CC} = V ₊ = V ₋ = 0V, V _{OUT} = ±2V		300			Ω
Transmitter Output Short-Circuit Current			±10			mA

Note 5: All units are production tested at hot except for the MAX240, which is production tested at T_A = +25°C. Specifications over temperature are guaranteed by design.

MAX220–MAX249

+5V-Powered, Multichannel RS-232 Drivers/Receivers

Typical Operating Characteristics



ATmega32(L)

The Atmel®AVR®AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega32 provides the following features: 32Kbytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 1024bytes EEPROM, 2Kbyte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega32 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The Atmel AVR ATmega32 is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Pin Descriptions

VCC Digital supply voltage.

GND Ground.

Port A (PA7..PA0) Port A serves as the analog inputs to the A/D Converter.

Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.



ATmega32(L)

Port B (PB7..PB0) Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATmega32 as listed on [page 57](#).

Port C (PC7..PC0) Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.

The TDO pin is tri-stated unless TAP states that shift out data are entered.

Port C also serves the functions of the JTAG interface and other special features of the ATmega32 as listed on [page 60](#).

Port D (PD7..PD0) Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega32 as listed on [page 62](#).

RESET Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in [Table 15 on page 37](#). Shorter pulses are not guaranteed to generate a reset.

XTAL1 Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2 Output from the inverting Oscillator amplifier.

AVCC AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.

AREF AREF is the analog reference pin for the A/D Converter.





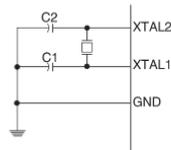
ATmega32(L)

Crystal Oscillator

XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier which can be configured for use as an On-chip Oscillator, as shown in Figure 12. Either a quartz crystal or a ceramic resonator may be used. The CKOPT Fuse selects between two different Oscillator amplifier modes. When CKOPT is programmed, the Oscillator output will oscillate with a full rail-to-rail swing on the output. This mode is suitable when operating in a very noisy environment or when the output from XTAL2 drives a second clock buffer. This mode has a wide frequency range. When CKOPT is unprogrammed, the Oscillator has a smaller output swing. This reduces power consumption considerably. This mode has a limited frequency range and it can not be used to drive other clock buffers.

For resonators, the maximum frequency is 8MHz with CKOPT unprogrammed and 16 MHz with CKOPT programmed. C1 and C2 should always be equal for both crystals and resonators. The optimal value of the capacitors depends on the crystal or resonator in use, the amount of stray capacitance, and the electromagnetic noise of the environment. Some initial guidelines for choosing capacitors for use with crystals are given in Table 4. For ceramic resonators, the capacitor values given by the manufacturer should be used.

Figure 12. Crystal Oscillator Connections



The Oscillator can operate in three different modes, each optimized for a specific frequency range. The operating mode is selected by the fuses CKSEL3..1 as shown in Table 4.

Table 4. Crystal Oscillator Operating Modes

CKOPT	CKSEL3..1	Frequency Range (MHz)	Recommended Range for Capacitors C1 and C2 for Use with Crystals (pF)
1	101 ⁽¹⁾	0.4 - 0.9	—
1	110	0.9 - 3.0	12 - 22
1	111	3.0 - 8.0	12 - 22
0	101, 110, 111	1.0 ≤	12 - 22

Note: 1. This option should not be used with crystals, only with ceramic resonators.

ATmega32(L)

Power Management and Sleep Modes

Sleep modes enable the application to shut down unused modules in the MCU, thereby saving power. The AVR provides various sleep modes allowing the user to tailor the power consumption to the application's requirements.

To enter any of the six sleep modes, the SE bit in MCUCR must be written to logic one and a SLEEP instruction must be executed. The SM2, SM1, and SM0 bits in the MCUCR Register select which sleep mode (Idle, ADC Noise Reduction, Power-down, Power-save, Standby, or Extended Standby) will be activated by the SLEEP instruction. See Table 13 for a summary. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU wakes up. The MCU is then halted for four cycles in addition to the start-up time, it executes the interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the Register File and SRAM are unaltered when the device wakes up from sleep. If a Reset occurs during sleep mode, the MCU wakes up and executes from the Reset Vector.

Figure 11 on page 24 presents the different clock systems in the ATmega32, and their distribution. The figure is helpful in selecting an appropriate sleep mode.

MCU Control Register – MCUCR

The MCU Control Register contains control bits for power management.

Bit	7	6	5	4	3	2	1	0	MCUCR
Read/Write	R/W								
Initial Value	0	0	0	0	0	0	0	0	

• **Bit 7 – SE: Sleep Enable**

The SE bit must be written to logic one to make the MCU enter the sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the sleep mode unless it is the programmers purpose, it is recommended to write the Sleep Enable (SE) bit to one just before the execution of the SLEEP instruction and to clear it immediately after waking up.

• **Bits [6:4] – SM2..0: Sleep Mode Select Bits 2, 1, and 0**

These bits select between the six available sleep modes as shown in Table 13.

Table 13. Sleep Mode Select

SM2	SM1	SM0	Sleep Mode
0	0	0	Idle
0	0	1	ADC Noise Reduction
0	1	0	Power-down
0	1	1	Power-save
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Standby ⁽¹⁾
1	1	1	Extended Standby ⁽¹⁾

Note: 1. Standby mode and Extended Standby mode are only available with external crystals or resonators.





ATmega32(L)

Idle Mode

When the SM2..0 bits are written to 000, the SLEEP instruction makes the MCU enter Idle mode, stopping the CPU but allowing SPI, USART, Analog Comparator, ADC, Two-wire Serial Interface, Timer/Counters, Watchdog, and the interrupt system to continue operating. This sleep mode basically halts clk_{CPU} and clk_{FLASH} while allowing the other clocks to run.

Idle mode enables the MCU to wake up from external triggered interrupts as well as internal ones like the Timer Overflow and USART Transmit Complete interrupts. If wake-up from the Analog Comparator interrupt is not required, the Analog Comparator can be powered down by setting the ACD bit in the Analog Comparator Control and Status Register – ACSR. This will reduce power consumption in Idle mode. If the ADC is enabled, a conversion starts automatically when this mode is entered.

ADC Noise Reduction Mode

When the SM2..0 bits are written to 001, the SLEEP instruction makes the MCU enter ADC Noise Reduction mode, stopping the CPU but allowing the ADC, the External Interrupts, the Two-wire Serial Interface address watch, Timer/Counter2 and the Watchdog to continue operating (if enabled). This sleep mode basically halts $clk_{I/O}$, clk_{CPU} , and clk_{FLASH} while allowing the other clocks to run.

This improves the noise environment for the ADC, enabling higher resolution measurements. If the ADC is enabled, a conversion starts automatically when this mode is entered. Apart from the ADC Conversion Complete interrupt, only an External Reset, a Watchdog Reset, a Brown-out Reset, a Two-wire Serial Interface Address Match Interrupt, a Timer/Counter2 interrupt, an SPM/EEPROM ready interrupt, an External level interrupt on INT0 or INT1, or an external interrupt on INT2 can wake up the MCU from ADC Noise Reduction mode.

Power-down Mode

When the SM2..0 bits are written to 010, the SLEEP instruction makes the MCU enter Power-down mode. In this mode, the External Oscillator is stopped, while the External interrupts, the Two-wire Serial Interface address watch, and the Watchdog continue operating (if enabled). Only an External Reset, a Watchdog Reset, a Brown-out Reset, a Two-wire Serial Interface address match interrupt, an External level interrupt on INT0 or INT1, or an External interrupt on INT2 can wake up the MCU. This sleep mode basically halts all generated clocks, allowing operation of asynchronous modules only.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. Refer to "External Interrupts" on page 66 for details.

When waking up from Power-down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same CKSEL fuses that define the reset time-out period, as described in "Clock Sources" on page 25.

Power-save Mode

When the SM2..0 bits are written to 011, the SLEEP instruction makes the MCU enter Power-save mode. This mode is identical to Power-down, with one exception:

If Timer/Counter2 is clocked asynchronously, that is, the AS2 bit in ASSR is set, Timer/Counter2 will run during sleep. The device can wake up from either Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK, and the Global Interrupt Enable bit in SREG is set.

If the Asynchronous Timer is NOT clocked asynchronously, Power-down mode is recommended instead of Power-save mode because the contents of the registers in the Asynchronous Timer should be considered undefined after wake-up in Power-save mode if AS2 is 0.

This sleep mode basically halts all clocks except clk_{ASY} , allowing operation only of asynchronous modules, including Timer/Counter2 if clocked asynchronously.

ATmega32(L)

Standby Mode

When the SM2..0 bits are 110 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Standby mode. This mode is identical to Power-down with the exception that the Oscillator is kept running. From Standby mode, the device wakes up in six clock cycles.

Extended Standby Mode

When the SM2..0 bits are 111 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Extended Standby mode. This mode is identical to Power-save mode with the exception that the Oscillator is kept running. From Extended Standby mode, the device wakes up in six clock cycles..

Table 14. Active Clock Domains and Wake Up Sources in the Different Sleep Modes

Sleep Mode	Active Clock domains					Oscillators		Wake-up Sources							
	clk_{CPU}	clk_{FLASH}	$clk_{I/O}$	clk_{BOD}	clk_{ASY}	Main Clock Source Enabled	Timer Oscillator Enabled	INT2	INT1	INT0	TWI Address Match	Timer 2	SPM / EEPROM Ready	ADC	Other I/O
Idle		X	X	X	X	X	X ⁽²⁾	X	X	X	X	X	X	X	X
ADC Noise Reduction			X	X	X	X	X ⁽²⁾	X ⁽³⁾	X	X	X	X	X		
Power-down								X ⁽³⁾	X						
Power-save					X ⁽²⁾		X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾					
Standby ⁽¹⁾						X		X ⁽³⁾	X						
Extended Standby ⁽¹⁾					X ⁽²⁾	X	X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾					

Notes: 1. External Crystal or resonator selected as clock source.
 2. If AS2 bit in ASSR is set.
 3. Only INT2 or level interrupt INT1 and INT0.

Minimizing Power Consumption

There are several issues to consider when trying to minimize the power consumption in an AVR controlled system. In general, sleep modes should be used as much as possible, and the sleep mode should be selected so that as few as possible of the device's functions are operating. All functions not needed should be disabled. In particular, the following modules may need special consideration when trying to achieve the lowest possible power consumption.

Analog to Digital Converter

If enabled, the ADC will be enabled in all sleep modes. To save power, the ADC should be disabled before entering any sleep mode. When the ADC is turned off and on again, the next conversion will be an extended conversion. Refer to "Analog to Digital Converter" on page 201 for details on ADC operation.



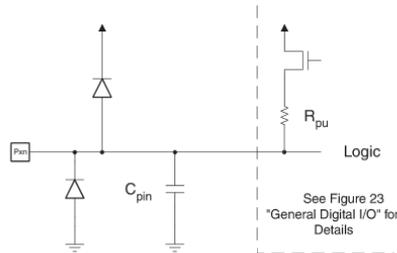
ATmega32(L)

I/O Ports

Introduction

All AVR ports have true Read-Modify-Write functionality when used as general digital I/O ports. This means that the direction of one port pin can be changed without unintentionally changing the direction of any other pin with the SBI and CBI instructions. The same applies when changing drive value (if configured as output) or enabling/disabling of pull-up resistors (if configured as input). Each output buffer has symmetrical drive characteristics with both high sink and source capability. The pin driver is strong enough to drive LED displays directly. All port pins have individually selectable pull-up resistors with a supply-voltage invariant resistance. All I/O pins have protection diodes to both V_{CC} and Ground as indicated in Figure 22. Refer to "Electrical Characteristics" on page 287 for a complete list of parameters.

Figure 22. I/O Pin Equivalent Schematic



All registers and bit references in this section are written in general form. A lower case "x" represents the numbering letter for the port, and a lower case "n" represents the bit number. However, when using the register or bit defines in a program, the precise form must be used, that is, PORTB3 for bit no. 3 in Port B, here documented generally as PORTxn. The physical I/O Registers and bit locations are listed in "Register Description for I/O Ports" on page 64.

Three I/O memory address locations are allocated for each port, one each for the Data Register – PORTx, Data Direction Register – DDRx, and the Port Input Pins – PINx. The Port Input Pins I/O location is read only, while the Data Register and the Data Direction Register are read/write. In addition, the Pull-up Disable – PUD bit in SFIOR disables the pull-up function for all pins in all ports when set.

Using the I/O port as General Digital I/O is described in "Ports as General Digital I/O" on page 50. Most port pins are multiplexed with alternate functions for the peripheral features on the device. How each alternate function interferes with the port pin is described in "Alternate Port Functions" on page 54. Refer to the individual module sections for a full description of the alternate functions.

Note that enabling the alternate function of some of the port pins does not affect the use of the other pins in the port as general digital I/O.

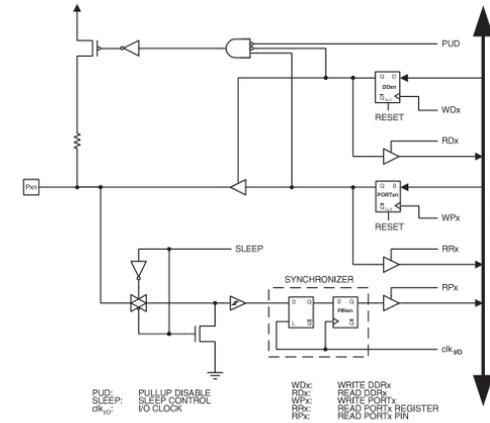


ATmega32(L)

Ports as General Digital I/O

The ports are bi-directional I/O ports with optional internal pull-ups. Figure 23 shows a functional description of one I/O-port pin, here generically called Pxn.

Figure 23. General Digital I/O⁽¹⁾



Note: 1. WPx, WDX, RFX, RPx, and RDx are common to all pins within the same port. clk_{IO}, SLEEP, and PUD are common to all ports.

Configuring the Pin

Each port pin consists of three register bits: DDRx, PORTxn, and PINxn. As shown in "Register Description for I/O Ports" on page 64, the DDRx bits are accessed at the DDRx I/O address, the PORTxn bits at the PORTx I/O address, and the PINxn bits at the PINx I/O address.

The DDxn bit in the DDRx Register selects the direction of this pin. If DDxn is written logic one, Pxn is configured as an output pin. If DDxn is written logic zero, Pxn is configured as an input pin.

If PORTxn is written logic one when the pin is configured as an input pin, the pull-up resistor is activated. To switch the pull-up resistor off, PORTxn has to be written logic zero or the pin has to be configured as an output pin. The port pins are tri-stated when a reset condition becomes active, even if no clocks are running.

If PORTxn is written logic one when the pin is configured as an output pin, the port pin is driven high (one). If PORTxn is written logic zero when the pin is configured as an output pin, the port pin is driven low (zero).

When switching between tri-state ((DDxn, PORTxn) = 0b00) and output high ((DDxn, PORTxn) = 0b11), an intermediate state with either pull-up enabled ((DDxn, PORTxn) = 0b01) or output low ((DDxn, PORTxn) = 0b10) must occur. Normally, the pull-up enabled state is fully acceptable, as a high-impedant environment will not notice the difference between a strong high driver





ATmega32(L)

and a pull-up. If this is not the case, the PUD bit in the SFIOR Register can be set to disable all pull-ups in all ports.

Switching between input with pull-up and output low generates the same problem. The user must use either the tri-state ((DDxn, PORTxn) = 0b00) or the output high state ((DDxn, PORTxn) = 0b11) as an intermediate step.

Table 20 summarizes the control signals for the pin value.

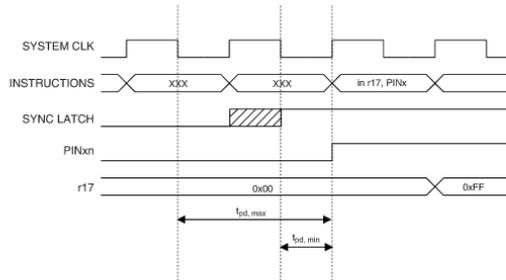
Table 20. Port Pin Configurations

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Reading the Pin Value

Independent of the setting of Data Direction bit DDxn, the port pin can be read through the PINxn Register bit. As shown in Figure 23, the PINxn Register bit and the preceding latch constitute a synchronizer. This is needed to avoid metastability if the physical pin changes value near the edge of the internal clock, but it also introduces a delay. Figure 24 shows a timing diagram of the synchronization when reading an externally applied pin value. The maximum and minimum propagation delays are denoted $t_{pd,max}$ and $t_{pd,min}$ respectively.

Figure 24. Synchronization when Reading an Externally Applied Pin Value

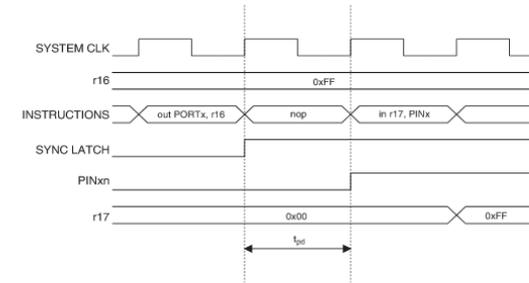


Consider the clock period starting shortly after the first falling edge of the system clock. The latch is closed when the clock is low, and goes transparent when the clock is high, as indicated by the shaded region of the "SYNC LATCH" signal. The signal value is latched when the system clock goes low. It is clocked into the PINxn Register at the succeeding positive clock edge. As indicated by the two arrows $t_{pd,max}$ and $t_{pd,min}$, a single signal transition on the pin will be delayed between 1/2 and 1 1/2 system clock period depending upon the time of assertion.

ATmega32(L)

When reading back a software assigned pin value, a *nop* instruction must be inserted as indicated in Figure 25. The *out* instruction sets the "SYNC LATCH" signal at the positive edge of the clock. In this case, the delay t_{pd} through the synchronizer is one system clock period.

Figure 25. Synchronization when Reading a Software Assigned Pin Value





ATmega32(L)

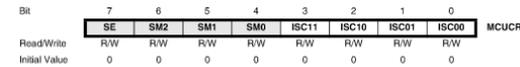
External Interrupts

The External Interrupts are triggered by the INTO, INT1, and INT2 pins. Observe that, if enabled, the interrupts will trigger even if the INTO..2 pins are configured as outputs. This feature provides a way of generating a software interrupt. The external interrupts can be triggered by a falling or rising edge or a low level (INT2 is only an edge triggered interrupt). This is set up as indicated in the specification for the MCU Control Register – MCUCR – and MCU Control and Status Register – MCUCSR. When the external interrupt is enabled and is configured as level triggered (only INTO/INT1), the interrupt will trigger as long as the pin is held low. Note that recognition of falling or rising edge interrupts on INTO and INT1 requires the presence of an I/O clock, described in "Clock Systems and their Distribution" on page 24. Low level interrupts on INTO/INT1 and the edge interrupt on INT2 are detected asynchronously. This implies that these interrupts can be used for waking the part also from sleep modes other than Idle mode. The I/O clock is halted in all sleep modes except Idle mode.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. This makes the MCU less sensitive to noise. The changed level is sampled twice by the Watchdog Oscillator clock. The period of the Watchdog Oscillator is 1 μs (nominal) at 5.0V and 25°C. The frequency of the Watchdog Oscillator is voltage dependent as shown in "Electrical Characteristics" on page 287. The MCU will wake up if the input has the required level during this sampling or if it is held until the end of the start-up time. The start-up time is defined by the SUT fuses as described in "System Clock and Clock Options" on page 24. If the level is sampled twice by the Watchdog Oscillator clock but disappears before the end of the start-up time, the MCU will still wake up, but no interrupt will be generated. The required level must be held long enough for the MCU to complete the wake up to trigger the level interrupt.

MCU Control Register – MCUCR

The MCU Control Register contains control bits for interrupt sense control and general MCU functions.



Bit 3, 2 – ISC11, ISC10: Interrupt Sense Control 1 Bit 1 and Bit 0

The External Interrupt 1 is activated by the external pin INT1 if the SREG I-bit and the corresponding interrupt mask in the GICR are set. The level and edges on the external INT1 pin that activate the interrupt are defined in Table 34. The value on the INT1 pin is sampled before detecting edges. If edge or toggle interrupt is selected, pulses that last longer than one clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt.

Table 34. Interrupt 1 Sense Control

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Any logical change on INT1 generates an interrupt request.
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.



ATmega32(L)

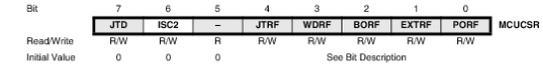
Bit 1, 0 – ISC01, ISC00: Interrupt Sense Control 0 Bit 1 and Bit 0

The External Interrupt 0 is activated by the external pin INTO if the SREG I-flag and the corresponding interrupt mask are set. The level and edges on the external INTO pin that activate the interrupt are defined in Table 35. The value on the INTO pin is sampled before detecting edges. If edge or toggle interrupt is selected, pulses that last longer than one clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt.

Table 35. Interrupt 0 Sense Control

ISC01	ISC00	Description
0	0	The low level of INTO generates an interrupt request.
0	1	Any logical change on INTO generates an interrupt request.
1	0	The falling edge of INTO generates an interrupt request.
1	1	The rising edge of INTO generates an interrupt request.

MCU Control and Status Register – MCUCSR



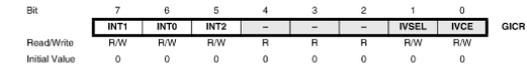
Bit 6 – ISC2: Interrupt Sense Control 2

The Asynchronous External Interrupt 2 is activated by the external pin INT2 if the SREG I-bit and the corresponding interrupt mask in GICR are set. If ISC2 is written to zero, a falling edge on INT2 activates the interrupt. If ISC2 is written to one, a rising edge on INT2 activates the interrupt. Edges on INT2 are registered asynchronously. Pulses on INT2 wider than the minimum pulse width given in Table 36 will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. When changing the ISC2 bit, an interrupt can occur. Therefore, it is recommended to first disable INT2 by clearing its Interrupt Enable bit in the GICR Register. Then, the ISC2 bit can be changed. Finally, the INT2 Interrupt Flag should be cleared by writing a logical one to its Interrupt Flag bit (INTF2) in the GIFR Register before the interrupt is re-enabled.

Table 36. Asynchronous External Interrupt Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
^t _{INT}	Minimum pulse width for asynchronous external interrupt			50		ns

General Interrupt Control Register – GICR



Bit 7 – INT1: External Interrupt Request 1 Enable

When the INT1 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control1 bits 1/0 (ISC11 and ISC10) in the MCU General Control Register (MCUCR) define whether the External Interrupt is activated on rising



ATmega32(L)

and/or falling edge of the INT1 pin or level sensed. Activity on the pin will cause an interrupt request even if INT1 is configured as an output. The corresponding interrupt of External Interrupt Request 1 is executed from the INT1 interrupt Vector.

• Bit 6 – INT0: External Interrupt Request 0 Enable

When the INT0 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control0 bits (ISC01 and ISC00) in the MCU General Control Register (MCUCR) define whether the External Interrupt is activated on rising and/or falling edge of the INT0 pin or level sensed. Activity on the pin will cause an interrupt request even if INT0 is configured as an output. The corresponding interrupt of External Interrupt Request 0 is executed from the INT0 interrupt vector.

• Bit 5 – INT2: External Interrupt Request 2 Enable

When the INT2 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control2 bit (ISC2) in the MCU Control and Status Register (MCUCSR) defines whether the External Interrupt is activated on rising or falling edge of the INT2 pin. Activity on the pin will cause an interrupt request even if INT2 is configured as an output. The corresponding interrupt of External Interrupt Request 2 is executed from the INT2 Interrupt Vector.

General Interrupt Flag Register – GIFR

Bit	7	6	5	4	3	2	1	0	
	INTF1	INTF0	INTF2	–	–	–	–	–	GIFR
Read/Write	R/W	R/W	R/W	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

• Bit 7 – INTF1: External Interrupt Flag 1

When an edge or logic change on the INT1 pin triggers an interrupt request, INTF1 becomes set (one). If the I-bit in SREG and the INT1 bit in GICR are set (one), the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical one to it. This flag is always cleared when INT1 is configured as a level interrupt.

• Bit 6 – INTF0: External Interrupt Flag 0

When an edge or logic change on the INT0 pin triggers an interrupt request, INTF0 becomes set (one). If the I-bit in SREG and the INT0 bit in GICR are set (one), the MCU will jump to the corresponding interrupt vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical one to it. This flag is always cleared when INT0 is configured as a level interrupt.

• Bit 5 – INTF2: External Interrupt Flag 2

When an event on the INT2 pin triggers an interrupt request, INTF2 becomes set (one). If the I-bit in SREG and the INT2 bit in GICR are set (one), the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical one to it. Note that when entering some sleep modes with the INT2 interrupt disabled, the input buffer on this pin will be disabled. This may cause a logic change in internal signals which will set the INTF2 Flag. See "Digital Input Enable and Sleep Modes" on page 53 for more information.



ATmega32(L)

8-bit Timer/Counter2 with PWM and Asynchronous Operation

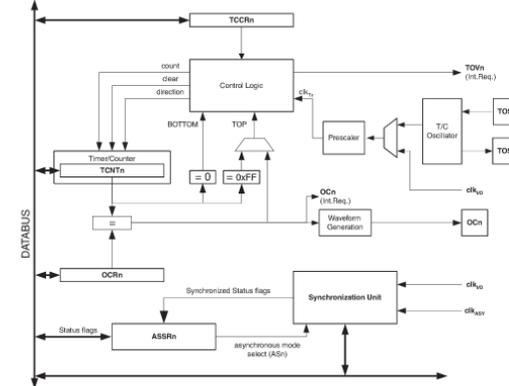
Timer/Counter2 is a general purpose, single compare unit, 8-bit Timer/Counter module. The main features are:

- Single Compare unit Counter
- Clear Timer on Compare Match (Auto Reload)
- Glitch-free, Phase Correct Pulse Width Modulator (PWM)
- Frequency Generator
- 10-bit Clock Prescaler
- Overflow and Compare Match Interrupt Sources (TOV2 and OCF2)
- Allows clocking from External 32kHz Watch Crystal Independent of the I/O Clock

Overview

A simplified block diagram of the 8-bit Timer/Counter is shown in Figure 53. For the actual placement of I/O pins, refer to "Pinout ATmega32" on page 2. CPU accessible I/O Registers, including I/O bits and I/O pins, are shown in bold. The device-specific I/O Register and bit locations are listed in the "8-bit Timer/Counter Register Description" on page 125.

Figure 53. 8-bit Timer/Counter Block Diagram



Registers

The Timer/Counter (TCNT2) and Output Compare Register (OCR2) are 8-bit registers. Interrupt request (shortened as Int.Req.) signals are all visible in the Timer Interrupt Flag Register (TIFR). All interrupts are individually masked with the Timer Interrupt Mask Register (TIMSK). TIFR and TIMSK are not shown in the figure since these registers are shared by other timer units.

The Timer/Counter can be clocked internally, via the prescaler, or asynchronously clocked from the TOSC1/2 pins, as detailed later in this section. The asynchronous operation is controlled by the Asynchronous Status Register (ASRR). The Clock Select logic block controls which clock source the Timer/Counter uses to increment (or decrement) its value. The Timer/Counter is inactive when no clock source is selected. The output from the Clock Select logic is referred to as the timer clock (clk_{T2}).





ATmega32(L)

The double buffered Output Compare Register (OCR2) is compared with the Timer/Counter value at all times. The result of the compare can be used by the waveform generator to generate a PWM or variable frequency output on the Output Compare Pin (OC2). See "Output Compare Unit" on page 116. For details. The compare match event will also set the Compare Flag (OCF2) which can be used to generate an output compare interrupt request.

Definitions

Many register and bit references in this document are written in general form. A lower case "n" replaces the Timer/Counter number, in this case 2. However, when using the register or bit defines in a program, the precise form must be used (that is, TCNT2 for accessing Timer/Counter2 counter value and so on). The definitions in Table 49 are also used extensively throughout the document.

Table 49. Definitions

BOTTOM	The counter reaches the BOTTOM when it becomes zero (0x00).
MAX	The counter reaches its MAXimum when it becomes 0xFF (decimal 255).
TOP	The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR2 Register. The assignment is dependent on the mode of operation.

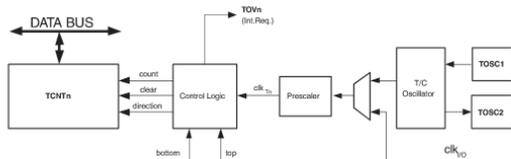
Timer/Counter Clock Sources

The Timer/Counter can be clocked by an internal synchronous or an external asynchronous clock source. The clock source clk_{T2} is by default equal to the MCU clock, clk_{IO} . When the AS2 bit in the ASSR Register is written to logic one, the clock source is taken from the Timer/Counter Oscillator connected to TOSC1 and TOSC2. For details on asynchronous operation, see "Asynchronous Status Register – ASSR" on page 128. For details on clock sources and prescaler, see "Timer/Counter Prescaler" on page 131.

Counter Unit

The main part of the 8-bit Timer/Counter is the programmable bi-directional counter unit. Figure 54 shows a block diagram of the counter and its surrounding environment.

Figure 54. Counter Unit Block Diagram



Signal description (internal signals):

- count** Increment or decrement TCNT2 by 1.
- direction** Selects between increment and decrement.
- clear** Clear TCNT2 (set all bits to zero).
- clk_{T2}** Timer/Counter clock.
- top** Signals that TCNT2 has reached maximum value.



ATmega32(L)

8-bit Timer/Counter Register Description

Timer/Counter Control Register – TCCR2

Bit	7	6	5	4	3	2	1	0	
	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	TCCR2
Read/Write	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	

• Bit 7 – FOC2: Force Output Compare

The FOC2 bit is only active when the WGM bits specify a non-PWM mode. However, for ensuring compatibility with future devices, this bit must be set to zero when TCCR2 is written when operating in PWM mode. When writing a logical one to the FOC2 bit, an immediate compare match is forced on the waveform generation unit. The OC2 output is changed according to its COM21:0 bits setting. Note that the FOC2 bit is implemented as a strobe. Therefore it is the value present in the COM21:0 bits that determines the effect of the forced compare.

A FOC2 strobe will not generate any interrupt, nor will it clear the timer in CTC mode using OCR2 as TOP.

The FOC2 bit is always read as zero.

• Bit 6, 3 – WGM21:0: Waveform Generation Mode

These bits control the counting sequence of the counter, the source for the maximum (TOP) counter value, and what type of waveform generation to be used. Modes of operation supported by the Timer/Counter unit are: Normal mode, Clear Timer on Compare match (CTC) mode, and two types of Pulse Width Modulation (PWM) modes. See Table 50 and "Modes of Operation" on page 118.

Table 50. Waveform Generation Mode Bit Description⁽¹⁾

Mode	WGM21 (CTC2)	WGM20 (PWM2)	Timer/Counter Mode of Operation	TOP	Update of OCR2	TOV2 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR2	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

Note: 1. The CTC2 and PWM2 bit definition names are now obsolete. Use the WGM21:0 definitions. However, the functionality and location of these bits are compatible with previous versions of the timer.

• Bit 5:4 – COM21:0: Compare Match Output Mode

These bits control the Output Compare pin (OC2) behavior. If one or both of the COM21:0 bits are set, the OC2 output overrides the normal port functionality of the I/O pin it is connected to. However, note that the Data Direction Register (DDR) bit corresponding to OC2 pin must be set in order to enable the output driver.





ATmega32(L)

When OC2 is connected to the pin, the function of the COM21:0 bits depends on the WGM21:0 bit setting. Table 51 shows the COM21:0 bit functionality when the WGM21:0 bits are set to normal or CTC mode (non-PWM).

Table 51. Compare Output Mode, non-PWM Mode

COM21	COM20	Description
0	0	Normal port operation, OC2 disconnected.
0	1	Toggle OC2 on compare match
1	0	Clear OC2 on compare match
1	1	Set OC2 on compare match

Table 52 shows the COM21:0 bit functionality when the WGM21:0 bits are set to fast PWM mode.

Table 52. Compare Output Mode, Fast PWM Mode⁽¹⁾

COM21	COM20	Description
0	0	Normal port operation, OC2 disconnected.
0	1	Reserved
1	0	Clear OC2 on compare match, set OC2 at BOTTOM, (non-inverting mode)
1	1	Set OC2 on compare match, clear OC2 at BOTTOM, (inverting mode)

Note: 1. A special case occurs when OCR2 equals TOP and COM21 is set. In this case, the compare match is ignored, but the set or clear is done at TOP. See "Fast PWM Mode" on page 120 for more details.

Table 53 shows the COM21:0 bit functionality when the WGM21:0 bits are set to phase correct PWM mode

Table 53. Compare Output Mode, Phase Correct PWM Mode⁽¹⁾

COM21	COM20	Description
0	0	Normal port operation, OC2 disconnected.
0	1	Reserved
1	0	Clear OC2 on compare match when up-counting. Set OC2 on compare match when downcounting.
1	1	Set OC2 on compare match when up-counting. Clear OC2 on compare match when downcounting.

Note: 1. A special case occurs when OCR2 equals TOP and COM21 is set. In this case, the compare match is ignored, but the set or clear is done at TOP. See "Phase Correct PWM Mode" on page 121 for more details.



ATmega32(L)

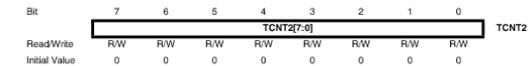
• Bit 2:0 – CS22:0: Clock Select

The three Clock Select bits select the clock source to be used by the Timer/Counter, see Table 54.

Table 54. Clock Select Bit Description

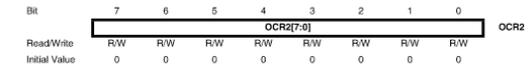
CS22	CS21	CS20	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{T20} /(No prescaler)
0	1	0	clk _{T20} /8 (From prescaler)
0	1	1	clk _{T20} /32 (From prescaler)
1	0	0	clk _{T20} /64 (From prescaler)
1	0	1	clk _{T20} /128 (From prescaler)
1	1	0	clk _{T20} /256 (From prescaler)
1	1	1	clk _{T20} /1024 (From prescaler)

Timer/Counter Register – TCNT2



The Timer/Counter Register gives direct access, both for read and write operations, to the Timer/Counter unit 8-bit counter. Writing to the TCNT2 Register blocks (removes) the compare match on the following timer clock. Modifying the counter (TCNT2) while the counter is running, introduces a risk of missing a compare match between TCNT2 and the OCR2 Register.

Output Compare Register – OCR2



The Output Compare Register contains an 8-bit value that is continuously compared with the counter value (TCNT2). A match can be used to generate an output compare interrupt, or to generate a waveform output on the OC2 pin.



ATmega32(L)

Asynchronous Operation of the Timer/Counter

Asynchronous Status Register – ASSR

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R/W	R	R	R
Initial Value	0	0	0	0	0	0	0	0

• Bit 3 – AS2: Asynchronous Timer/Counter2

When AS2 is written to zero, Timer/Counter 2 is clocked from the I/O clock, clk_{IO} . When AS2 is written to one, Timer/Counter2 is clocked from a Crystal Oscillator connected to the Timer Oscillator 1 (TOSC1) pin. When the value of AS2 is changed, the contents of TCNT2, OCR2, and TCCR2 might be corrupted.

• Bit 2 – TCN2UB: Timer/Counter2 Update Busy

When Timer/Counter2 operates asynchronously and TCNT2 is written, this bit becomes set. When TCNT2 has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that TCNT2 is ready to be updated with a new value.

• Bit 1 – OCR2UB: Output Compare Register2 Update Busy

When Timer/Counter2 operates asynchronously and OCR2 is written, this bit becomes set. When OCR2 has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that OCR2 is ready to be updated with a new value.

• Bit 0 – TCR2UB: Timer/Counter Control Register2 Update Busy

When Timer/Counter2 operates asynchronously and TCCR2 is written, this bit becomes set. When TCCR2 has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that TCCR2 is ready to be updated with a new value.

If a write is performed to any of the three Timer/Counter2 Registers while its update busy flag is set, the updated value might get corrupted and cause an unintentional interrupt to occur.

The mechanisms for reading TCNT2, OCR2, and TCCR2 are different. When reading TCNT2, the actual timer value is read. When reading OCR2 or TCCR2, the value in the temporary storage register is read.

Asynchronous Operation of Timer/Counter2

When Timer/Counter2 operates asynchronously, some considerations must be taken.

- Warning: When switching between asynchronous and synchronous clocking of Timer/Counter2, the Timer Registers TCNT2, OCR2, and TCCR2 might be corrupted. A safe procedure for switching clock source is:
 1. Disable the Timer/Counter2 interrupts by clearing OCIE2 and TOIE2.
 2. Select clock source by setting AS2 as appropriate.
 3. Write new values to TCNT2, OCR2, and TCCR2.
 4. To switch to asynchronous operation: Wait for TCN2UB, OCR2UB, and TCR2UB.
 5. Clear the Timer/Counter2 Interrupt Flags.
 6. Enable interrupts, if needed.



ATmega32(L)

1. Write any value to either of the registers OCR2 or TCCR2.
2. Wait for the corresponding Update Busy Flag to be cleared.
3. Read TCNT2.

- During asynchronous operation, the synchronization of the Interrupt Flags for the asynchronous timer takes three processor cycles plus one timer cycle. The timer is therefore advanced by at least one before the processor can read the timer value causing the setting of the Interrupt Flag. The output compare pin is changed on the timer clock and is not synchronized to the processor clock.

Timer/Counter Interrupt Mask Register – TIMSK

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W							
Initial Value	0	0	0	0	0	0	0	0

• Bit 7 – OCIE2: Timer/Counter2 Output Compare Match Interrupt Enable

When the OCIE2 bit is written to one and the I-bit in the Status Register is set (one), the Timer/Counter2 Compare Match interrupt is enabled. The corresponding interrupt is executed if a compare match in Timer/Counter2 occurs, that is, when the OCF2 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

• Bit 6 – TOIE2: Timer/Counter2 Overflow Interrupt Enable

When the TOIE2 bit is written to one and the I-bit in the Status Register is set (one), the Timer/Counter2 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter2 occurs, that is, when the TOV2 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

Timer/Counter Interrupt Flag Register – TIFR

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W							
Initial Value	0	0	0	0	0	0	0	0

• Bit 7 – OCF2: Output Compare Flag 2

The OCF2 bit is set (one) when a compare match occurs between the Timer/Counter2 and the data in OCR2 – Output Compare Register2. OCF2 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, OCF2 is cleared by writing a logic one to the flag. When the I-bit in SREG, OCIE2 (Timer/Counter2 Compare match Interrupt Enable), and OCF2 are set (one), the Timer/Counter2 Compare match Interrupt is executed.

• Bit 6 – TOV2: Timer/Counter2 Overflow Flag

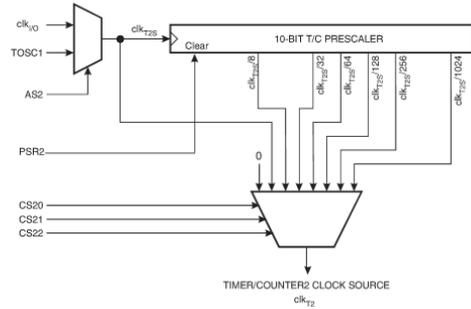
The TOV2 bit is set (one) when an overflow occurs in Timer/Counter2. TOV2 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV2 is cleared by writing a logic one to the flag. When the SREG I-bit, TOIE2 (Timer/Counter2 Overflow Interrupt Enable), and TOV2 are set (one), the Timer/Counter2 Overflow interrupt is executed. In PWM mode, this bit is set when Timer/Counter2 changes counting direction at \$00.



ATmega32(L)

Timer/Counter Prescaler

Figure 64. Prescaler for Timer/Counter2



The clock source for Timer/Counter2 is named clk_{T2S} . clk_{T2S} is by default connected to the main system I/O clock $clk_{I/O}$. By setting the AS2 bit in ASSR, Timer/Counter2 is asynchronously clocked from the TOSC1 pin. This enables use of Timer/Counter2 as a Real Time Counter (RTC). When AS2 is set, pins TOSC1 and TOSC2 are disconnected from Port C. A crystal can then be connected between the TOSC1 and TOSC2 pins to serve as an independent clock source for Timer/Counter2. The Oscillator is optimized for use with a 32.768kHz crystal. Applying an external clock source to TOSC1 is not recommended.

For Timer/Counter2, the possible prescaled selections are: $clk_{T2S}/8$, $clk_{T2S}/32$, $clk_{T2S}/64$, $clk_{T2S}/128$, $clk_{T2S}/256$, and $clk_{T2S}/1024$. Additionally, clk_{T2S} as well as 0 (stop) may be selected. Setting the PSR2 bit in SFIOR resets the prescaler. This allows the user to operate with a predictable prescaler.

Special Function IO Register – SFIOR

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	

• Bit 1 – PSR2: Prescaler Reset Timer/Counter2

When this bit is written to one, the Timer/Counter2 prescaler will be reset. The bit will be cleared by hardware after the operation is performed. Writing a zero to this bit will have no effect. This bit will always be read as zero if Timer/Counter2 is clocked by the internal CPU clock. If this bit is written when Timer/Counter2 is operating in asynchronous mode, the bit will remain one until the prescaler has been reset.



Datasheet Modem GSM

wavecom
Make it wireless

Fastrack M1306B User Guide
Basic Features and Services

4 Basic Features and Services

Basic features of the M1306B and available services are summarized in the table below.

Table 3: Basic features of the M1306B

Features	GSM	DCS
Open AT*	Open AT* programmable: Native execution of embedded standard ANSI C applications, Custom AT command creation, Custom application library creation, Standalone operation.	
Standard	900 MHz. E-GSM compliant. Output power: class 4 (2W). Fully compliant with ETSI GSM phase 2 + small MS.	1800 MHz Output power: class 1 (1W). Fully compliant with ETSI GSM phase 2 + small MS.
GPRS	Class 10. PBCCH support. Coding schemes: CS1 to CS4. Compliant with SMG31bis. Embedded TCP/IP stack (optional).	
Interfaces	RS232 (V.24/V.28) Serial interface supporting: <ul style="list-style-type: none"> ▪ Baud rate (bits/s): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, ▪ Autobauding (bits/s): 2400, 4800, 9600, 19200, 38400, 57600. 2 General Purpose Input/Output gates (GPIOs) available. 3 V SIM interface. AT command set based on V.25ter and GSM 07.05 & 07.07. Open AT* interface for embedded application.	
SMS	Text & PDU. Point to point (MT/MO). Cell broadcast.	

wavecom confidential © Page: 25 / 65

wavecom
Make it wireless

Fastrack M1306B User Guide
Basic Features and Services

Features	GSM	DCS
Data	Data circuit asynchronous. Transparent and Non Transparent modes. Up to 14.400 bits/s. MNP Class 2 error correction. V42.bis data compression.	
Fax	Automatic fax group 3 (class 1 and Class 2).	
Audio	Echo cancellation Noise reduction Telephony. Emergency calls. Full Rate, Enhanced Full Rate and Half Rate operation (FR/EFR/HR). Dual Tone Multi Frequency function (DTMF).	
GSM supplement services	Call forwarding. Call barring. Multiparty. Call waiting and call hold. Calling line identity. Advice of charge. USSD	
Other	DC power supply Real Time Clock with calendar Complete shielding	

For other detailed technical characteristics, refer to Chapter 8.

wavecom confidential © Page: 26 / 65

This document is the sole and exclusive property of WAVECOM. Not to be distributed or divulged without prior written agreement.

WM_PRJ_M13_UGD_001 -003 November 11, 2006



Fastrack M1306B User Guide
Technical Characteristics

8.2 Electrical Characteristics

8.2.1 Power Supply

Table 13: Electrical characteristics

Operating Voltage ranges	5.5 V to 32 V DC (GSM or DCS or GPRS).
Maximum current	480 mA Average at 5.5V. 2.1 A Peak at 5.5 V.

Note:

The M1306B is permanently powered once the power supply is connected. The following table describes the consequences of over-voltage and under-voltage with the Fastrack M1306B.

Warning:

All the input voltages specification described in this chapter are at the M1306B input. While powering the M1306B, take into account the input drop caused by the power cable. With the delivered cable, this input drop is around 700 mV at 5.5 V and 220 mV at 32V.

Table 14: Effects of power supply defect

If the voltage	then
falls below 5.5 V,	the GSM communication is not guaranteed.
is over 32 V (Transient peaks),	the M1306B guarantees its own protection.
is over 32 V (continuous overvoltage)	the protection of the M1306B is done by the fuse (the supply voltage is disconnected).

The fuse is a 800 mA / 250 V FAST-ACTING 5*20mm. See Chapter 10 for recommended references.

The following table provides information on power consumption of the Fastrack M1306B, assuming an operating temperature of +25 °C and using a 3 V SIM card.



Fastrack M1306B User Guide
Technical Characteristics

Table 15: Power consumption (1*)

Power Consumption in E-GSM 900/DCS 1800 MHz - GPRS class 10		E-GSM 900	DCS 1800	
Input Peak Supply Current Power = 33,10 dBm GSM 900 During 2TX bursts @Pcl5 Power = 30,00 dBm GSM 1800 During 2TX bursts @Pcl0	@ 5.5 V	2.1 A	1,88 A	
	@ 13,2 V	1.4 A	1.1 A	
	@ 32 V	610 mA	390 mA	
Input average supply current in communication mode Power = 33,10 dBm GSM 900 Average 3Rx/2Tx @Pcl5 Power = 30,00 dBm GSM 1800 Average 3Rx/2Tx @Pcl0	@ 5.5 V	500 mA	390 mA	
	@ 13,2 V	500 mA	160 mA	
	@ 32 V	95 mA	80 mA	
Input Peak Supply Current Power = 33,10 dBm GSM 900 During 1TX bursts @Pcl5 Power = 30,00 dBm GSM 1800 During 1TX bursts @Pcl0	@ 5.5 V	2.1 A mA	1,88 A	
	@ 13,2 V	1.4 A	1.1 A	
	@ 32 V	570 mA	380 mA	
Input average supply current in communication mode Power = 33,10 dBm GSM 900 Average 1Rx/1Tx @Pcl5 Power = 30,00 dBm GSM 1800 Average 1Rx/1Tx @Pcl0	@ 5.5 V	300 mA	240 mA	
	@ 13,2 V	125 mA	100 mA	
	@ 32 V	60 mA	50 mA	
Input average supply current in idle mode with active RS232 link (2*)	@ 5.5 V	34 mA	34 mA	
	@ 13,2 V	18 mA	18 mA	
	@ 32 V	9 mA	9 mA	
Input average supply current in idle mode with no RS232 link (3*)	disconnection by software	@ 5.5 V	31 mA	31 mA
		@ 13,2 V	16,5 mA	16,5 mA
		@ 32 V	8,9 mA	8,9 mA
	physical disconnection	@ 5.5 V	31 mA	31 mA
		@ 13,2 V	6,5 mA	6,5 mA
		@ 32 V	4,5 mA	4,5 mA

(1*): The power consumption might vary by 5 % over the whole operating temperature range (-20 °C to +55 °C).

(2*): In this Mode, the RF function is active and the M1306B synchronized with the network, but there is no communication.

(3*): In this Mode, the RF function is disabled, but regularly activated to keep the synchronization with the network. This Mode works only if the serial link is disconnected either physically or by the software (DTE turns DTR in inactive state).