

Generating I2C Using LabVIEW Through Card Expansion

Ria Saptarika¹, Insannul Kamil², Abdussalam Silmi Rabbani³

Program Studi Profesi Insinyur, Pasca Sarjana, Universitas Andalas

E-mail: rsaptarika@yahoo.com¹, insannulkamil@eng.unand.ac.id², alamsilmir@gmail.com³

ABSTRAK

I2C sebenarnya terdiri dari perangkat keras, perangkat lunak, dan protokol, sedangkan I2C membuat segalanya lebih sederhana, efisien, efektif, kompak, dan ekonomis, awalnya untuk mengendalikan tugas yang menggunakan banyak jalur seperti Data 8-bit dan Alamat 16-bit dalam sistem Mikroprosesor akan menggunakan minimal 24 kabel, jalur atau track PCB, sekarang dengan I2C cukup hanya memiliki 2 kabel, dimana satu kabel disebut SDA (Serial Data Address) dan satu kabel lainnya disebut SCL (Serial Clock) Bus. I2C ditemukan oleh Philips Semiconductors pada awal tahun 80-an, tetapi sekarang beberapa perusahaan semikonduktor seperti Xicor, SGS-Thomson, Siemens, Intel, Texas Instrument, Maxim, Amtel dan Analog Devices sudah mengadopsi nya. Artikel ini menjelaskan cara membuat perintah Protokol I2C menggunakan LabVIEW melalui Ekspansi Kartu Personal Computer (PC). Ada dua hal yang akan dijelaskan mengenai pembangkitan I2C, yang pertama adalah Protocol I2C dan yang terakhir adalah cara membangkitkannya menggunakan software LabView.

Keyword: I2C, Labview, Mikrokontroller, PC

ABSTRACT

I2C actually consists of hardware, software, and protocols, whit the I2C makes everything more simple, efficient, effective, compact, and economical, initially for controlling tasks that use multiple paths such as 8-bit Data and 16-bit Address in Microprocessor systems will using at least 24 wire, path or PCB track, now with I2C, it is enough to only have 2 wires, where one wire is called SDA (Serial Data Address) and one other wire is called SCL (Serial Clock) Bus. I2C was invented by Philips Semiconductors in the early '80s, but now some semiconductor companies like Xicor, SGS-Thomson, Siemens, Intel, Texas Instrument, Maxim, Amtel dan Analog Devices were already adopted. This article describes how to generate the I2C Protocol command using LabVIEW through Personal Computer (PC) Card Expansion. There are two things that will be described regarding generating the I2C, the first is Protocol I2C and the last is the LabView software and how to generate it

Keyword: I2C, LabVIEW, Microcontroller, PC

PENDAHULUAN

I²C (Inter Integrated Circuit) atau sarana komunikasi antar IC adalah suatu temuan yang cukup revolusioner pada masanya, dimana banyak menggantikan sistem-sistem sebelumnya yang terlihat, tidak efisien, efektif dan ekonomis. Sebelum ditemukan I2C dalam setiap system pengendalian didalam perangkat elektronik yang

menggunakan Microprocessor atau Microcontroller sara komunikasinya selalu menggunakan Port I/O maupun Address yang memiliki jalur PCB atau kabel sebanyak 8 - 64 jalur sesuai banyak Bit yang digunakan atau biasa disebut dgn Bus sehingga Bus data dan Bus Address sehingga sebuah PCB dengan jalur yang rumit tidak

dapat terelakkan, ditambah lagi dengan banyaknya komponen-komponen lain sebagai interface.

I²C yang sebenarnya terdiri dari hard ware, soft ware, dan protocolnya, telah membuat sesuatunya lebih simple, efisien, efektif, kompak dan ekonomis, dimana untuk tugas pengontrolan yang menggunakan banyak jalur seperti 8-bit Data dan 16-bit Address pada sistem Microprocessor sebelumnya, dengan I²C cukup hanya dengan 2 jalur saja, dimana, 1 jalur di sebut dengan SDA (Serial Data Address) dan 1 jalur lagi disebut dengan SCL (Serial Clock) Bus. I²C ditemukan oleh Philips Semiconductors di awal tahun 80-an, sekarang teknologi I²C, sudah di adopsi oleh beberapa Pabrik semikonduktor terkenal lain, diantaranya Xicor, SGS-Thomson, Siemens, Intel, Texas Instrument, Maxim, Amtel dan Analog Devices, sehingga tidak heran kalau sistem I²C sekarang telah tersedia (build in) dalam berbagai IC dan aplikasi antara lain; LCD, Memory, Audio/Video, PLL, Microcontroller, Data converter (DA/AD), I/O expander dll.

Dalam tulisan ini Pembangkitan I2C dilakukan dengan menggunakan Software LabVIEW sebuah software pemerograman berbasis visual artinya dengan cara drag & drop bagan-bagan yang sudah disediakan oleh LabVIEW dapat menjadi sebuah pemerograman untuk membangkitkan I2C. Sementara Port keluaran dari komputer dapat menggunakan dua cara yakni; melalui Card Expansion atau melalui LPT Port, yang membedakannya hanya pada Address nya, untuk LPT menggunakan Address 378h - 37Fh sementara Card Expansion menggunakan 220h - 2F7h, sedangkan untuk menjaga potensi short circuit dari kedua sisi atau untuk penggunaan lainnya ditambahkan rangkaian Logic Gate yang berfungsi sebagai buffer pada keluaran Card Expansion tersebut.

TINJAUAN PUSTAKA

MEMAHAMI I2C

Saluran, SDA dan SCL memiliki kemampuan komunikasi dua arah (bidirectional), maksudnya saluran SDA dan SCL dapat di kendalikan dari dalam ke luar IC I2C, maupun dari luar ke dalam. Masing-masing saluran tersebut adalah open drain atau open collector¹, tergantung teknologi pembuatan IC tersebut (Bipolar, CMOS atau NMOS), untuk itu setiap saluran tersebut dibutuhkan resistor pull-up, pada kondisi ini dapat

disimpulkan bahwa bila saluran SDA dan SCL sedang tidak aktif, maka saluran tersebut akan selalu belogika tinggi.

Hal yang menarik dari konsep ini adalah, bila salah satu IC I2C didalam suatu bus active (membuat level logika rendah), maka semua IC tidak akan dapat melakukan apa-apa lagi karena semua saluran di buat berimpedansi rendah terhadap ground, dasar pengertian inilah yang disebut dengan arbitration.

Walaupun demikian konsep “open collector” ini juga memiliki kekurangan. Dimana ia akan membatasi panjangnya penggunaan Bus.

Bila bus terlalu panjang maka akan menurunkan kecepatan transfer data I2C, sebab panjangnya bus akan membuat beban kapasitansi bus membesar atau besarnya kapasitansi ini akan mempengaruhi konstanta waktu (time constant) bus membesar pula. Hal inilah yang secara langsung mempengaruhi waktu naik (rise time) dan waktu jatuh (fall time) setiap sinyal I2C, untuk lebih memperjelasnya lihatlah Gambar-1 dan karakteristik jalur bus pada Tabel-1

The I²C-bus specification

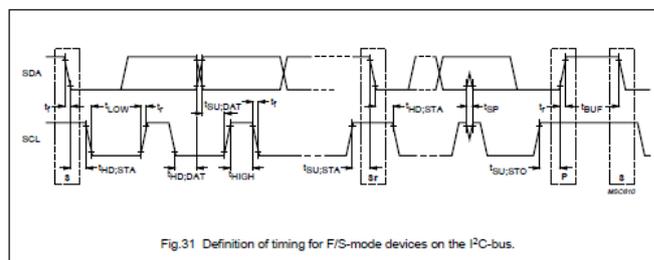


Fig.31 Definition of timing for F/S-mode devices on the I²C-bus.

Gambar 1 Definisi timming pada i2c bus
Sumber: The I2c-Bus Specification Ver.2.1 J
anuary 2000, p.32

Tabel 1 Karakteristik SDA & SCL Bus
 Sumber: The I2c-Bus Specification Ver.2.1 January 2000, p.33

Table 5 Characteristics of the SDA and SCL bus lines for F/S-mode (PC-bus devices⁽¹⁾)

PARAMETER	SYMBOL	STANDARD-MODE		FAST-MODE		UNIT
		MIN.	MAX.	MIN.	MAX.	
SCL clock frequency	f _{SCL}	0	100	0	400	kHz
Hold time (repeated) START condition. After this period, the first clock pulse is generated	t _{HD,STA}	4.0	–	0.6	–	µs
LOW period of the SCL clock	t _{LOW}	4.7	–	1.3	–	µs
HIGH period of the SCL clock	t _{HIGH}	4.0	–	0.6	–	µs
Set-up time for a repeated START condition	t _{SU,STA}	4.7	–	0.6	–	µs
Data hold time: for CBUS compatible masters (see NOTE, Section 10.1.3) for PC-bus devices	t _{HD,DAT}	5.0 (0 ⁽²⁾)	– 3.45 ⁽³⁾	– 0 ⁽²⁾	– 0.9 ⁽³⁾	µs µs
Data set-up time	t _{SU,DAT}	250	–	100 ⁽⁴⁾	–	ns
Rise time of both SDA and SCL signals	t _r	–	1000	20 + 0.1C _b ⁽⁵⁾	300	ns
Fall time of both SDA and SCL signals	t _f	–	300	20 + 0.1C _b ⁽⁵⁾	300	ns
Set-up time for STOP condition	t _{SU,STO}	4.0	–	0.6	–	µs
Bus free time between a STOP and START condition	t _{BUF}	4.7	–	1.3	–	µs
Capacitive load for each bus line	C _b	–	400	–	400	pF
Noise margin at the LOW level for each connected device (including hysteresis)	V _{NL}	0.1V _{DD}	–	0.1V _{DD}	–	V
Noise margin at the HIGH level for each connected device (including hysteresis)	V _{NH}	0.2V _{DD}	–	0.2V _{DD}	–	V

Pada keadaan tertentu pengaruh waktu naik dan waktu turun sinyal ini akan lebih parah karena dapat mengaburkan perbedaan antara logika TINGGI dan logika RENDAH.

PROTOCOL I2C

Berikut adalah protokol I2C secara sederhana dapat dilihat pada gambar telegram I2C dibawah ini, perhatikan pula nomor urut yang menandakan urutan pengiriman dari sebuah telegram:

1	2	3	4	5	6
Start	Address	Acknowledge	Data	Acknowledge	Stop

Gambar 3 Urutan pengiriman telegram dengan 2 acknowledge

Bila kita lihat urutan keadaan didalam telegram tersebut maka masing-masing dapat dijelaskan sebagai berikut:

Start: Merupakan sinyal “pemberitahuan” dari semua slave (IC) agar bersiap-siap untuk menerima sebuah data.

Address: Merupakan sinyal berikut setelah start (sepanjang 8 pulsa SCL), berguna untuk mengirimkan alamat (Address) kepada slave. Dalam keadaan ini semua slave akan membandingkan masing-masing alamat miliknya. Bagi slave yang address-nya cocok, maka slave tersebut akan mengeluarkan respon yang disebut dengan Acknowledge.

Acknowledge: Adalah sebuah respon dari slave, dalam bentuk membuat SDA menjadi berlogika rendah, sesaat setelah Master mengirimkan satu byte Address/data, tepatnya pada pulsa clock SCL ke-9 nya. Bila bus sudah menerima acknowledge dari slave maka Master akan memulai mengirimkan data berikutnya.

Data: adalah sebuah sinyal data yang memiliki panjang sebanyak 8 pulsa SCL atau sama dengan 1 byte. Setelah selesai menerima 8 bit data, maka slave memberi Acknowledge lagi.

Stop: Merupakan sinyal pertanda bahwa Master bus sudah mengakhiri pengiriman data, sehingga dalam hal ini semua IC kembali dalam kondisi diam (tidak siap menerima data, sampai bila ada sinyal start berikutnya).

START DAN STOP I2C

Untuk membuat kondisi START caranya membuat transisi sinyal SDA dari kondisi berlogika TINGGI menuju RENDAH sementara keadaan itu, SCL dipertahankan berlogika TINGGI, untuk mempermudah memahaminya lihatlah tabel kebenaran kondisi START dibawah ini.

Tabel 2 Tabel Kebenaran Kondisi Start

Tahap	SCL	SDA
1	1	1
2	1	0
3	0	0

Sedangkan untuk membuat kondisi STOP ialah, dengan cara membuat transisi sinyal SDA dari kondisi berlogika RENDAH menuju TINGGI sementara keadaan itu, SCL dipertahankan berlogika TINGGI, sebagaimana tabel kebenaran START diatas lihat pula tabel kebenaran kondisi STOP dibawah ini.

Tabel 3 Tabel Kebenaran Kondisi Stop

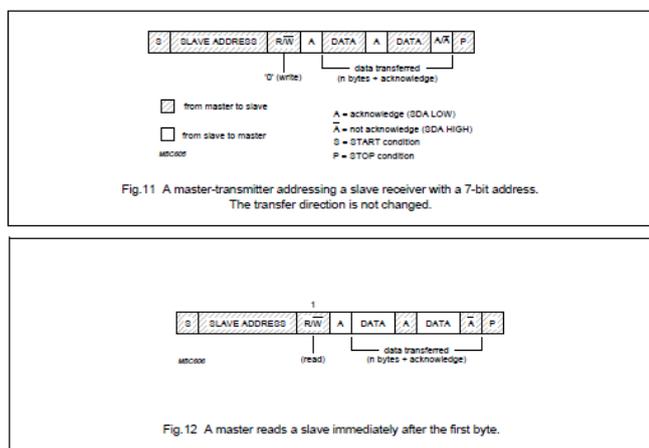
Tahap	SCL	SDA
1	1	0
2	1	1
3	1	1

Kondisi START dan STOP selalu dibangkitkan oleh Master, dimana bila sinyal START sudah dikirimkan, berarti Bus dalam kondisi terpakai (sibuk) dan Bus akan

bebas kembali bila telah dikirimkan sinyal STOP, dalam prakteknya sinyal START boleh dikirim ulang didalam satu siklus sinyal STOP atau biasa disebut dengan Repeated START (Sr), sedangkan kegunaannya adalah sama.

ADDRESSING I2C

Prosedur pengalamatan (addressing) didalam I2C, dilakukan pada pengiriman byte pertama setelah pengiriman signal START sedangkan Bit address yang dipakai untuk pengalamatannya hanya 7-Bit dan sisa 1-Bit lagi berguna untuk membedakan antara mode baca atau mode tulis (Read/Write) dari sebuah telegram I2C, lihatlah kembali contoh dari telegram I2C yang sudah digambarkan pada bahasan Protocol I2C Bus dan



konfigurasi byte address pada Gambar-3

Gambar 3 Bentuk Telegram Kondisi Baca & Tulis Master I2C ke Slave, Sumber: The I2c-Bus Specification Ver.2.1 January 2000, p.14

Seperti pengertian Addressing umumnya bertujuan untuk mengaktifkan chip tertentu yang terhubung pada Bus I2C, meskipun demikian didalam I2C Bus terdapat sebuah address yang di kenal dengan “General Call Address” [1 HAL.16] yaitu sebuah address yang dapat mengaktifkan seluruh chip yang terhubung pada Bus I2C secara bersamaan. Sedangkan beberapa kekhususan lain lihatlah Tabel-4 dari tabel dibawah dapat disimpulkan bahwa semua address yang dibawah $1F_{(hex)}$ adalah dipersiapkan untuk keperluan khusus

Tabel 3 Definisi Bit Address, Sumber: The I2c-Bus Specification Ver.2.1 January 2000, p.16

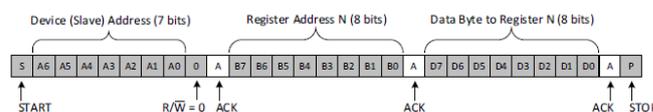
Table 2 Definition of bits in the first byte

SLAVE ADDRESS	R/W BIT	DESCRIPTION
0000 000	0	General call address
0000 000	1	START byte ⁽¹⁾
0000 001	X	CBUS address ⁽²⁾
0000 010	X	Reserved for different bus format ⁽³⁾
0000 011	X	Reserved for future purposes
0000 1XX	X	Hs-mode master code
1111 1XX	X	Reserved for future purposes
1111 0XX	X	10-bit slave addressing

Sebuah alamat Slave terkadang juga terdiri dari alamat yang secara permanen sudah ditetapkan oleh pabrik (Fixed) dan alamat yang dapat diubah (secara hardware) melalui kombinasi logika yang diberikan pada Pin-pin IC Slave, dengan demikian membolehkan beberapa IC yang sama dipasang dalam Bus yang sama, sedangkan jumlah alamat yang dapat diprogram, tergantung jumlah pin yang tersedia pada IC tersebut. Sebagai contoh, bila sebuah IC Slave memiliki 3-pin alamat berarti ia memiliki 3-bit alamat yang dapat diprogram, maka jumlah slave yang sama yang dapat dihubungkan bersamaan dalam Bus yang sama adalah sebanyak 8 buah (2^3).

DATA VALIDITY

Proses transfer data dilakukan bit per bit dengan mendahulukan yang MSB (Most Significant Bit) dan seterusnya hingga LSB. Setiap byte yang dikirim SDA bus haruslah memiliki panjang 8-bit (sebanyak 8-pulsa clock SCL), sedangkan banyaknya jumlah byte yang dapat dikirimkan tidaklah terbatas, namun setiap byte harus dipisahkan dengan acknowledge lihat Gambar-2 dan Gambar-4 dibawah

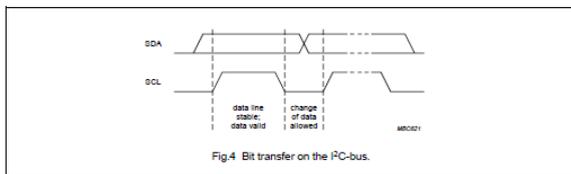


Gambar 4 Penempatan Tiga Acknowledge Sumber: Understanding the I 2C Bus, Texas Instruments, p.7

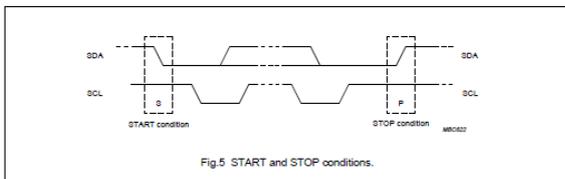
Untuk menghindari kesalahan dalam proses Transfer Bit, maka bila SCL berlogika TINGGI, SDA tidak boleh

berubah (harus konsisten TINGGI atau konsisten RENDAH). Dengan kata lain, setiap perubahan transfer Bit dari TINGGI ke RENDAH (atau sebaliknya), hanya boleh dilakukan bila SCL dalam kondisi RENDAH, proses ini dapat dilihat pada Gambar-5 & Gambar-6 sebagai penjelasan tambahan, Pelarangan perubahan logika SDA pada keadaan SCL TINGGI ini sesungguhnya dikarenakan oleh dua faktor yaitu:

1. Bila SDA berubah dari TINGGI ke RENDAH, maka hal ini akan diartikan sebagai START,
2. Bila SDA berubah dari RENDAH ke TINGGI, maka hal ini akan diartikan sebagai STOP,



Gambar 5 Transfer perubahan Bus SDA pada saat SCL Low,
 Sumber: The I2c-Bus Specification Ver.2.1 January 2000, p.9 [1]



Gambar 6 Kondisi SDA & SCL Pada Start & Stop
 Sumber: The I2c-Bus Specification Ver.2.1 January 2000, p.9 [1]

LabVIEW

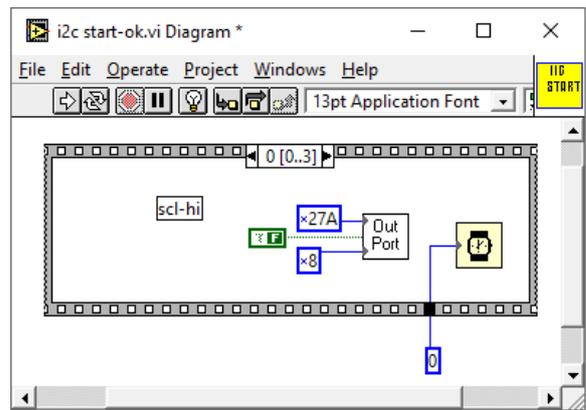
LabVIEW adalah perangkat lunak yang berguna untuk mengembangkan system pengujian dan pengukuran, akuisisi data, kontrol instrumen, pencatatan data, analisis pengukuran, dan aplikasi pembuatan laporan. untuk artikel ini, kami menggunakan untuk menghasilkan I2C. [3]

Dengan menganggap kita sudah familiar menggunakan Labview maka kita langsung membahas langkah per langkah bagaimana cara membangkitkan I2C menggunakan LabVIEW saja.

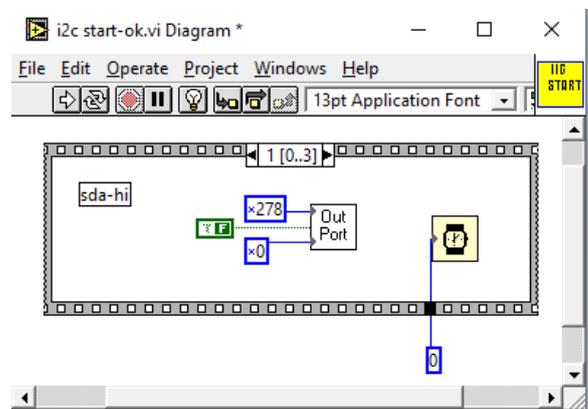
PEMBAHASAN

MEMBANGKITKAN I2C MENGGUNAKAN LabVIEW

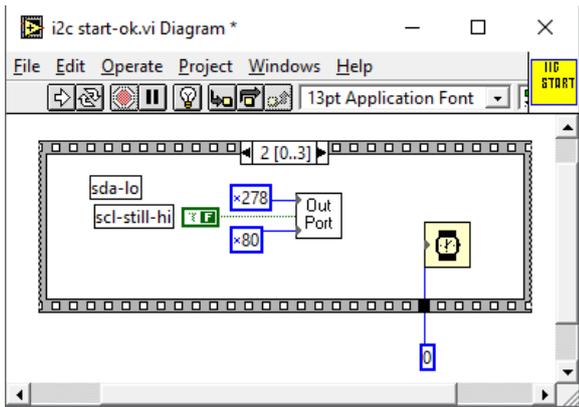
Langkah pertama yg kita lakukan adalah membuat perintah Start & Stop pada LabVIEW lihatlah pada Gambar-7 sampai Gambar-10 dibawah, ada 4 gambar yang menampilkan 4-langkah yang dibuat untuk melakukan perintah Start pada LabVIEW. Perlu pula dijelaskan bahwa untuk untuk address Port SCL adalah 27Ah dan byte data Port nya diberi 8h. (artinya Bit ke-4 dari LSB diberi “hi”) sedangkan address Port SDA adalah 278h dan byte data Port nya diberi 80h dan 0h.



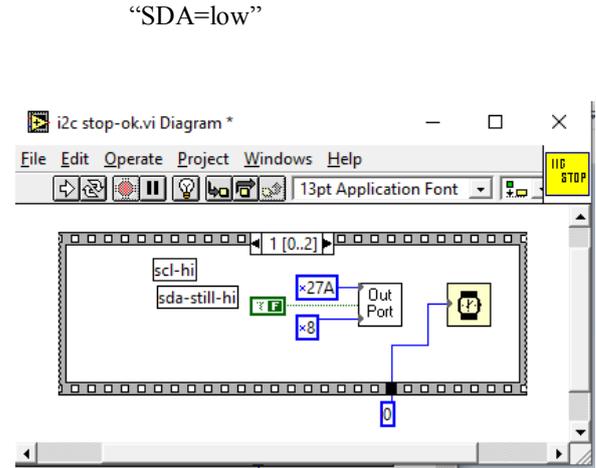
Gambar 7 Step Nol Langkah Membangkitkan Start “SCL=hi”



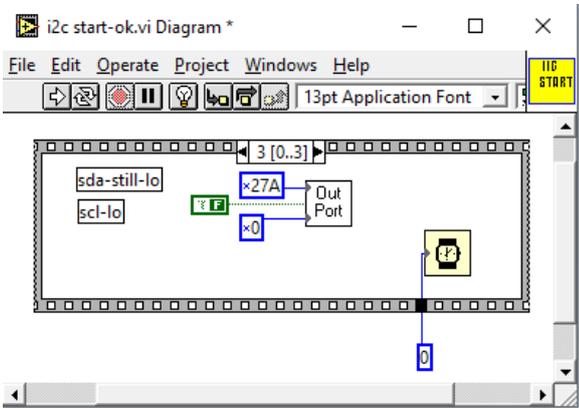
Gambar 8 Step 1 Langkah Membangkitkan Start “SDA=hi”



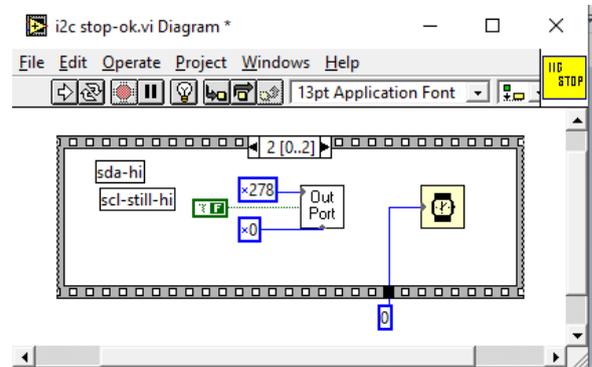
Gambar 4 Step 2 Langkah Membangkitkan Start
 “SCL=hi, SDA=low”



Gambar 12 Step 1 Langkah Membangkitkan Stop
 “SCL=hi, SDA=hi”



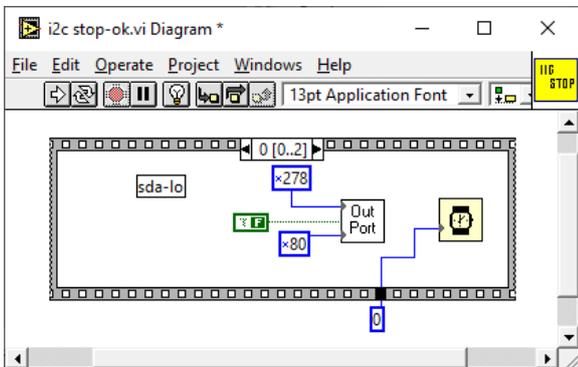
Gambar 10 Step 3 Langkah Membangkitkan Start
 “SDA=low, SCL=low”



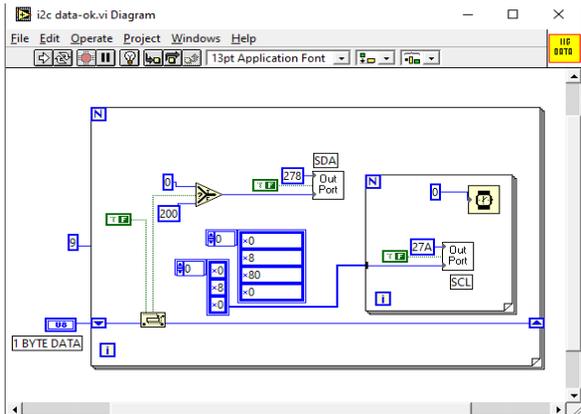
Gambar 13 Step 2 Langkah Membangkitkan Stop
 “SDA=hi, SCL=hi,”

Berikutnya adalah langkah untuk melakukan perintah Stop lihatlah pada Gambar-11 sampai Gambar-13 dimana untuk perintah Stop ini ada tiga step LabVIEW

Berikut ini adalah langkah untuk melakukan perintah Pengiriman DATA lihatlah LabVIEW Gambar-14 dimana untuk perintah ini hanya ada satu step namun dilakukan berulang (looping) sebanyak 9 kali dan 1 tahap terakhir itu sebagai acknowledge.

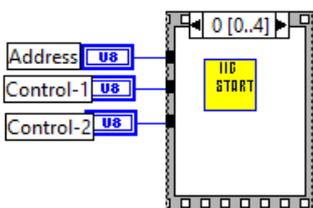


Gambar 11 Step Nol Langkah Membangkitkan Stop

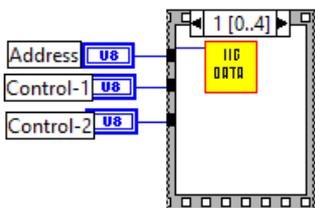


Gambar 14 Perintah Pengiriman Data I2C

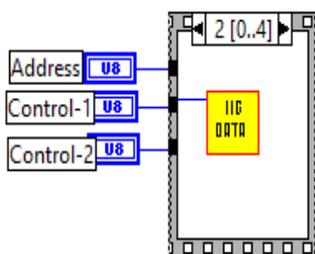
Terakhir dibawah ini Pada Gambar-15 sampai Gambar-19 adalah perintah I2C satu paket lengkap terdiri dari Signal Start kemudian diikuti dua pengulangan pengiriman data, byte data pertama adalah Address Byte data kedua adalah byte Data (dapat berupa kontrol atau Data itu sendiri). Dalam praktek dilapangan pengirimsn byte data ini dapat dilakukan beberapa kali sesuai keperluan namun tentunya setiap satu pengirimannya dibuat Acknowledge.



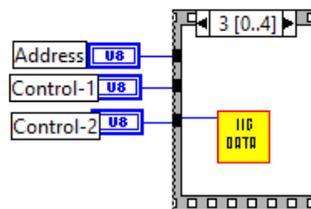
Gambar 5 Step Nol I2C mengirim Start



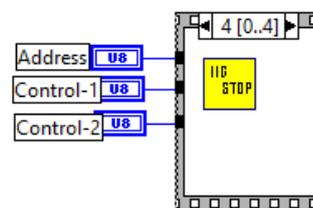
Gambar 6 Step 1 I2C Mengirim Address



Gambar 7 Step 2 I2C Mengirim Data ke-2



Gambar 8 Step 3 I2C Mengirim Data ke-3



Gambar 9 Step 4 I2C Mengirim Stop

KESIMPULAN DAN SARAN

Kesimpulan:

1. I2C dapat di bagkitkan dengan Komputer dengan bantuan software pendukung seperti LabVIEW.
2. Dengan Komputer eksperimen-experimen perangkat perngkat yang menggunakan I2C dilakukan dengan mudah

Saran:

1. Pada artikel ini menggunakan LabVIEW versi 4.1 dengan perkembangan zaman software juga berkembang sehingga perlu dilakukan untuk versi terbarunya.
2. Perlu juga di lakukan menggunakan software tingkat tinggi seperti C++, Visual Basic dll.

DAFTAR PUSTAKA

- [1] Philips Semiconductors, *The I2c-Bus Specification Version 2.1*, January 2000
- [2] Jonathan Valdez, Jared Becker, *Understanding the I2C Bus*, Texas Instruments, Texas 2018 available at: <https://www.ti.com/lit/pdf/slva704> hal.7
- [3] National Instruments, Malan Shiralkar, *LabVIEW Graphical Programming Course*, CONNEXIONS, Rice University, Houston, Texas, October 25, 2012. hal.2
available at: <http://cnx.org/content/col10241/1.4/>