

PC-Link **PC-LINK** *Application Notes* AN135 – Kontrol Motor DC Melalui **Port USB**

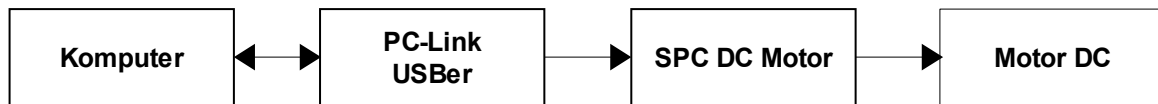
Oleh: Tim IE

Application note ini akan memanfaatkan fitur *bit bang* dari PC-Link USBer sama seperti halnya AN122. Fitur *bit bang* kali ini akan dimanfaatkan untuk mengirim sinyal PWM ke modul SPC DC Motor yang akan mengendalikan 1 buah motor DC. Aplikasi ini masih bisa dikembangkan agar dapat mengendalikan 2 motor DC sekaligus.

Modul atau komponen yang diperlukan dalam AN ini adalah:

- 1 buah PC-Link USBer
- 1 buah SPC DC Motor
- 1 buah motor DC

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN135

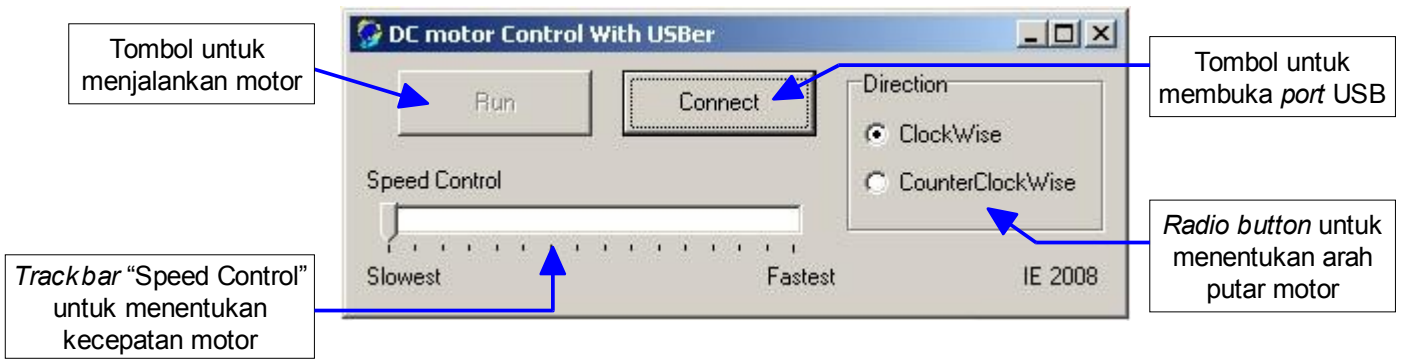
Hubungan antara modul-modul tersebut adalah sebagai berikut:

PC-Link USBer	SPC DC Motor
GND (J2-Pin 1)	GND (J1)
VCC (J2-Pin 2)	+5V (J1)
TXD (J2-Pin 3)	Run 1 (J5-S1)
RXD (J2-Pin 4)	Dir 1 (J5-S2)

Tabel 1
Hubungan PC-Link USBer dengan SPC DC Motor

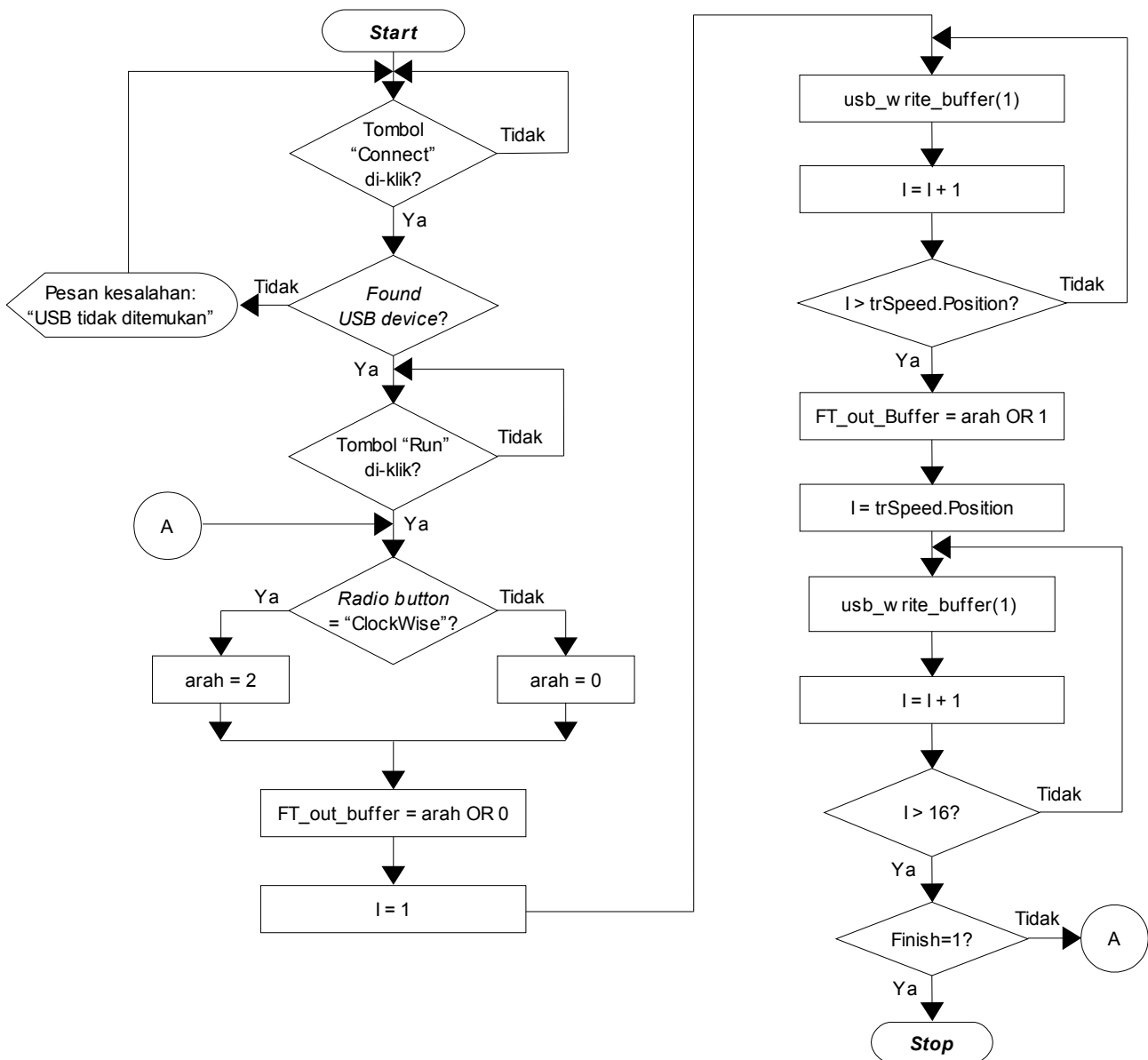
Hubungkan motor DC ke terminal M1+ dan M1- pada J6 SPC DC Motor, polaritas boleh dibolak-balik tetapi risikonya arah putaran motor dapat tidak sesuai dengan asumsi di dalam program. Gunakan catu daya yang sesuai dengan kebutuhan motor DC, lalu hubungkan catu daya tersebut ke GND dan Vmotor pada terminal J6 SPC DC Motor. Berilah catu daya 5 Volt DC dengan arus minimal 500mA pada terminal J1 SPC DC Motor.

Gunakan kabel USB yang tersedia dalam paket penjualan PC-Link USBer untuk menghubungkan modul ini ke *port* USB komputer. Lakukan instalasi driver D2XX agar aplikasi ini dapat bekerja dengan benar, cara instalasi terdapat dalam CD produk PC-Link USBer. Setelah semua modul dan sumber tegangan terhubung dengan baik dan benar, jalankan program **DCMotor.exe** yang terdapat dalam **AN135.zip**. Program **DCMotor.exe** dikembangkan dengan Borland® Delphi™ versi 7.0 dan menggunakan bantuan komponen TaUSBDirect untuk menangani komunikasi *bit bang* PC-Link USBer. Tampilan program DCMotor.exe ini tampak seperti Gambar 2.



Gambar 2
Tampilan Program DCMotor.exe

Flowchart dari listing program utama DC.PAS adalah sebagai berikut:



Gambar 3
Flowchart Program

Penjelasan program secara garis besar adalah sebagai berikut:

1. Program akan melakukan deklarasi variabel dan kemudian menunggu pengguna meng-klik tombol "Connect".
2. Jika tombol "Connect" di-klik maka program akan memeriksa apakah *USB device* terdeteksi. Hal ini dilakukan dengan memeriksa nilai *usb.GetFTDeviceCount* apakah sama dengan *FT_OK*.
3. Jika *USB device* terdeteksi maka tombol "Run" diaktifkan dan melanjutkan ke langkah selanjutnya. Jika *USB device* tidak terdeteksi maka program menetapkan *FT_HANDLE=0*, menampilkan pesan kesalahan "USB tidak ditemukan" dan program kembali ke langkah 1.
4. Program menunggu penekanan tombol "Run". Jika tombol "Run" ditekan maka program akan masuk perulangan *while-do* pada langkah berikutnya.
5. Program akan memeriksa kondisi *radio button* pada panel "Direction". Jika *radio button* yang dipilih adalah "ClockWise" maka variabel arah akan diberi nilai 2, tetapi jika tidak maka variabel arah akan diberi nilai 0.
6. Program mengisi *buffer* keluaran *FT_Out_Buffer[0]* dengan nilai tertentu, bit 1 dari nilai ini berkaitan dengan pin Dir 1 dan bit 0 berkaitan dengan pin Run 1. Bit 1 pada nilai tersebut ditentukan oleh variabel arah dan akan membuat motor berputar ke kiri atau ke kanan sesuai dengan pilihan pada *radio button* di panel "Direction". Sedangkan bit 0 pada nilai tersebut akan menentukan apakah motor aktif atau tidak. Bit 0 ini akan diisi dengan nilai *low* dan *high* secara bergantian, lamanya nilai *low* dan lamanya nilai *high* ditentukan oleh nilai variabel *trSpeed.Position* (*trackbar* "Speed Control"). Semakin tinggi kecepatan yang dipilih pada *trackbar* "Speed Control", bit 0 (pin Run 1) akan bernilai *low* lebih lama daripada bernilai *high*, demikian juga sebaliknya. Penjelasan detail *listing* programnya sebagai berikut:
 - Ketika bit 0 (pin Run 1) bernilai *low*:
FT_Out_Buffer[0]:=arah OR 0;
for i:=1 to trSpeed.Position do Usb.Write_USB_Device_Buffer(1);
 - Ketika bit 0 (pin Run 1) bernilai *high*:
FT_Out_Buffer[0]:=arah OR 1 ;
for i:=trSpeed.Position to 16 do usb.Write_USB_Device_Buffer(1);
 - Keterangan:
 - Nilai variabel arah dapat bernilai 0 (bit 1 = *low*) untuk arah CCW atau bernilai 2 (bit 1 = *high*) untuk arah CW.
 - Bit 0 pada *FT_Out_Buffer[0]* akan menentukan kondisi logika pada pin Run 1. Ketika bit ini berlogika *low* maka motor akan aktif, dan ketika bit ini berlogika *high* maka motor akan mati.
 - Proses perulangan *for-to-do* dari *i=1* sampai *i=trSpeed.Position* akan menentukan lama waktu pin Run 1 berlogika *low*.
 - Proses perulangan *for-to-do* dari *i=trSpeed.Position* sampai *i=16* akan menentukan lama waktu pin Run 1 berlogika *high*.
 - Nilai *high* dan *low* secara bergantian pada bit 0 ini akan membentuk sinyal PWM dengan *duty cycle* yang sesuai dengan posisi *trackbar* "Speed Control".
 - Nilai biner 8 bit pada variabel *FT_Out_Buffer[0]* merupakan nilai yang akan dikeluarkan pada pin TXD, RXD, RTS, CTS, dst pada PC-Link USBer. Dalam AN ini hanya digunakan 2 bit saja, yaitu bit 0 (pin TXD) dan bit 1 (pin RXD).
7. Program memeriksa nilai variabel *Finish*. Jika *Finish=1* maka program akan keluar dari proses perulangan dan kembali ke langkah 1. Sedangkan jika *Finish<>1* maka program kembali ke langkah 5.

Keterangan lebih lanjut mengenai komponen TaUSBDirect, terdapat pada *application note AN122* yang dapat diperoleh dari situs www.InnovativeElectronics.com.

Listing program terdapat pada **AN135.ZIP**.

Selamat berinovasi!

All trademarks, trade names, company names, and product names are the property of their respective owners.
All softwares are copyright by their respective software publishers and/or creators.