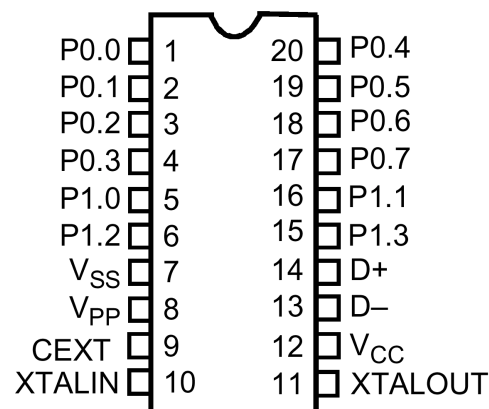


FAN Solution

- Kan de snelheid van vier ventilatoren onafhankelijk van elkaar in 255 stappen regelen.
- Gebruikt pulsbreedte modulatie met een herhalingsstijd van 4ms. Is ook bruikbaar om lampen te dimmen, gelijkstroom motortjes aan te sturen, etc.
- 8 Extra I/O lijnen, elk bruikbaar als ingang of als uitgang.
- Programmeerbare pull-up weerstand op elke lijn, per lijn apart te kiezen.
- In 16 stappen programmeerbare uitgangs stroom, per lijn individueel te programmeren.
- Vormt met slechts twee externe componenten een complete bouwsteen voor op de USB bus. Om een ventilator aan te sturen is daarnaast nog slechts één transistor nodig.
- Software geschikt voor Windows 98SE, ME, 2000 en XP.



1	Poort 0 bit 0	Poort 0 bit 4	20
2	Poort 0 bit 1	Poort 0 bit 5	19
3	Poort 0 bit 2	Poort 0 bit 6	18
4	Poort 0 bit 3	Poort 0 bit 7	17
5	Fan1	Fan2	16
6	Fan3	Fan4	15
7	Vss (0 Volt)	D+ (USB bus)	14
8	Vpp (verbind deze met 0V)	D- (USB bus)	13
9	Cext (open-drain suspend uitgang)	Vcc (+5 Volt)	12
10	XTALIN aansluiten op resonator	XTALOUT aansluiten op resonator	11

Inleiding

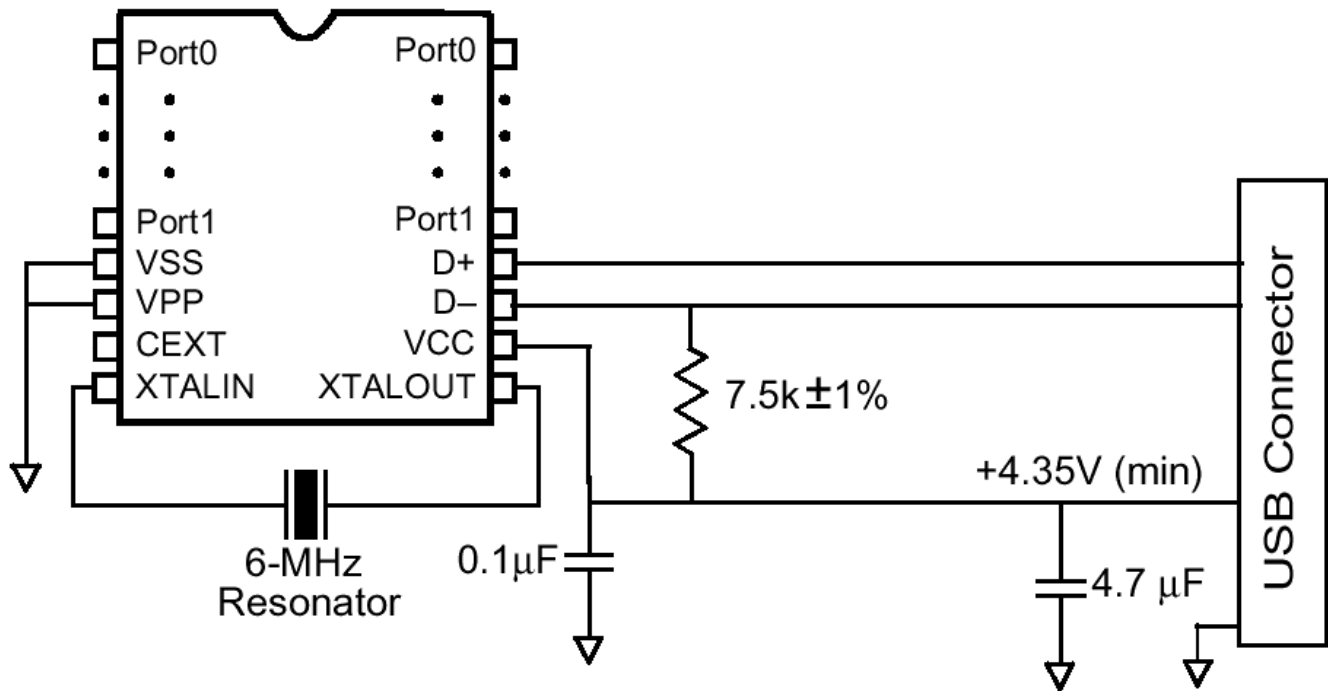
De FAN Solution is gebaseerd op een CY7C63001A microcontroller van Cypress (www.cypress.com) Deze microcontroller heeft een speciaal programma gekregen om zijn werk als FAN Solution te kunnen doen. Het programma voor de microcontroller kan niet gewijzigd worden.

Bij aansluiten aan een PC zorgt het ingebouwde programma van de FAN Solution er voor, dat deze door de PC automatisch wordt herkend. De FAN Solution heeft geen specifieke drivers nodig, de benodigde drivers maken standaard deel uit van Windows 98 (SE), ME, 2000 en XP. Windows 95 ondersteunt de FAN Solution niet.

De FAN Solution kan vanuit de PC eenvoudig worden aangestuurd.

Aansluiten

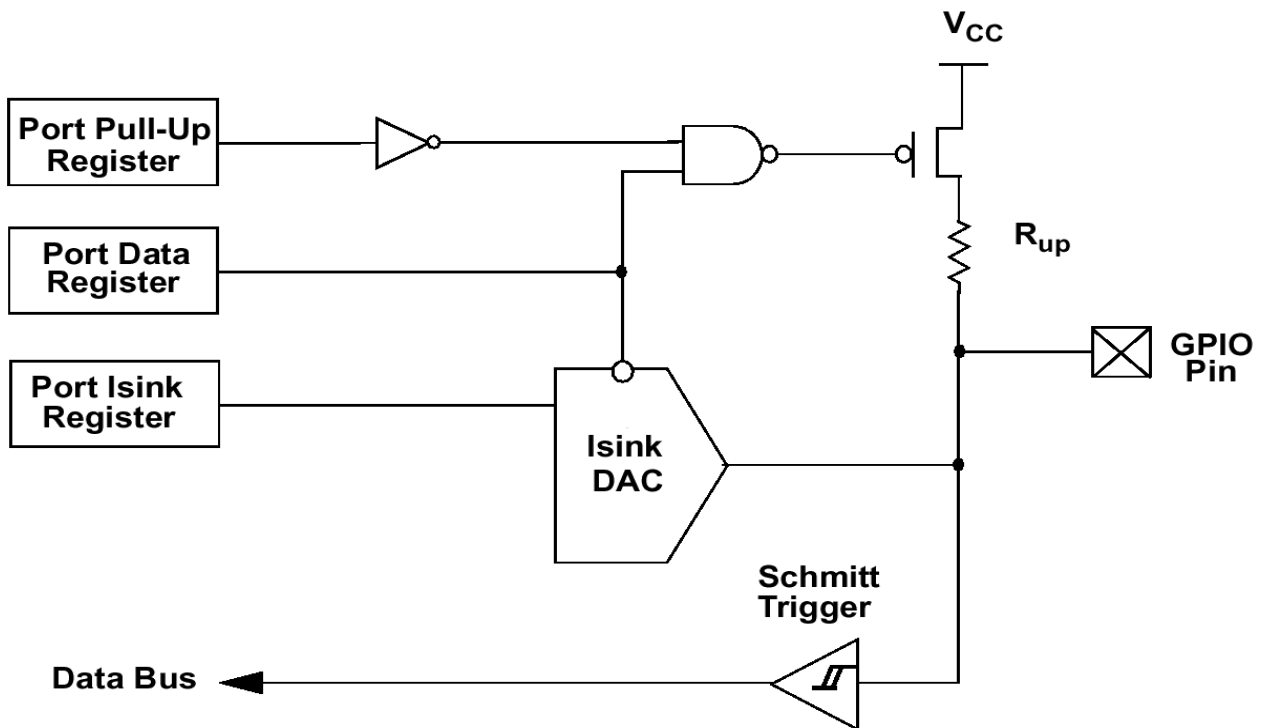
Om de FAN Solution op de USB bus aan te sluiten is niet meer nodig dan een +5 Volt voeding, die van de USB bus kan worden afgenomen, en de twee USB datalijnen. Een weerstand van 7.5 kΩ tussen +5 Volt en de D- lijn van de USB bus, vertelt de USB bus dat er een low-speed apparaat is aangesloten. Een keramische resonator van 6MHz is genoeg om de FAN Solution te laten werken. Een paar ontkoppel condensatoren kunnen worden toegevoegd om het geheel storingsvrij te laten functioneren. De ventilatoren of andere te regelen componenten worden aangesloten op de pinnen P1.0 t/m P1.3. Als de belasting wat groter is, kan een (darlington) NPN transistor voor de nodige power zorgen. Een basisweerstand is niet nodig.



I/O poorten

Er zijn vier uitgangen gereserveerd om de ventilatoren (of lampen of motortjes, etc...) mee te regelen. Deze uitgangen kunnen direct de basis van een NPN transistor aansturen, zodat heel eenvoudig voor voldoende stroom voor de ventilatoren kan worden gezorgd.

Er is daarnaast ook nog een vrij te gebruiken I/O poort van 8 bits. Het interne schema van een lijn van deze poort ziet er als volgt uit:



Hierin zijn de volgende functies te herkennen:

1. Een schmitt-trigger ingangspoort die wordt gebruikt om het werkelijke logische niveau van de pin te meten.
2. Met een pull-up weerstand, die naar believen is te activeren, kan de pin intern een gedefinieerd 'hoog' niveau gegeven worden.
3. Met een DAC kan de stroom worden vastgelegd die de poort in vloeit, als deze via het data register 'laag' gemaakt wordt.

Pull-up register	Data register	
0	0	De pin is laag, de stroom die de pin in loopt richting 0V is vastgelegd door de instelling van de DAC (16 standen) ⁻
0	1	De pin wordt intern naar +5 V getrokken door de inwendige pull-up weerstand.
1	0	De pin is laag, de stroom die de pin in loopt richting 0V is vastgelegd door de instelling van de DAC (16 standen) ⁻
1	1	De pin is een hoog-ohmige ingang.

De poort accepteert ingangs stromen tot nominaal 1mA. De stroom voor 4 pinnen van poort 0 (P0.7, P0.6, P0.5 en P0.4) staat vast ingesteld op de maximale waarde. De stroom voor de andere 4 pinnen kan via de PC voor elke pin apart geprogrammeerd worden.

Voorbeelden:

- De basis van een PNP transistor kan worden aangesloten op een I/O pin en de emitter van de PNP transistor op de +5 Volt. De basisstroom die door de transistor moet gaan lopen kan in 16 standen worden ingesteld. De stroom is in elk geval begrensd, zodat geen extra weerstand in de basis nodig is. Met de collector van de transistor kan een wat zwaardere belasting worden geschakeld.

Pull-up register	Data register	Uitgangs stroom
1	0: LED aan, 1: LED uit	Instellen op basis stroom

- Een drukknop of een open-collector uitgang van bijvoorbeeld een opto-coupler kan zonder meer worden aangesloten op elk van de pinnen. De interne pull-up weerstand zorgt voor een gedefinieerd logisch niveau als de ingang niet geactiveerd is.

Pull-up register	Data register	Uitgangs stroom
0	1	nvt

- Een ingang van een (HC)TTL-compatibel logisch IC kan rechtstreeks worden aangestuurd door elk van de pinnen.

Pull-up register	Data register	Uitgangs stroom
0	1= logisch '1', 0 = logisch '0'	maximaal

- Een uitgang van een (HC)TTL-compatibel logisch IC kan rechtstreeks worden ingelezen door elk van de pinnen

Pull-up register	Data register	Uitgangs stroom
1	1	nvt

- Een NPN darlington transistor kan direct worden aangesloten op een pin naar keuze. Met de emitter van de transistor aan 0 V, zorgt de interne pull-up weerstand voor basis stroom. Met de transistor kan een zwaardere belasting geschakeld worden. Deze belasting wordt aangebracht tussen een externe voeding en de collector van de transistor.

Pull-up register	Data register	Uitgangs stroom
0	1: Transistor geleidt, 0: Transistor spert	maximaal

Uitgangen voor de ventilatoren

Op elk van deze uitgangen kan direct de basis van een NPN (darlington) transistor worden aangesloten. Er is geen serieweerstand nodig. Met de emitter van de transistor aan 0 V, zorgt een interne pull-up weerstand voor basis stroom. Met de transistor kan een zwaardere belasting geschakeld worden. Deze belasting wordt aangebracht tussen een externe voeding en de collector van de transistor. Voor een PC ventilator zal de externe voedingsspanning meestal +12 Volt zijn.

Het is ook heel goed mogelijk de uitgang, voorzien van een transistor, te gebruiken om bijvoorbeeld een gloeilampje te dimmen, een gelijkstroom motortje in snelheid te regelen, of een weerstand meer- of minder warm te laten worden.

De uitgang wordt geregeld doordat de periode dat deze actief is, over de totale periode van ongeveer 4ms, is in te stellen in 255 stapjes. Bij een lage instelling komen er relatief korte pulsjes uit, waardoor de ventilator langzaam zal draaien. Bij hogere instelling komen er bredere pulsen uit, zodat de ventilator sneller zal draaien. Bij stand 0 komen er helemaal geen pulsen uit, en bij stand 255 is de uitgang continue aan.

Specificaties I/O poorten

Parameter	Min	Max	Eenheid	Opmerkingen
Pull-up weerstand	8	25	kΩ	
Minimum stroom (instelling 0) vrije poort	0.1	0.3	mA	Vuit = 2.0 Volt
Maximum stroom (instelling 15) vrije poort	0.5	1.5	mA	Vuit = 2.0 Volt
Maximum ingangsstroom FAN uitgang	8	24	mA	Vuit = 2.0 Volt
Maximum ingangsstroom FAN uitgang	5		mA	Vuit = 0.4 Volt
Max. stroom voor alle pinnen van vrije poort		10	mA	Alle stromen gesommeerd
Threshold spanning Schmitt-triggers	45%	65%	V _{cc}	2.25 ... 3.25V bij V _{cc} =5V
Hysterese Schmitt-triggers	6%	12%	V _{cc}	0.3 ... 0.6V bij V _{cc} =5V
Cext uitgang		0.4	V	I=2mA
Cext uitgang		2.0	V	I=5mA

Voeding uit de USB bus

De USB bus kan voeding leveren voor de FAN Solution, en daarnaast voor eventuele extra hardware mits de totale stroombehoefte niet te groot is. De FAN Solution reserveert 100mA stroom uit de USB bus. Daarvan zijn 25mA voor de FAN Solution nodig, dus er blijven maximaal 75mA over voor additionele hardware.

Suspend

Als een PC in stand-by staat, moeten alle apparaten die erop zijn aangesloten, ook in een energiezuinige toestand staan. Deze 'suspend' mode wordt door de FAN Solution herkend doordat de USB bus geen dataverkeer meer heeft. De FAN Solution gaat dan zelf in een zuinige toestand, maar de eventuele extra hardware die ook zijn voeding uit de USB bus betreft, moet officieel ook in een low-power toestand gaan. De CEXT pin van de FAN Solution kan worden gebruikt om externe hardware aan- en uit te schakelen. CEXT is een open-drain uitgang. Als deze uitgang 'laag' is, is de USB bus in vol bedrijf en mag de externe hardware de gereserveerde 75mA gebruiken (maximaal). Als de uitgang open is, moet de externe hardware uitgeschakeld worden en mag geen stroom meer gebruiken.

Pas op: Als de PC uit stand-by komt, kan de CEXT uitgang eerst een paar keer kortstondig laag worden, alvorens definitief laag te blijven.

Communicatie met de PC

De FAN Solution communiceert met de PC via blokken van elk 8 bytes. De blokken hebben de volgende betekenis:

Byte	Vanuit PC	Betekenis
1	Schrijven	Stroominstelling P0.3 (4 meest significante bits) en P0.2 (4 minst significante bits)
2	Schrijven	Stroominstelling P0.1 (4 meest significante bits) en P0.0 (4 minst significante bits)
3	Schrijven	Instelling pull-up weerstanden poort 0 (een '0' betekent de pull-up inschakelen)
4	Schrijven	Data voor de vrije poort P0
5	Schrijven	Fan1 instelling
6	Schrijven	Fan2 instelling
7	Schrijven	Fan3 instelling
8	Schrijven	Fan4 instelling
1	Lezen	Data van de vrije poort
2	Lezen	Data van de vrije poort
3	Lezen	Data van de vrije poort
4	Lezen	Data van de vrije poort
5	Lezen	Data van de vrije poort
6	Lezen	Data van de vrije poort
7	Lezen	Data van de vrije poort
8	Lezen	Data van de vrije poort

Als er data binnenkomt verwerkt de FAN solution deze in de volgorde waarin de bytes staan in de bovenstaande tabel. De volgorde van verwerken is dus:

1. De stroom voor P0.3 wordt ingesteld
2. De stroom voor P0.2 wordt ingesteld
3. De stroom voor P0.1 wordt ingesteld
4. De stroom voor P0.0 wordt ingesteld
5. De pull-ups voor poort 0 worden ingesteld
6. De data voor poort 0 wordt geschreven
7. De instelling voor Fan1 wordt overgenomen
8. De instelling voor Fan2 wordt overgenomen
9. De instelling voor Fan3 wordt overgenomen
10. De instelling voor Fan4 wordt overgenomen

Tussen elke stap zit een aantal micro-seconden. Voor de meeste toepassingen zal het niet nodig zijn daar rekening mee te houden.

Als er data opgevraagd wordt leest de FAN solution de niveaus op de I/O pinnen acht maal snel achter elkaar uit. Tussen de lees acties zit steeds een paar micro-seconden. Als het ingangs signaal binnen die tijd niet stabiel is, bijvoorbeeld omdat een pin juist bezig is van toestand te veranderen, is dit dus terug te zien in de data die gelezen wordt. Voor de meeste toepassingen zal het niet nodig zijn daar rekening mee te houden, en is alleen het eerste byte interessant.

Software

Vanuit een eigen programma kan de FAN Solution worden bereikt door middel van de standaard CreateFile, ReadFile en WriteFile routines. In Delphi:

```
var
  HdDevice: THandle;
  BytesWritten: DWORD;
  Success: Boolean;
  Buffer: Array[0..8] of Byte;
  BufSize: Integer;

// handle naar het device openen
HdDevice := CreateFile( DeviceName,
                      GENERIC_READ or GENERIC_WRITE,
                      FILE_SHARE_READ or FILE_SHARE_WRITE,
                      Nil, OPEN_EXISTING, 0, 0);

// handle gebruiken voor schrijf actie
Buffer[ 0 ] := 0; // Dit altijd op 0 zetten
Buffer[ 1 ] := $FF; // Instellen stroom P03 en P02
Buffer[ 2 ] := $FF; // Instellen stroom P01 en P00
Buffer[ 3 ] := $00; // Instellen pull-ups poort 0
Buffer[ 4 ] := $FF; // Data voor poort 0
Buffer[ 5 ] := $00; // Instellen snelheid fan1
```

```

Buffer[ 6] := $56; // Instellen snelheid fan2
Buffer[ 7] := $80; // Instellen snelheid fan3
Buffer[ 8] := $FF; // Instellen snelheid fan4

Success := WriteFile( FHandle, Buffer, SizeOf( Buffer), BytesWritten, NIL);

// handle gebruiken voor lees actie
BufSize := SizeOf( Buffer);
ReadFile( HidDevice, Buffer, BufSize, BytesRead, nil;

// Buffer[ 1] bevat nu de data die werd gelezen van poort 0

// handle weer sluiten
CloseHandle( HidDevice);

```

In Visual Basic of C++ zijn soortgelijke routines te gebruiken. Uiteraard is dit maar summere code, waar nog de nodige fout afhandeling aan dient te worden toegevoegd. Ook is de naam van het device bekend verondersteld. Die naam wordt echter door Windows aan het device gegeven op het moment dat dit wordt aangesloten. Dat is dus een steeds veranderend gegeven, en de naam die gebruikt moet worden moet dan ook eerst worden opgezocht in de lijst met aangesloten devices, die door Windows wordt bijgehouden. Een voorbeeld van hoe om hier te behandelen, maar voldoende informatie hierover kunt u vinden op de website: www.elomax.nl Daar zijn ook kant- en klare componenten te vinden voor Delphi, die het programmeren van de FAN Solution vanuit Delphi of C++ Builder sterk vereenvoudigen.