

Web Server Unit (WSU) - Preliminary

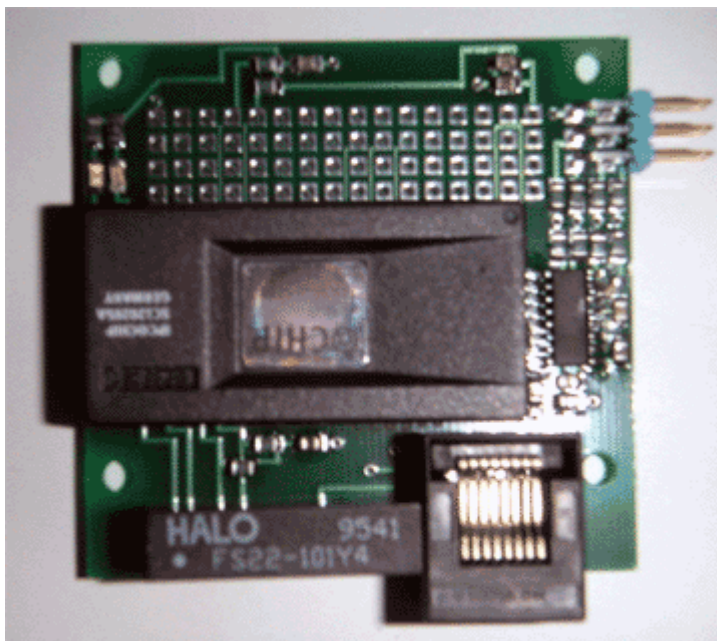
De MultiMotions Web Server Interface (WSI) is een piggyback bordje met een complete Web Server. Deze Web Server heeft een direct interface met het Universal Processor Board (UPB) en kan hierdoor een directe LAN verbinding met de microcontroller implementeren.

MultiMotion's UPB is een flexibel multi-purpose board met een busstructuur die het mogelijk maakt een verscheidenheid van apparaten aan te sluiten.

Met de WSI kunnen gebruikers van een Local Area Network (LAN) of Internet gebruikers in het algemeen een TCP/IP verbinding maken met een microcontroller op de UPB. Een standaard Web Browser kan op die manier worden gebruikt om toegang tot de Web Server en alle functionaliteit van de UPB te krijgen.

De WSI ondersteunt HTML, HTTP, DHCP, FTP en SMTP. De WSI heeft twee seriële poorten, die worden gebruikt als console verbinding en als communicatie tussen de UPB en de Web Server. De WSI kan ook via het I2C interface met de UPB communiceren.

De Web server heeft de beschikking over 256K Ram en 256K Flash geheugen, dat als RAM disk is ingericht en gebruikt wordt om web pagina's op te slaan.



De unit wordt geplaatst als piggyback bordje op de UPB. Communicatie met de UPB vindt plaats via het I2C interface of via RS232.

De unit bevat twee RS232 poorten en een UTP aansluiting.

HTML Interface

Een generieke tag binnen de embedded controller maakt het gemakkelijk om een direct interface met de UPB te realiseren zonder dat de WSI hoeft te worden geprogrammeerd. Hoewel dit uiteraard mogelijk is, wordt zoveel mogelijk geprobeerd om de opzet van het systeem zo te maken dat zonder programmeren gebruik kan worden gemaakt van het interface. Met deze speciale tag is het mogelijk om absolute of relatieve geheugenposities van de UPB processor te benaderen.

Door een adres, lengte en een datastring te gebruiken, kan informatie met de controller worden uitgewisseld door alleen een HTML pagina te maken en deze in het geheugen van de

WSI op te slaan. De WSI kan worden benaderd via een Web Browser, via TelNet of een terminal connectie via de RS232 poort.

TCP/IP Interface

Naast het HTML interface kan een aangesloten computer HTTP commando's sturen naar de Web Server direct het geheugen van de embedded controller benaderen met standaard TCP/IP interfaces.

Interfaces

De WSI bevat twee standard RS232 poorten die primair worden gebruikt als console en om wijzigingen in de BIOS aan te brengen. De poorten kunnen ook direct vanuit de Web Server worden gebruikt maar moeten apart worden geprogrammeerd als dit gewenst is.

Het standaard TCP/IP interface gebruikt een 10BaseT interface met een standaard UTP kabel. De processor wordt via het bus interface op de UPB aangesloten. Communicatie tussen beide processoren vindt plaats via RS232 of I2C.

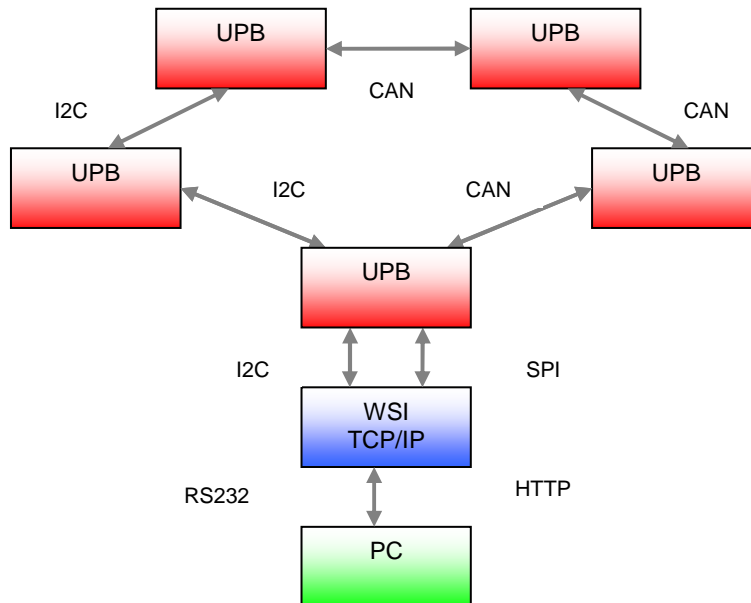
Software Interface

Het interface tussen de UPB en de WSI houdt rekening met de volgende factoren:

De omzetting van de CAN messages naar de PIC processor wordt zoveel mogelijk zonder conversie gerealiseerd om de uitwisseling van data tussen een PC en het netwerk te vereenvoudigen.

Een interface dat via tags de inhoud van HTML pagina's kan beheren zonder dat er op de WSI processor extra programma's moeten worden geschreven.

Het uitgangspunt is dat er generiek interface wordt ontwikkeld waarmee alle communicatie tussen de PIC, de WSI en een PC kan worden verricht. Hierbij geldt het volgende schema:



Communicatie

Communicatie tussen UPB's kan in principe gebeuren via RS232, I2C en CAN.

Communicatie tussen de WSI en de UPB gebeurt via RS232 of I2C.
Communicatie tussen de WSI en een PC gebeurt via HTTP over TCP/IP maar kan ook via RS232 plaatsvinden.

UPB/UPB communicatie

In de huidige opzet is de communicatiemethode tussen UPB's via RS232 of I2C. Er is een multidrop faciliteit aanwezig, die echter inhoudt dat de UPB's elkaar moeten pollen om van de gemeenschappelijke RS232 bus gebruik te kunnen maken. Daarnaast heeft iedere UPB een tweede RS232 die eveneens gebruikt kan worden.

De uiteindelijke situatie is dat alle UPB's daarnaast een USB of een CAN aansluiting hebben. USB is bedoeld voor directe communicatie met een PC, CAN is bedoeld als communicatie voor UPB's onderling. USB zal pas in een later stadium worden aangepakt. Voor USB is het wachten op de aangekondigde PCI versie met een CAN interface, voor CAN is er een piggyback bordje dat een CAN interface bevat. Daarnaast is er ook een PIC versie gepland die CAN ondersteunt.

WSI/UPB communicatie

De communicatie tussen de WSI en de UPB kan in principe via RS232 en I2C worden gedaan.

De primaire RS232 poort op de Beck is gereserveerd voor het terminal interface en voor het low-level programmeren van de BIOS.

I2C heeft als voordeel dat er slechts 2 pennen van de UPB voor gebruikt worden. Het nadeel is dat de snelheid te wensen overlaat en dat er incidentele hardware problemen tussen WSI en PIC processoren worden gemeld.

De WSI heeft ook een parallel interface, maar dit vereist de gehele D poort plus 3 A/D pennen. Om deze reden wordt dit interface niet gebruikt en is vrij voor gebruik door de gebruiker. De parallelle output van de WSI is wel beschikbaar op het bordje maar wordt niet naar de bus gebracht.

WSI/PC communicatie

Alle communicatie tussen een PC en de WSI vinden plaats ofwel via het terminal protocol of direct via RS232 ofwel via TCP/IP.

De PC kan rechtstreeks HTML data sturen, die door de WSI rechtstreeks aan de UPB moet worden doorgezonden. Daarnaast moet er via een Browser met de WSI kunnen worden gecommuniceerd, waarbij er tags moeten worden gedefinieerd, waarmee de WSI voor de gebruiker transparant met de UPB gegevens uitwisselt.

Hiervoor wordt het volgende protocol voorgesteld:

- < I2C >** Deze tag zorgt ervoor dat vanaf nu alle communicatie via de I2C bus verloopt.
- < SER >** Deze tag zorgt voor seriële communicatie met de COM poort van de WSI naar de 1^e of 2^e COM poort van de UPB.
- < PIC >** Deze tag verzorgt de feitelijke communicatie. Via commando's in deze tag kan de WSI data tussen velden onderling uitwisselen. De commandostructuur wordt onderstaand uitgewerkt.

Een < PIC > tag dient om een invoer/uitvoerveld in een HTML pagina te specificeren, waarmee via een Web Browser rechtstreeks contact met de PIC kan worden gelegd op een transparante manier. De maker van de web pagina hoeft verder niets anders te doen, dan de definitie van een veld op te nemen. Hierdoor is er geen programmering van de WSI nodig om een interface te kunnen gebruiken.

HTML Tag commando's

Een <PIC> tag definieert een edit box of een text veld in een pagina, die automatisch gekoppeld wordt aan data in de PIC processor. Om dit te kunnen realiseren wordt de volgende commando structuur voorgesteld:

F BB AA LL C DD

- F** *Format.* Hiermee wordt aangegeven of de gegevens in de PIC een String, Decimaal of Hexadecimaal veld zijn (S, D, X) Een String wordt ongewijzigd doorgegeven. Een D veld wordt als decimale waarde doorgegeven. Een hexadecimaal veld wordt als 2 character velden per byte doorgegeven.
- BB** *Bank adres.* Dit is een Hex waarde, die het bank adres in de PIC aangeeft.
- AA** *Adres.* Dit is een Hex waarde die het adres binnen de aangegeven bank aangeeft/
- LL** *Lengte.* Het aantal bytes dat moet worden uitgewisseld. Bij Strings vindt geen conversie plaats. Bij D en X moeten de waarden worden omgevormd tot strings, die door de PIC direct als ASCII waarden kunnen worden verwerkt. Alle conversie vindt in de Beck processor plaats.
- C** *Command.* De waarde kunnen worden gelezen (R), geschreven (W) of worden schoongemaakt (C). Verder kan de PIC processor worden ge-reset (X).
- DD** *DataString* met data, die door de twee processoren worden uitgewisseld. De conversie zorgt ervoor dat de ingetikte gegevens in een veld automatisch worden geconverteerd. Gegevens die uit de PIC worden opgevraagd worden automatisch geconverteerd.

Adres onafhankelijk protocol

Het voorgaande protocol is adres afhankelijk. Dit geeft een grote mate van vrijheid, doordat de maker van de pagina alle adressen in de PIC kan bereiken. Hiermee kan in principe het gehele geheugen worden uitgelezen.

Het houdt echter wel in dat alle aanpassingen in het PIC programma die leiden tot adreswijzigingen een aanpassing van de HTML pagina vereist. Om dit te voorkomen, kan ook voor een adres onafhankelijk protocol worden gekozen. Dit wordt gedaan door de Bank code van 00-FF te vervangen door de letters FD voor Field Definition. De tweecijferige code die hierop volgt wordt dan niet beschouwd als een adres maar moet door de PIC worden beschouwd als een veldcode en de PIC dient dan zelf het juiste adres hiervoor aan te houden.

De commandoregel wordt dan als volgt:

F FD AA LL C DD

Hoewel de lengte strikt genomen niet meer nodig is, wordt deze voor de compatibiliteit nog steeds opgenomen. De PIC begrenst echter het aantal tekens dat wordt verzonden.

Piggyback Modules zijn kleine, standaard bordjes, die via de UPB bus gekoppeld kunnen worden met de UPB en andere modules. De standaard stackable 32 pins bus wordt in alle modules doorgevoerd zodat meerdere bordjes kunnen worden gebruikt.

Pic	ID	Naam	Omschr	Conn	Prijs €
	WSI	Webserver Interface	Via een embedded web server kan de processor een externe internet aansluiting maken en communiceren via de parallelle poort. De embedded controller kan HTML pages bevatten en gegevens verzamelen en deze uitwisselen met het processor bordje. De webserver bevat zelf een 80186 processor met 512K ram en 512K flash geheugen, dat als RAM disk kan worden gebruikt voor het opslaan van files zoals HTML pages.	RE0-2	

Alle PiggyBack bordjes sluiten aan op de standaard 32 pins UPB bus. Voor de compleetheid is onderstaand een overzicht van de busaansluitingen opgenomen.

Bus Pen	Bus ID	Proc Pen	Omschr	Alt Functie	Bus Pen	Bus ID	Proc Pen	Omschr	Alt Functie
1	Vcc		+5v Reg		2	AD0	RA0		
3		RA4	T0CKI – Timer 0		4	AD1	RA1		
5	Int	RB0	Int		6	AD2	RA2	Vref-	
7	LD1	RB1		Led Red	8	AD3	RA3	Vref +	
9	LD2	RB2		Led Yel	10	AD4	RA5	SS	Slave Select
11	LD3	RB3		Led Grn	12	AD5	RE0		PSP RD
13	SW0	RB4		Switch 0 / RB6 = SW2	14	AD6	RE1		PSP WR
15	SW1	RB5		Switch 1 / RB7 = SW3	16	AD7	RE2		PSP CS
17	TIM1	RC0	T1OSO/T1CKI	Timer1 I/O	18	RD0	RD0		PSP0
19	PWM2	RC1	T1OSI/CCP2	Timer1 I / Capt2 / PWM2	20	RD1	RD1		PSP1
21	PWM1	RC2	CCP1	Capt1 / PWM1	22	RD2	RD2		PSP2
23	SPIC	RC3	SCK/SCL	SPI / I2C Clk – USB Clk	24	RD3	RD3		PSP3
25	SPII	RC4	SDI/SDA	SPI / I2C Data – D-USB	26	RD4	RD4		PSP4
27	SPIO	RC5	SD0	SPI Data out – D+ USB	28	RD5	RD5		PSP5
29	nc				30	RD6	RD6		PSP6
31	Vss		Ground		32	RD7	RD7		PSP7

Voor meer informatie neem contact op met:

MultiMotions
 Waterlandlaan 120
 1441RW Purmerend
 0299-471617
 e-mail: info@lithp.nl
 website: www.multimotions.com

MultiMotions is een handelsmerk van Lithp Systems bv