

Řídicí jednotka s mikrokontrolérem MC68376

Pavel Píša (pisa@cmp.felk.cvut.cz)

12. července 2001

1 Úvod

Cílem projektu bylo vybrat a zavést vhodnou mikroprocesorovou platformu, která by nahradila dřívější osmibitové architektury, byla vhodná pro řízení složitějších přístrojů a středně složitých robotických systémů. Důležitým kritériem byla otevřenost architektury, snadná dostupnost vývojových a ladících prostředků a podpora ze strany real-time operačních systémů.

Článek předkládá jedno z možných řešení vytčeného cíle včetně důvodů, které vedly k výběru navrženého řešení. Při řešení projektu bylo nutné vyřešit množství dílčích problémů. Například problém ladění mikroprocesorového systému v cílovém zapojení. Návod a zdokonalená verze použitého debuggeru jsou k dispozici na autorových webových stránkách. Cena vývojového prostředí i použitých součástek byla také podstatným kritériem výběru architektury a návrhu hardware. Návrh se ovšem nesnaží o snížení ceny na úkor průmyslové použitelnosti výsledného řešení.

2 Výběr mikrokontroléru a kompilátoru

Dostupnost kompilátoru a celého vývojového řetězce je jedním z významných faktorů při výběru procesorové architektury. Většina výrobců sice nabízí levnou variantu vývojového prostředí pro školní a nekomerční využití, ale po bližším prozkoumání vyjde většinou najevo, že se jedná o demoverze nebo omezené verze, případně staré verze bez technické podpory. Ceny plné verze vývojových prostředí bývají velmi vysoké a o harwarovém emulátoru si v mnoha případech může český student nebo malá česká firma pouze snít. Protože se v dnešní době stále více uplatňují i jiné hostitelské architektury než DOS a Windows, byla i přenositelnost křížového překladače podstatnou podmínkou. Podpora širokého spektra architektur může být z hlediska přechodu na výkonnější architektury v budoucnosti též výhodná.

Z výše uvedených požadavků vyplývá, že nejvhodnějším dostupným autorovi známým kompilátorem je GNU Compiler Collection (GCC). Tento otevřeně vyvíjený a dostupný

kompilátor je k dispozici pro velké množství cílových i hostitelských architektur. Do optimalizace kompilátoru GCC přispívá velmi široký okruh vývojářů i firem (Intel, RedHat, Cygnus a další). Vývojový řetězec GNU je velice často i základem prestižních komerčních vývojových prostředí. Protože systém programů GNU vznikal jako lepší náhrada za UNIX (GNU is not UNIX), jsou hostitelské systémy UNIXového typu nejlepší pro jeho přípravu a provozování (GNU kompilátory lze sice také provozovat na platformách Windows 95/NT, ovšem pro přípravu křížového překladače není tato architektura nejvhodnější). Z výše uvedeného důvodu a z důvodu ceny hostitelského operačního systému a počítače byl zvolen systém Linux na platformě PC. Kvalita výsledného kódu generovaná kompilátorem GCC bývá většinou uspokojivá a připravovaná verze 3.0 slibuje výrazné vylepšení optimalizací výsledného kódu.

Výběrem kompilátoru se i zúžil prostor pro výběr mikrokontroléru. Z architektur podporovaných GCC jsou jen některé vhodné pro embedded aplikace. Uvažované výhody a nevýhody jednotlivých architektur :

alpha velmi výkonná a elegantní architektura, při jejím nasazení v embedded aplikacích je nevýhodou větší paměťová náročnost a nedostupnost čipů s vhodnou kombinací integrovaných periférií, budoucnost této architektury není též zcela jasná

arm architektura často nasazovaná v embedded aplikacích především v oblasti sítí, existují i verze s integrovanými perifériemi, ale jsou opět zaměřené spíše pro palmtopy

c4x signálové procesory TS320C3x/C4x, výkonné DSP procesory s plovoucí aritmetikou, jsou vhodné pro výpočty regulátorů a pohybů robotů a zpracování složitých signálů. Nejsou vhodné pro komunikace a uživatelský komfort přístrojů

h8300 architektura H8 a především její výkonnější verze H8S od společnosti Hitachi je vhodná pro řídicí aplikace, existují verze s velkou kapacitou paměti FLASH na čipu a s množstvím periférií

i386 existuje velké množství embedded verzí, ovšem většinou se jedná o integraci architektury PC, nejsou integrovány převodníky, rychlé vstupy a výstupy a časovací subsystemy

m68k architektura nejčastěji nasazovaná v složitějších řídicích systémech, verze s integrovaným velkým množstvím periférií pro řídicí a průmyslové aplikace 683xx, výkonné procesory 68060, architektura ColdFire pro spotřební průmysl a síťové komponenty

mips existují i verze pro embedded aplikace

rs6000 původní architektura Power od IBM je pouze pro sálové počítače, ale odvozený procesor PowerPC byl úspěšně integrován s perifériemi společností Motorola a je považován za nástupce architektury m68k.

sh SuperH - následník h8300 od společnosti Hitachi. Zatím více zaměřený perifériemi pro handheld zařízení

sparc existuje i ve verzích pro embedded aplikace

Po srovnání byla vybrána osvědčená architektura m68k. Mikrokontrolér MC68376 obsahuje nejvhodnější integrované periferie ze všech členů rodiny 683xx. Dvě další architektury byly vybrány za zajímavé pro budoucí projekty. Hitachi H8S byla již od té doby v jiném projektu úspěšně použita a práce s architekturou PowerPC MPC555 jsou v přípravné fázi.

Jedním z podstatných důvodů výběru mikrokontroléru z rodiny 683xx byla i podpora ladění v cílové aplikaci integrovaná přímo na čipu (BDM rozhraní) a naděje na získání a vybudování kvalitního debuggeru z dostupných zdrojů.

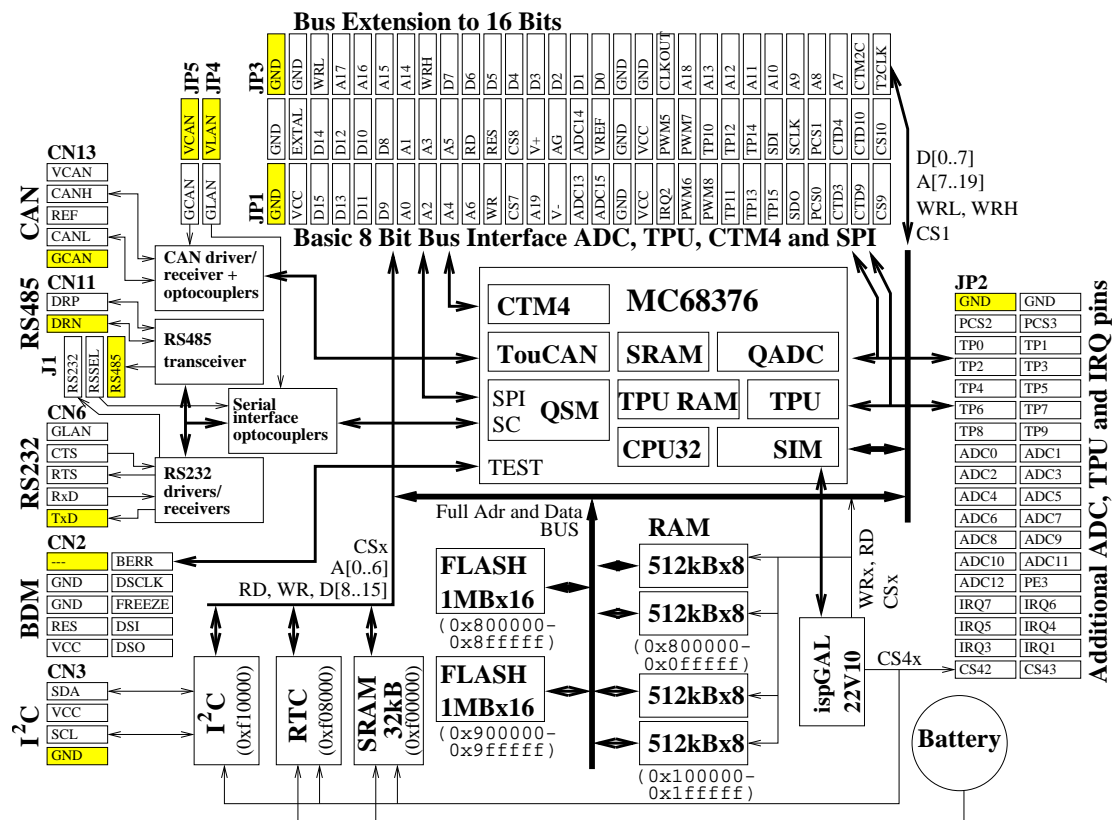
3 Vývojové prostředí

Na Linuxovém hostitelském počítači byl připraven křížový kompilátor pro procesory Motorola m68k. Pro první pokusy byl zakoupen za 20 tisíc korun původním majitelem již nepoužívaný vývojový kit Motorola EVK332. Na něm byla ověřena funkčnost kompilátoru GCC a nástrojů pro práci s assemblerem a objektovými soubory BINUTILS. Kit také posloužil pro vybudování debuggeru. Knihovny pro podporu ladícího BDM (Background Debug Mode) rozhraní mají dlouho historii. První verze pro DOS a později pro Linux napsal vývojář Motoroly Scott Howard. Další, kteří přispěli k vývoji a rozšíření debuggeru byli M. Schraut (Linux BDM driver), Gunter Magin (úprava GNU Debuggeru), W. Eric Norum (verze s podporou ColdFire). O úpravy a dokumentaci k původní verzi pro rodinu 683xx se nyní stará autor. Podrobný popis rozhraní a vše potřebné pro kompilaci debuggeru lze nalézt na adrese <http://cmp.felk.cvut.cz/~pisa/>. Současná podporovaná verze debuggeru je GDB-5.0 v konfiguraci `-target=m68k-bdm-coff` s podporou pro další objektové formáty `-enable-targets=m68k-linux-elf,m68k-coff,m68k-a.out-linux`. Kompilátor GCC byl úspěšně používán od verze 2.8.x až po současnou verzi 2.95.3 v konfiguracích `-target=m68k-coff` a `-target=m68k-rtms`. Podrobnější výklad přípravy křížového překladače je na výše uvedené stránce.

Dalším důležitým prvkem vývojového prostředí je operační systém pro cílovou architekturu a podpůrná knihovna jazyka C a C++. Část podpůrných knihoven je součástí distribuce kompilátoru. Hlavní použitá C knihovna je volně šiřitelná knihovna NewLib od společnosti Cygnus, která může být používána po dopsání několika málo funkcí i na platformách bez operačního systému. Pro aplikace vyžadující real-time operační systém byla vybrána real-time exekutiva RTEMS. Volně šiřitelný i použitelný systém se zdrojovými kódy, původně napsaný pro americkou armádu. V současné době je testována vývojová verze RTEMS 4.5 v konfiguraci `-target=m68k-rtms` využívající modernější objektový formát ELF.

4 Návrh procesorové desky

Z již popsaných důvodů byl vybrán mikrokontrolér Motorola 68376, mezi jeho hlavní výhody patří 32 bitové jádro odpovídající procesoru 68010 s některými rozšířeními z proce-



Obrázek 1: Blokové schéma desky s mikrokontrolérem MC68376

sořu 68020. Vyvedená je 16-ti bitová datová a 24-bitová adresová sběrnice. Na čipu je integrován řadič sběrnice CAN (TouCAN), 16-ti kanálový 10-bitový A/D převodník (QADC), 16-ti kanálový časovací koprocetor (TPU), dalších až 8 PWM výstupů, ze kterých 4 mohou sloužit jako dvojité jednotky komparátorů nebo záchytů (CTM4), řadič SPI rozhraní s možností až 16 automatických přenosů (QSM) a asynchronní sériové rozhraní UART. Mikrokontrolér navíc obsahuje bloky 4 kB a 3.5 kB RAM.

Navržená deska s označením MO_CPU1 obsahuje navíc rozhraní IIC, galvanické oddělení pro sběrnici CAN a opticky oddělené budiče RS232/485. Pro uložení zavaděče a aplikací slouží 1 nebo 2 MB paměti FLASH v 16-ti bitové konfiguraci. Paměť FLASH je inicializována a může být kdykoliv přeprogramována i přes rozhraní BDM. Data a další kód může být uložen v rychlé paměti SRAM o velikosti až 2 MB. Data, která mají být uchována i přes dobu vypnutí systému, lze uložit do paměti FLASH nebo do 32 kB baterií zálohované paměti CMOS SRAM. Systém je dále doplněn obvodem reálného času. Blokové schéma desky je na obrázku 1. Deska byla realizována s oboustranným osazením technologií SMD na čtyřvrstevném plošném spoji s rozměry 115 mm x 80 mm. Úplné schéma desky i výkresy rozmístění součástek byly uvolněny pro neomezené použití a budou zájemcům dodány.

Vzájemné rozmístění konektorů na obrázku 1 odpovídá rozmístění konektorů na navržené desce. Propojovací konektory jsou umístěny po horní a pravé straně. Na rozdíl od tradičního

rozmístění po obou delších stranách desky toto řešení umožňuje propojení sběrnic na další desky stejného nebo podobného rozměru, jako je rozměr procesorové desky, a nebrání vyvedení vstupně výstupních konektorů a galvanickému oddělení vedenému středem rozšiřujících desek. V levé části jsou vyvedeny konektory pro komunikace CAN a RS232/RS485. Budiče obou komunikací mohou být napájeny z vzájemně nezávislých zdrojů.

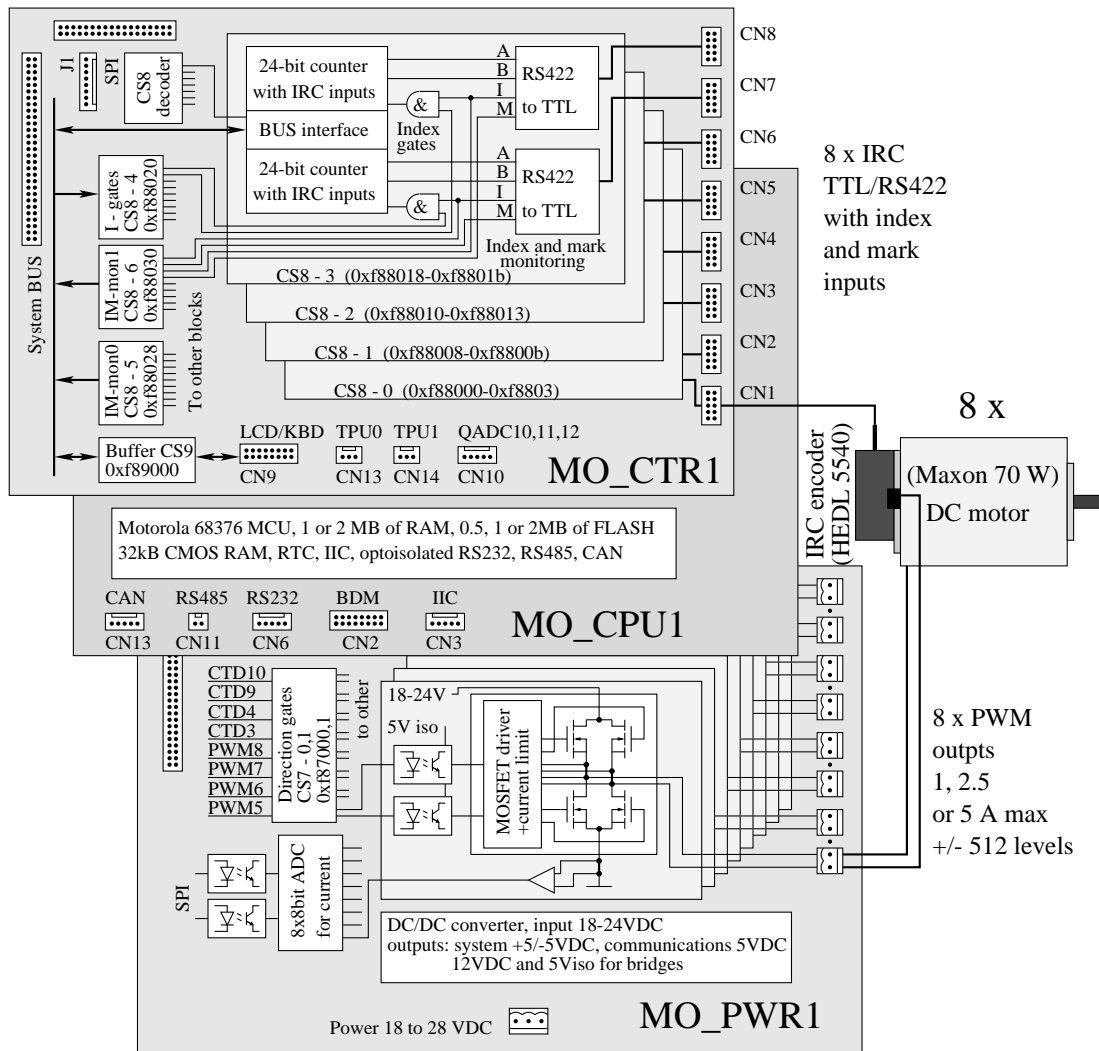
Mapování pamětí do paměťového prostoru je plně konfigurovatelné, protože výběrové signály jsou odvozeny od programovatelných chip-select generátorů mikrokontroléru. Zvolené mapování pamětí i periférií je v obrázku vyznačeno v závorkách. Pro ušetření počtu nutných signálů chip-select pro dekodování způsobu přístupu k pamětem byla logika výběru čtení, zápisu horního a dolního byte a výběru jednotlivých periférií na desce realizována v aplikaci programovatelným logickým obvodem ispGAL22V10.

5 Servo systém

Pro řízení manipulátorů a polohovacích systémů byla procesorová deska MO_CPU1 doplněna deskou MO_CTR1 s dekodéry pro osm IRC vstupů a deskou MO_PWR1 s výkonovými výstupy. Na desce s IRC vstupy byly kromě přijímačů diferenciálních signálů normy RS422 přidány samostatné dekodéry s 24 bitovými čítači. Osm dekodérů a 16-ti bitových čítačů lze sice realizovat jednotkou TPU mikrokontroléru, ale časová analýza činnosti mikrořadiče TPU ukazuje, že by při vyšších rychlostech otáčení motorů mohlo docházet ke ztrátám polohy. Pro pomalejší rychlosti otáčení a pouze čtyři kanály IRC by bylo možné návrh zjednodušit a využít vstupů TPU. Rozšíření čítačů na 24 nebo 32 bitů je pro reálné rychlosti snadno realizovatelné v obslužném software.

Výkonové budiče PWM jsou řízeny přímo výstupy z rychlé časovací jednotky CTM4 mikrokontroléru. Aby nebylo nutné řídit 16 půlmůstků šestnácti signály, prochází osm šířkově modulovaných logických signálů hradlovací logikou, která řídí polaritu (směr otáčení motoru) každého výstupu. Informace o požadovaných polaritách jsou mikrokontrolérem zapisovány do dvou osmibitových registrů. Výstupní budiče v můstkovém zapojení obsahují nadproudovou ochranu na horních tranzistorech půlmůstků. Celkový proud můstky je snímán, zesílen a převeden osmibitovým AD převodníkem připojeným přes optoizolátory k sběrnici SPI mikrokontroléru. Blokové schéma sestavy všech tří desek se nalézá na obrázku 2.

Software je interně navržen modulárně pro volitelný počet řízených os. Navržený hardware omezuje maximální použitý počet os na osm, což odpovídá i výkonu mikroprocesorového systému, který je při nejnáročnějších fázích operací hledání počáteční polohy a koordinovaných pohybů na současně na všech osách špičkově zatížen na 50%. Polohová regulace a výpočty požadované polohy probíhají se vzorkovací frekvencí 1 kHz. Činnost každé osy je určena informacemi uloženými v její stavové struktuře. Zde jsou též uloženy ukazatele na funkce, které obstarávají načtení aktuální polohy z hardware, výpočet regulačního zásahu, výstup zásahu na výkonové výstupy, monitorování polohy pro účely ladění a přípravu nové požadované polohy. Jednotlivé funkce lze pro každou osu za běhu individuálně povolovat, zakazovat, parametrizovat a i měnit, pokud jsou dodrženy určité



Obrázek 2: Navržený servosystém

podmínky.

V současné době je k dispozici široce parametrizovatelný PSD regulátor, funkce pro čtení polohy z externích IRC registrů nebo TPU koprocessoru a nastavování PWM výstupů přes CTM4 a vnější hradlovací registry. Pro generování požadované polohy je možné používat množství generátorů trajektorií. Základní jsou:

- generátor pro přejetí nastavenou maximální rychlostí do cílové polohy s lichoběžníkovým profilem rychlosti
- generátor pro nalezení referenční nulové polohy s množstvím různých režimů
- generátor konstantní rychlosti otáčení s plynulým přechodem na požadovanou rychlost a zastavením

Systém dále umožňuje dynamicky definovat skupinu os (v případě potřeby lze rozšířit i na více skupin), které mají provádět pohyby koordinovaně. Skupina umožňuje přesuny koordinované přesuny mezi zadanými body v prostoru zatím pouze s konstantními poměry rychlostí mezi motory v každém úseku. Systém je schopen optimalizovat napojení více zadaných úseků tak, aby zpomalení při průjezdu hran bylo co nejmenší. V budoucnosti se počítá s koordinovanými pohyby definovanými spline funkcemi.

6 Závěr

Projekt vedl úspěšnému návrhu řídicího systému pro robotické a polohovací aplikace. Dvě jednotky jsou nyní používány pro řízení výukových manipulátorů (robotů BlueBot) na katedře kybernetiky ČVUT FEL, další systém modifikovaný pro výkony motorů do 1 kW je nyní testován s mechanikou BOSH s kinematikou SCARA. Projekt zároveň ukazuje, jakým způsobem lze postupovat při návrhu složitějších řídicích jednotek bez nutnosti velkých investic do komerčních vývojových prostředků. Veřejně publikované zkušenosti s přípravou vývojového prostředí pro mikrokontroléry Motorola 683xx a archivy s potřebnými úpravami software již v noha případech posloužily a dalším zájemcům mohou pomoci k přechodu z osmibitových mikrokontrolérů na výkonnější architektury.

Navržené jednotky jsou již využívány, ale tím se pouze otevírá množství zajímavých oblastí pro další projekty. Například vývoj software do jednotek i do připojeného počítače pro nahrazení komunikace po lince RS-232 za komunikaci po sběrnici CAN s protokolem CanOPEN. Další projekt se zabývá návrhem modulu pro připojení mikrokontrolérů k síti Ethernet s protokolem TCP/IP. Pro další na katedře řídicí techniky vyvíjené zařízení se předpokládá využití výkonnějších mikrokontrolérů s architekturou PowerPC.