

KitLoader

UŽIVATELSKÁ PŘÍRUČKA

Příručka uživatele a programátora



SofCon[®] spol. s r.o.
Střešovická 49
162 00 Praha 6
tel/fax: +420 220 180 454
E-mail: sofcon@sofcon.cz
www: <http://www.sofcon.cz>

Informace v tomto dokumentu byly pečlivě zkontrolovány a SofCon věří, že jsou spolehlivé, přesto SofCon nenese odpovědnost za případné nepřesnosti nebo nesprávnosti zde uvedených informací.

SofCon negarantuje bezchybnost tohoto dokumentu ani programového vybavení, které je v tomto dokumentu popsáno. Uživatel přebírá informace z tohoto dokumentu a odpovídající programové vybavení ve stavu, jak byly vytvořeny a sám je povinen provést validaci bezchybnosti produktu, který s použitím zde popsaného programového vybavení vytvořil.

SofCon si vyhrazuje právo změny obsahu tohoto dokumentu bez předchozího oznámení a nenese žádnou odpovědnost za důsledky, které z toho mohou vyplynout pro uživatele.

Datum vydání: 29.03.2006

Datum posledního uložení dokumentu: 29.03.2006

(Datum vydání a posledního uložení dokumentu musí být stejné)

Upozornění:

V dokumentu použité názvy výrobků, firem apod. mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

Obsah :

1.	O dokumentu.....	4
1.1.	Revize dokumentu.....	4
1.2.	Účel dokumentu	4
1.3.	Rozsah platnosti	4
1.4.	Související dokumenty	4
2.	Termíny a definice	4
3.	Úvod.....	5
3.1.	Vývojový nástroj.....	5
3.2.	Možnosti/schopnosti	5
3.3.	Co je nového oproti ReTOS Debugger	5
4.	Instalace a spuštění	6
4.1.	Instalace	6
4.2.	Hostitelský operační systém.....	6
4.3.	Spuštění	6
5.	Vzhled a základní ovládání	8
5.1.	Horní roletové menu	8
5.2.	Panel nástrojů	9
5.3.	Stavový řádek.....	9
6.	Práce s KitLoaderem.....	10
6.1.	Termíny, pojmy a názvosloví.....	10
6.2.	Vytváření a práce s projekty	10
6.3.	Vytváření paměťového rozložení.....	11
6.4.	Nahrání aplikace do systému KIT a její spuštění.....	13
7.	Nastavení komunikace s KITem.....	13
8.	Parametry příkazové řádky programu KitLoader	14
8.1.	Zadávané parametry	14
9.	Setup (konfigurace) KITu.....	14
9.1.	Setup pro Kit386EXR	15
9.2.	Setup pro Kit188ER.....	16
10.	Identifikace BIOSu KITu.....	17
10.1.	KitV40.....	17
10.2.	Kit386EXR.....	18
10.3.	Kit188ER	18
11.	Uživatelé, přihlašování a přístupové úrovně.....	18
12.	Čtení a zápis uživatelských .BIN souborů do paměti KITu	19
12.1.	Vyčítání obsahu paměti do binárního souboru	19
12.2.	Zápis binárních dat do paměti KITu	20
13.	Zobrazení a editace části paměti KITu	20

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
01.00	01.XX	Bin	10.09.2004	První vydání
01.01	01.XX	Net	11.05.2005	Konfigurace Kit188ER
01.02	02.XX	Bin	10.07.2005	
02.00	02.XX	Wil	29.03.2006	Kompletní přepracování celého dokumentu

1.2. Účel dokumentu

Tento dokument slouží jako popis programu KitLoader používaného pro vytváření binárních souborů spustitelných aplikací a pro nahrávání těchto binárních souborů do paměti řídicích systémů Kit (KitV40, Kit386EXR, Kit188ER).

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu není nezbytně nutné číst žádný další manuál, ale je potřeba se orientovat v používání programového vybavení SofCon pro programování řídicích systémů KIT. To Vám usnadní následující dokumenty:

- [1] „Termíny a definice“ – Termíny.pdf
- [2] „Programujeme řídicí systémy Kit s BP7 a o.s. Retos“ – Zas_s_Kit_v_BP.pdf
- [3] „Přehled adresního prostoru“ – KitAdr.pdf
- [4] „Retos Debugger, Integrovaný ladící nástroj“ – RTD.pdf

2. Termíny a definice

Některé používané termíny a definice jsou popsány v samostatném dokumentu „Termíny a definice“, který obsahuje základní terminologii používanou v SW firmy SofCon.

3. Úvod

3.1. Vývojový nástroj

KitLoader je vývojový nástroj sloužící ke konfiguraci řídicích jednotek KIT a nahrávání řídicích aplikací. Program KitLoader umožňuje po sériové komunikační lince nahrát do paměti EEPROM řídicího systému aplikaci vytvořenou ve vývojovém prostředí Borland Pascal 7.0. Vytvořený aplikační .EXE soubor je možné pomocí KitLoaderu převést do podoby jednotlivých binárních souborů, které jsou určeny pro naprogramování paměti EPROM/FLASH.

3.2. Možnosti/schopnosti

Hlavní činnost programu je rozdělena do dvou bloků:

- 1) Načtení .EXE souboru řídicí aplikace, rozmístění jednotlivých paměťových modulů, vygenerování příslušných .BIN souborů.
- 2) Nahrání .BIN souborů do paměti KITu a jejich spuštění.

Další podpůrné činnosti, které KitLoader umožňuje:

- 1) Setup (konfigurace) KITů
- 2) Vyhledávání adres run-time chyb aplikací
- 3) Prohlížení obsahu paměti KITů
- 4) Nahrávání a vyčítání souborů do/z KITů

3.3. Co je nového oproti ReTOS Debugger

Dříve se pro nahrávání aplikací do KITů používal nástroj ReTOS Debugger (zkráceně RTD), který byl postaven na DOS platformě a svým vzhledem připomínal prostředí Borland Pascalu 7.0 pro DOS.

Pro programátory přecházející na KitLoader z RTD zde v několika bodech uvedeme základní odlišnosti:

- KitLoader je program pod Windows používající standardní vzhled Windows aplikací.
- Nepodporuje Kernel Monitor.
- Podporuje Kit188ER.
- Stále se vyvíjí a nabízí celou řadu nových užitečných funkcí.

4. Instalace a spuštění

4.1. Instalace

Program KitLoader nevyžaduje zvláštní instalaci. Stačí pouze na disk zpravidla do adresáře **KitLoader** zkopírovat všechny níže uvedené soubory:

- **KitLoader.exe** – Hlavní spouštěcí soubor.
- **KitLoader.ini** – Nastavení programu. Nedoporučuje se editovat tento soubor, neboť neodborná manipulace s tímto inicializačním souborem může vést k nefunkčnosti aplikace. Pro koncového uživatele nejsou v tomto souboru žádná důležitá nastavení.
- **Rtdw.ini** – Uživatelská nastavení pro konkrétní projekt, zejména COM port apod. Tento soubor je automaticky vytvářen rovněž v pracovním adresáři každého projektu.
- **KitLoader.psw** – Definice uživatelů a jejich přístupové úrovně pro některé speciální funkce. Viz kapitola „1.1 Uživatelé, přihlašování a přístupové úrovně“.

4.2. Hostitelský operační systém

KitLoader je aplikace psaná pro platformu Win32 – tedy pro os. Microsoft Windows 95, 98, ME, 2000, XP.

4.3. Spuštění

Program se spouští zpravidla z pracovního adresáře daného projektu příkazem **[Drive:]\Path\KitLoader.exe Kitloader.ini App.exe**, kde App je název vaší řídicí aplikace (projektu).

Pokud se program KitLoader spouští prostřednictvím DOS aplikace (např. z prostředí Borland Pascal 7.0 pro DOS), nesmí se zapomenout, že program má dlouhé jméno souboru KitLoader.exe, které se z DOS prostředí volá zpravidla jako KitLoa~1.exe. Příkazová řádka pro nástroj spuštění KitLoaderu s daným projektem z editoru Borland Pascal 7.0 pro DOS by měla následující tvar: Program path = **[Drive:]\Path\KitLoa~1.exe**, Command line = **Kitloader.ini \$EXENAME** + další parametry (viz následující obrázek). Prostředí Pascalu za \$EXENAME automaticky před spuštěním KitLoader doplní název vašeho projektu (název souboru zadaného v menu editoru Pascalu Compile – Primary file ...).



Pozn.: Význam parametrů /G /L /R je blíže popsán v kapitole „8 Parametry příkazové řádky programu“.

Pozn.: Parametr **kitloader.ini** se nemusí uvádět v jeho zkrácené DOS formě.

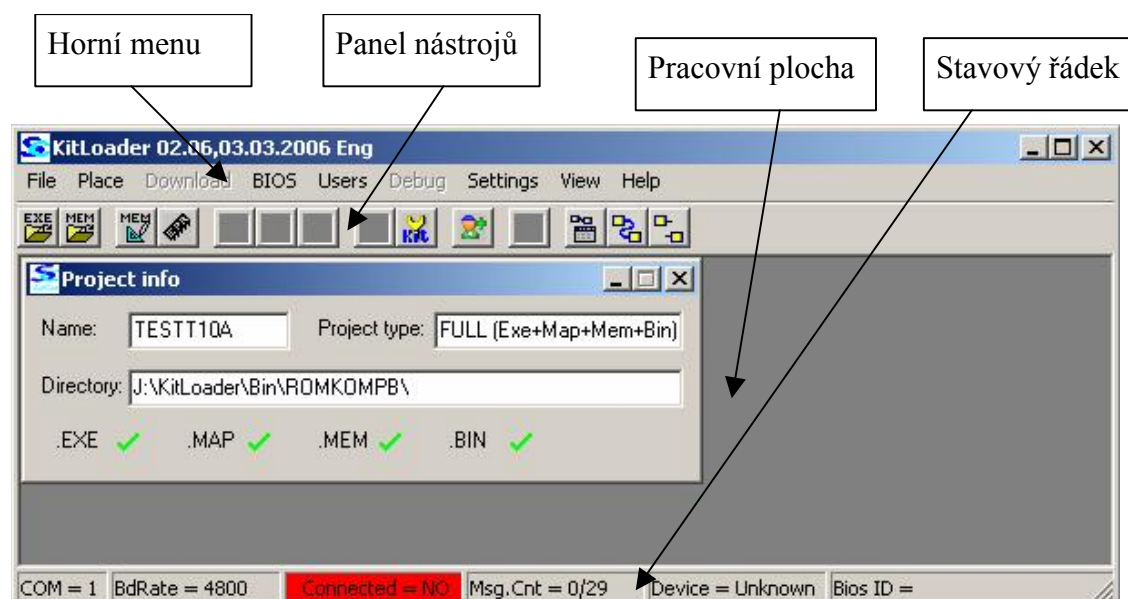
Při takovéto konfiguraci je možné klávesovou zkratkou Shift+F8 vyvolat KitLoader přímo z editoru Borland Pascalu a zároveň nahrát a spustit danou aplikaci.

Po spuštění KitLoaderu se program snaží automaticky navázat komunikaci s připojeným Kitem. Nastavení COM portu je dáno v souboru Rtdw.ini a lze ho změnit příkazem **Settings – COM** (viz „7 Nastavení komunikace“).

Z praktického důvodu je možno spustit jen jednu instanci programu KitLoader pod o.s.Windows. Při případném pokusu o spuštění další instance se zaktivní KitLoader, který již spuštěný je.

5. Vzhled a základní ovládání

Aplikace sestává z pracovní plochy, na které se zobrazují jednotlivá okna aplikace. Dále z horního menu, panelu nástrojů a stavového řádku.



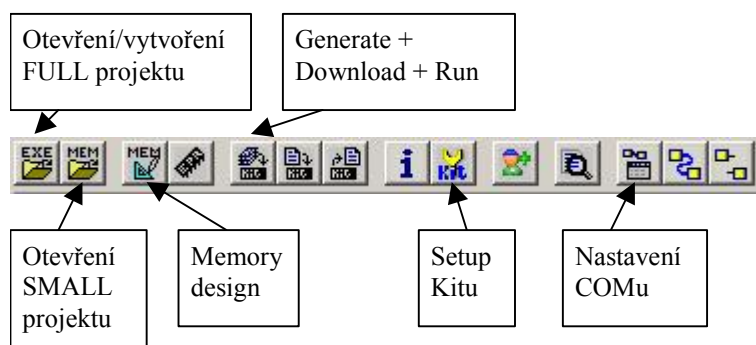
5.1. Horní roletové menu

Struktura horního menu je následující:

- **File** – příkazy pro otvírání a ukládání jednotlivých projektů (viz kapitola „6.2 Vytváření a práce s projekty“) a příkaz pro ukončení programu.
- **Place** – příkazy pro vytvoření paměťového rozložení, generaci binárních souborů a pro zpětné vyhledání adres run-time chyb aplikací.
- **Download** – příkazy pro nahrání a spuštění projektu a dále pro vyčítání a zápis souborů do KITu.
- **Bios** – příkazy zobrazující identifikaci BIOSu KITu a jeho nastavení (SETUP) v závislosti na připojeném KITu atd.
- **Users** – přihlášení/odhlášení a definice uživatelů s různými úrovněmi přístupu (některé funkce jsou dostupné pouze uživatelům s dostatečným oprávněním).
- **Debug** – pomocné příkazy určené pro ladění aplikací (zobrazování paměťového prostoru KITu apod.).
- **Settings** – nastavení parametrů programu KitLoader (nastavení COM portu).
- **View** – možnosti zobrazení několika informativních oken.
- **Help** – nápověda a informace o verzi KitLoaderu.

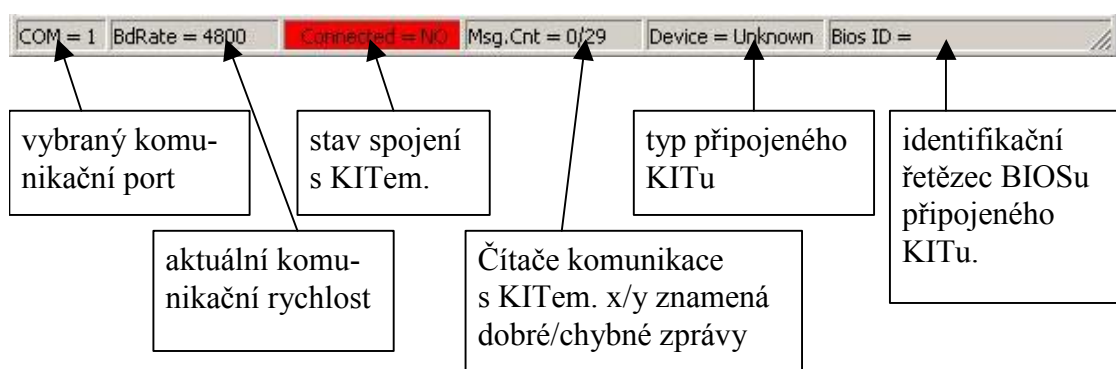
5.2. Panel nástrojů

V panelu nástrojů jsou pro snazší dostupnost vytaženy některé často používané příkazy z horního roletového menu. Obsahují stejné grafické ikony jako příkazy v horním menu a „Hint“ (nápočedný text, který se objeví pokud uživatel pouze najede myší nad nástrojové tlačítko) je stejný, jako název příkazu horního menu.



5.3. Stavový řádek

Stavový řádek je u spodního okraje okna KitLoader. Zobrazuje zejména stav komunikace s KITem.



6. Práce s KitLoaderem

6.1. Termíny, pojmy a názvosloví

KIT – souhrnný název pro jeden z těchto řídicích systémů: KitV40, Kit386, Kit188ER.

Bios Monitor – základní programové vybavení umístěné v paměti FLASH/EPROM KITu. Zajišťuje komunikaci při nahrávání/konfiguraci. Na každém KITu je propojka, která udává, zdali po zapnutí napájení má být spuštěna aplikace, nebo Bios Monitor.

Jednotka – Unit z Pascalu. Pro účely tohoto manuálu se pod pojmem jednotka také uvažují speciální „jednotky“ – a to **--LOADER--**, **--DATA--**, **--STACK--**, **--HEAP--** a **--INITDATA--**.

Modul – paměťový modul = část paměti obsahující jednotky (units) řídicí aplikace. Tato aplikace se skládá z jednoho nebo více modulů. Moduly se dělí na RAM modul (modul umístěný v prostoru paměti RAM a obsahující jednotky **--DATA--**, **--STACK--** a **--HEAP--**) a ROM moduly (moduly umístěné v prostoru ROM, tj. FLASH či EPROM, obsahující ostatní jednotky aplikace).

6.2. Vytváření a práce s projekty

Pod pojmem projekt je zde míněna právě jedna řídicí aplikace, kterou je možné nahrát do KITu. Každý projekt má své jméno (totožné se jménem .EXE souboru aplikace). Do projektu náleží soubor .EXE, soubor .MAP, soubor .MEM a jeden nebo více souborů .BIN.

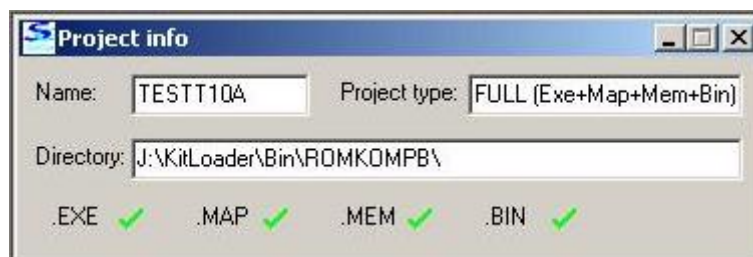
Soubor .EXE je výstupem z překladače Borland Pascalu.

Soubor .MAP je výstupem z linkeru Borland Pascalu (standardně Pascal tento soubor nevytváří, proto musíme v menu prostředí Borland Pascalu 7.0 nastavit jeho vytváření položkami Options – Linker a ve sloupci Map file zvolíme položku Detailed).

Soubor .MEM je hlavním souborem projektu, neboť obsahuje paměťové rozložení jednotlivých modulů a jednotek aplikace. .MEM soubor je vytvářen KitLoaderem (z menu Place – Memory design).

Soubory .BIN jsou binární soubory určené pro nahrání do paměti KITu.

Stav aktuálního projektu je zobrazován v okně **Project info**. Okno je možné vyvolat z menu View - Project.



V tomto okně se zobrazuje jméno projektu, typ projektu (FULL / SMALL), umístění projektu na disku a v dolní části je znázorněna existence (zelená fajfka) / neexistence (červený křížek) příslušného souboru.

Výše popsáný projekt je typu FULL, který má správně obsahovat všechny čtyři typy souborů (.MEM a .BIN až po „Vytváření paměťového rozložení“). Projekt typu SMALL se liší v tom, že neobsahuje .EXE a .MAP soubory, ale pouze vygenerované .MEM a .BIN. Takový SMALL projekt má další omezení vyplývající z jeho podstaty – např. protože neexistuje .EXE ani .MAP, nelze vygenerovat nové .BIN soubory.

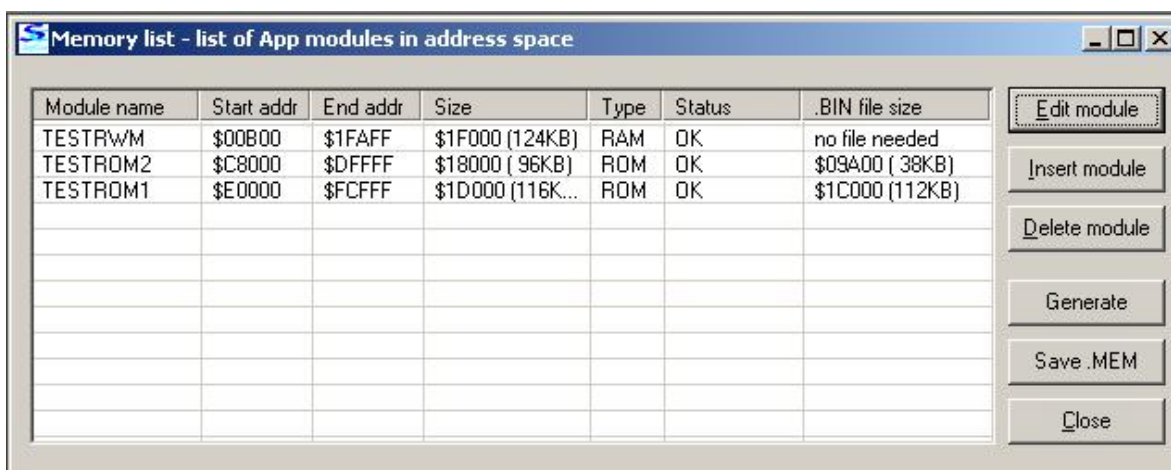
Zvolení projektu se provádí z menu **File – Set .EXE** (pro FULL projekt), nebo **File – Set .MEM** (pro SMALL projekt).

Uzavření projektu se provádí z menu **File - Close project**.

6.3. Vytváření paměťového rozložení

Máme-li zvolený projekt (příkazem **File – Set .EXE**), je možné začít s vytvářením paměťového rozložení. Paměťové rozložení je seskupení několika paměťových modulů, přičemž každý modul obsahuje počáteční adresu, délku, typ a jednotky v něm obsažené.

Po příkazu **Place – Memory design** se otevře okno **Memory list**:



Každý řádek popisuje právě jeden paměťový modul. Význam jednotlivých sloupců je následující:

Module name – jméno konkrétního modulu.

Start addr. – absolutní adresa, na které modul začíná.

End addr. – absolutní adresa, na které modul končí.

Size – velikost modulu v bytech.

Type – typ modulu (RAM/ROM...).

Status – informace o správnosti a platnosti daného modulu.

.BIN file size – velikost .BIN souboru v bytech.

Sada tlačítek má následující význam:

Edit module – editace vybraného modulu (viz níže).

Insert module – vložení nového prázdného modulu.

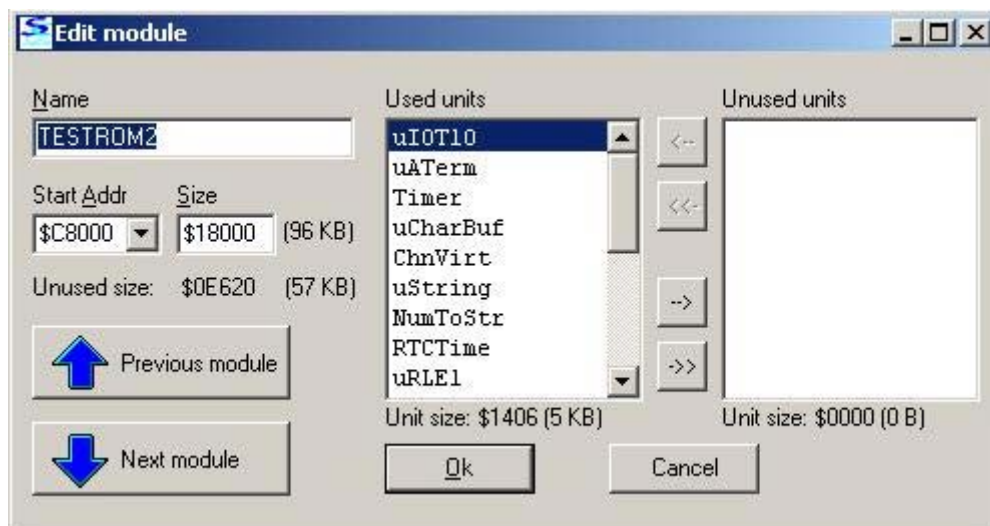
Delete module – odstranění modulu ze seznamu (vyžaduje potvrzení).

Generate – vygenerování .BIN souborů jednotlivých modulů.

Save .MEM – uložení .MEM souboru.

Close – zavření okna.

Editace jednotlivých modulů se provádí v okně **Edit module**, které se vyvolá stisknutím tlačítka **Insert** – vložení nového modulu, nebo tlačítka **Edit** – editace existujícího modulu.



Okno zobrazuje:

- **Name** - jméno modulu (max. 20znaků)
- **Start Addr** - počáteční adresu (musí být dělitelná 512 (= \$200)). O přípustných adresách umístění jednotlivých modulů pojednává dokument KitAdr.
- **Size** - velikost místa vyhrazeného pro paměť modulu (musí být dělitelná 256 (= \$100)). Délka paměťového modulu typu RAM (tj. modulu obsahujícího jednotku DATA, STACK, HEAP) je bez omezení. Délka paměťového modulu ROM (modul s kódem) může být max. \$20000 (= 128KB).
- **Unused size** - zbývající volné místo v modulu
- **Used units** - seznam jednotek řídicí aplikace umístěných v tomto modulu
- **Unused units** – seznam jednotky řídicí aplikace, které ještě nejsou umístěné v žádném modulu.

Pod seznamy **Used units** a **Unused units** jsou zobrazeny délky právě vybraných jednotek ze seznamů. Pro přidání/ubrání jednotlivých jednotek je možné použít tlačítka →, ←, případně přetáhnout jednotku myši pomocí „drag & drop“, nebo též použít poklepání (dvoj-klik) na jednotku. Tlačítko <<- umožňuje místo postupného přidávání jednotek přidat co nejvíce jednotek najednou, až do maximálního využití celého modulu. Naopak tlačítko ->> způsobí ubrání všech jednotek z modulu.

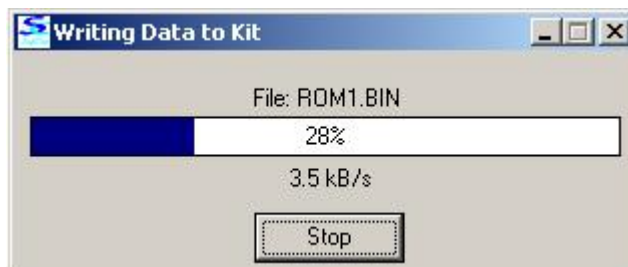
Tlačítka ↑ **Previous module** a ↓ **Next module** slouží pro snazší přechod mezi jednotlivými moduly aplikace.

Tlačítkem **Ok** se okno **Edit module** zavře a zpět se zpřístupní okno **Memory list**. Jsou-li začleněny všechny jednotky řídicí aplikace do jednotlivých modulů, je možno pro tuto aplikaci vygenerovat příkazem **Generate** jednotlivé .BIN soubory, které jsou určeny pro nahrání do řídicího systému KIT.

Pozn: Příkaz **Generate** před vlastním vytvořením .BIN souborů zkontroluje umístění jednotlivých jednotek řídicí aplikace v modulech a správné zadání těchto modulů (jestli se moduly navzájem nepřekrývají, jestli jsou na správných adresách apod.).

6.4. Nahrání aplikace do systému KIT a její spuštění

Pro nahrání aplikace do systému KIT a její spuštění slouží položka menu **Download – (Gen)+Download+Run**. Tímto příkazem se znovu zkontroluje a provede vygenerování jednotlivých modulů aplikace a poté se tyto moduly začnou aktuální komunikační rychlostí nahrávat do řídicího systému KIT. Průběh nahrávání je zobrazován v následujícím okně:



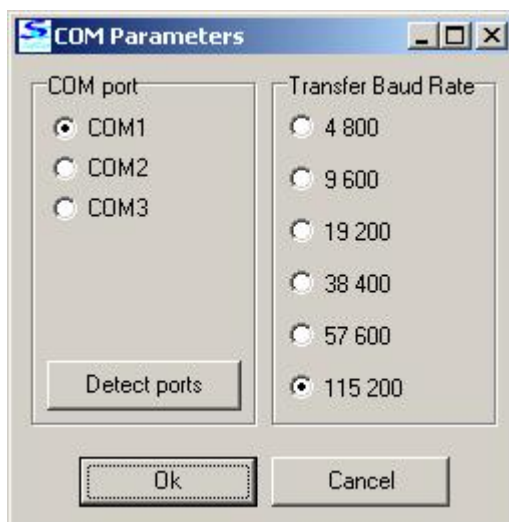
Po úspěšném nahrání všech modulů aplikace se provede její spuštění. Tím se přeruší komunikace mezi KitLoaderem a KITEM, tj. objeví se tato systémová zpráva:



která v tomto případě tedy neznamená chybu.

7. Nastavení komunikace s KITEM

KIT může být připojen na libovolný dostupný COM port.. Pro nastavení příslušného portu slouží položka menu **Settings – COM**.



Pole **COM port** informuje o dostupných COM portech na PC a umožňuje jeden z nich vybrat. Nabízené COM porty mohou být i za běhu přidáné emulace COM portů, např. přes USB, ETHERNET apod. Pro případné opakované vyhledání všech existujících a dostupných portů je určeno tlačítko **Detect ports**.

Pole **Transfer Baud Rate** udává použitou (vyšší) rychlost pro komunikaci s KITem. Po zapnutí KITa je jeho komunikační rychlost nastavena standardně na 4 800Bd, přičemž KitLoader po navázání komunikace na této standardní rychlosti přepne komunikaci na případnou vyšší rychlost nastavenou ve sloupci **Transfer Baud Rate**. Pokud se komunikace s KITem přeruší, snaží se KitLoader navázat komunikaci střídavě na standardní rychlosti 4 800Bd nebo na vyšší nastavené rychlosti.

Upozornění: pro Kit V40/8MHz je maximální použitelná rychlost 19 200Bd, pro KitV40/16MHz je to rychlost 38 400Bd.

8. Parametry příkazové řádky programu KitLoader

8.1. Zadávané parametry

Program umožňuje při spuštění z příkazové řádky zadat parametry. Na pořadí zadání parametrů nezáleží (výjimkou je první parametr KitLoader.ini). Syntaxe příkazového řádku je následující:

[Drive:]Path\KitLoader.exe KitLoader.ini [App.exe [/G][/L][/R]]

- Hranaté závorky znamenají nepovinný parametr.
- App je název projektu (aplikace) nahrávaného do KITu. V případě zadání .EXE souboru se po spuštění KitLoaderu otevře FULL projekt. Pokud se ale do příkazové řádky zadá .MEM soubor, otevře se SMALL projekt.

Význam parametrů G, L a R je totožný s ReTOS Debuggerem a to:

/G – po otevření zadaného FULL projektu provést vygenerování binárních souborů

/L – po navázání komunikace s KITem nahrát binární soubory na příslušné adresy do paměti KITu

/R – po nahrání aplikace do paměti KITu ji spustit

Jednotlivé operace se provádějí vždy v pořadí /G, /L, /R, pokud byl daný parametr v příkazové řádce uveden.

9. Setup (konfigurace) KITu

Pro nastavení konfigurace Kit386EXR nebo Kit188ER (pozn: KitV40 žádnou konfiguraci nemá) použijte příkaz z horního menu **Bios – Setup KitXxx**. Je-li navázána komunikace s Kit386EXR, zobrazí se okno popsané v kapitole „9.1 Setup pro Kit386EXR“. Komunikuje-li se s Kit188ER, zobrazí se okno popsané v kapitole „9.2 Setup pro Kit188ER“.

Není-li navázána komunikace s žádným KITem, otevře se dialogové okno pro načtení souboru konfigurace některého z KITů z disku. Tímto způsobem je možné modifikovat (nebo jen zobrazit) konfiguraci bez nutnosti připojeného KITu, tzv. nastavení

Offline. Po otevření souboru s uloženou konfigurací se tato konfigurace zobrazí v některém z oken pro setup daného KITu (viz následující kapitoly).

9.1. Setup pro Kit386EXR

Dialogové okno pro nastavení konfigurace Kit386EXR vypadá následovně:

	Chip size	Operat. Mem.	Acc. Time	Mem. Type	Ext. disk
Socket 1 (ROM LOW)	256 kB	512 kB	92/116 nS	FLASH	0 kB
Socket 2 (RAM LOW)	512 kB	512 kB	92/116 nS	RAM	256 kB
Socket 3 (ROM HIGH)	256 kB	512 kB	92/116 nS	FLASH	0 kB
Socket 4 (RAM HIGH)	512 kB	512 kB	92/116 nS	RAM	256 kB

	Begin addr.	Size
High ROM space	\$C0000	256 kB
Low ROM space	\$80000	128 kB
Cut ROM space	\$C0000	32 kB

Configuration

<input type="checkbox"/> SSIO/COM2	<input type="checkbox"/> INT3/SMM	<input checked="" type="checkbox"/> 33/25 MHz
<input type="checkbox"/> DMA1/COM2	<input type="checkbox"/> DMA2/COM2	<input type="checkbox"/> Video Wr

Setup status
Ok

Buttons: Save to Kit, Cancel, Load from disk ..., Save to disk ...

Význam jednotlivých parametrů je popsán v manuálu k Biosu „MCP BIOS řídící jednotky Kit386EXR“.

POZOR: Položka **Acc. Time** má stejný význam jako u dříve používaného programu RTD (pro zachování compatibility), tj. délka vložených čekacích stavů (wait states). Ve skutečnosti tedy nejde o přístupovou dobu k daným pamětem.

Konfiguraci Kit386 je možné uložit tlačítkem **Save to disk ...** v binární podobě do souboru na disk. Naopak uloženou konfiguraci je možné načíst z disku tlačítkem **Load from disk ...**

9.2. Setup pro Kit188ER

Dialogové okno pro nastavení konfigurace Kit188ER vypadá následovně:

Význam jednotlivých položek je následující:

- **EEPROM/FLASH chip memory size** - Konfigurace paměťového prostoru:
 - **Bank HIGH / Bank LOW** – velikost osazené paměti FLASH/EEPROM, zvlášť spodní a horní bank.
 - **Start of code memory** – hodnota určující adresu konce datové paměti a současně adresu začátku paměti kódu.
 - **Run program in RAM** – při spuštění aplikace z ROM paměti dojde k překopírování obsahu ROM od adresy dané položkou **Start of code memory** do paměti RAM. Tato paměť RAM je pak použita jako paměť kódu a není přístupná pro zápis (tzv. Write protected).
- **CF Booting** - Start aplikace z Compact Flash karty:
 - **Run from CF card** – povolení startu aplikace z Compact Flash karty, respektive při startu Kit188ER bude aplikace nejprve hledána na CF kartě. pokud nebude nalezena, bude proveden start aplikace z paměti ROM podle konfigurace paměťového prostoru položkami v části konfigurace **EEPROM/FLASH chip memory size**.
 - **Segment** – hodnota segmentu paměti RAM, na kterém bude spuštěna aplikace nahraná z CF karty.
 - **Boot file name** - jméno souboru aplikace uložené v kořenovém adresáři na CF kartě.
- **IOBUS settings** - Konfigurace časování na sběrnici IOBUS:
 - **Wait State** – počet čekacích stavů přidanych do přístupových cyklů sběrnice IOBUS.
 - **Extended access time** – počet čekacích stavů přidanych mezi dva přístupové cykly na sběrnici IOBUS.

- **NMI Enabled from** - Generování nemaskovatelného přerušení:
 - **Write to protected (code) memory** – povoluje generovat nemaskovatelné přerušení od zápisu do chráněné části paměti.
 - **Power Fail** – povoluje generovat nemaskovatelné přerušení při výpadku (poklesu) napájecího napětí.

Konfiguraci Kit188 je možné uložit tlačítkem **Save to disk ...** v binární podobě do souboru na disk. Naopak uloženou konfiguraci je možné načíst z disku tlačítkem **Load from disk ...**

10. Identifikace BIOSu KITu

Pro identifikaci BIOSu připojeného KITu je určena položka z horního menu **Bios - Bios info**. Po vyvolání a provedení tohoto příkazu se na obrazovce objeví zpráva s identifikačním řetězcem BIOSu, identifikátorem paměti FLASH a typem paměti FLASH. Příklad pro Kit188ER je uveden na následujícím obrázku:



Pozn: Identifikační řetězec BIOSu je také zobrazován na dolním stavovém řádku.

10.1. KitV40

Struktura identifikačního řetězce pro řídicí jednotku KitV40 je následující:

SofCon (C) [8/16]-[B/P][V][xx][<MCP>/<T10>/<T03>]

```
| | | |  
|--- určení BIOSu.  
|----- číslo verze BIOSu.  
|----- video služby BIOSU  
          provádějí zápis do  
          VIDEO RAM. (EPROM)  
          V případě, že BIOS  
          neobsahuje tento flag  
          pak video služby  
          neprovádí zápis do  
          VIDEO RAM. (FLASH)  
|----- test JP4 a následné  
          spuštění BIOS monitoru,  
          případně aplikačního  
          programu.  
|----- taktovací frekvence  
          procesoru V40
```

10.2. Kit386EXR

Struktura identifikačního řetězce pro řídicí jednotku Kit386 je následující:

```
386EXR/[a]-[b]-[c]:[d]-[e][f]<KIT>-[g]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- nastavení konfiguračního
|   |   |   |   |   |   |   |       registru BIOS386EXR [O/M]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- číslo verze BIOSu
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- test APL/BM a následné
|   |   |   |   |   |   |   |       spuštění BIOS monitoru,
|   |   |   |   |   |   |   |       případně aplikačního
|   |   |   |   |   |   |   |       programu (JP33) [B/P]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- přístup k paměti RAM [8/16]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- přístup k paměti ROM [8/16]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- test DEF/CFG a následné
|   |   |   |   |   |   |   |       nastavení registrů
|   |   |   |   |   |   |   |       procesoru 386EX (JP30)
|   |   |   |   |   |   |   |       [Def/Cfg]
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |----- taktovací frekvence
|   |   |   |   |   |   |   |       procesoru 386EX [25/33]
```

10.3. Kit188ER

Struktura identifikačního řetězce pro řídicí jednotku Kit188 je následující:

```
KIT188ER [a]-v[h1]
|   |
|   |----- číslo verze BIOSu ve formátu „h.l“
|   |
|   |----- taktovací frekvence procesoru Am188
|   |       [40/44/48/49]
```

Poznámka: detekce paměti typu FLASH u řídicí jednotky Kit188ER se provádí pouze v horním paměťovém banku (ROM Hi).

11. Uživatelé, přihlašování a přístupové úrovně

Některé funkce využívané pro ladění aplikací vyžadují hlubokou znalost řídicích systémů KIT, BIOSu apod. Pro uživatele používající KitLoader pouze pro nahrání nových verzí aplikací by případné chybné užití těchto funkcí mohlo vést k poškození a nefunkčnosti nejen jejich aplikací, ale i BIOSu a potažmo i řídicího systému. Proto byly formou uživatelů (Users) zavedeny uživatelské úrovně pro jednotlivé funkce. To znamená, že každý uživatel má přiděleno své jméno, heslo a uživatelskou úroveň, která ho opravňuje k používání určité skupiny funkcí.

Po spuštění programu KitLoader není přihlášen žádný uživatel, tj. je aktivní přístupová úroveň 0 a jsou dostupné funkce nevyžadující zvláštní přístup. Do této úrovně patří všechny standardně používané funkce, běžný uživatel tedy může program používat bez přihlášení.

Pro přihlášení uživatele s vyšší úrovní přístupu k některým funkcím slouží příkaz z horního menu **Users – User Login** (viz obrázek). Příkaz **Users – Logout** slouží k odhlášení přihlášeného uživatele a příkaz **Users – Users define** slouží k definici uživatelů a jejich přístupových úrovní. Tento příkaz je však přístupný pouze uživateli s nejvyšší úrovní a tím je výhradně pověřená osoba fy. *SofCon*.

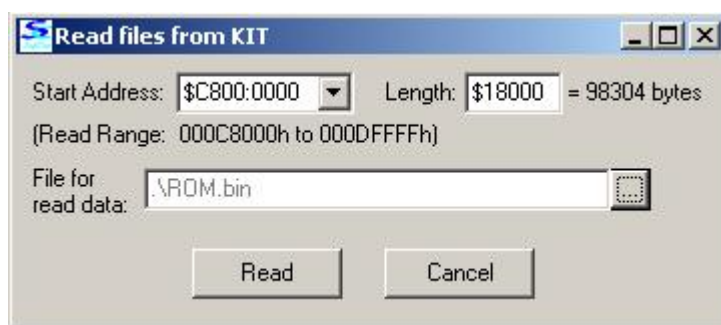


12. Čtení a zápis uživatelských .BIN souborů do paměti KITu

12.1. Vyčítání obsahu paměti do binárního souboru

Pozn: Tato funkce je přístupná pouze přihlášeným uživatelům s přístupovou úrovní 1.

K vyčítání obsahu paměti KITu do souboru slouží okno vyvolané pomocí menu **Download – Read .BIN from KIT**.

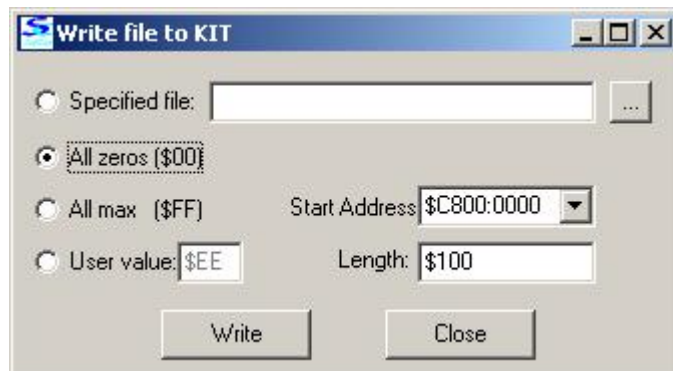


Po zadání počáteční adresy **Start Address** (absolutní - lineární nebo ve tvaru segment:offset), délky vyčítané paměti v bytech **Length** a jména cílového souboru **File for read data** je vyčítání spuštěno tlačítkem **Read**. Jméno souboru je zadáváno kliknutím na tlačítko „...“, které otevře standardní dialogové okno ukládání souboru, ve kterém je možno zvolit již existující soubor nebo zadat název nového souboru, do kterého se načtená data uloží.

12.2. Zápis binárních dat do paměti KITu

Pozn: Tato funkce je přístupná pouze přihlášeným uživatelům s přístupovou úrovní 1.

K zápisu binárního obrazu souboru do paměti KITu je možné použít okno vyvolané pomocí menu **Download – Write .BIN to KIT**.



Uživatel si může zvolit, zapíše-li do KITu konkrétní soubor uložený na disku (zadaný v položce **Specified file**), a nebo zapíše-li na danou adresu pole konstantních znaků (jedna z položek **All zeroes**, **All max**, **User value**).

Položka **Start Address** určuje počáteční adresu v paměťovém prostoru KITu, od které se budou zapisovat zvolená data.

V případě, že uživatel zapisuje konkrétní soubor (**Specified file**), je položka **Length** vyplněna aktuální velikostí souboru a uživatel ji může případně zkrátit (provede se zápis pouze části souboru).

V případě, že uživatel zapisuje pole konstantních hodnot, musí položku **Length** zadat ručně.

Spuštění zápisu se provede tlačítkem **Write**.

Pozn: Tato funkce zápisu binárních dat do paměti KITu se částečně překrývá s nahráváním aplikace do KITu. Pro nahrávání aplikací však doporučujeme použít projekt a položku menu **Download – Download App**.

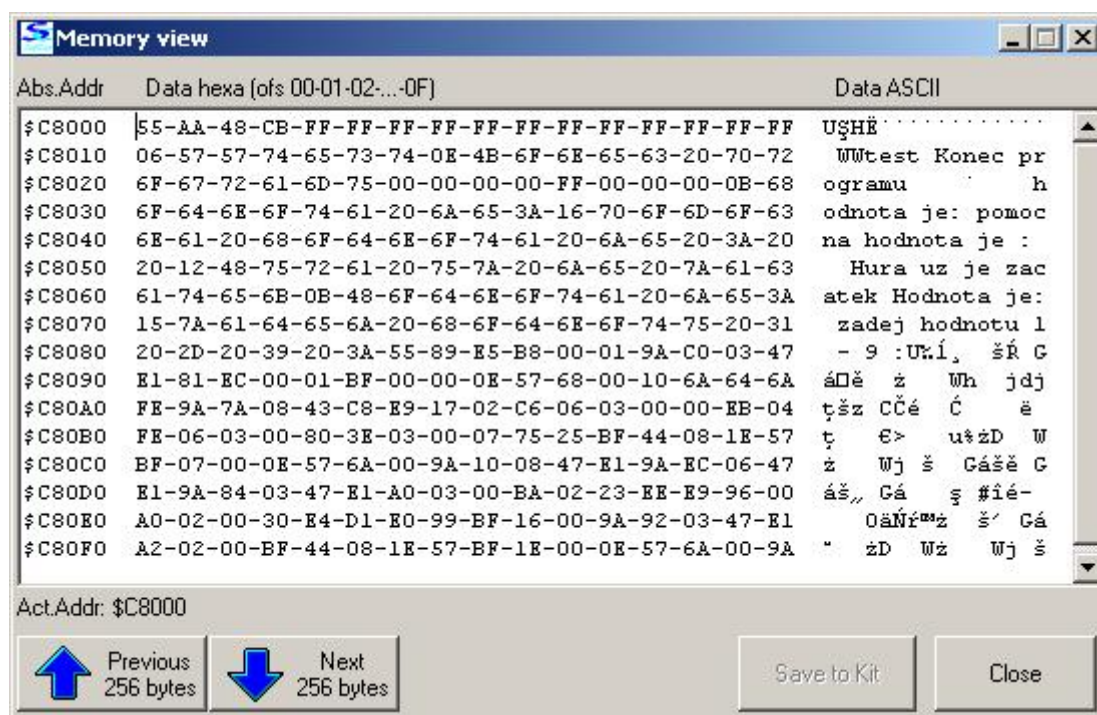
13. Zobrazení a editace části paměti KITu

Pozn: Funkce pro zobrazení je přístupná pouze přihlášeným uživatelům s přístupovou úrovní 1. Editace zobrazených dat je přístupná pouze přihlášeným uživatelům s přístupovou úrovní 2.

Po příkazu z horního menu **Debug – Memory view** se otevře nejprve okno pro zadání počáteční adresy výpisu paměti. Je možno zadat jak absolutní (lineární) adresu, tak adresu ve tvaru segment:offset:



Po zadání adresy a potvrzení tlačítkem **Ok** se vyvolá okno (viz následující obrázek), ve kterém je zobrazeno 256 bytů od zadané adresy:



V levém sloupci **Abs.Addr** je pro informaci zobrazena absolutní (lineární) adresa načtených dat. V sekci **Data hexa** jsou hexadecimálně zobrazeny byty načtené z příslušných adres a v sekci **Data ASCII** jsou stejná data zobrazena textově podle ASCII tabulky. Kurzorovými šipkami se lze pohybovat po jednotlivých znacích (bytech), přičemž aktuální adresa znaku, na kterém se nachází kurzor, je zobrazena dole v položce **Act.Addr**. Kurzor lze přepínat (kliknutím myši) do sekce **Data ASCII** nebo do **Data hexa** a pohybovat se v daných sekcích.

Je-li přihlášen uživatel s přístupovou úrovní 2, je možno editovat znaky (byty) na aktuální pozici kurzoru. Změněné znaky se zobrazují červeně. Tlačítkem **Save to Kit** se změny uloží na příslušné adresy do paměti KITu.

Pomocí tlačítek **↑ Previous 256 bytes** a **↓ Next 256 bytes** lze zobrazovat předchozí resp. následující byty v paměti. Provedl-li uživatel editaci zobrazovaných dat, je před přechodem na jiných 256 bytů dotázán na jejich uložení do KITu.