

Začínáme s vizualizací

JAK POSTUPOVAT PŘI NÁVRHU
APLIKACE S VIZUALIZACÍ

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: 20.05.2003

Datum posledního uložení dokumentu: 20.05.2003

(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	5
1.1. Revize dokumentu	5
1.2. Účel dokumentu	5
1.3. Rozsah platnosti	5
1.4. Související dokumenty	5
2.Termíny a definice	5
3.Úvod	6
4.Adresářová struktura a kopírování souborů	6
5.Používané direktivy	8
6.Dodávané příklady	10
6.1. Example1	10
6.2. Example2	11
6.3. Example3	12
7.Postup návrhu aplikace	13
7.1. Vytvoření CSV souboru	13
7.2. Tvorba oken pro vizualizaci	14
7.3. Aplikace pro řídicí systémy KIT	14
7.3.1. Úpravy šablon a nastavení konstant popisující paměť	14
7.3.2. Předefinování metod funkčních objektů	15
7.4. Nahrání aplikace do řídicího systému KIT	15

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Hv		První vydání.
1.10	1.XX	Tu	20.05.2003	Úprava dokumentu dle ISO9000.

1.2. Účel dokumentu

Tento dokument popisuje postup při návrhu nové aplikace komunikující s programem TheKing, který vizualizuje data z řídicího systému KIT v nadřazeném počítači s operačním systémem Windows..

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu je potřeba seznámit se s manuály popisujícími vizualizační knihovny a archivy - KitKing a Archivy.

2. Termíny a definice

Používané termíny a definice jsou popsány v samostatném dokumentu Termíny a definice.

3. Úvod

Účelem tohoto dokumentu je popsat postup při návrhu nové aplikace komunikující s programem TheKing, který vizualizuje data z řídicího systému KIT v nadřazeném počítači s operačním systémem Windows.

V několika následujících kapitolách budou popsány jednotlivé kroky při návrhu aplikace. Pořadí kroků vychází ze zkušeností programátorů aplikací s podporou vizualizace tzn. komunikujících s programem TheKing. Dále v těchto kapitolách najdete popis dodávaných příkladů a stručný popis šablon případně odkazy na další manuály.

Protože cílem tohoto dokumentu je poskytnout postup při návrhu aplikace vizualizující svá data, nejsou v tomto manuálu popisovány žádné knihovny a programy. Tyto knihovny a programy jsou popsány v samostatných manuálech.

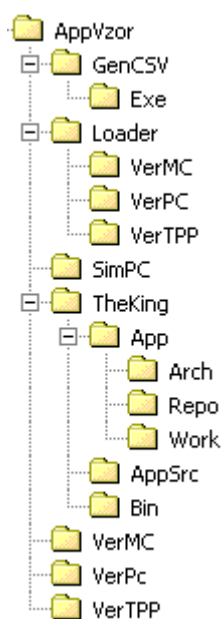
4. Adresářová struktura a kopírování souborů

Pro usnadnění vytvoření doporučené adresářové struktury a získání souborů vyžadovaných aplikací s podporou vizualizace, můžete použít dodávanou šablonu v adresáři **Vzor\AppVzor**. Při jejím používání doporučujeme provést kontrolu a případně aktualizaci tohoto souboru:

Turbo.tpl v adresáři AppVzor\VerMc a AppVzor\Loader\VerMc.

Název (označení) aplikace doporučujeme max. 7 písmen, protože první znak se v některých případech používá pro prefix **x (Exchange)** označující sdílené jednotky mezi více programy.

Pokud budete chtít vše provést sami, musíte nejprve vytvořit následující adresářovou strukturu.



Po vytvoření této adresářové struktury nakopírujte z příloženého příkladu následující soubory. Soubory, které v názvu obsahují **AppVzor** přejmenujte dle označení (názvu) Vaší aplikace.

Do adresáře Loader umístěte následující soubory:

uxDesc.pas. Pokud požadavky aplikace nebo řídicího systému vyžadují rozmístění sdílených struktur a programu LOADER na jinou adresu, zkopírujte také soubor Loader.pas.

Do adresáře GenCSV umístěte následující soubory:

AppVzor.pas, uAppVzor.pas a xAppVzor.pas. (Pozn. Tyto adresáře přejmenujete dle označení (názvu) Vaší aplikace.) Tyto soubory budete později upravovat dle požadavků Vaší aplikace a to pouze v případě, že budete používat online parametry.

Do adresářů SimPc, VerPc, VerTpp a VerMc umístěte soubor Sets.inc, ve kterém se musí správně nastavit direktivy překladače pro jednotlivé adresáře. Do adresáře VerMc je potřeba ještě nakopírovat Turbo.tpl pro řídicí systém KIT. Tento soubor získáte v dodávaných knihovnách v adresáři MCP7 při používání Borland Pascal verze 7.x. V těchto adresářích budete později provádět vlastní překlad aplikace.

Popis adresářů:

AppVzor	označení (název) Vaší aplikace. Doporučujeme používat max. 7 písmen.
Loader	upravená verze programu LOADER nebo obsahuje minimálně soubor uxDesc.pas popisující rozložení paměti.
GenCSV	v adresáři je umístěn program generující CSV soubor, který se používá v programu Cr_TheKing. Dále je v tomto adresáři soubor xAppVzor (označení-název Vaší aplikace), jenž je sdílen s Vaší aplikací.
VerMC	adresář pro překlad aplikace určené pro KIT. Ze začátku obsahuje upravený soubor Sets.inc a Turbo.tpl.
VerPC	adresář pro překlad aplikace určené pro PC s připojeným HW prostřednictvím PC Kit.
VerTPP	adresář pro překlad aplikace pro PC pracující v chráněném režimu s připojeným HW prostřednictvím PC Kit..
SimPC	adresář pro překlad aplikace určené pro PC bez připojeného HW. HW je programově simulován, zpravidla pomocí extra procesu nebo procesů.

TheKing adresář obsahující INI soubory, soubory s navrženými okny pohledů online parametrů a další pracovní adresáře.

Pozn.: Jména souborů a adresářů odpovídající ukázkové aplikaci (šabloně) se mohou lišit dle Vašich požadavků. Popis těchto adresářů je popsán v samostatném manuálu programu TheKing.

V adresáři App je soubor s koncovkou VIE, který obsahuje navržená okna s pohledem na online parametry. Dále v tomto adresáři mohou být různé soubory s obrázky, které jsou použity při vykreslování navržených oken a soubory generované programem Cr_TheKing (viz Tvorba oken pro vizualizaci)

V adresáři AppSrc jsou uloženy soubory s výčtovými typy zobrazovanými v programu TheKing a soubor s koncovkou CSV generovaný programem v adresáři GenCSV. Soubor s CSV a soubory s výčtovými typy tvoří vstup programu Cr_TheKing, který z těchto souborů vytvoří soubor s koncovkou APP.

5. Používané direktivy

Šablona a dodávané příklady obsahují soubory Sets.inc, pomocí kterých vybíráte, jak se příklad vytvoří. V tomto souboru mohou být následující direktivy.

- volba komunikačního kanálu
 - V40 komunikační port na procesoru V40
 - COM standardní komunikační port
 - UDP přenos dat probíhá pomocí Ethernet protokol UDP
- volba nastavení překladu
 - SimPC verze pro počítač PC s monitorem a klávesnicí a simulací HW
 - VerPC verze pro počítač PC s monitorem a klávesnicí a reálným HW
 - VerTPP verze pro počítač PC s monitorem a klávesnicí a reálným HW pracujícím v chráněném módu
 - VerMC verze pro řídicí systém Kit s HW
- volba překládaného příkladu
 - FILE1 příklad se dvěma jednoduchými souborovými bankami. Pomocí těchto bank se přenáší struktura tFixData (menší než 4KB, tzn. přenáší se v jednom komunikačním paketu). V tomto příkladu index dotazované banky souborů odpovídá indexu banky v řídicím systému KIT.
 - CONVERT_FILENO and FILE1 příklad se dvěma jednoduchými souborovými bankami. Pomocí těchto bank se přenáší struktura tFixData (menší než 4KB, tzn. přenáší se v jednom komunikačním paketu). V tomto

- příkladu index dotazované banky souborů se liší od indexu banky v řídicím systému KIT, proto se provádí jeho konverze.
- FILE2 příklad se dvěma souborovými bankami implementovanými jako oblast paměti o velikosti 4KB, tzn. přenáší se v několika komunikačních paketech. V tomto příkladu index dotazované banky souborů odpovídá indexu banky v řídicím systému KIT.
- LOADER příklad pro podporu příkazů programu LOADER, který umožňuje nahrát novou verzi firmware řídicího systému.
- PARAM příklad s nastavováním jednotlivých i blokových online parametrů v řídicím systému, čtení parametrů a přenos příkazů. Při spuštění příkladu s online parametry můžete v programu TheKing tyto parametry nastavovat, číst a zobrazovat pomocí oken **Prenos jednotlivých parametru a Blokovy prenos parametru** v menu **Views\Controller1\OnLine View**.
- ARCHIVE_INIT při nastavení této direktivy může dojít k poškození archivu. Proto by se měla používat pouze, pokud jste provedli změny struktury archivu případně jste je umístily do jiného paměťového prostoru. Při jejím použití se nastaví příznak `af_NotCheckAStruct`, což způsobí jejich nastavení podle předaných parametrů. Pokud tento příznak není nastaven, provádí se kontroly popisovačů archivu v paměti s požadovanými parametry. Pokud se zjistí rozdíl, vrací se chyba, viz `GetLastError`.
- ARCHIVE1 příklad práce se dvěma archivy používající paměť s kruhovým bufferem s podporou zobrazení tabulky, tzn. nepodporuje zobrazení grafů. Při čtení dat z archivu se vrací data uložená v archivu. (Pozn. Při zvolení jiného typu archivu se může vykonávat jejich modifikace.) V tomto příkladu index dotazované banky archivů odpovídá indexu banky v řídicím systému KIT.
- CONVERT_BANKNO and ARCHIVE1 příklad práce se dvěma archivy používající paměť s kruhovým bufferem s podporou zobrazení tabulky, tzn. nepodporuje zobrazení grafů. Při čtení dat z archivu se vrací data uložená v archivu. (Pozn. Při zvolení jiného typu archivu se může vykonávat jejich modifikace.) V tomto příkladu index dotazované banky archivů se liší od indexu banky v řídicím systému KIT, proto se provádí jeho konverze.
- ARCHIVE2 příklad práce se dvěma archivy používající paměť s kruhovým bufferem s podporou zobrazení grafů. Při čtení dat z archivu se vrací data uložená v archivu. (Pozn. Při zvolení jiného typu archivu se může vykonávat jejich modifikace.) V tomto příkladu index dotazované banky archivů odpovídá indexu banky v řídicím systému KIT.

6. Dodávané příklady

Součástí vizualizace jsou následující příklady, které implementují jednotlivé funkce vizualizace.

Abychom usnadnili přidávání těchto funkcí, je každá funkce implementována v samostatném souboru. Proto je možné po editaci funkcí s prefixem **fn**, tyto soubory přidat přímo k Vaší aplikaci.

Pozn. Při spuštění příkladů na počítači PC se může vygenerovat chyba. Tato chyba se generuje, pokud soubory pro ukládání statických struktur (koncovka SAV) nejsou vytvořeny nebo jsou poškozeny. Pokud chyba při opakovaném spuštění nezmizí, smažte nejdříve soubory s koncovkou SAV a dvakrát zopakujte spuštění příkladu. Nedaří-li se příklad spustit, kontaktujte firmu SofCon s.r.o.

6.1. Example1

Pomocí výše uvedených direktiv můžete vytvořit příklad dle Vašich požadavků. Dále bude následovat popis souborů s implementovanou funkcí aplikace.

Pozn.: V rámci zjednodušení tohoto kódu se ve verzi pro PC neuvolňuje všechna alokovaná paměť a nekontrolují se ukazatele na hodnotu NIL. Tento příklad byl později doplněn o přenos dat pomocí UDP a rozšířen o dříve uvedené kontroly a uvolňování paměti a je dodáván pod označením Example3.

Param.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace přenos online parametrů ve formě jednoduchých nebo blokových přenosů a implementaci příkazů.

Při první editaci tohoto souboru upravujte pouze funkce `fnGetParamVal`, `fnPutParamVal`, `fnGetParamSize`, `Init`, `SetResultCode` a `ShowState`. Funkce `fnGetFirmwareParamVal` a `fnPutFirmwareParamVal` byste měnit neměli, protože Vám ve spolupráci s jednotkou `AppDesc` zajišťují čtení a inicializaci struktury pro zachytávání chyb aplikace, které můžete v programu `TheKing` zobrazit pomocí **Views\Controller 1\Restart structure**.

Podrobný popis metod objektu najdete v manuálu `KitKing` v kapitole *Funkční objekt* a podkapitole *Objekt TWinTickParam*.

Loader.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace podporu pro nahrávání nové verze do řídicího systému KIT pomocí vizualizace.

Při první editaci tohoto souboru upravujte pouze funkci `ShowState`.

Podrobný popis metod objektu najdete v manuálu `KitKing` v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickLoader*.

File1.pas nebo File2.pas

V těchto souborech je předvedeno, jak doplnit do Vaší aplikace funkce vizualizace pro přenos souborů (viz Používané direktivy)

Při první editaci těchto souborů upravujte funkce `InitBankDescUserData`, `ShowState`, `fnConvertFileNo`, `fnGetFileCnt`, `fnGetBankListItem`,

fnGetBankDataOpen, fnGetBankDataBlock, fnPutBankDataOpen a
fnPutBankDataBlock.

Podrobný popis metod objektu najdete v manuálu KitKing v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickFile*.

Archive1.pas nebo Archive2.pas

V těchto souborech je předvedeno, jak doplnit do Vaší aplikace funkce vizualizace pracující s archivy (viz Používané direktivy)

Při první editaci těchto souborů upravujte funkce TextModif, ShowState, WriteToArchive, Init, fnConvertArchiveNo, fnGetArchiveCnt, fnClose, fnGetListOpen, fnGetListItem, fnGetState, fnOpenArchive, fnReadArchive a fnClearArchive.

Pokud zachováte použití pole při správě instancí objektů archivu stačí upravit pouze funkce TextModif, ShowState, WriteToArchive a Init.

Podrobný popis metod objektu najdete v manuálu KitKing v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickFile*.

6.2. Example2

Tento příklad implementuje pouze práci s online parametry a nahrávání nové verze aplikace. Dále bude následovat popis souborů s implementovanou funkcí aplikace.

Param.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace přenos online parametrů ve formě jednoduchých nebo blokových přenosů a implementaci příkazů.

Při první editaci tohoto souboru upravujte pouze funkce fnGetParamVal, fnPutParamVal, fnGetParamSize, Init, SetResultCode a ShowState. Funkce fnGetFirmwareParamVal a fnPutFirmwareParamVal byste měnit neměli, protože Vám ve spolupráci s jednotkou AppDesc zajišťují čtení a inicializaci struktury pro zachytávání chyb aplikace, které můžete v programu TheKing zobrazit pomocí **Views\Controllor 1\Restart structure**.

Podrobný popis metod objektu najdete v manuálu KitKing v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickParam*.

Loader.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace podporu pro nahrávání nové verze do řídicího systému KIT pomocí vizualizace.

Při první editaci tohoto souboru upravujte pouze funkci ShowState.

Podrobný popis metod objektu najdete v manuálu KitKing v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickLoader*.

6.3. Example3

Pomocí výše uvedených direktiv můžete vytvořit příklad dle Vašich požadavků. Dále bude následovat popis souborů s implementovanou funkcí aplikace.

Pozn.: Tento příklad vychází z Example1 a je rozšířen o výpisy stavu HEAP, uvolňování alokované paměti, kontroly ukazatelů na NIL a možnost přenosu dat pomocí UDP.

Param.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace přenos online parametrů ve formě jednoduchých nebo blokových přenosů a implementaci příkazů.

Při první editaci tohoto souboru upravujte pouze funkce `fnGetParamVal`, `fnPutParamVal`, `fnGetParamSize`, `Init`, `SetResultCode` a `ShowState`. Funkce `fnGetFirmwareParamVal` a `fnPutFirmwareParamVal` byste měnit neměli, protože Vám ve spolupráci s jednotkou `AppDesc` zajišťují čtení a inicializaci struktury pro zachytávání chyb aplikace, které můžete v programu `TheKing` zobrazit pomocí **Views\Controllor 1\Restart structure**.

Podrobný popis metod objektu najdete v manuálu `KitKing` v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickParam*.

Loader.pas

V tomto souboru je předvedeno, jak doplnit do Vaší aplikace podporu pro nahrávání nové verze do řídicího systému KIT pomocí vizualizace.

Při první editaci tohoto souboru upravujte pouze funkci `ShowState`.

Podrobný popis metod objektu najdete v manuálu `KitKing` v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickLoader*.

File1.pas nebo File2.pas

V těchto souborech je předvedeno, jak doplnit do Vaší aplikace funkce vizualizace pro přenos souborů (viz Používané direktivy)

Při první editaci těchto souborů upravujte funkce `InitBankDescUserData`, `ShowState`, `fnConvertFileNo`, `fnGetFileCnt`, `fnGetBankListItem`, `fnGetBankDataOpen`, `fnGetBankDataBlock`, `fnPutBankDataOpen` a `fnPutBankDataBlock`.

Podrobný popis metod objektu najdete v manuálu `KitKing` v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickFile*.

Archive1.pas nebo Archive2.pas

V těchto souborech je předvedeno, jak doplnit do Vaší aplikace funkce vizualizace pracující s archivy (viz Používané direktivy)

Při první editaci těchto souborů upravujte funkce TextModif, ShowState, WriteToArchive, Init, fnConvertArchiveNo, fnGetArchiveCnt, fnClose, fnGetListOpen, fnGetListItem, fnGetState, fnOpenArchive, fnReadArchive a fnClearArchive.

Pokud zachováte použití pole při správě instancí objektů archivu stačí upravit pouze funkce TextModif, ShowState, WriteToArchive a Init.

Podrobný popis metod objektu najdete v manuálu KitKing v kapitole *Funkční objekty* a podkapitole *Objekt TWinTickFile*.

7. Postup návrhu aplikace

7.1. Vytvoření CSV souboru

Upravte soubory v adresáři GenCSV podle následujících poznámek.

Pozn. Názvy popisovaných souborů jsou původními názvy před jejich přejmenováním dle předchozí kapitoly.

Pozn. Pokud přejmenujete jednotky dle Vašich požadavků, musíte upravit jména používaných jednotek v dále uvedených souborech.

AppVzor.pas	hlavní tělo programu generující CSV soubor. V těle tohoto programu se musí změnit cesty a jméno generovaného souboru.
xAppVzor.pas	soubor obsahující popis přenášených parametrů a je sdílen s Vaší aplikací pro řídicí systém KIT.
uAppVzor.pas	v souboru jsou příkazy, které generují vlastní CSV soubor. Popis těchto příkazů najdete v samostatném manuálu popisujícím Cr_TheKing.

Po spuštění vytvořeného programu odpovídající požadavkům aplikace na přenos online parametrů se vytvoří na zadané cestě CSV soubor. Tento soubor by se měl generovat do adresáře AppSrc, viz nastavení Cr_TheKing.ini.

7.2. Tvorba oken pro vizualizaci

Před vytvářením oken ve vizualizaci se musí spustit program Cr_TheKing, který vygeneruje soubory s koncovkou LST a APP. Soubor s koncovkou APP obsahuje jak systémové tak i aplikační popisovače datových struktur, nad kterými pracuje vizualizační program TheKing. Popis podrobného postupu pro vytváření těchto oken a těchto souborů najdete v samostatném manuálu popisujícím aplikaci TheKing a Cr_TheKing.

Pozn. Pro bezchybnou práci programu TheKing musí být vždy vytvořen soubor s koncovkou APP.

Nejjednodušší postup jak získat soubory pro vizualizaci je editací souboru Cr_TheKing.ini, kde změníte následující položky (upravíte cesty) dle požadavků Vaší aplikace.

```
AppSrcFile='o:\Vzor\AppVzor\TheKing\AppSrc\AppVzor.CSV'
```

```
AppDestFile='o:\Vzor\AppVzor\TheKing\App\AppVzor.app'
```

7.3. Aplikace pro řídicí systémy KIT

Pokud píšete první aplikaci s vizualizací, tak ji nejjednodušeji vytvoříte, pokud vycházíte z dodávaných příkladů případně můžete použít návod uvedený v samostatném manuálu popisujícím vizualizační knihovny případně archivy.

7.3.1. Úpravy šablon a nastavení konstant popisující paměť

Po změně názvů souborů obsažených v šabloně dle Vašich požadavků byste měli nastavit konstanty popisující rozložení paměti Vaší aplikace.

- **uxDesc.pas:**

Popis konstant popisující paměť v souboru **uxDesc.pas** najdete v samostatném manuálu popisujícím program LOADER:

Pozn.: Pokud měníte soubor uxDesc.pas, tak musíte přeložit program LOADER a ten poté zapsat do KIT. Nahrání nové verze LOADER můžete provést buď nahráním pomocí RTD, pomocí programátoru nebo pomocí programu LdrPatch.

- **AppDesc.pas:**

V tomto souboru se musí doplnit konstanty popisující struktury uložené v zachovávané oblasti statické paměti RAM. Tuto jednotku neměňte více než je nutné, protože v jejím těle jsou obsaženy funkce a definice pro zpracování chyb aplikace. Při jejich poškození byste přišli o užitečný nástroj při vývoji a sledování chodu aplikace, protože při použití objektu TWinTickLoader (jednotka LOADER) můžete chyby zobrazovat v programu TheKing pomocí okna **Restart structure** v menu **Views\Controller 1** jak při spuštění aplikace tak programu LOADER.

Pro účely ladění je vhodné doplnit inicializace oblastí těchto konstant do funkce fnAppInit, která slouží především na PC. Pokud jsou tyto konstanty nastavovány dle dodaných příkladů a je doplněna kontrola ve funkci fnAppInit, tak se už ve verzi pro

počítač PC nechají zjistit chyby při nastavení adres nebo změnách těchto struktur. Především se jedná o přepisy jednotlivých struktur.

7.3.2. Předefinování metod funkčních objektů

Pokud nastavíte paměťové konstanty a vytvoříte okna ve vizualizaci, můžete přikročit k poslednímu kroku - doplnění vizualizace do hotové aplikace případně můžete začít tuto aplikaci vytvářet. V rámci tohoto kroku doporučujeme vycházet z dodaných příkladů, viz Dodávané příklady.

Podrobný popis přidání vizualizace najdete v samostatném manuálu KitKing.

7.4. Nahrání aplikace do řídicího systému KIT

Pro nahrání aplikace do řídicího systému můžete použít program RTD nebo program TheKing. Při použití posledně jmenovaného programu musíte do řídicího systému nejdříve nahrát program LOADER, který najdete v podadresáři LOADER. Podrobný popis tohoto programu najdete v manuálu LOADER.

Při použití tohoto programu se Vám podstatně zjednoduší ladění aplikace, protože zobrazování dat a nahrávání vlastní aplikace provádíte z jednoho programu. Překlad a rozmístění programu LOADER provádíte jak jste zvyklý, tj. programem BP a RTD (rozmístění modulů aplikace).