# XG5000

# Export of 패키지 SW 설계

Produced by **이상훈 E sanghun.lee**

Exported from **HEAD**

Dec 20 2019 11:56

http://localhost:8080/cb/images/newskin/header/cblogo-xl.png

## Table of contents

[XG5000 1](#_Toc27735413)

[Export of 패키지 SW 설계 1](#_Toc27735414)

[Table of contents 2](#_Toc27735415)

[패키지 SW 설계 overview 3](#_Toc27735416)

[패키지 SW 설계 3](#_Toc27735417)

[4.1.2 XML 파일 구조 3](#_Toc27735418)

[XML의 파일의 대략적인 구조 4](#_Toc27735419)

[컨피그레인 별 구조 5](#_Toc27735420)

[1 속성 6](#_Toc27735421)

[2 하위노드 7](#_Toc27735422)

[글로벌 변수 구조 8](#_Toc27735423)

[POU 구조 10](#_Toc27735424)

[Program 11](#_Toc27735425)

[3 속성 11](#_Toc27735426)

[4 하위노드 12](#_Toc27735427)

[로컬 변수 구조 13](#_Toc27735428)

[프로그램 별 구조 13](#_Toc27735429)

[5 LD 편집기 13](#_Toc27735430)

[5.1 노드 및 속성 13](#_Toc27735431)

[5.2 LD 기본 구성 15](#_Toc27735432)

## 패키지 SW 설계 overview

패키지 소프트웨어 설계

## 패키지 SW 설계

##### 4.1.2 XML 파일 구조

|  |  |
| --- | --- |
| **CodeBeamer reference:** | [PDESIGN-5188506](#[ISSUE:5188506]) |
| **Revision:** | 48 |
| **Priority:** | -- |
| **Severity:** | -- |
| **Status:** | 오픈 |

Upstream references (0):

Downstream references (0):

Associations (0):

|  |  |  |
| --- | --- | --- |
| **From** | **Association Type** | **To** |

Description:

**Table of Contents**

* [**XML의 파일의 대략적인 구조**](#section-XML_EC_9D_98+_ED_8C_8C_EC_9D_BC)
* [**컨피그레인 별 구조**](#section-_EC_BB_A8_ED_94_BC_EA_B7_B8_EB_)
* [속성](#section-_EC_86_8D_EC_84_B1)
* [하위노드](#section-_ED_95_98_EC_9C_84_EB_85_B8_EB_)
* [**글로벌 변수 구조**](#section-_EA_B8_80_EB_A1_9C_EB_B2_8C+_EB)
* [**POU 구조**](#section-POU+_EA_B5_AC_EC_A1_B0)
* [**Program**](#section-Program)
* [속성](#section-1-_EC_86_8D_EC_84_B1)
* [하위노드](#section-1-_ED_95_98_EC_9C_84_EB_85_B8_E)
* [**로컬 변수 구조**](#section-_EB_A1_9C_EC_BB_AC+_EB_B3_80_EC)
* [**프로그램 별 구조**](#section-_ED_94_84_EB_A1_9C_EA_B7_B8_EB_)
* [LD 편집기](#section-LD+_ED_8E_B8_EC_A7_91_EA_B8_B0)
* [노드 및 속성](#section-_EB_85_B8_EB_93_9C+_EB_B0_8F+_E)
* [LD 기본 구성](#section-LD+_EA_B8_B0_EB_B3_B8+_EA_B5_AC)

## XML의 파일의 대략적인 구조

XML프로젝트는 기본적으로 다음과 같은 구조로 되어있다.

![http://165.243.151.32:8080/cb/displayDocument/1576799203910.png?task_id=5188506&artifact_id=767396&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

|  |  |
| --- | --- |
| 노드 | 설명 |
| Project | 프로젝트 주 노드로, 속성 및 버전 등의 정보 포함 |
| NetworkConfiguration | 프로젝트 내부의 네트워크 정보 구성 |
| SystemVariable | 시스템 변수(네트워크에서 사용된 변수) |
| Configurations | 컨피그레이션 목록 (멀티 PLC) |
| Configuration | 각 컨피그레이션별 속성 |

## 컨피그레인 별 구조

XG5000내에서 컨피그레이션(Configuration)은 PLC, HMI, DRIVE 등 기기 단위의 구성을 의미한다. PLC를 제외한 항목의 경우에는 XG5000 내부적으로 구성을 변경할 수 있는 항목은 아니므로, 본 문서에서 추가적인 설명은 하지 않는다.

![http://165.243.151.32:8080/cb/displayDocument/d6d490a66480e29f829e54e844ca3fc7.png?task_id=5188506&artifact_id=767410&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

### 속성

|  |  |
| --- | --- |
| 속성 | 설명 |
| Version | PLC 구조에 대한 버전으로, 과거 호환성 때문에 사용됨  현재는 0x103으로 고정 |
| Attribute | PLC에 특정 기능이 있는지에 대한 속성값으로, 다음의 값의 OR된 값이 저장된다.  #define HAS\_CONFIG\_GLOBAL\_VAR 0x00000001  #define HAS\_CONFIG\_ACCESS\_VAR 0x00000002  #define HAS\_CONFIG\_DIRECT\_COMMENT 0x00000004   #define HAS\_RESOURCE\_GLOBAL\_VAR 0x00000010  #define HAS\_USER\_FUN\_FB 0x00000020  #define HAS\_USER\_TASK 0x00000040   #define HAS\_USER\_LIBRARY 0x00000100  #define HAS\_USER\_DATA\_TYPE 0x00000200  #define HAS\_LOCAL\_VAR 0x00000400   #define HAS\_BASIC\_PARA 0x00001000  #define HAS\_IO\_PARA 0x00002000  #define HAS\_INTERNAL\_PARA 0x00004000 // 2005.7 XGB 추가  #define HAS\_LOCAL\_ETH\_PARA 0x00008000 // NGP2000 Ethernet   #define HAS\_REDUNDANCY\_PARA 0x00010000 // 2007.8.7 이중화 para  #define HAS\_PB\_POOL 0x00020000 // 2011.5.26  #define HAS\_NETWORK\_PARA 0x00020000 // 2010.10.27 CANopen 추가  #define HAS\_EXTRA\_INFO 0x00080000 // serialize extension   #define IEC\_CONFIG 0x10000000 // IEC 형  #define SAFETY\_CONFIG 0x20000000 //SAFETY 형 |
| Kind | 컨피그레이션에 대한 유형 값   enum E\_SYSCON\_ITEM\_TYPE  {  SYSCON\_TYPE\_UNKNOWN = 0,  SYSCON\_TYPE\_PLC ,  SYSCON\_TYPE\_HMI ,  SYSCON\_TYPE\_INV ,  SYSCON\_TYPE\_NETWORK ,  SYSCON\_TYPE\_MOTION ,  SYSCON\_TYPE\_EXT\_ADAPTER ,   SYSCON\_TYPE\_FAKE = 100,  }; |
| Type | PLC에 대한 타입 값으로 XG5000\l.kor\symbol.mdb 파일의 [PLC\_TYPE\_LIST].nPLCID에 해당함 |

### 하위노드

|  |  |
| --- | --- |
| 노드 | 설명 |
| Parameters | PLC 내부의 파라미터 구조  유형에 따라 기본 파라미터, I/O 파라미터, 자동할당 파라미터, 내장 이더넷 파라미터 등이 구성된다 |
| GlobalVariables | 글로벌 변수 목록에 대한 최상위 노드 |
| Tasks | 태스크에 대한 최상위 노드 |
| POU | 프로그램 구성요소에 대한 최상위 노드 |
| OnlineElements | PLC에 다운로드 되어, 오프라인 프로젝트와의 일치성을 확인하기 위하여 사용되는 데이터. 프로그램 테이블, 테스크 테이블 등이 이에 포함됨 |
| XGPD | 통신에 대한 파라미터로 기본 파라미터, P2P 파라미터 및 고속링크 파라미터 저장 |
|  |  |

## 글로벌 변수 구조

![http://165.243.151.32:8080/cb/displayDocument/431d20ba93cc4637dce6388de5ac5e40.png?task_id=5188506&artifact_id=767513&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

|  |  |
| --- | --- |
| 속성/노드 | 설명 |
| Version | 글로벌 변수 버전 |
| Count | 하위 심볼에 대한 개수 |
| Symbols | 심볼 목록에 대한 상위 항목 |
| Symbol | 심볼에 대한 정의 |
| Name | 변수 명 |
| Kind | 변수 유형  typedef enum \_var\_kind\_2 { NULL1 = 0 , // NEVER change the enum value of NULL1 !!!  // It influences GlobalVarMem allocation!  TYPE = 1,  VAR\_IN\_OUT = 2 ,  VAR\_IN = 3 ,  VAR\_OUT = 4 ,  VAR\_OUT\_RETAIN = 5 ,  VAR = 6 ,  VAR\_CONSTANT = 7 ,  VAR\_CONSTANT\_RETAIN = 8 ,  VAR\_RETAIN = 9 ,  VAR\_EXTERN = 10 , // if PLC\_TYPE != 1  VAR\_GLOBAL = 11 ,  VAR\_GLOBAL\_CONSTANT = 12,  I\_DIRECT\_ADDRESS = 13, // used only in BODY compile  Q\_DIRECT\_ADDRESS = 14, // used only in BODY compile  M\_DIRECT\_ADDRESS = 15, // used only in BODY compile  FUNCTION\_KIND = 16, // used only in BODY compile  FUNCNAME = 17, // used as Function block name & output of FUN  SFC\_TRANS = 18,  FUN\_STRCONST = 19, // when string constant is used in FUN  VAR\_EXT\_IN = 20, // following are for GM1 extern var!!  VAR\_EXT\_OUT = 21,  VAR\_EXT\_IN\_OUT = 22,  ARY\_EXT\_IN = 23,  ARY\_EXT\_OUT = 24,  ARY\_EXT\_IN\_OUT = 25,  SFC\_STEP1 = 26 // 98/2/12 초기스텝으로 돌아가는 GOTO\_S1 변수에만 쓰이는 타입 .  } var\_kind\_gmwin ; |
| Type | 변수 타입 종류 (FB명, UDT명, ARRAY[1..2,1..3] OF BOOL, WORD, DWORD,...) |
| State | 변수의 내부 속성  const UINT STATE\_RETAIN = 0x0001 ;  const UINT STATE\_USEDIT = 0x0002 ; // 사용 유무는 PLC 에 다운로드하지 않는다 . ( 체크섬이 변경되는 현상발생 )(2016.12.08)  const UINT STATE\_READONLY = 0x0004 ;  const UINT STATE\_SPECIAL = 0x0008 ; |
| Address | 글로벌 변수일 경우, IEC 메모리 할당 (NULL, <AUTO>, %IXO, %QX1.1.2, %MX45,...) |
| Trigger | 트리거 속성 (사용자 펑션 및 펑션블록) |
| InitValue | 초기 값 |
| Comment | 설명문 |
| Device | 디바이스명 (P, M, L, ..) |
| DevicePos | 디바이스 위치 |
| TotalSize | struct, fb/fun인 경우 메모리 할당시 멤버 사이즈 |
| OrderIndex | struct, fb/fun인 경우 메모리 할당시 순서 |
| HMI | HMI 공유 속성 |
| EIP | EIP 공유 속성 |
| SturctureArrayOffset | 구조체 배열 옵셋값 |
| ModuleInfo | 특수/통신 모듈 변수 정보 |
| ArrayPointer | 사용자 펑션 및 펑션 블록의 배열 인덱스 유무 설정 |
| MemberAddresses | <이름, 값> |
| MemberRetains | <이름, 값> |
| MemberInitValues | <이름, 값> |
| MemberComments | |

## POU 구조

POU(Program Organization Unit)은 프로그램, 사용자 펑션/펑션블록, 사용자 데이터 타입 및 사용자 라이브러리로 구성된다. 그 중 Programs는 프로젝트 내의 프로그램에 대한 목록이며, 프로젝트 트리의 순서로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/27cec88d56d90028c9ca23650d37da0a.png?task_id=5188506&artifact_id=767422&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

## Program

프로그램은 프로그램 자체와 로컬 변수로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/0158575556d3bbe475817993e92855cc.png?task_id=5188506&artifact_id=767427&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

### 속성

|  |  |
| --- | --- |
| 속성 | 설명 |
| Task | 프로그램이 속한 태스크 명으로, Tasks의 하나 노드를 참조한다. |
| Version | 프로그램에 대한 버전으로 현재는 0x100 |
| LocalVraiable | 로컬 변수 유무를 나타내는 속성 |
| Kind | 프로그램 유형  #define LD\_EDITOR 0  #define IL\_EDITOR 1  #define SFC\_EDITOR 2  #define SFC\_MANAGER\_EDITOR 3  #define ST\_EDITOR 4  #define FBD\_EDITOR 5  #define PD\_EDITOR 6  #define GCODE\_EDITOR 7  #define LIB\_EDITOR 8  4define ILT\_EDITOR 9 // IL Text editor |

### 하위노드

|  |  |
| --- | --- |
| 속성 | 설명 |
| Task | 프로그램이 속한 태스크 명으로, Tasks의 하나 노드를 참조한다. |
| Version | 프로그램에 대한 버전으로 현재는 0x100 |
| LocalVraiable | 로컬 변수 유무를 나타내는 속성 |
| Kind | 프로그램 유형  #define LD\_EDITOR 0  #define IL\_EDITOR 1  #define SFC\_EDITOR 2  #define SFC\_MANAGER\_EDITOR 3  #define ST\_EDITOR 4  #define FBD\_EDITOR 5  #define PD\_EDITOR 6  #define GCODE\_EDITOR 7  #define LIB\_EDITOR 8  4define ILT\_EDITOR 9 // IL Text editor |

## 로컬 변수 구조

<글로벌 변수 구조 참조>

## 프로그램 별 구조

### LD 편집기

LD 프로그램은 LDRoutine으로 시작하며 Rung 및 Rung에 속한 Element로 구성된다. 아래 그림에서 OnlineUploadData의 경우에는 PLC에 다운로드 되는 데이터로 바이너리로 인코딩 되어 있으며, 본 문서에서 상세 설명은 하지 않는다.

![http://165.243.151.32:8080/cb/displayDocument/f229842af700028fe7282d790d420dc5.png?task_id=5188506&artifact_id=767446&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

#### 노드 및 속성

|  |  |
| --- | --- |
| 항목 | 설명 |
| Rung | 하나의 LDRoutine은 N개의 Rung으로 구성된다. |
| BlockMask | 비실행문 설정 유무 |
| Element | LD 개별 구성 요소 |
| ElementType | LD 프로그램 상에서의 구성요소에 대한 식별자  typedef enum {  LDElementMode\_Start = 0,  LineType\_Start = 0,  VertLineMode = LineType\_Start , //'|'  HorzLineMode , // '-'  MultiHorzLineMode , // '-->>'  //add, only here, additional line type device.  LineType\_End = 5,   ContactType\_Start = 6,  ContactMode = ContactType\_Start , // '-| |-'  ClosedContactMode , // '-|/|-'  PulseContactMode , // '-|P|-'  NPulseContactMode , // '-|N|-'  //add, only here, additional contact type device.  ContactType\_End = 13,   CoilType\_Start = 14,  CoilMode = CoilType\_Start , // '-( )-'  ClosedCoilMode , // '-(/)-'  SetCoilMode , // '-(S)-'  ResetCoilMode , // '-(R)-'  PulseCoilMode , // '-(P)-'  NPulseCoilMode , // '-(N)-'  //add, only here, additional coil type device.  CoilType\_End =30,   FunctionType\_Start = 31,  FuncMode ,  FBMode , // '-[F]-'  FBHeaderMode , // '-[F]-' : Header  FBBodyMode , // '-[F]-' : Body  FBTailMode , // '-[F]-' : Tail  FBInputMode ,  FBOutputMode ,  //add, only here, additional function type device.  FunctionType\_End = 45,   BranchType\_Start = 51,  SCALLMode ,  JMPMode ,  RetMode , //  SubroutineMode ,  BreakMode ,  ForMode ,  NextMode ,  //add, only here, additional branch type device.  BranchType\_End = 60,   CommentType\_Start = 61,  InverterMode , // '-\*-'  RungCommentMode , // 'rung comment'  OutputCommentMode , // 'output comment'  LabelMode ,  EndOfPrgMode ,  RowCompositeMode , // 'row'  ErrorComponentMode ,  NullType ,  VariableMode ,  CellActionMode ,  //add, only here, additional comment type device.  CommentType\_End = 90,    // vertical function(function & function block) related  VertFunctionType\_Start = 100,  VertFuncMode ,  VertFBMode ,  VertFBHeaderMode ,  VertFBBodyMode ,  VertFBTailMode ,  // add additional vertical function type device here  VertFunctionType\_End = 109,  LDElementMode\_End = 110,   Misc\_Start = 120,  ArrowMode ,  Misc\_End ,   } LDComponentType ; |
| Coordninate | 각 항목이 위치하는 좌표로, 좌표계의 구성은 문서 하단 참조 |
| Text | 화면상에 표시되는 텍스트 |
| Param | 각 항목별 요소마다 다름  연결선: 끝좌표  명령어: 명령어, 파라미터1, 파라미터2, … 파라미터 N  펑션/펑션블록:  FNAME: [이름]\n  TYPE: [function][function\_block][,user]\n  INSTANCE: [인스턴스 명]\n  INDEX: [펑션에 대한 index (XG5000\l.kor\cmdd.mdb)]\n  COL\_PROP: [칼럼 너비]\n  SAFETY: [Safety 속성]\n  VAR\_IN: 이름, 데이터 타입, 추가 데이터, 트리거 속성\n  …  VAR\_IN\_OUT:  …  VAR\_OUT |

#### LD 기본 구성

프로그램: LD 프로그램 전체

![http://165.243.151.32:8080/cb/displayDocument/91671da60e577a1bb721682ac0f126c5.png?task_id=5188506&artifact_id=767448&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

렁: 수직선 또는 펑션/펑션으로 연결된 라인의 집합

![http://165.243.151.32:8080/cb/displayDocument/86ea5d196a8c48a1ae67f7841799d1a3.png?task_id=5188506&artifact_id=767447&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

라인: 하나의 라인에 대한 데이터

![http://165.243.151.32:8080/cb/displayDocument/648dca9e5d3c1fe08aebe17ccb144429.png?task_id=5188506&artifact_id=767449&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

명령: XGK 혹은 XGI의 확장 명령어로 하나의 명령은 header + tail 및 body 블록으로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/7a3e7693b46a289f330abe7565086f37.png?task_id=5188506&artifact_id=767450&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

펑션/펑션블록: 펑션/펑션블록은 header + tail 블록 및 body 블록로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/a0de226cf1d33cbd7978436098cada98.png?task_id=5188506&artifact_id=767458&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

연결선: 시작과 끝을 갖는 긴 수평 라인

![http://165.243.151.32:8080/cb/displayDocument/427fc8e81f1a1b73613e44af4a8389c6.png?task_id=5188506&artifact_id=767459&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

수직선: 수직선은 2개의 라인을 연결하는 것으로, 상위/하위 라인으로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/e7abdd2e685037699901fd8523d73df9.png?task_id=5188506&artifact_id=767457&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)

LD 좌표계: LD의 좌표계는 32비트의 정수로 나타내지며, 수평셀은 1, 4, 7의 형태로 증가하며 라인은 1024 단위로 증가한다. 수직라인의 경우 상위로 라인으로의 연결은 셀 + 1, 하위로의 연결은 셀 + 2로 구성된다.

![http://165.243.151.32:8080/cb/displayDocument/e06631f9fcbb24a7f1a238332eb7ebb9.png?task_id=5188506&artifact_id=767463&version=1&raw=true&history=false&notification=false](data:application/xhtml+xml;base64,)