

In Circuit Scope Plus (ICS++) 取扱説明書

リアルタイム・変数波形表示ツール

要旨

In Circuit Scope Plus はターゲットシステム上で実行されるプログラムのグローバル変数値をリアルタイムにパソコン上に波形表示することができるツールです。

◎対象 ICS++

本取り扱い説明書は、下記の機種に対応しています。

ICS++シリーズ

W1004, W2001, W2002, W2901

ICS++下位互換機能を持つスタータキット T2001C, T2006A

旧 ICS シリーズ

W1001, W1002, W1003

ICS 下位互換機能を持つスタータキット T2001, T2001A, T2001B, T2002, T2002A, T2002B

◎対象 PC 側ソフトウェア

DTLScope.exe Ver.1.5.0.0

DTLScope.exe Ver.1.0.0.0 に対して上位互換性があります。

本ソフトウェアは、旧世代の ICS / ICS++でも動作します。

DTLScope.exe Ver.1.0.0.0 の操作法は、P00301-A2-023_In Circuit_Scope 取り扱い説明書をご覧ください

◎対象ターゲット側 ICS ライブラリ

どの CPU バージョンの ICS ライブラリでも新しいシステムで使用可能です。ただし、新しいライブラリソフトウェアは、機能や性能が向上しているため、新しいライブラリをご使用になることを推奨します。

◎対象開発環境

弊社では、下記の環境で動作する環境を整備中です。詳細バージョンなどがあるため、具体的な環境については、個別に弊社にお問い合わせください。

RENESAS CS+ CA78K0R コンパイラ

RENESAS CS+ CC-RL コンパイラ

RENESAS CS+ CC-RX コンパイラ

IAR Embedded Workbench

◎対象デバイス

下記のデバイスで動作実績があります。

○16bit 整数型デバイス：

RENESAS

RL78G1F, RL78F13, RL78F14, RL78G14, RL78/G12, RL78/G13, 78K0RIC3, 78K0R/ID3, 78K0R/IE3

○32bit 整数型デバイス：

RENESAS

RX111, RX210, RX220, V850/IA3, V850/IA4, V850E2/IG3, V850E/FJ3, SH7047, SH7146, SH7149, SH7237,

○32bit 浮動小数点型デバイス：

RENESAS

RX23T, RX24T, RX62T, RX63T, RX631, RX64M, RX71M, V850E2M/FJ4, V850E2/ML4, SH7216, SH7239,

その他のデバイスのサポートについては、株式会社デスクトップラボ <http://desktoplab.co.jp> にお問い合わせください。ターゲット側ライブラリの開発により、UART と DMA/DTC を持つ 16bit 以上のほとんどの CPU に対応可能です。デスクトップラボでは、有償でお客様仕様のカスタムライブラリを受託開発しております。

目次

1.	概要	5
1.1	概要	5
1.2	特徴	5
1.3	接続構成	6
1.4	ソフトウェア構成	6
2.	ICS++仕様	7
2.1	製品構成	7
2.2	動作環境	7
2.3	ツール仕様	7
3.	ICS++使用方法	9
3.1	使用の流れ	9
3.2	起動	10
3.3	変数情報ファイル読み込み	12
3.4	画面の見方	14
3.5	Menu Window	19
3.5.1	File メニュー	19
3.5.2	Settings メニュー	20
3.6	Tools メニュー	21
3.6.1	Array Editor	22
3.6.2	Custom Control Panel	23
3.6.3	Timetable player 機能 (Function generator 機能)	24
3.7	Help メニュー	26
3.8	COM ポート選択	27
3.9	プロジェクトを開く	27
3.10	変数情報の更新	28
3.11	オシロスコープ機能	29
3.11.1	波形表示変数設定	29
3.11.2	横軸レンジとサンプリング周期設定	31
3.11.3	縦軸レンジとオフセット設定	32
3.11.4	トリガ設定	32
3.11.5	カーソル設定	35
3.11.6	ZOOM 画面設定	36
3.11.7	波形データ保存／読み込み	38
3.12	変数値読み込み／書き込み機能	39
4.	ICS++ PC ソフトウェア(DTLScope.exe) のインストール	41
4.1	DTLScope.exe (ICS++用ソフトウェア) のインストール	41
4.1.1	PC 側ソフトウェアのインストール	41
4.1.2	USB ドライバーインストール (W2001, W2002, T2001C, T2006 以降の ICS バージョン)	41
4.1.3	USB ドライバーのインストール(W1003, W1004, T2001B, T2002B)	41
5.	ICS++システムの接続	46
5.1	ICS++システムの接続	46
5.2	PC と ICS++ユニットとの接続	46
5.3	ターゲット I/F ユニットと ICS++ユニットとの接続	47
5.4	ターゲット CPU とターゲット I/F ユニットとの接続	48
5.4.1	電氣的仕様概要	48
5.4.2	端子処理具体例	49
5.5	ICS++ユニット、ターゲット IF ユニット、ターゲット CPU 接続例	49
6.	注意事項	50
6.1	ICS++のサポート CPU について	50

6.2	ICS++の通信クロック周波数計算方法	50
6.2.1	RX24T の場合	50
7.	ホームページとサポート窓口	51
7.1	サポート	51
8.	改訂記録	51

1. 概要

1.1 概要

In Circuit Scope Plus は、パワーエレクトロニクス向けのリアルタイムデバッグツールです。



図 1-1 ICS++ (PC 側ソフトウェア外観 : DTL_Scope Ver.1.5.0.0)

1.2 特徴

グローバル変数に対して下記の操作が可能です。

- (1) リアルタイムにパソコン上に波形表示することが可能です。
- (2) レンジの変更やトリガの設定など、オシロスコープと同じ感覚で使用可能です。
- (3) ICS++上から、任意の値をグローバル変数に書き込むことができるため、プログラム実行中にゲインの調整等が可能です。
- (4) ICS++上から、変数に対する時系列の値を書き込むことができるため、運転パターンなどの生成機能があります。
- (5) 表示した値を長時間測定しファイルとして残す、データロガーのような使い方が可能です。
- (6) 波形の周波数解析機能 (FFT)

1.3 接続構成

ターゲット・プロセッサから ICS++ユニットを経由してパソコンに接続します。接続の際は本紙の3.1節の使用の流れをご参考ください。また、ICS++ユニットに関しては4章をご参考ください。

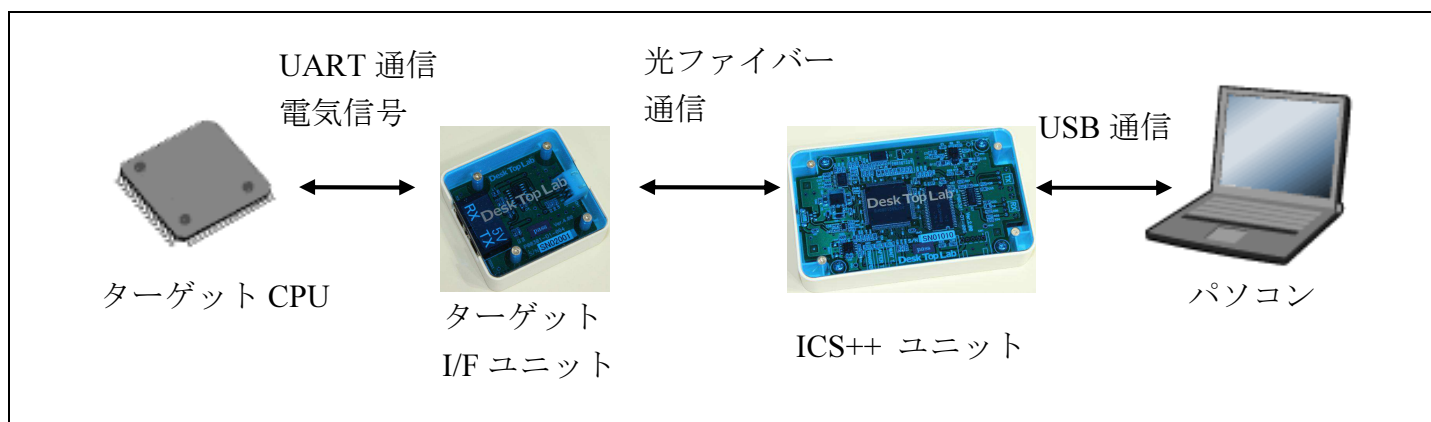


図 1-2 ハードウェア構成

1.4 ソフトウェア構成

ICS++ は変数情報ファイルに格納されている変数名およびメモリアドレス情報に基づき、ターゲット CPU から ICS++ボードを経由して送信されるデータを処理し、パソコン上に波形表示します。

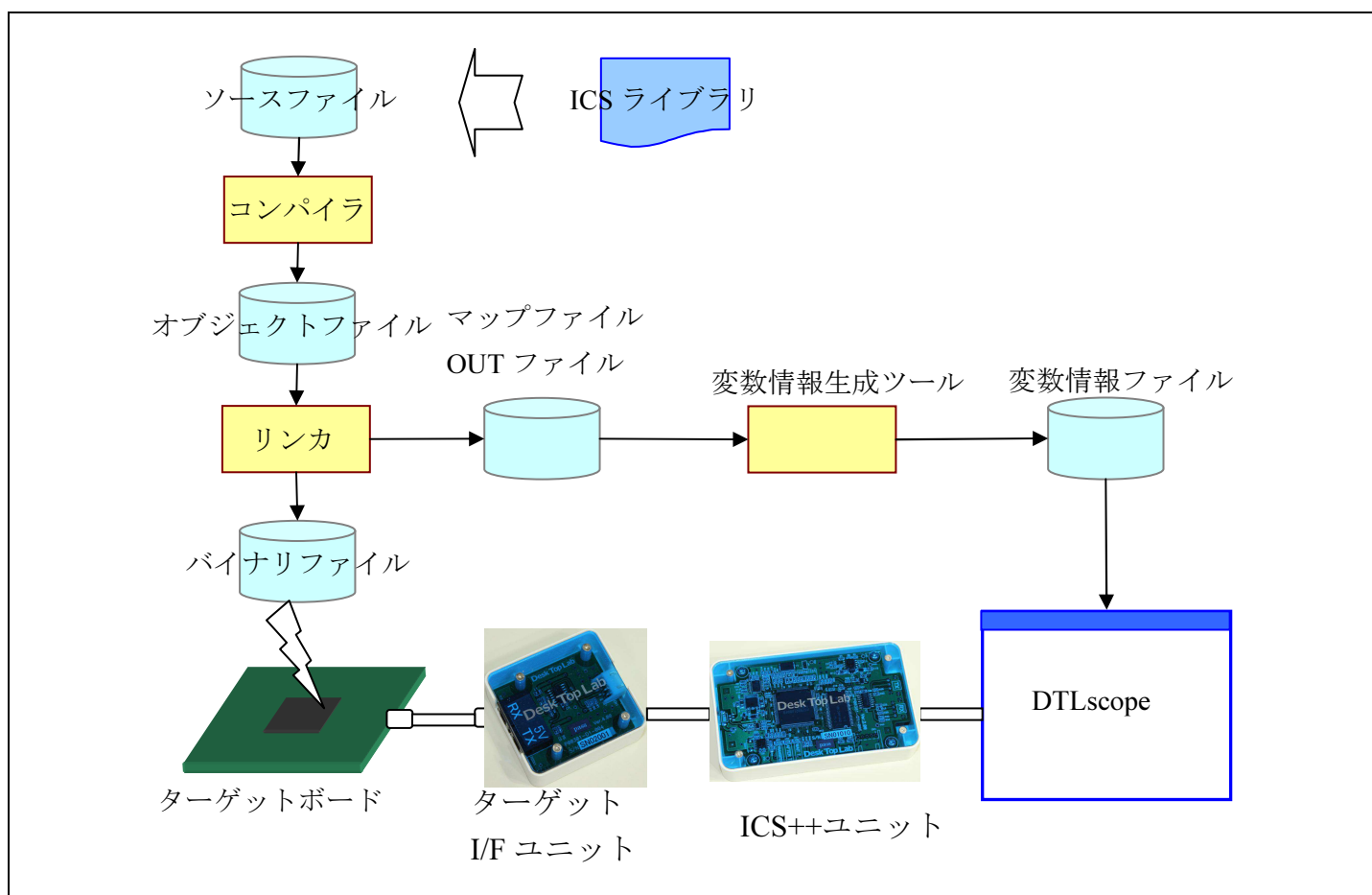


図 1-3 ソフトウェア構成

2. ICS++仕様

2.1 製品構成

本製品には以下のものが含まれています。

表 2-1 製品構成

No.	項目	
1	In Circuit Scope Plus ユニット	1 台
2	ターゲット I/F ユニット (5V CPU 用)	1 台
3	ターゲット I/F ユニット (3.3V CPU 用)	1 台
4	光ファイバー 2m	2 本
5	USB ケーブル	1 本
6	ターゲット I/F ケーブル 1 片側 XHP-4	1 本
7	ターゲット I/F ケーブル 2 両側 XHP-4	1 本
8	DVD-ROM	1 枚

2.2 動作環境

ICS++をインストールする上で必要なスペックを以下に示します。

ただし、高速な PC を推奨します。波形の更新速度は、PC の性能に依存します。

表 2-2 動作環境

No.	項目	内容
1	パソコン	空きディスク容量: 100M [byte] 以上 RAM: 1G [byte] 以上 USB ポート : 1 [チャンネル](USB2.0, USB3.0) .Net framework client 4.5 インストール済
2	OS	Windows 7, Windows 8, Windows 10 x86, x64

2.3 ツール仕様

ICS++ (W2002)の仕様概要一覧を以下に示します。

表 2-3 ICS++仕様概要

項目	内容
測定周期	ターゲット CPU に組みこむライブラリに依存
波形測定チャンネル	最大 12 [チャンネル] 使用ライブラリに依存
測定変数型名	ターゲット CPU に組みこむライブラリに依存
トリガモード	Single / Auto / Normal
トリガソース	測定チャンネル 12 [チャンネル]
数値表示変更チャンネル	最大 2 4 変数
データ管理	波形データ Save / Load 機能
波形表示画面	メイン画面 + ZOOM 画面(1 画面)
使用ポート	TXD, RXD
使用周辺機能	UART 1 [チャンネル], DMA or DTC, (FIFO)

対応 CPU	<p>16bit RL78G14, RL78F14, RL78G1F, RL78G12, RL78G13, 78K0R/Ix3</p> <p>32bit 浮動小数点 : RX62T, RX63T, RX64M, V850E2M/FJ4, V850E2/ML4, SH7216, SH7239,</p> <p>32bit 固定小数点 RX111, RX210, RX220, V850E2/IG3, SH7047, SH7146, SH7149, SH7237,</p> <p>16bit 以上の DMA, DTC を内蔵するほぼ全ての CPU に対応可能。具体的な CPU については、デスクトップラボにお問い合わせください</p> <p>※2017 年 3 月 23 日時点で確認している使用できない CPU</p> <p>RL78/G1G,</p>
--------	--

3. ICS++使用方法

3.1 使用の流れ

ICS++をご使用の際は、(1)～(9)の順に準備してください。

- (1) ICS++ unit とターゲットボードを接続 (4 章を参照)
- (2) ICS++ unit とパソコンを USB ケーブルで接続
- (3) (初回のみ) USB ドライバーのインストール
- (4) ターゲットボードへ電源を供給
- (5) ターゲット CPU へプログラムをダウンロード
- (6) プログラム実行
- (7) ICS++起動 (3.2節)
- (8) 通信の確立を確認 (3.2節)
- (9) 変数情報ファイル読み込み (3.3節)
- (10) 準備完了

3.2 起動

DTLScope.exe ファイルを実行し、ICS++を起動してください。起動するとメイン画面が表示されます。

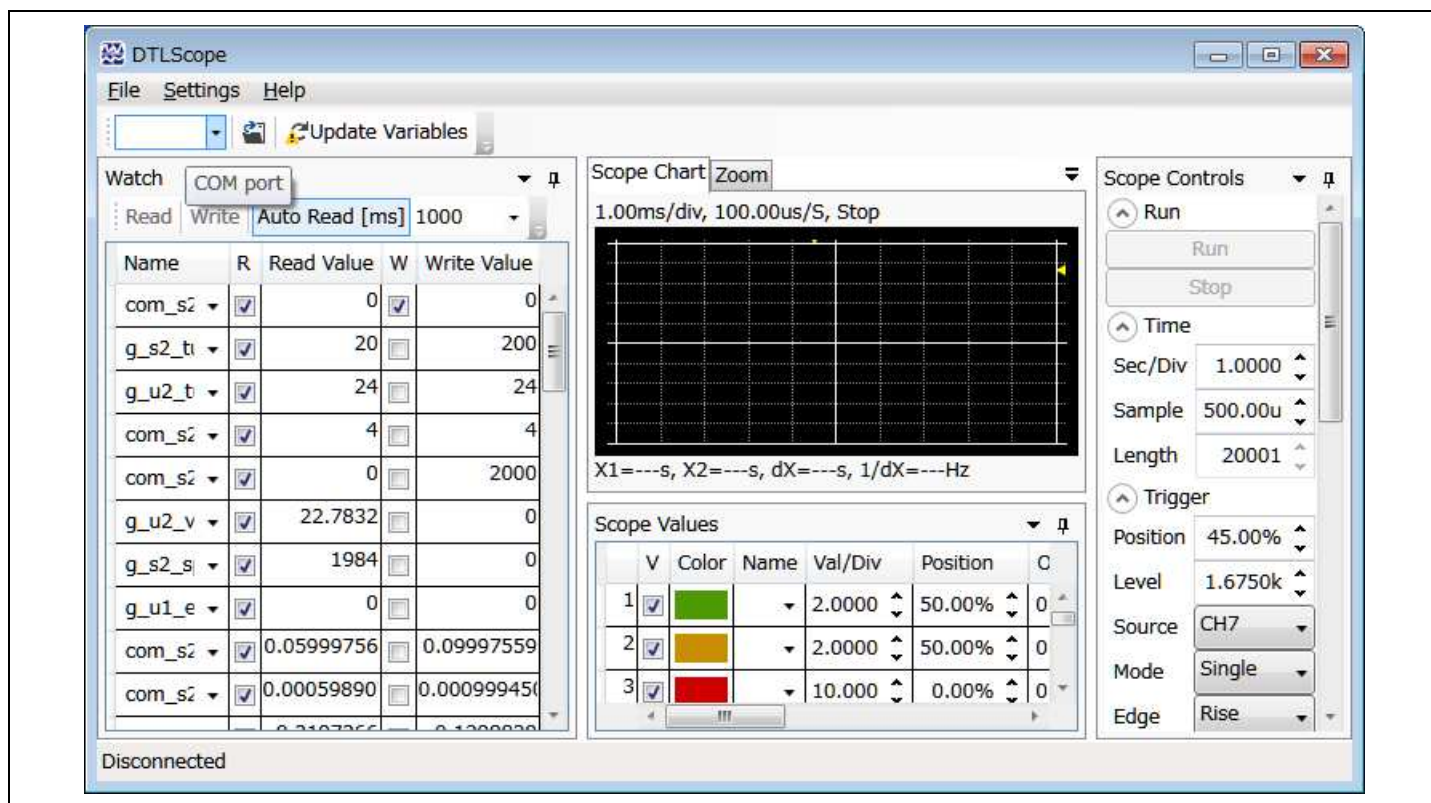


図 3-1 メイン画面

正しく接続されている場合、パソコン側に接続された COM ポートは自動選択され、ターゲット・プロセッサとパソコン間の通信を開始します。画面下部には接続状態が表示され、以下のようになります。

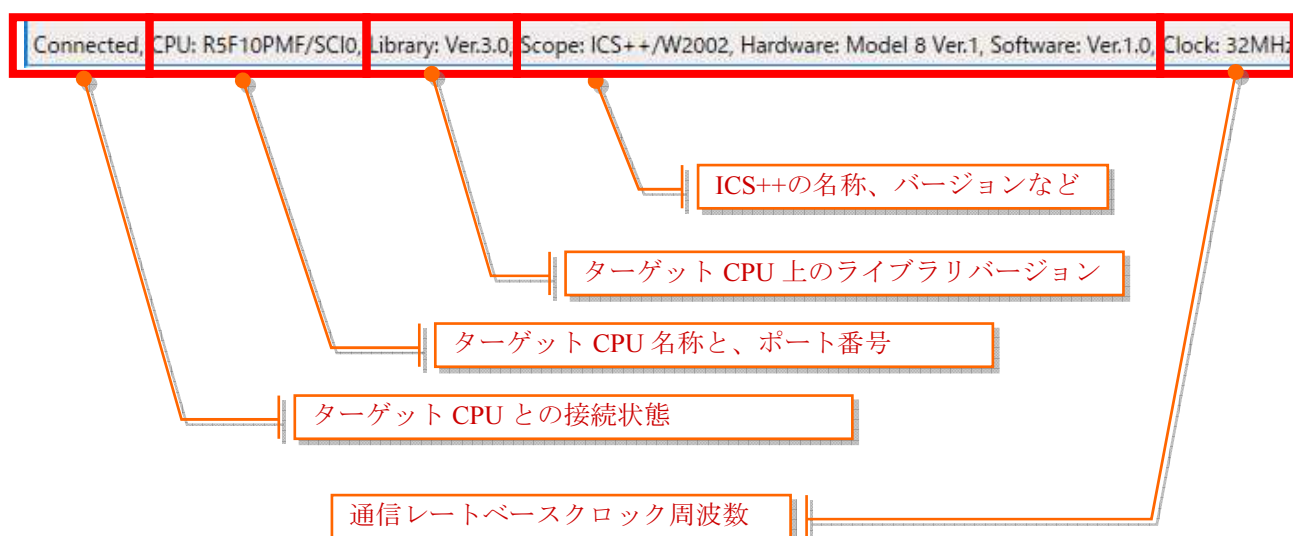
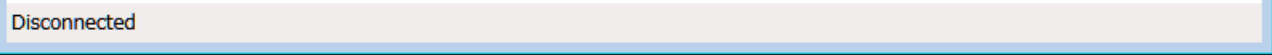


図 3-2 ステータスバー

ICS++ユニットを認識できない場合、ステイタスバーに “Disconnected” と表示されます。この場合、再度3.1節の使用の流れをご確認ください。特に、PC と ICS++ユニット間の接続と PC ソフト上の COM ポートの設定のご確認を御願います。



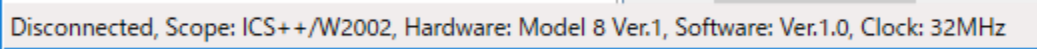
Disconnected

図 3-3 ステイタスバー（ICS++を認識しない場合）

ICS++ユニットのみを認識し、ターゲット CPU を認識できない場合

“Disconnected, Scope: I.C.Scope++, Hardware: Model 8, Ver.1, Software Ver.1.0, Clock: 8Mhz”

と ICS++のバージョンや名称のみが表示され、CPU 名が表示されません。



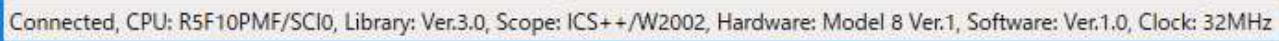
Disconnected, Scope: ICS++/W2002, Hardware: Model 8 Ver.1, Software: Ver.1.0, Clock: 32MHz

図 3-4 ステイタスバー（ICS++を認識し、ターゲット CPU を認識しない場合）

ICS++ユニットとターゲット CPU との両方とも認識した場合。

“Connected, CPU:R5F10PMF/SCI0, Library: Ver2.0, Scope: I.C.Scope++, Hardware: Model 7, Ver., Software Ver.1.35, Clock: 8Mhz”

と ICS++の情報、CPU の情報、ライブラリの情報を表示します。



Connected, CPU: R5F10PMF/SCI0, Library: Ver.3.0, Scope: ICS++/W2002, Hardware: Model 8 Ver.1, Software: Ver.1.0, Clock: 32MHz

図 3-5 ステイタスバー（ICS++、ターゲット CPU の両方を認識した場合）

3.3 変数情報ファイル読み込み

ICS++ PC ソフトウェアは、起動後にターゲット CPU に書き込んだプログラムの変数情報ファイル(拡張子: csv)を読み込む必要があります。ファイルメニュー⇒Load Variables をクリックして、変数情報ファイルをロードします。

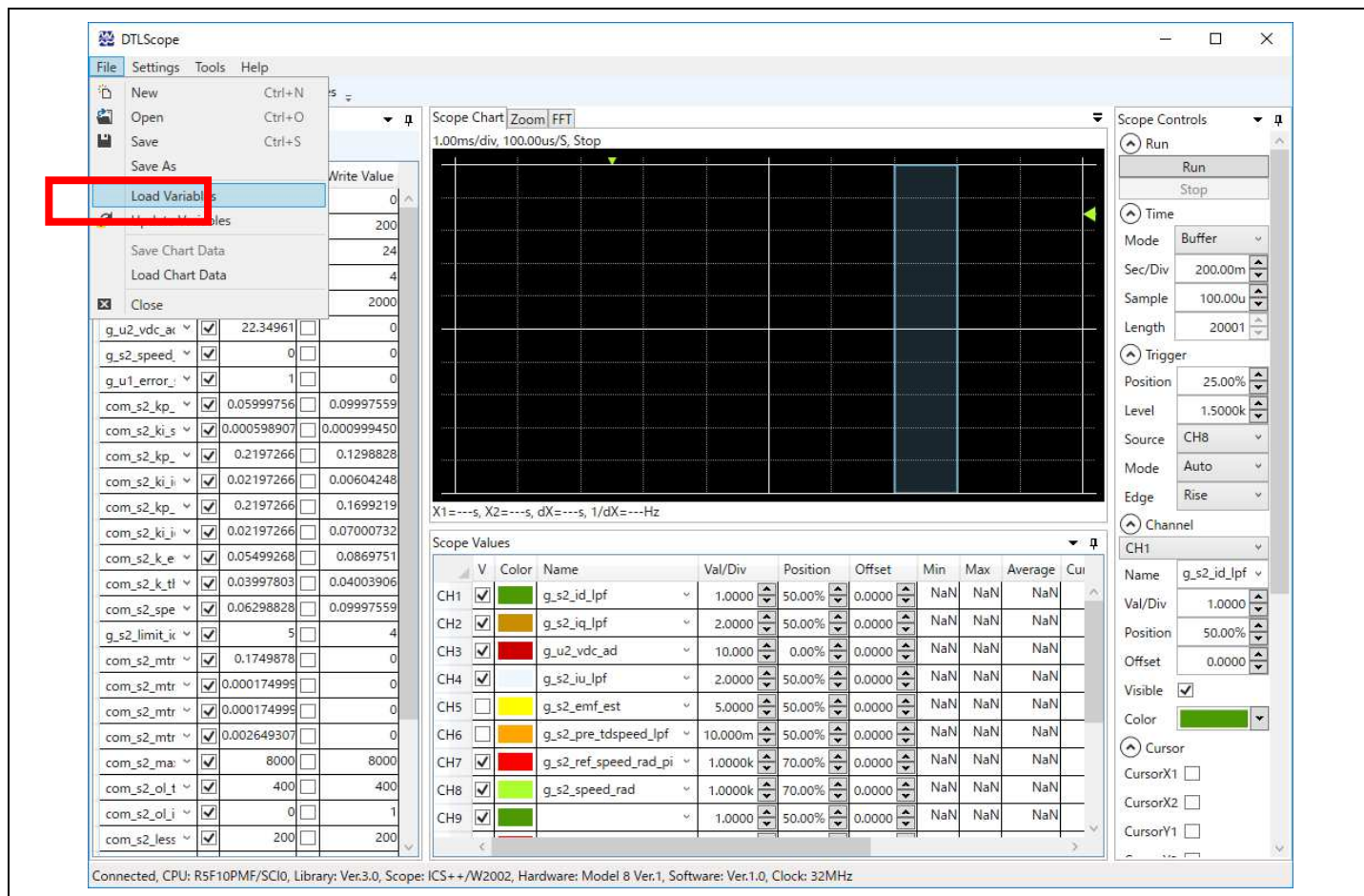


図 3-6 Load Variables

下図のような画面が表示されるので、生成した変数情報ファイル(拡張子: csv)を選択してください。

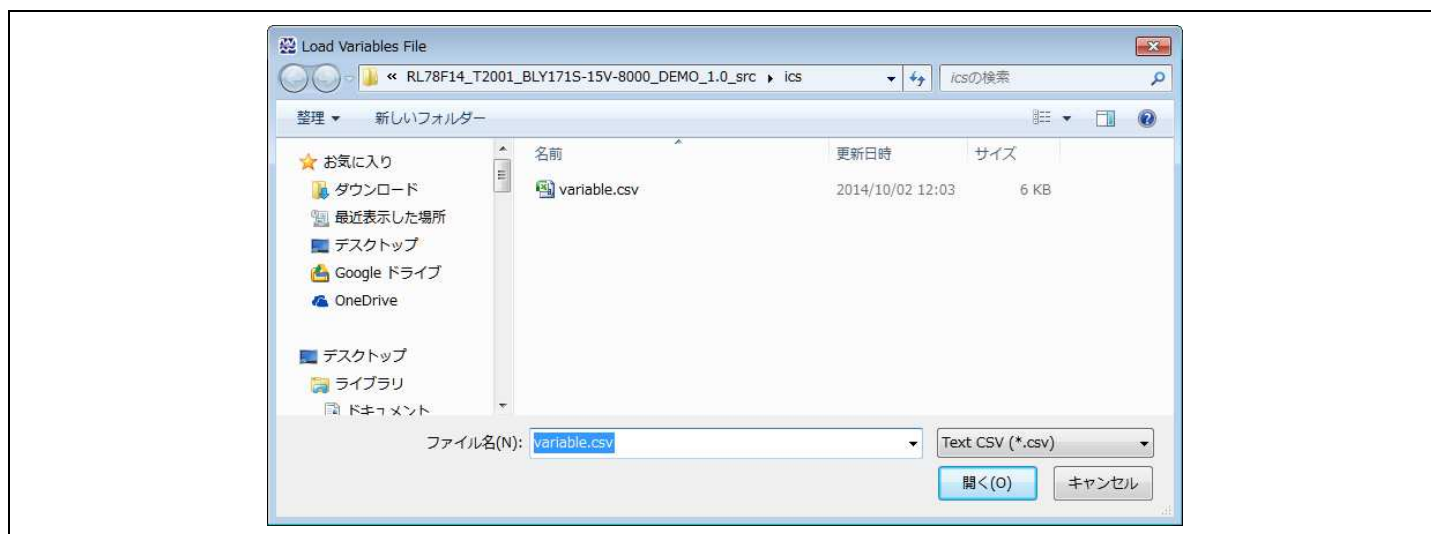


図 3-7 変数情報ファイル選択画面

”Variable information has been loaded” と表示されれば読み込み完了です。

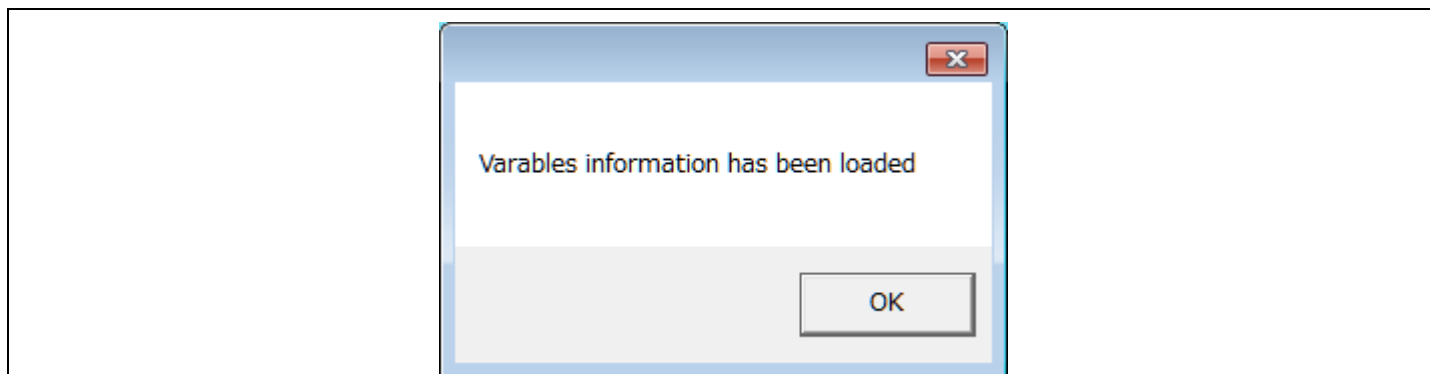


図 3-8 変数情報ファイル読み込み成功

3.4 画面の見方

ICS++の PC ソフトは 5 つの画面で構成されます。この節では、各画面の機能概要について説明します。各画面の詳細については以降の節を参照ください。

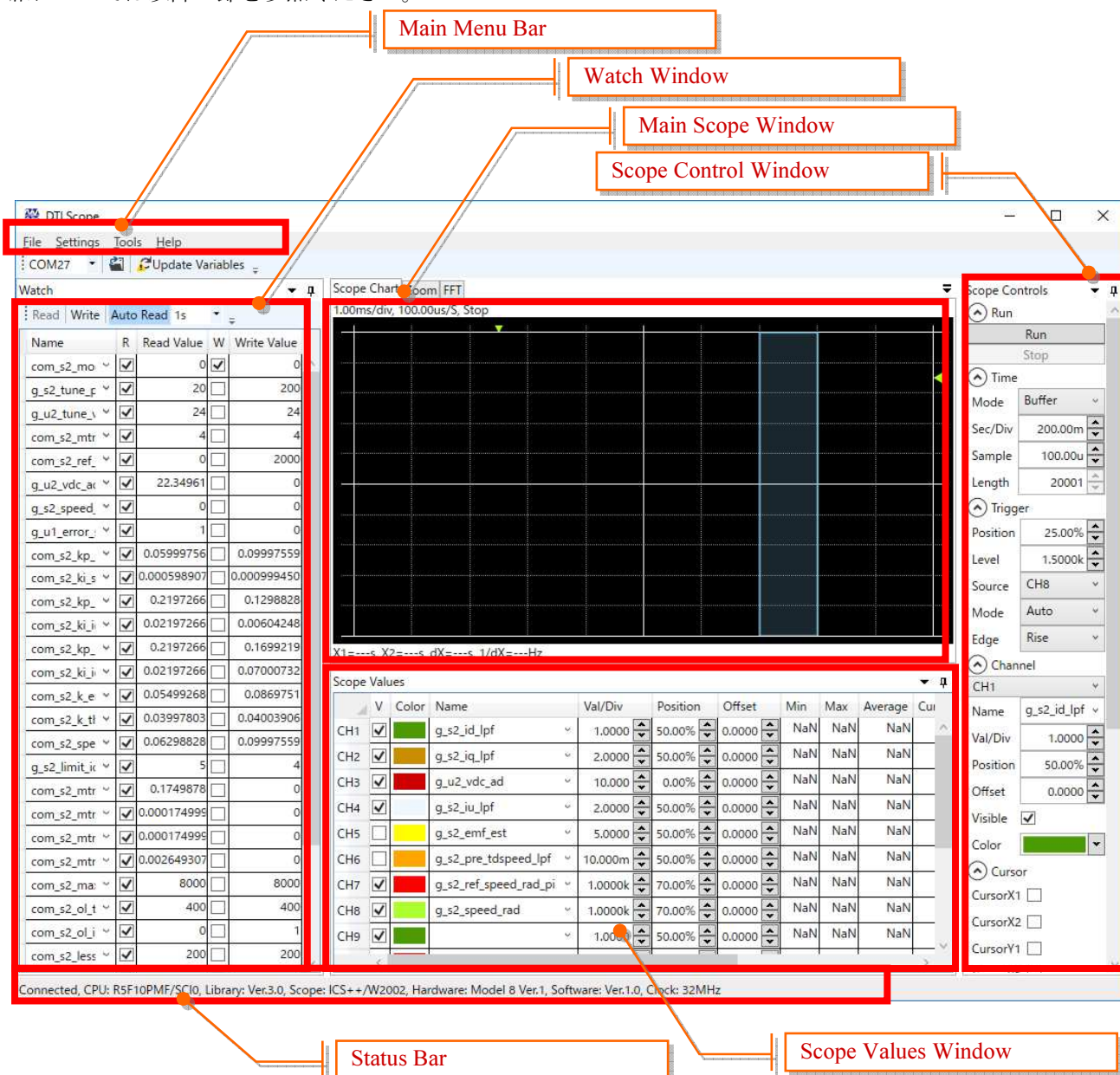


図 3-9 デフォルト画面

表 3-1 各画面の概要

画面名称	概要
(a) Main Menu Bar	ファイルの保存、読み込み、画面の表示設定
(b) Main Scope Window	Scope Control Window / Scope value window で設定した波形を表示します。
(c) Scope Control Window	レンジの設定やトリガ条件等を設定します。
(d) Scope Value Window	レンジの設定や数値情報の表示を行います。
(e) Watch Window	任意の変数値の読み出し、書き込みを行います。
(f) Status Bar	ターゲットとの接続情報を表示します。

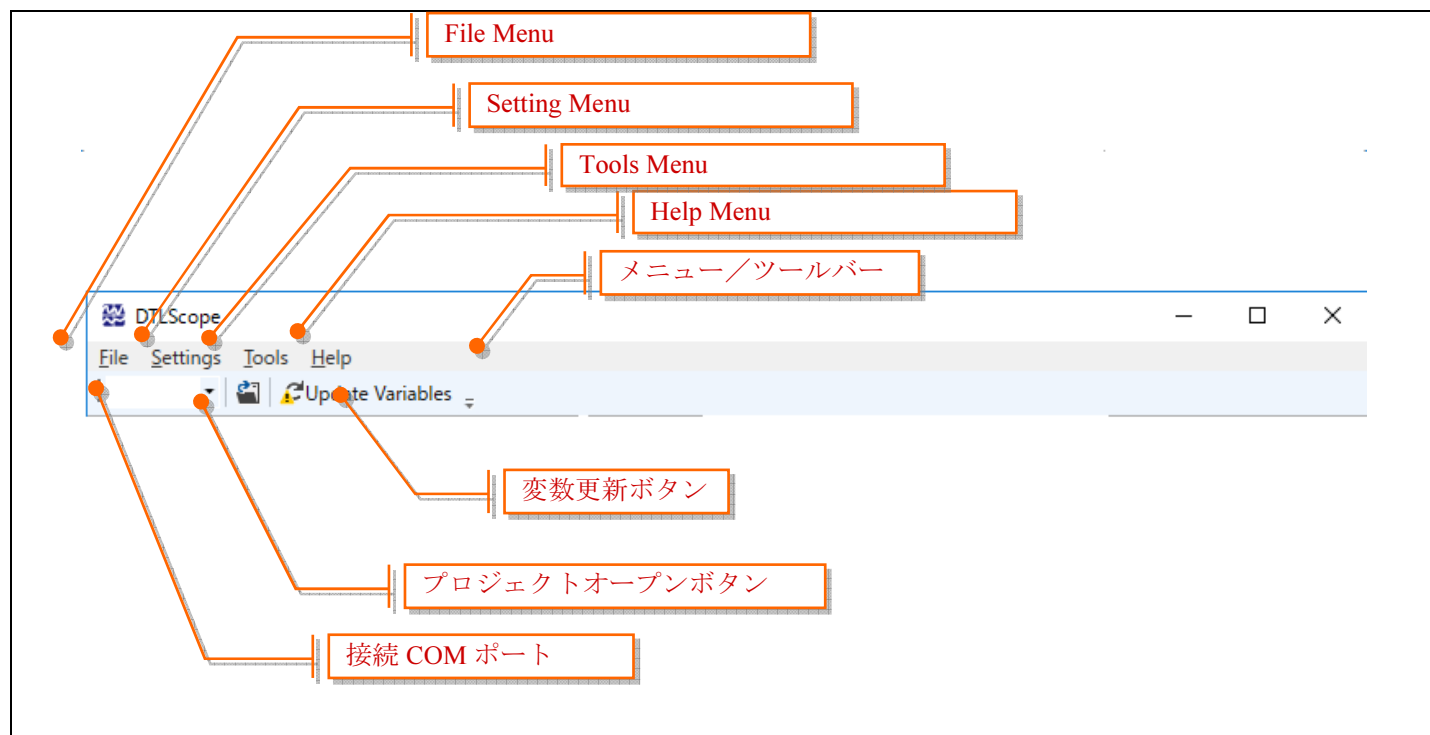


図 3-10 Main Menu Bar

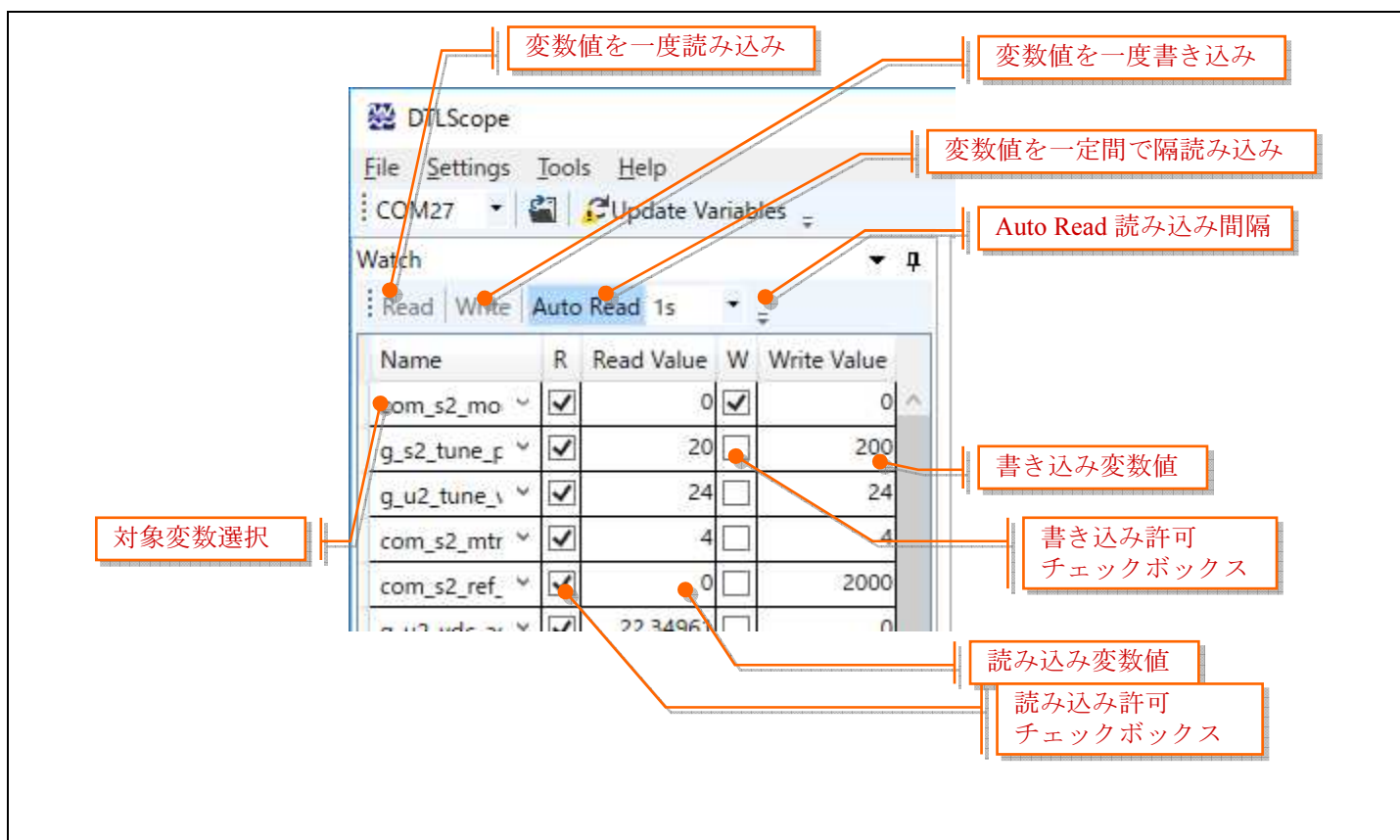


図 3-11 Watch window

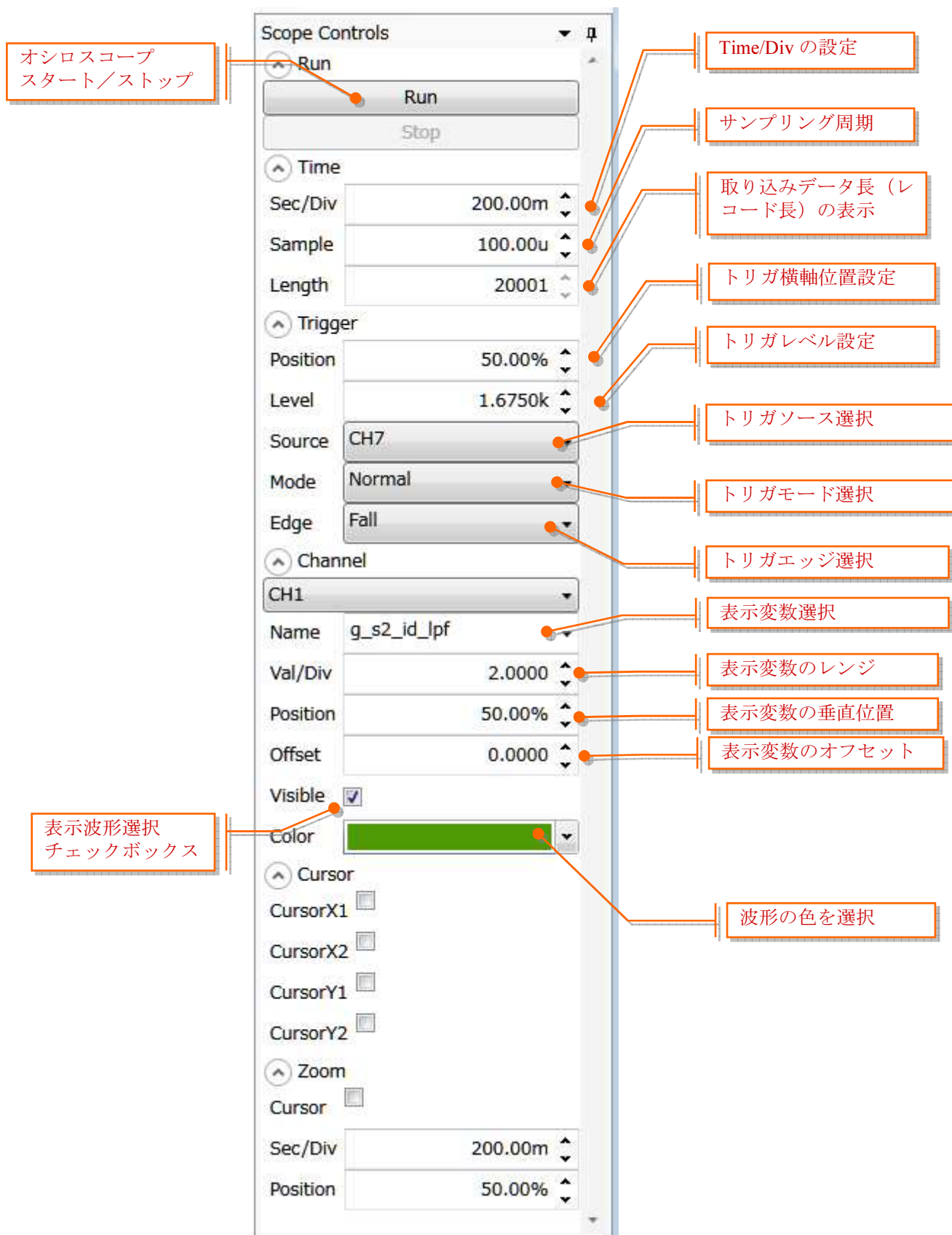


図 3-12 Scope controls window (part1)

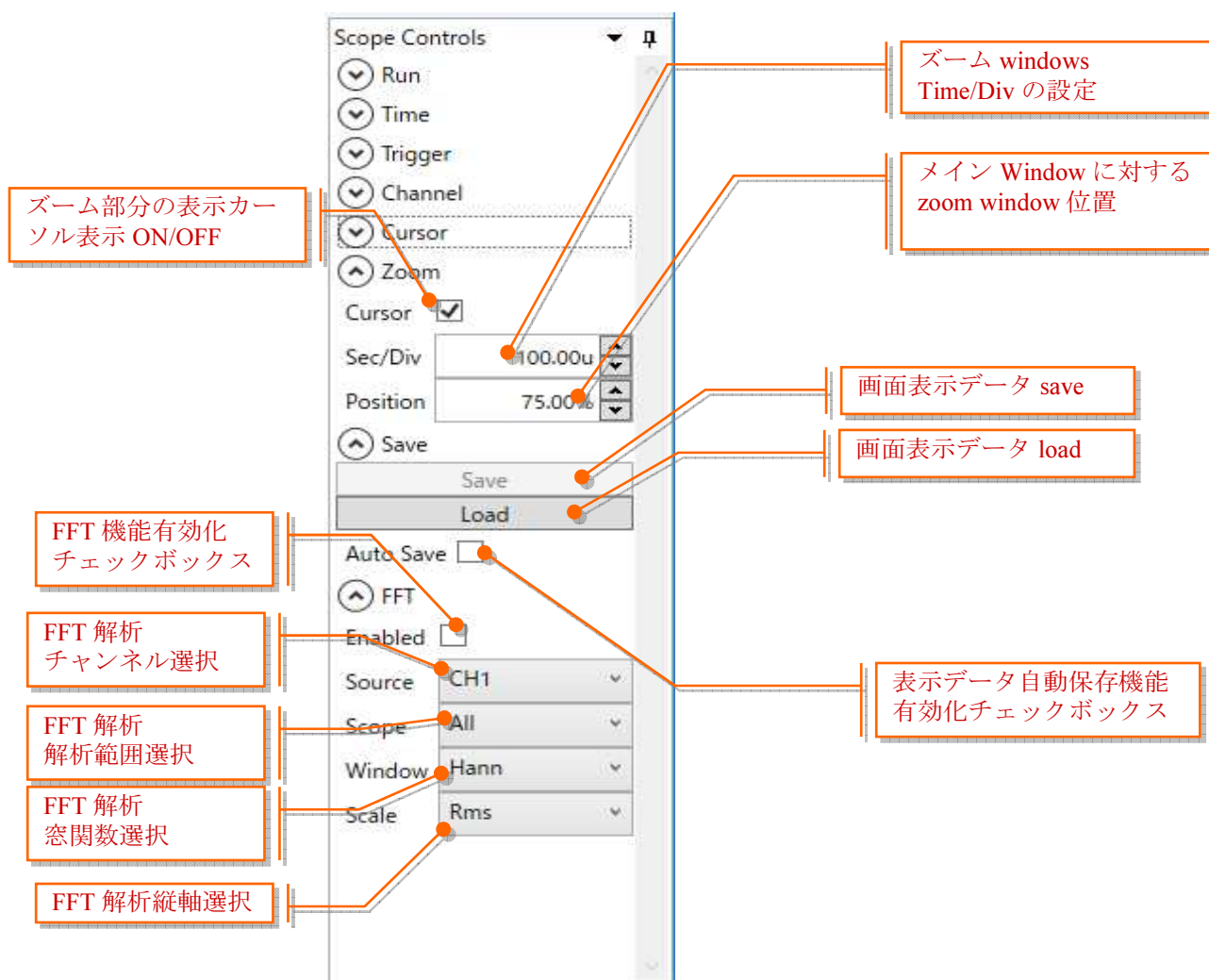


図 3-13 Scope controls window (part2)

Scope Values

V	Color	Name	Val/Div	Position	Offset	Min	Max	Average	CursorX1	CursorX2	X1-X2	CursorY1	CursorY2	Y1-Y2
1		g_s2_id_lpf	2.0000	50.00%	0.0000	-1.1094	1.0938	74.192m	-453.13m	-453.13m	0.0000	-10.000	-10.000	0.0000
2		g_s2_iq_lpf	2.0000	50.00%	0.0000	-4.7969	5.6563	233.38m	4.7813	4.7813	0.0000	-10.000	-10.000	0.0000
3		g_u2_vdc_ad	10.000	0.00%	0.0000	21.264	31.896	22.770	22.240	22.240	0.0000	0.0000	0.0000	0.0000
4		g_s2_iu_lpf	2.0000	50.00%	0.0000	-3.9883	4.2383	-56.531m	-1.6836	-1.6836	0.0000	-10.000	-10.000	0.0000
5		g_s2_emf_est	5.0000	50.00%	0.0000	2.1387	7.4629	5.0103	3.2246	3.2246	0.0000	-25.000	-25.000	0.0000
6		g_s2_pre_tdspeed_lpf	10.000m	50.00%	0.0000	-54.932m	-5.3711m	-42.940m	-36.621m	-36.621m	0.0000	-50.000m	-50.000m	0.0000
7		g_s2_ref_speed_rad_pi	1.0000k	70.00%	0.0000	837.75	2.5133k	1.6755k	1.2378k	1.2378k	0.0000	-7.0000k	-7.0000k	0.0000
8		g_s2_speed_rad	1.0000k	70.00%	0.0000	758.00	2.6110k	1.6752k	1.0325k	1.0325k	0.0000	-7.0000k	-7.0000k	0.0000

Callouts:

- 波形表示 ON/FF 設定
- 表示波形選択
- 表示波形レンジ設定
- 波形表示位置設定
- 波形数値オフセット設定

図 3-14 Scope Value Window

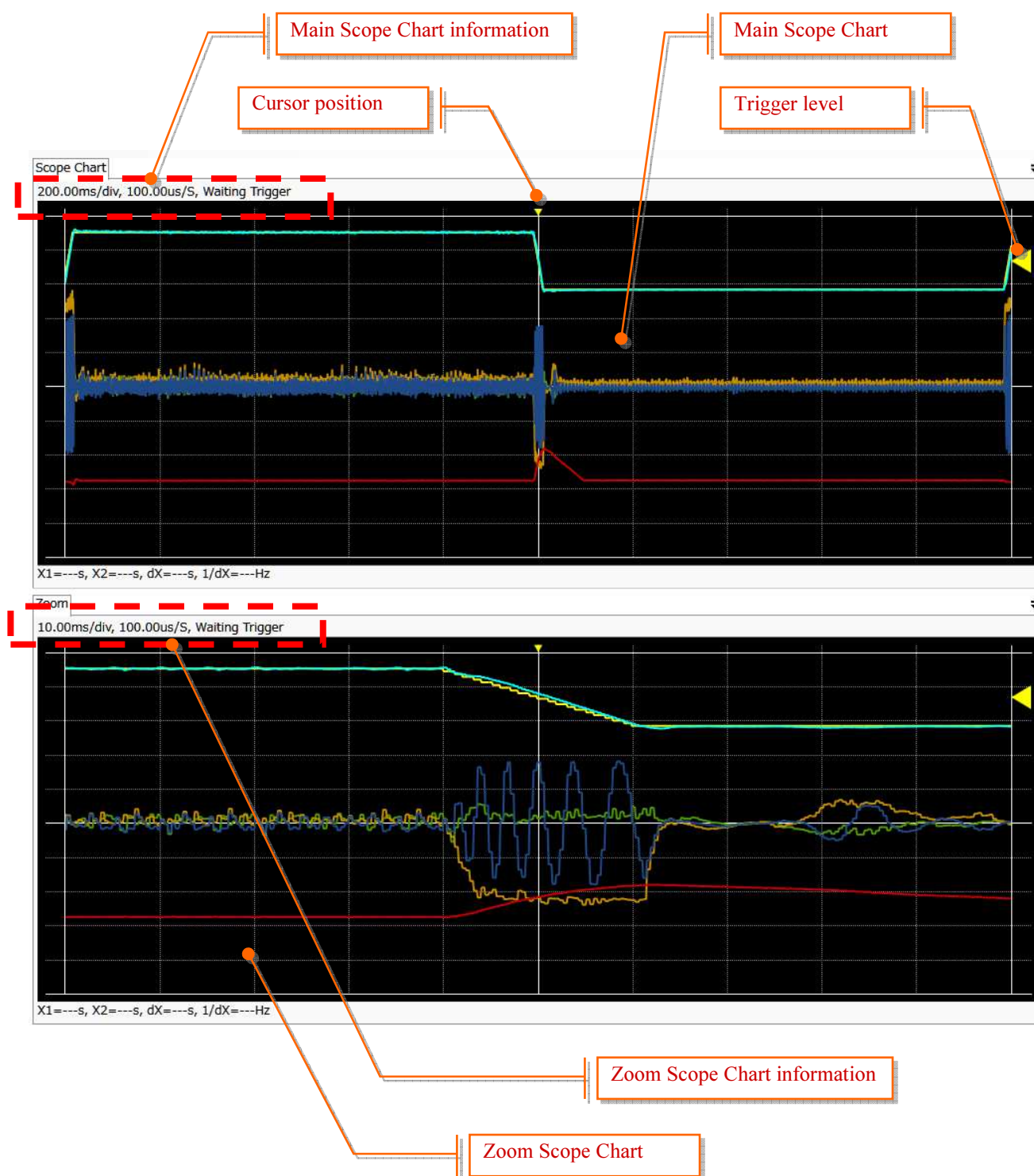


図 3-15 Main Scope Window

3.5 Menu Window

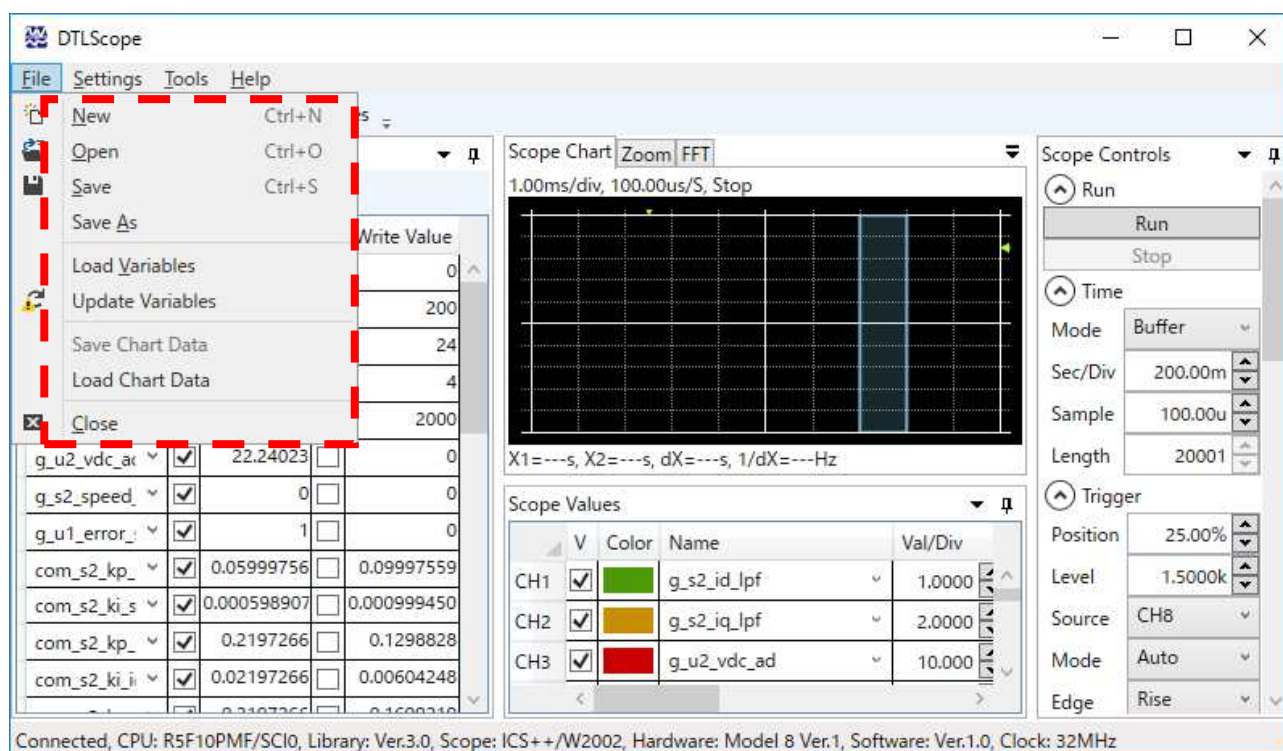


図 3-16 メニューバー

3.5.1 File メニュー

(1) New (Ctrl+N)

ICS++の変数情報などを全て初期化し、続けて、変数ファイルをロードします。

(2) Open (Ctrl + O)

ICS++のプロジェクト設定情報を開きます。

(3) Save (Ctrl + S)

ICS++のプロジェクト設定情報を保存します。

(4) Save As

ICS++のプロジェクト設定情報を、ファイル名を指定して保存します。

(5) Load Variables

変数情報を指定して、読み込みます。

(6) Update Variables

現在指定している変数ファイルを再度読み込みます。

(7) Save Chart Data

波形データを保存します。

(8) Load Chart Data

波形データを読み込みます。

(9) Close

ICS++を終了します。

3.5.2 Settings メニュー

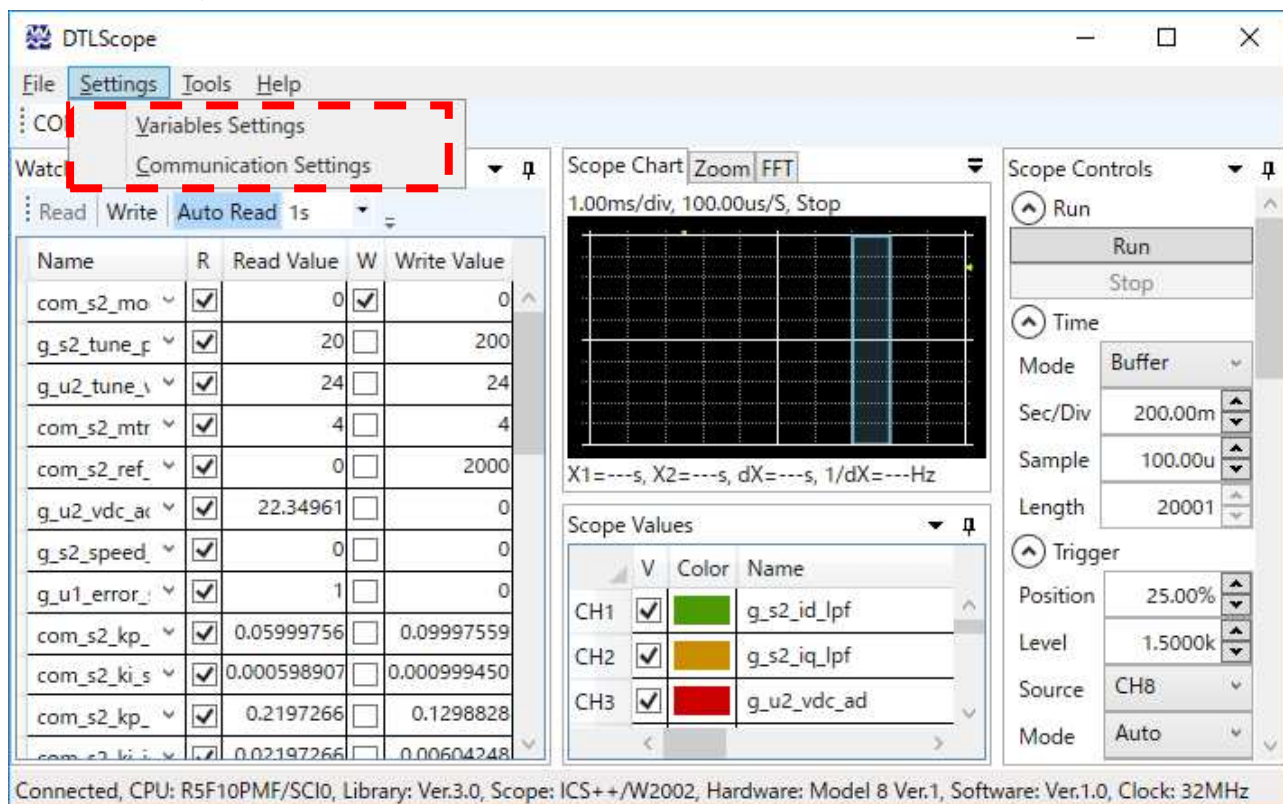


図 3-17 メニュー ⇒ Setting

(1) Variables Settings

必要に応じて、使用している変数の詳細を設定します。

Variable Settings では、下記の機能があります。

- A) 変数の型名を強制的に変更する
- B) 整数型の場合、スケーリングを行う。数値を入力してください。 **(数式の入力は禁止します)**
- C) 読込、書込みを許可、禁止します。
- D) 変数を別名定義します。
- E) 変数にコメントを付けます。Watch window カーソルを変数上に移動するとコメントを表示できます。

Name	Address	Type	Modified Type	Scale	Min	Max	R	W	Alias	Description
com_s2_sw_userif	FE9C2	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
g_s2_sw_userif	FE9C4	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_mode_system	FE9C6	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	COMMAND	0:停止, 1:ADJ, 2:実行, 4:VDCCHK
g_s2_mode_system	FE9C8	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_direction	FE9CA	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ref_speed_rpm	FE9CC	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_kp_speed	FE9CE	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ki_speed	FE9D0	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_kp_id	FE9D2	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ki_id	FE9D4	Int16		1-Infinity	Infinity		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

図 3-18 Variable Settings

(2) Communication Settings

通信レート設定用クロックを設定します。詳細は、ICS/ICS++の関数マニュアルを参照してください。ICS の関数マニュアルで指定されるクロックを設定してください。

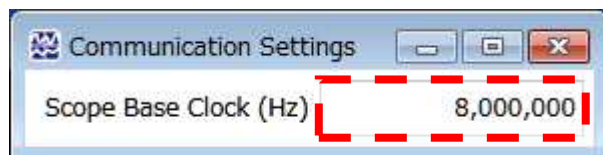


図 3-19 Communication Settings

※RENESAS 版 ICS では、水晶発振器を交換することで対応していましたが、ICS++からは、可変クロック機能を内蔵しています。このクロックを修正することで、5,000,000Hz から 10,000,000Hz までと、12,000,000Hz, 24,000,000Hz に対応が可能です。

3.6 Tools メニュー

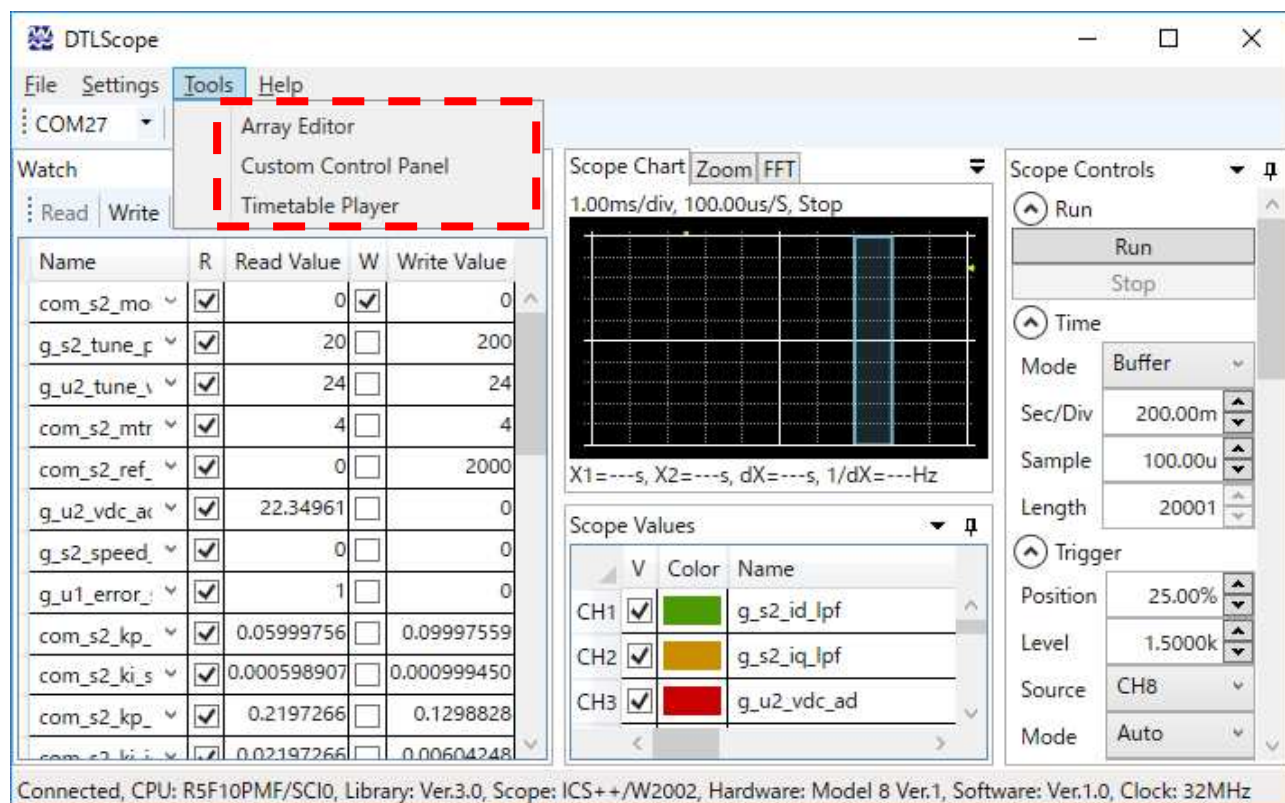


図 3-20 Tools menu

3.6.1 Array Editor

配列の要素を表示するためのツールです。配列は要素数も多いので、Watch Windows で表示するのは困難です。よって、特別な機能を用意しました。

Array Setting の Origin で変数を選択し、変数の方や Start で最初の配列要素番号を選択することで配列の値を表示することが可能です。

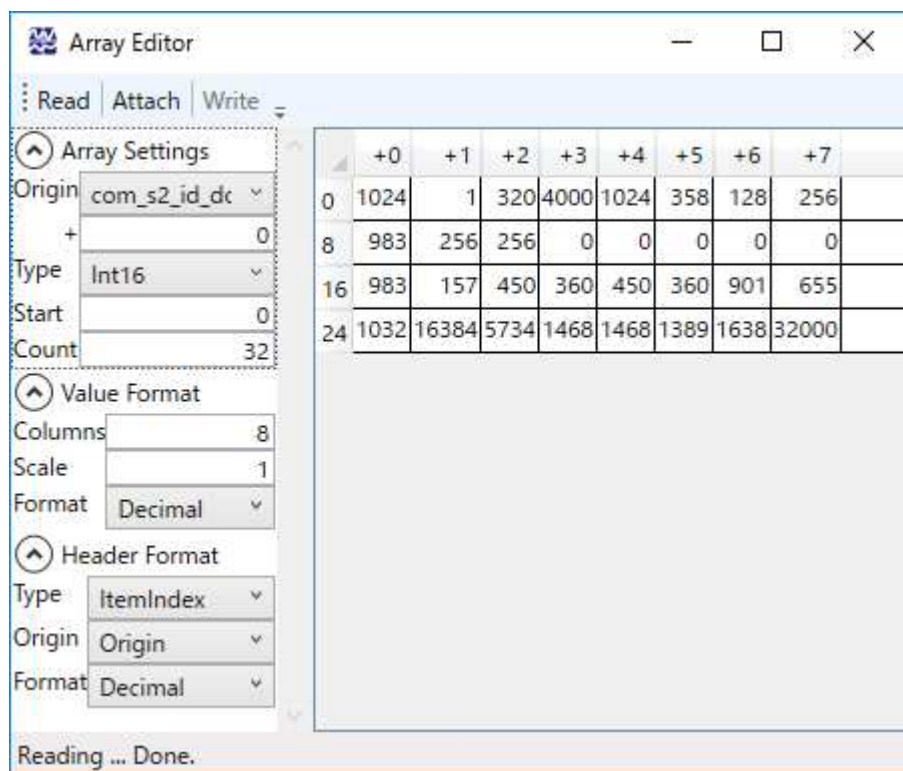


図 3-21 Array Editor Window

基本的な操作方法とソフトウェアの挙動

ウィンドウの左側の"Array Settings"等で設定をして、ツールバーの"Read"を押すとマイコンからデータを読み取り、ウィンドウ右側の配列データに表示されます。

"Read"をした後にツールバーの"Attach"を押すと"Array Settings"が固定されて配列データの編集ができるようになり、ツールバーの"Write"で配列データがマイコンに書き込まれます。

"Read"をした後に"Attach"できるようになりますが、"Array Settings"を変更すると"Attach"できなくなります。

"Attach"状態でツールバーの"Detach"を押すと"Array Settings"の固定が解除されて、配列データの編集と"Write"ができなくなります。

Array Settings

Origin

配列の起点を設定する。コンボボックスで起点にする変数を選択します。

下段のテキストボックスで変数のアドレスからのオフセットを設定することもできます。

配列要素ではなくバイト長のオフセットであることに注意してください。

基本的には変数を空白にして絶対アドレスで配列の起点を設定するときに使用します。

プレフィックス"0x"で 16 進数の入力が可能です。

Type

配列の型を設定します。変数を選択すると変数情報に従って自動的に設定されますが変更することもできます。

Start

配列にアクセスする開始インデックスを設定します。バイト長ではなく要素のインデックスであることに注意してください。

Count

配列から読み取る要素数（最大 1024）を設定します。バイト長ではなく要素数であることに注意してください。

Value Format

Columns

配列データを表示するときの列数（最大 1024）を設定します。

Scale

配列データに適用するスケーリング値を設定します。変数を選択すると変数情報に従って自動的に設定されますが変更することもできます。

スケーリングが適用されるのは 10 進数表示のときのみです。

Format

配列データの文字列表示形式（2/10/16 進数）を設定します。

Header Format

Type

行・列ヘッダのタイプ（要素数、行/列数、アドレス）を設定します。

Origin

行・列ヘッダの数値の起点（Array Settings の Origin/Array Settings の Start で指定した要素/アドレス 0）を設定します。

Format

行・列ヘッダの文字列形式（10/16 進数）を設定します。

3.6.2 Custom Control Panel

"Custom Control Panel"はデスクトップラボ社の受託品ユーザーに対するカスタム仕様の PC ソフトウェアを提供する機能です。弊社では、本機能の仕様を公開しておりません。

"Custom Control Panel"はマイコンの変数を操作・表示するためのユーザー定義のコントロールパネルを XAML により提供します。

"Custom Control Panel"ではマイコンの変数に対して以下の 2 つのことができます。

1. ボタンが押された時にマイコンの変数の値を書き換えます
2. マイコンの変数の値をコントロールのプロパティにバインドして操作・表示します

3.6.3 Timetable player 機能 (Function generator 機能)

FG 指令値生成機能はターゲットマイコンの変数にあらかじめ決められた経過時間に決められた値の書き込みを行う機能です。以下にデスクトップラボ製「DTLScope」の FG 指令値生成機能「Timetable Player」の仕様を記載します。

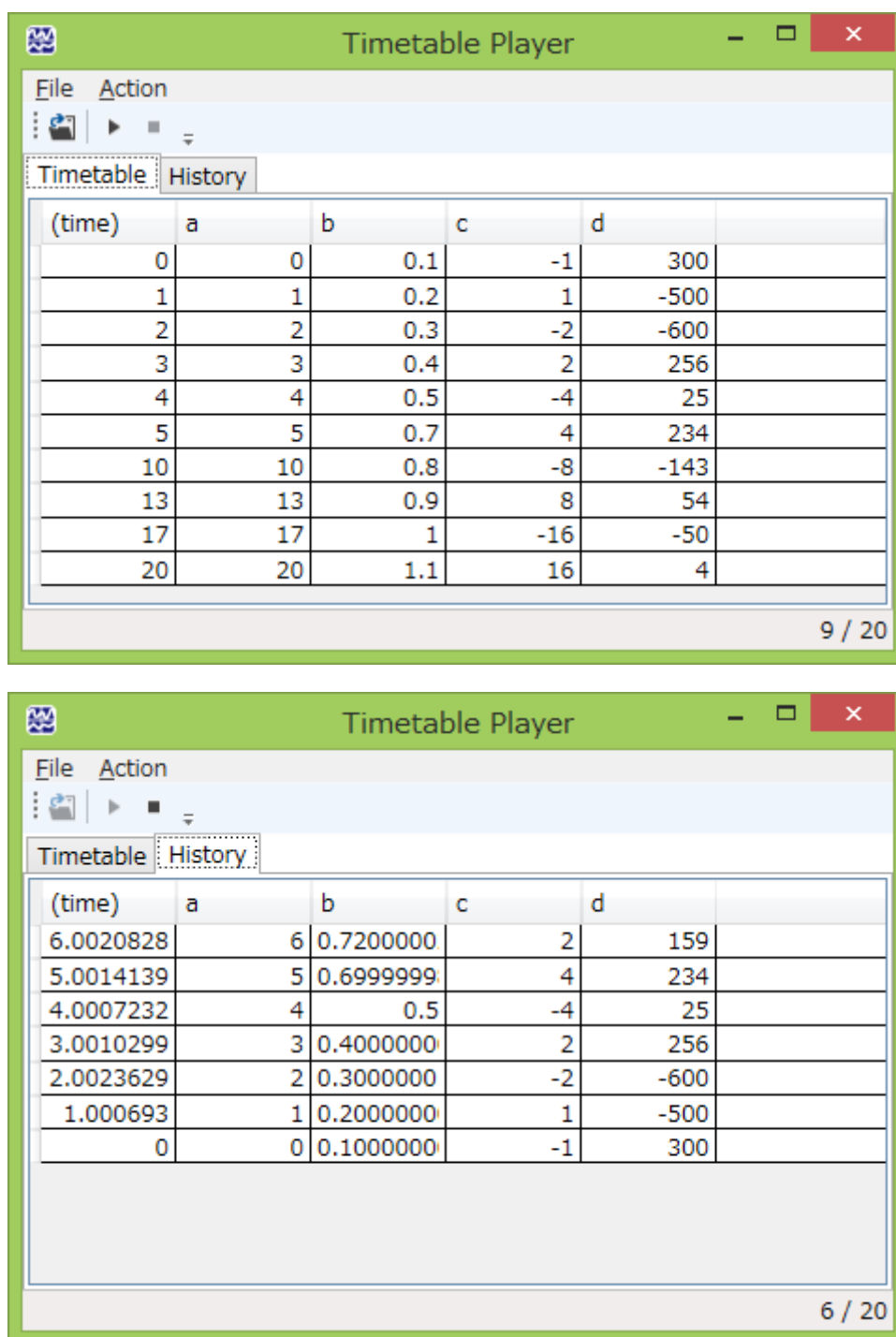


図 3-22 「DTLScope」の FG 指令値生成機能「Timetable Player」の画面

基本的な操作方法とソフトウェアの挙動

"Timetable Player" ウィンドウの"メニューバー" -> "File" -> "Open" (もしくは、ツールバーのフォルダアイコン) で CSV 形式の "Timetable データ" を開きます。

ファイルが正常に開くと "Timetable タブ" に表形式でデータが表示されます。

"Timetable データ" の書式は以下の CSV ファイル形式です。

例 :

(time),a,b,c,d

0,0,0.1,-1,300

1,1,0.2,1,-500

5,5,0.7,4,234

1 列目は時間、2 列目以降は変数です。

1 行目の 2 列目以降は変数名です。

時間の単位は秒、最小単位は 1 秒です。

時間は等間隔である必要はありません。

時間は 0 秒から単調に増加する必要があります。

"Timetable Player" ウィンドウの"メニューバー" -> "Action" -> "Play" (もしくは、ツールバーの三角アイコン) でマイコンへの値の書き込みが開始します。

"History タブ" に書き込み値の履歴が表形式で表示されます。(履歴は最大で直近 60 秒が表示されます。60 秒分になると 30 秒分消えます。)

書き込み間隔は "1 秒" で "Timetable データ" のデータの間は線形補間されたデータが書き込まれます。

書き込み間隔以内に処理が終わらない場合は、その間の時間間隔の処理はスキップされます。

書き込み間隔ごとにデータの上から経過時間より大きい行が検索されます。該当の行と 1 つ前の行との間で書き込み値の線形補間を行います。

通信エラーが発生してもデータの再送は行ないません。エラー内容は "History タブ" に書き込み値のかわりに表示されます。

プロジェクトに存在しない変数の場合は "Undefined variable" と表示されます

通信 BOX から NAK が返ってきた時は "NAK 8" のように表示されます

"Timetable Player" ウィンドウの"メニューバー" -> "Action" -> "Stop" (もしくは、ツールバーの四角アイコン) でマイコンへの値の書き込みを中止します。

3.7 Help メニュー

(1) About DTLScope

Product 名、Version, Copyright を表示します。



図 3-23 About DTL Scope

3.8 COM ポート選択

ICS++ユニットを接続する COM ポートを選択します。

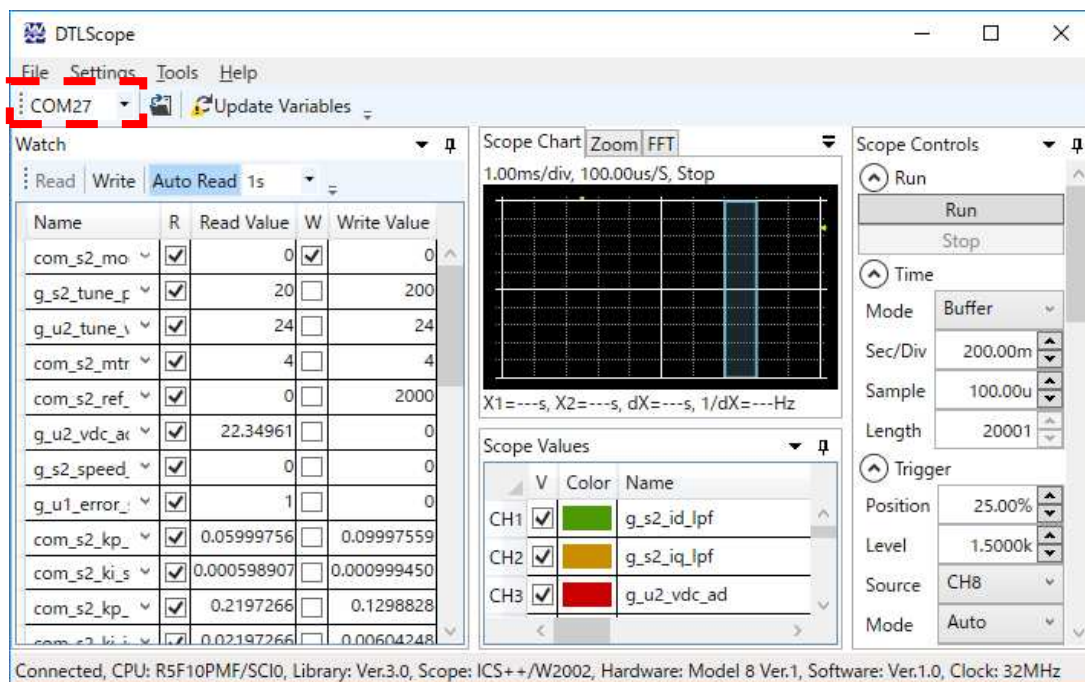


図 3-24 COM ポートの選択

3.9 プロジェクトを開く

設定済みプロジェクトを開きます。

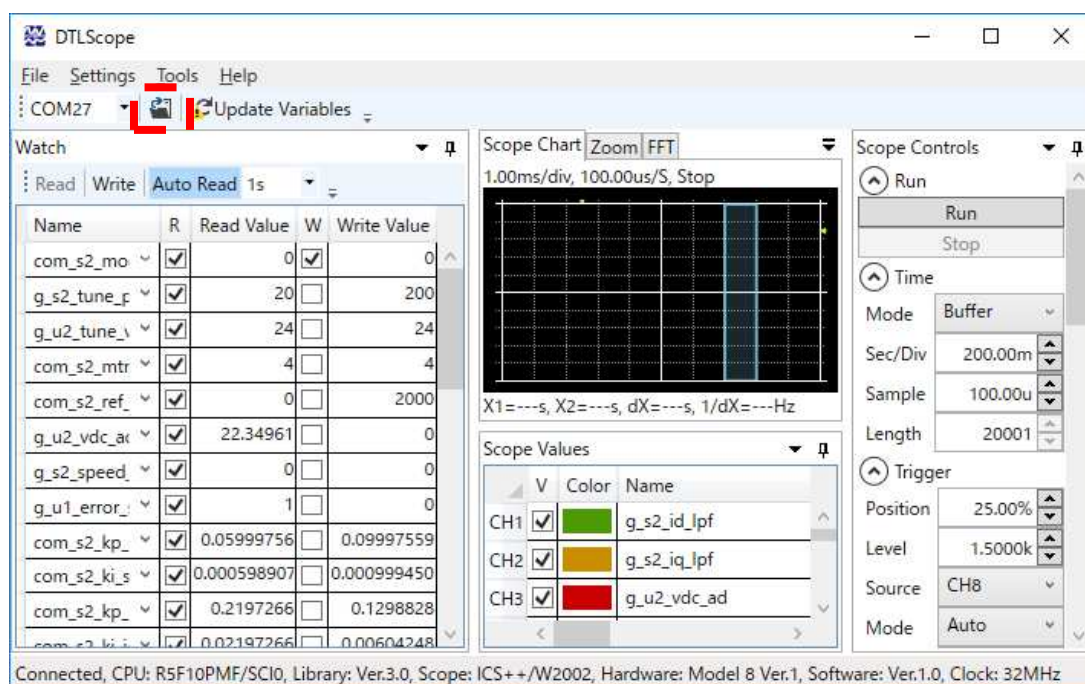


図 3-25 プロジェクトを開く

3.10 変数情報の更新

変数情報を更新します。

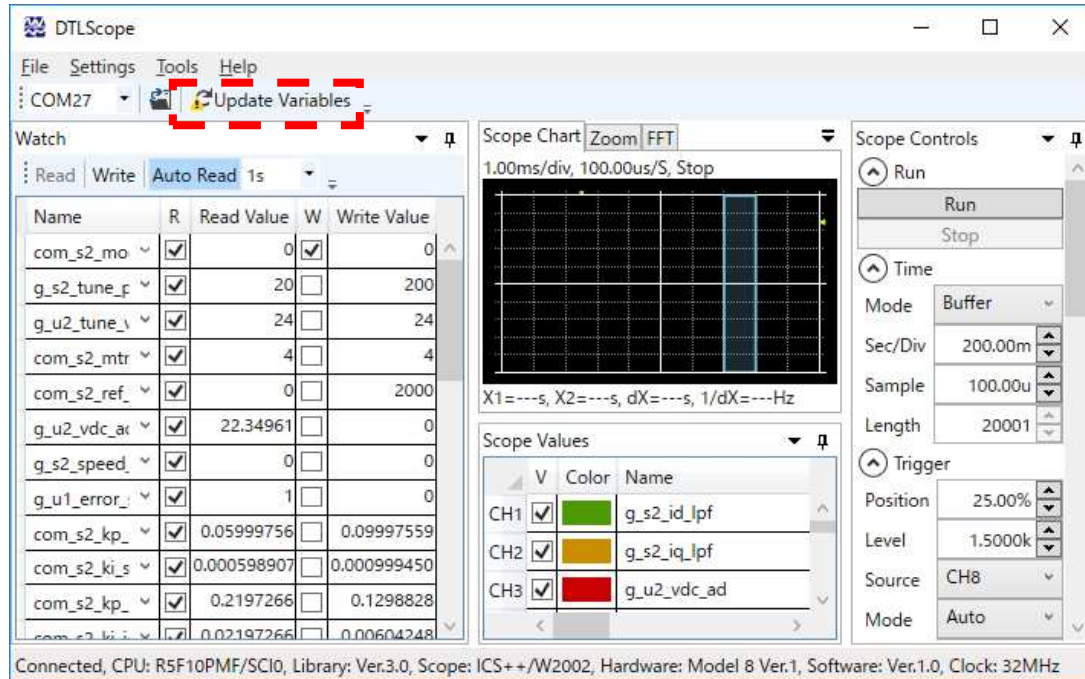


図 3-26 変数情報の更新

3.11 オシロスコープ機能

この節ではオシロスコープ機能を使用する上で必要な設定について説明します。本機能を用いて波形表示可能な変数の型は、CPU に組み込むライブラリの種類によって異なります。ご注意ください。

3.11.1 波形表示変数設定

Scope Controls Window もしくは、Scope Values Window の変数名のプルダウンボタンから、測定する変数名を選択します。また、直接キーボード入力することで、検索することも可能です。

変数リストは読み込んだ変数情報ファイル(拡張子: csv) に基づいています。

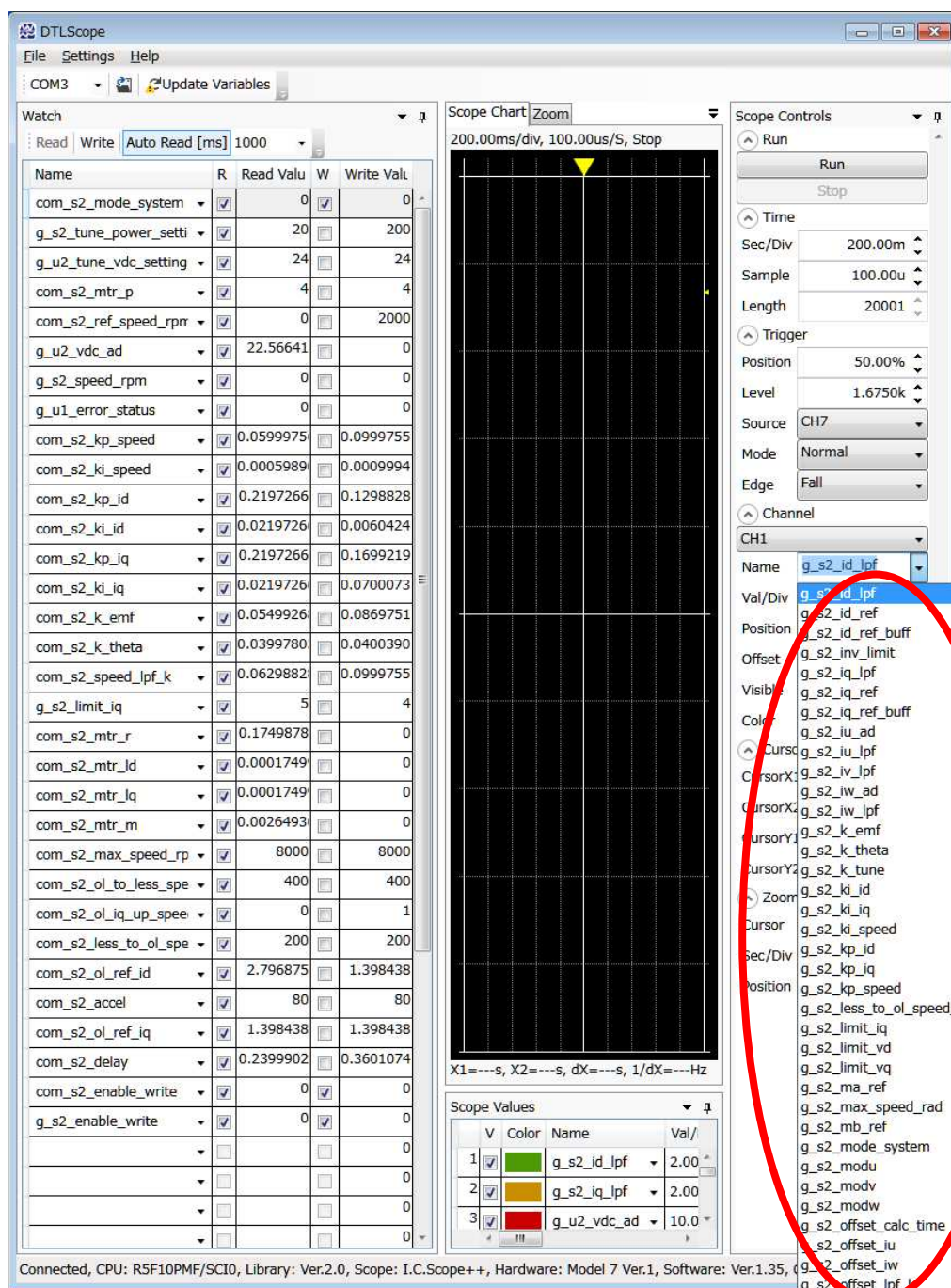


図 3-27 Scope Control Window の変数選択画面

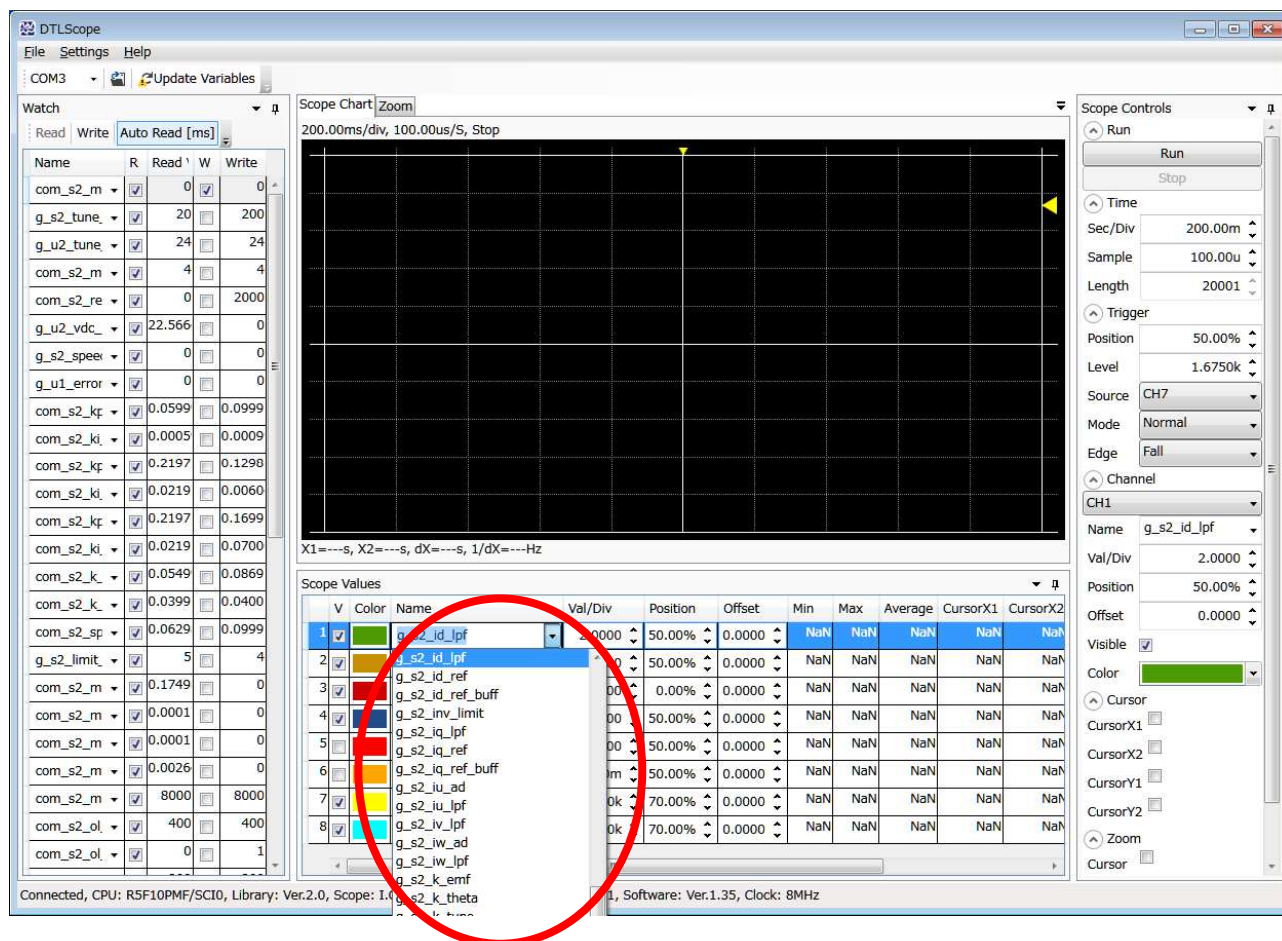


図 3-28 Scope Value Window の変数選択画面

波形表示の変数選択は、上記の2つの方法があります。通常は、Scope Value Window から選択するのが楽です。

3.11.2 横軸レンジとサンプリング周期設定

(1) 横軸レンジ

Main Scope Window と Zoom Window の横軸レンジである Time/Div は Scope Controls Window で設定します。Time/Div は横軸 1 目盛当たりの時間を表します。100 us/Div～262.14s/Div の範囲で設定可能です。

(2) サンプリング周期

変数データのサンプリング周期は Acquisition 欄の Sample で設定します。20 us～10 ms の範囲で設定可能です。ここでいうサンプリング周期は測定周期とは異なるものです。測定周期はターゲット CPU から ICS++ユニットへの変数データ転送周期を指します。一方、サンプリング周期は ICS++ボードのバッファに格納されている変数データをサンプリングする周期を指します。

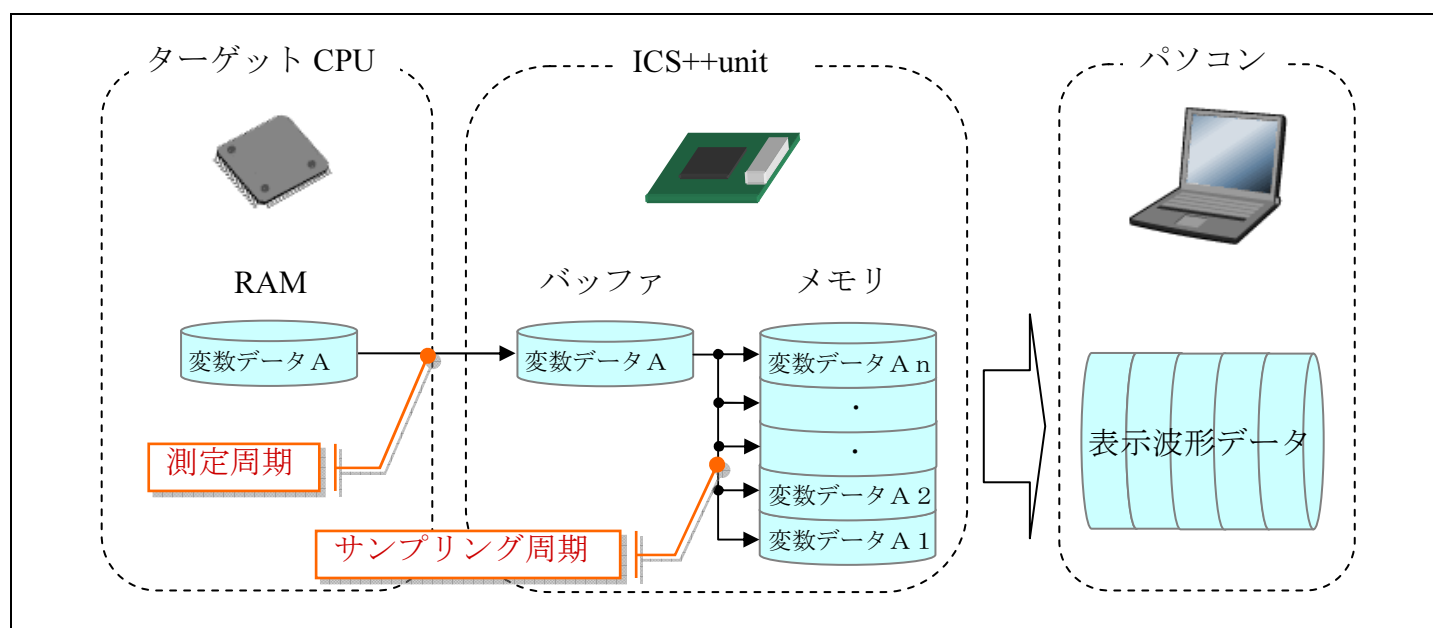


図 3-29 サンプリング周期と測定周期の違い

(3) レコード長

取得波形データのレコード長 Scope Controls Window の Length に表示されます。横軸レンジとサンプリング周期の値によって変化します。(Length = Time/Div ÷ Sample)

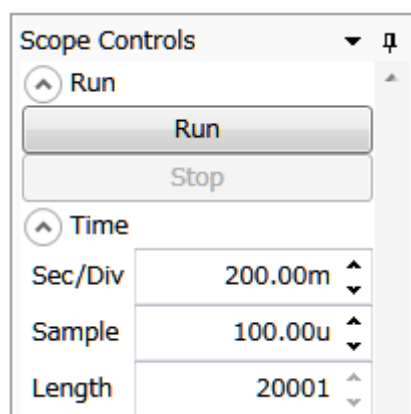


図 3-30 Time/Div と Sample 設定

3.11.3 縦軸レンジとオフセット設定

縦軸レンジとオフセットの設定はチャンネル毎に設定します。

(1) 縦軸レンジ

Main Scope Window と Zoom Window の縦軸レンジである Val/Div は、Scope Value の Val/Div 欄で設定します。Val/Div は縦軸 1 目盛りりの値を表します。1p/Div～1T/Div の範囲で設定可能です。

(2) オフセット

オフセットは Scope Value のオフセット欄で設定します。各変数にオフセットで設定した値分を足して波形を表示します。

(3) Position

画面上の表示位置を、画面全体を 100%として位置を指定します。

Scope Values														
V	Color	Name	Val/Div	Position	Offset	Min	Max	Average	CursorX1	CursorX2	X1-X2	CursorY1	CursorY2	Y1-Y2
1		g_s2_id_lpf	2.0000	50.00%	0.0000	-1.1094	1.0938	74.192m	-453.13m	-453.13m	0.0000	-10.000	-10.000	0.0000
2		g_s2_iq_lpf	2.0000	50.00%	0.0000	-4.7969	5.6563	233.38m	4.7813	4.7813	0.0000	-10.000	-10.000	0.0000
3		g_u2_vdc_ad	10.000	0.00%	0.0000	21.264	31.896	22.770	22.240	22.240	0.0000	0.0000	0.0000	0.0000
4		g_s2_iu_lpf	2.0000	50.00%	0.0000	-3.9883	4.2383	-56.531m	-1.6836	-1.6836	0.0000	-10.000	-10.000	0.0000
5		g_s2_emf_est	5.0000	50.00%	0.0000	2.1387	7.4629	5.0103	3.2246	3.2246	0.0000	-25.000	-25.000	0.0000
6		g_s2_pre_tdspeed_lpf	10.000m	50.00%	0.0000	-54.932m	-5.3711m	-42.940m	-36.621m	-36.621m	0.0000	-50.000m	-50.000m	0.0000
7		g_s2_ref_speed_rad_pi	1.0000k	70.00%	0.0000	837.75	2.5133k	1.6755k	1.2378k	1.2378k	0.0000	-7.0000k	-7.0000k	0.0000
8		g_s2_speed_rad	1.0000k	70.00%	0.0000	758.00	2.6110k	1.6752k	1.0325k	1.0325k	0.0000	-7.0000k	-7.0000k	0.0000

図 3-31 Val/Div, Offset, Position の設定

3.11.4 トリガ設定

(1) トリガチャンネル

Scope Controls Window の Trigger Source のプルダウンでトリガの対象とするチャンネルを選択します。

選択されたトリガチャンネルは Main Scope Window の上部に表示されます。

(2) トリガモード

以下の 3 種からトリガモードを選択します。

選択されたトリガモードは Main Scope Window の上部に表示されます。

- Auto モード : トリガ信号が発生する度に波形表示を更新します。また、一定時間トリガ信号が発生しない場合、自動的に波形表示を更新します。
- Single モード : トリガ信号が発生すると波形を一度表示し、STOP 状態になります。
- Normal モード : トリガ信号が発生する度に波形表示を更新します。

(3) トリガポジションとトリガレベル

トリガチャンネルに CH1～CH8 を選択した場合、トリガポジションとトリガレベルの交点にトリガポイントが設定されます。

設定されたトリガポジション値は Main Scope Window の上部に表示されます。

- トリガポジション : "Scope control Window" の Trigger 欄の Position で設定可能です。
(Max 値 : レコード長 * サンプルング周期、 Min 値 : サンプルング周期)
- トリガレベル : "Scope control Window" の Trigger 欄の Level で設定可能です。
(Max 値 : Val/Div * 5、 Min 値 : Val/Div * (-5))

(4) トリガエッジ

以下の 3 種からトリガエッジを選択します。

- Rise : 立ち上がりエッジ
- Fall : 立ち下がりエッジ
- Both : 立ち上がりと立ち下がりエッジとの両エッジ

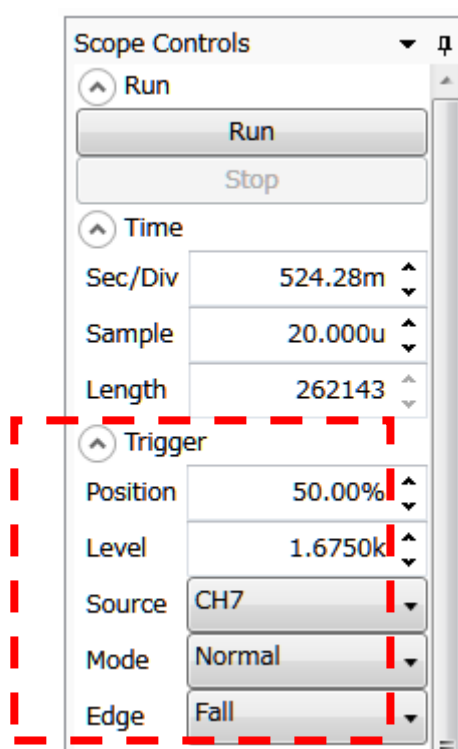


図 3-32 トリガ設定 (1)

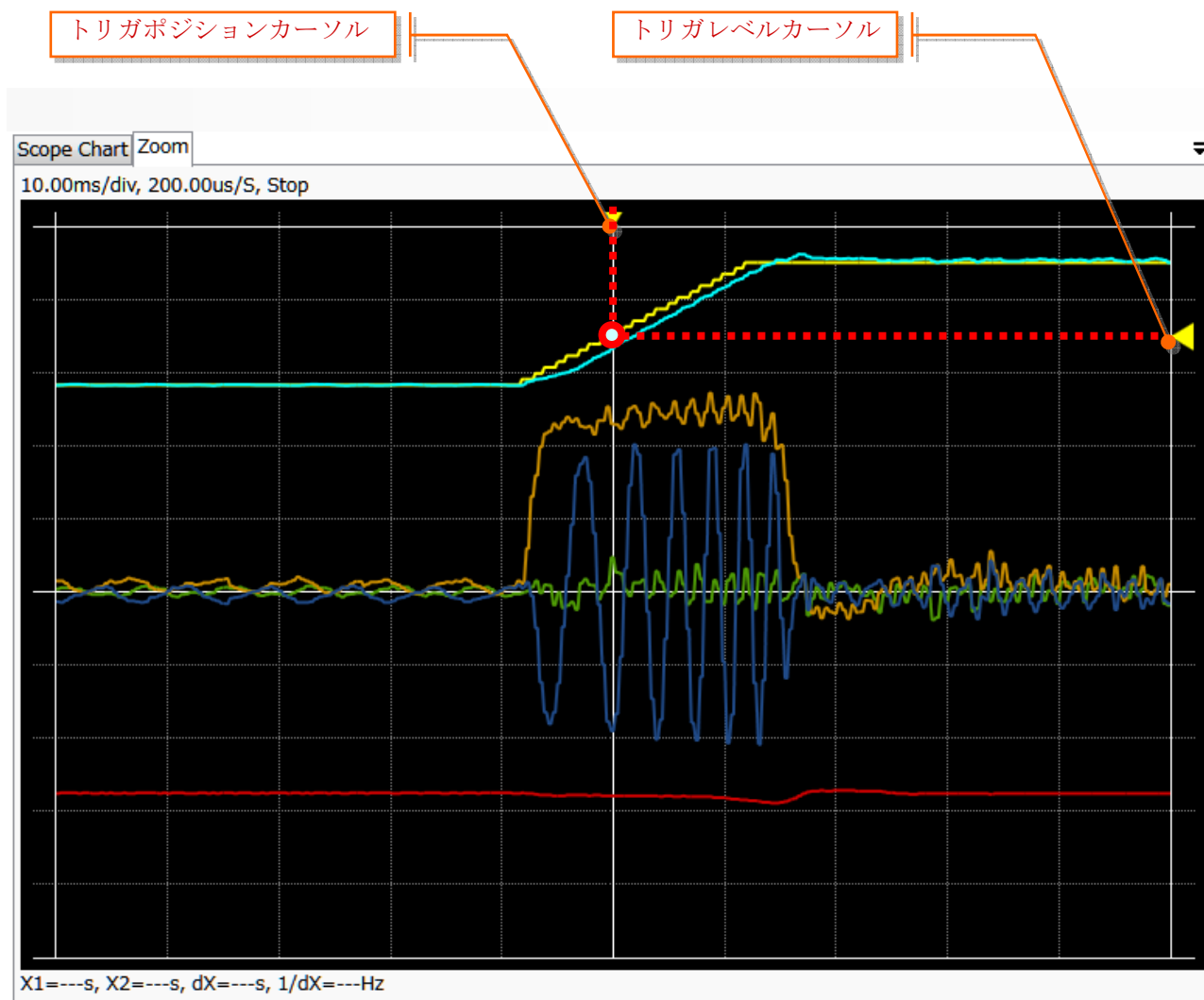


図 3-33 トリガ設定 (2)

3.11.5 カーソル設定

カーソル Cursor X1, Cursor X2, Cursor Y1, Cursor Y2 により波形との交点の Value 値を計測します。計測値は Scope Value Window 画面下部に表示されます。カーソルの表示／非表示は表示カーソル選択欄で選択します。



図 3-34 カーソル設定

3.11.6 ZOOM 画面設定

ICS++は Main Scope Window の一部を拡大表示する Zoom 画面を 1 画面搭載しています。Zoom 画面の表示範囲は Main Scope Window 上の枠で示されています。また、Scope controls Window 上のチェックボックスにより、表示範囲枠の表示／非表示を切り替えます。

(1) 拡大表示範囲幅

Scope Controls の Zoom \Rightarrow Sec/Div のスピンドルコントロールボタンを操作することにより設定します。

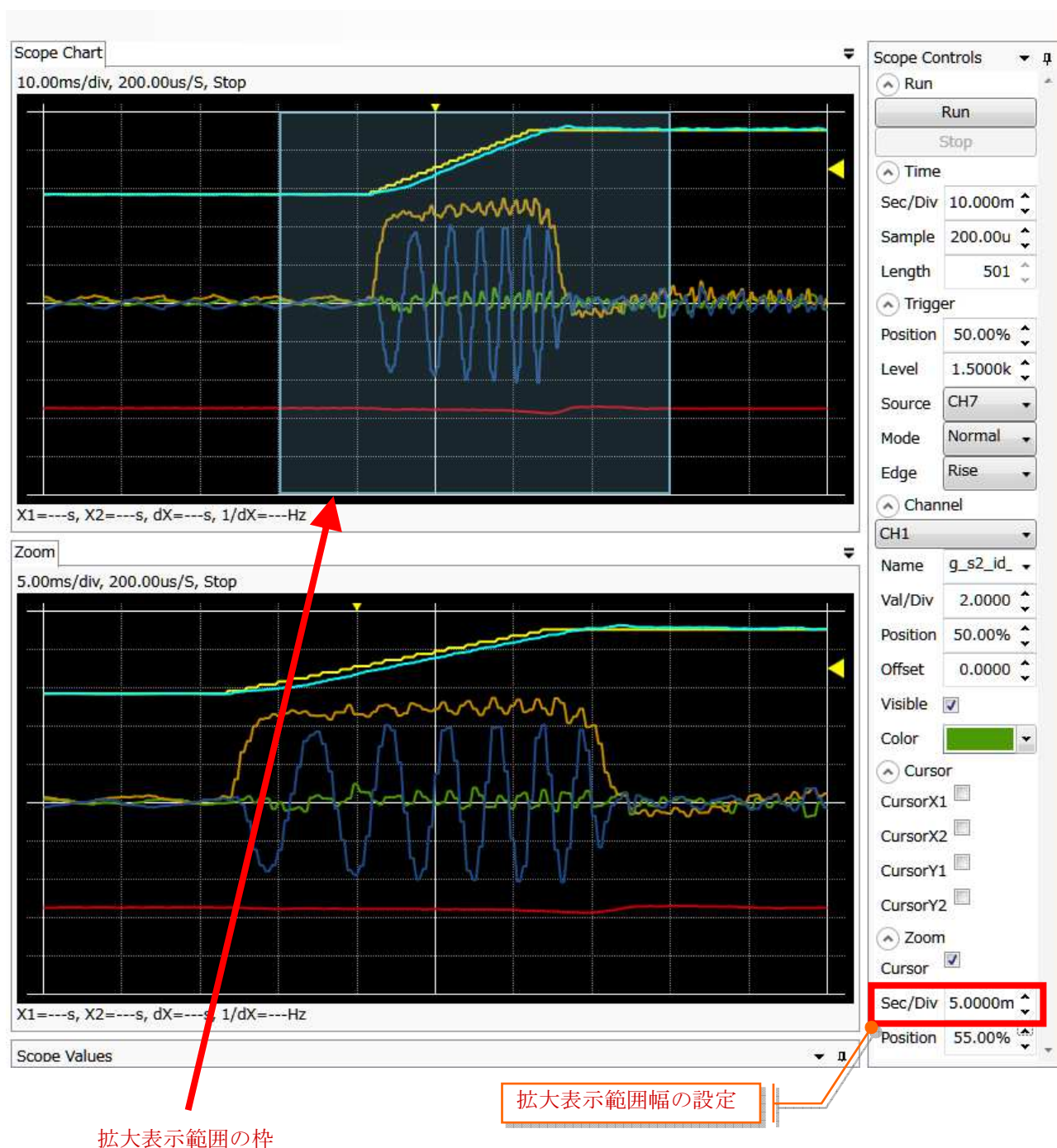


図 3-35 拡大表示範囲幅の設定

(2) 拡大表示開始位置

Scope Controls Window の ZOOM ⇒ Position の値を変更することで設定します。

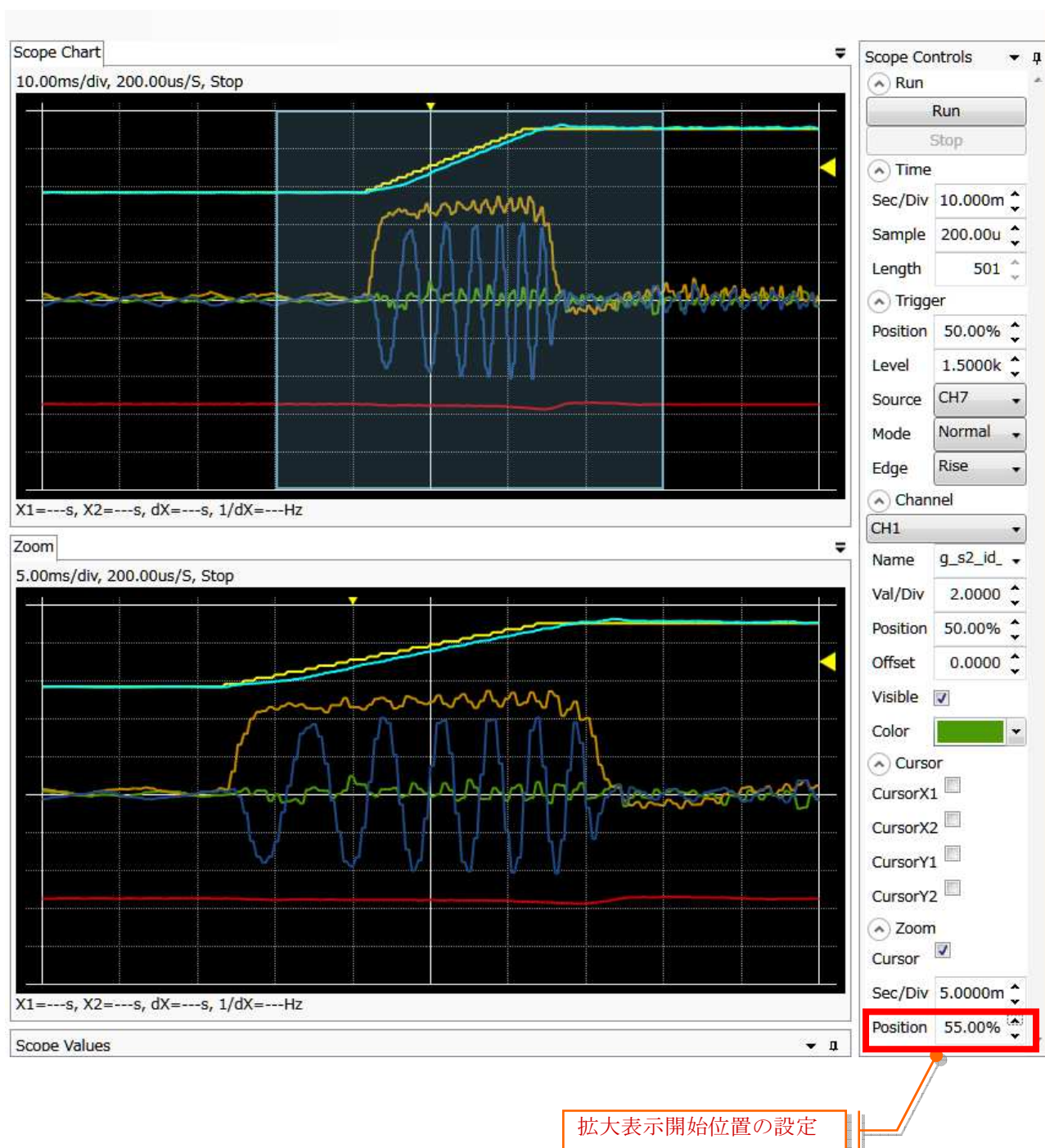


図 3-36 拡大表示開始位置の設定

3.11.7 波形データ保存／読み込み

波形データの保存および、読み込み機能を搭載しています。SAVE することにより、下記の 2 つのファイルが生成されます。

(1) ICS++ 波形データ設定 File(*.dtlcd) : 波形データの波形計測時の設定情報を保存します。

(2) ICS++ Chart Data File(*.dtlcd.csv) : 波形データの生データを保存します。



図 3-37 波形データ保存／読み込み

3.12 変数値読み込み／書き込み機能

この節では変数値読み込み／書き込み機能について説明します。

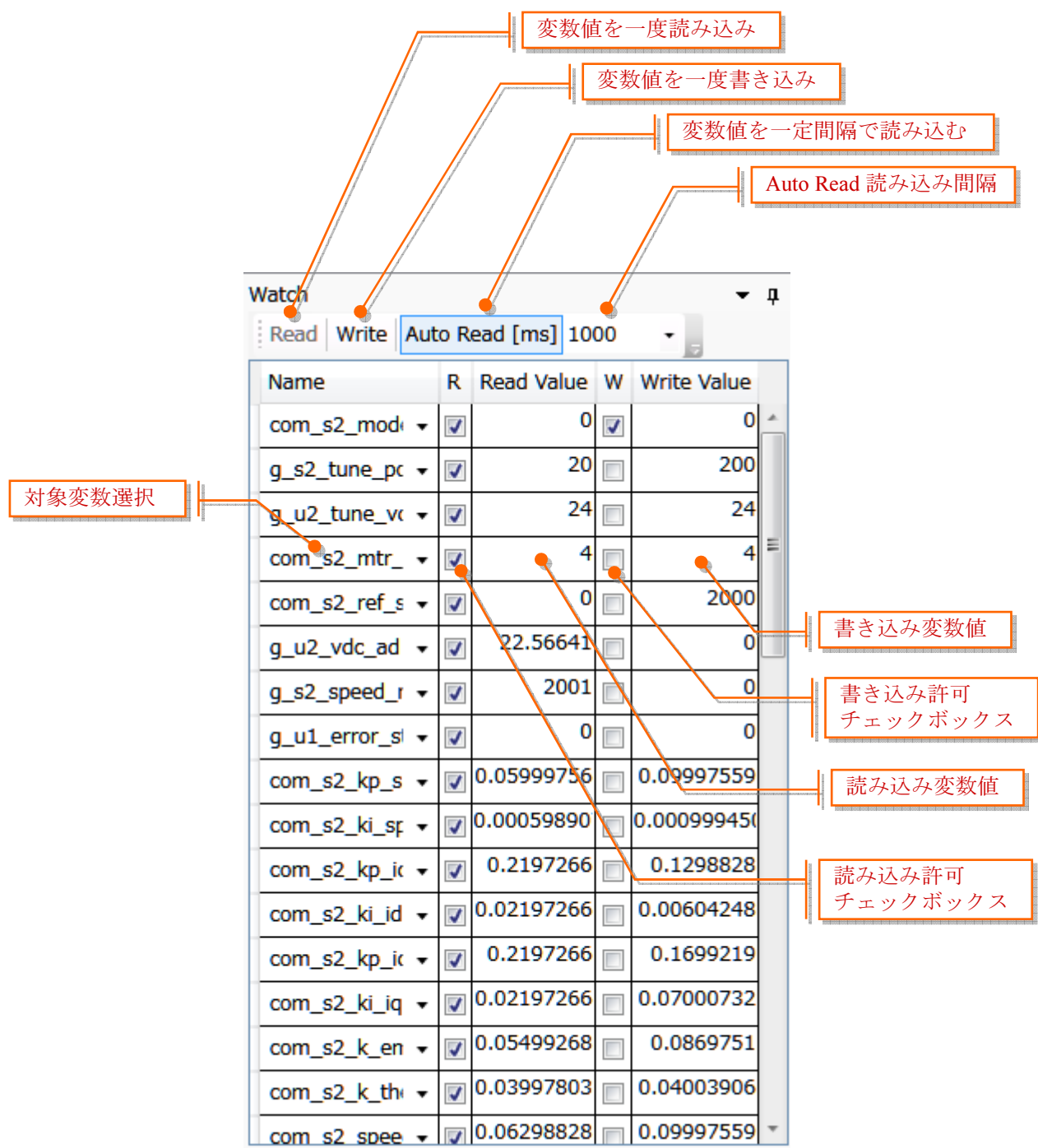


図 3-38 Watch Window

(1)～(5)の順に操作することで、変数値の読み込み／書き込み機能を使用することができます。

(1) 変数選択

Variable Name の列で読み込み／書き込み機能の対象とする変数を選択。

(2) 読み込み／書き込み許可

R、W の列のチェックボックスをチェックすることで、それぞれ読み込み、書き込みを許可します。

(3) Read／Write／Auto Read 実行

Read ボタンを押すと変数値を読み込み、Read の列に値が表示されます。

Write ボタンを押すと Write の列で設定した値を変数に書き込みます。

Auto Read ボタンを押すと Auto Read 読み込み間隔欄で設定した間隔で変数値を読み込みます。

読み込む際に、Setting ⇒ Variables Settings に変数情報を設定することにより、機能を拡張できます。

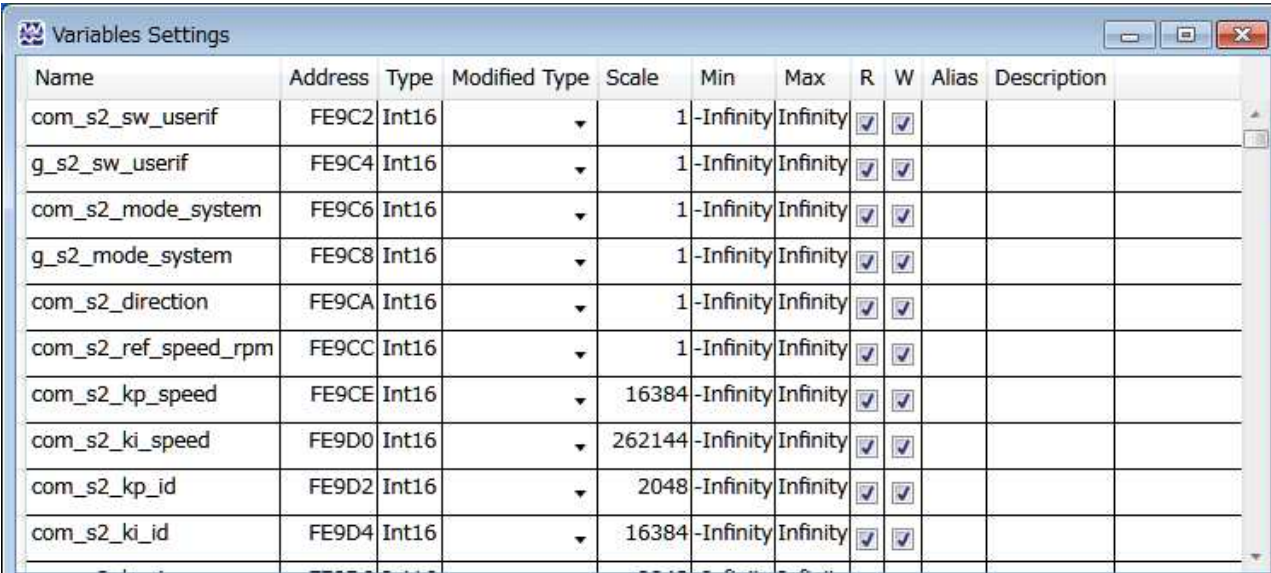
(4) データタイプ選択

Setting ⇒ Variables Settings の対応する変数の行で(1)で選択した変数のデータ型名を選択することにより、デフォルトの型と異なった指定が可能です。

(5) 倍率選択

Setting ⇒ Variables Settings の対応する変数の行で(1)で読み込み値／書き込み値の倍率を選択します。

指定した値で割った値が数値欄に表示されます。この指定は、波形表示に対しても適用されます。



Name	Address	Type	Modified Type	Scale	Min	Max	R	W	Alias	Description
com_s2_sw_userif	FE9C2	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
g_s2_sw_userif	FE9C4	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_mode_system	FE9C6	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
g_s2_mode_system	FE9C8	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_direction	FE9CA	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ref_speed_rpm	FE9CC	Int16		1	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_kp_speed	FE9CE	Int16		16384	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ki_speed	FE9D0	Int16		262144	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_kp_id	FE9D2	Int16		2048	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
com_s2_ki_id	FE9D4	Int16		16384	-Infinity	Infinity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

図 3-39 Variables Settings Window

4. ICS++ PC ソフトウェア(DTLScope.exe) のインストール

4.1 DTLScope.exe (ICS++用ソフトウェア) のインストール

4.1.1 PC 側ソフトウェアのインストール

USB メモリ/DVD-ROM 内の ICSPPlus_PC_Soft_V1.5.0.0 フォルダの中、もしくは、デスクトップラボ社の HP からダウンロードした DTLScope_1.5.0.0_installer.exe を実行して、DTLScope.exe をインストールしてください。本ソフトウェアをダウンロードするには、.Net framework 4.5 が必要です。

お使いになる PC に、.Net framework 4.5 がインストールされていない場合、DTLScope.exe をインストールする前に、Microsoft update から最新版をダウンロード、もしくは最新版に更新してください。

USB ドライバーのインストール

4.1.2 USB ドライバーインストール (W2001, W2002, T2001C, T2006 以降の ICS バージョン)

USB ドライバーのインストール方法は、Windows バージョンにより異なります。

USB ドライバーのインストールは、Windows の種類や、先に ICS を接続してしまっているなど、さまざまな状況が考えられるため、ここでは、Windows7 の例を示します。

付属の USB メモリー、DVD-ROM、もしくは、デスクトップラボの HP からダウンロードした USB ドライバーを実行してください。もしくは、FTDI 社の最新の VCP Drivers をインストールしてください。USB ドライバーが PC にコピーされます。

その後、ICS 用の USB ポートに付属の USB ケーブルを接続し、PC と接続すると、1～2 分程度で USB ドライバーがインストールされます。

4.1.3 USB ドライバーのインストール(W1003, W1004, T2001B, T2002B)

(注意、この部分は、過去の ICS モデル用の記述です。ICS++ W2002 には対応していません)

USB ドライバーのインストール方法は、Windows バージョンにより異なります。

USB ドライバーのインストールは、Windows の種類や、先に ICS を接続してしまっているなど、さまざまな状況が考えられるため、ここでは、Windows7 の例を示します。

ICS 用の USB ポートに付属の USB ケーブルを接続してください。



図 3-40 ICS++ ツール接続

②【スタート】→【コントロールパネル】→【ハードウェアとサウンド】→【デバイスマネージャー】を選択してください。例として、下記のようなビックリマークがついた、[CDC USB Demonstration]が現れます。

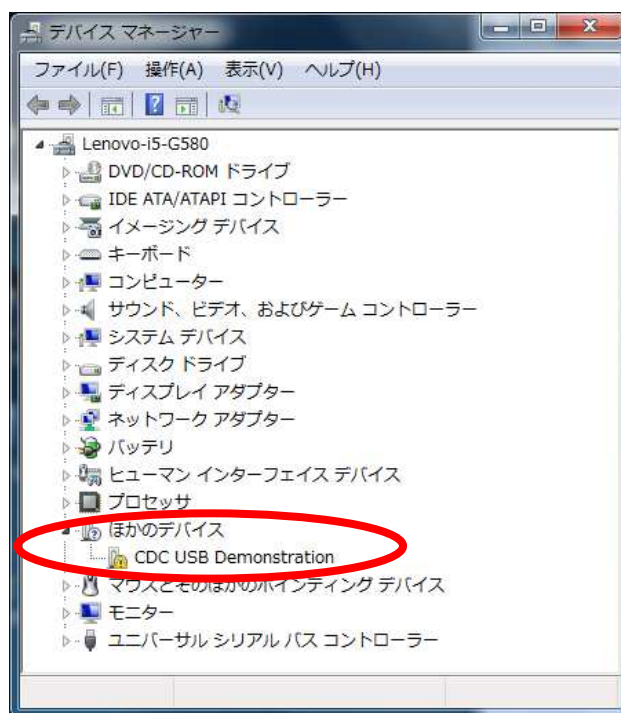


図 3-41 デバイスマネージャー

③[CDC USB Demonstration]を選択し、右クリックします。そうすると、【ドライバーソフトウェアの更新】が現れるので、クリックしてください。さらに、下のような画面が現れます。

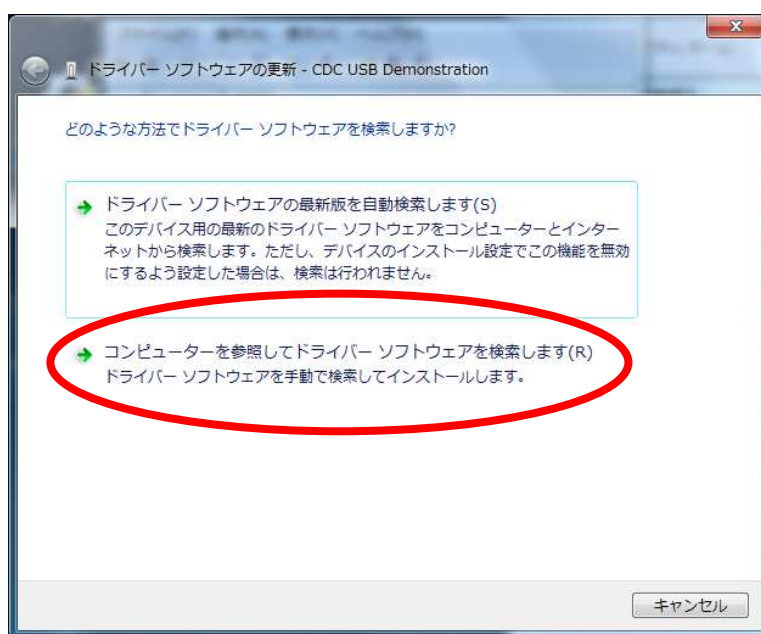
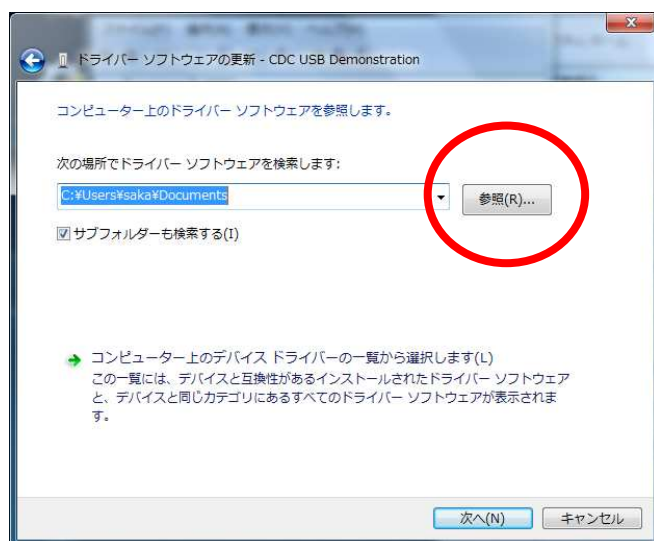


図 3-42 USB ドライバーソフトウェアの更新方法の選択

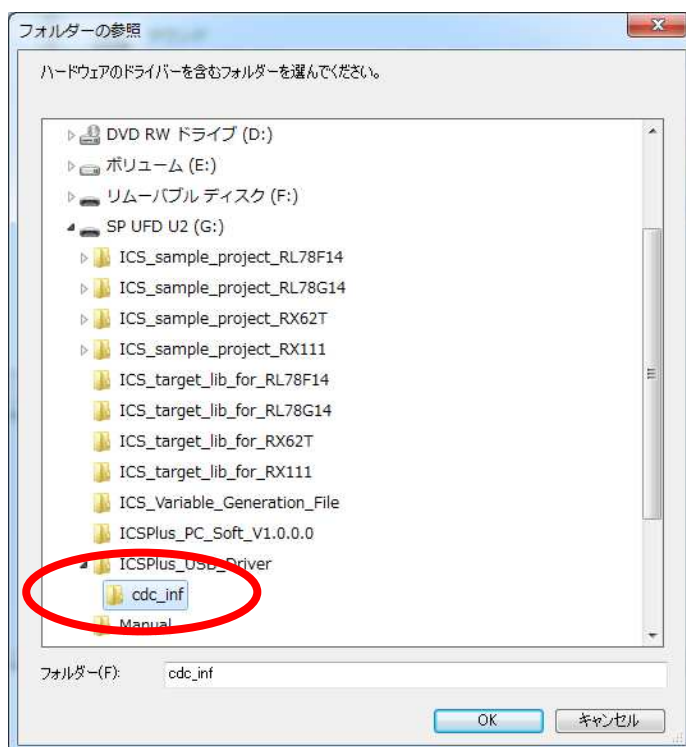
- ④上の画面で、【コンピューターを参照して、ドライバーソフトウェアを検索します】をクリックしてください。
下のような画面が現れます。



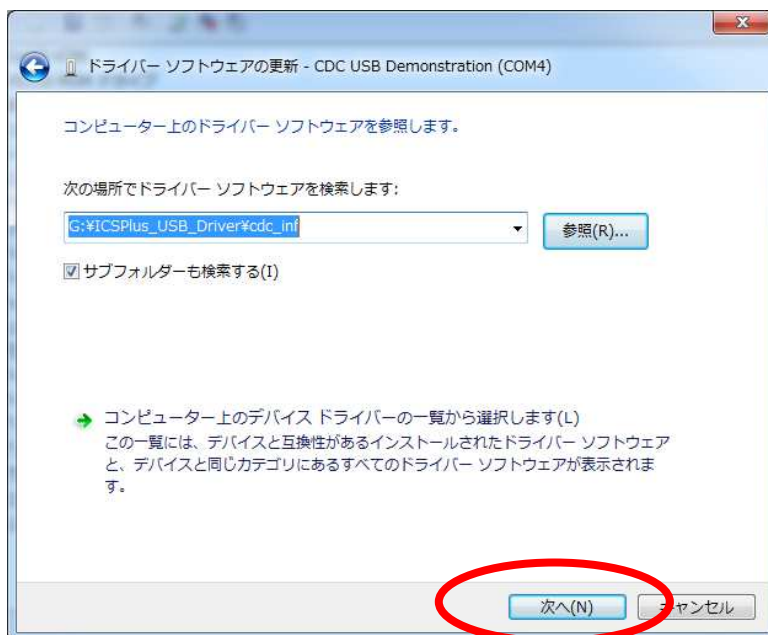
- ⑤上の画面で、『次の場所でドライバーソフトウェアを参照します』の参照をクリックし、添付されていた USB メモリの中の、

¥ICSPlus_USB_Driver¥cdc_inf

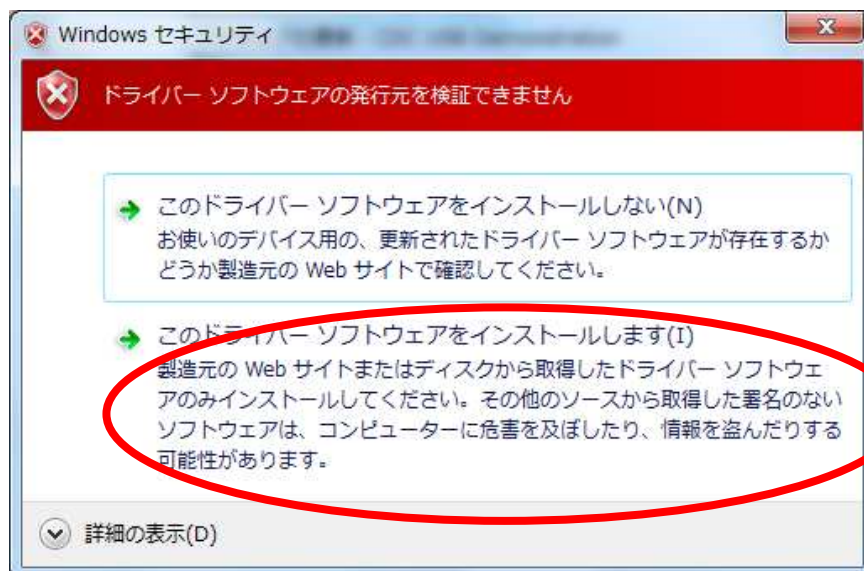
のフォルダーを選択してください。



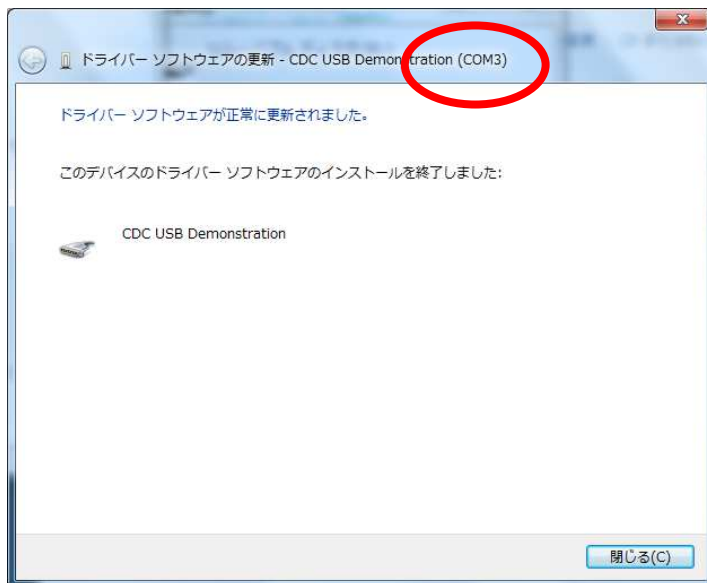
⑥選択すると、下のような画面になるので、【次へ】をクリックしてください。



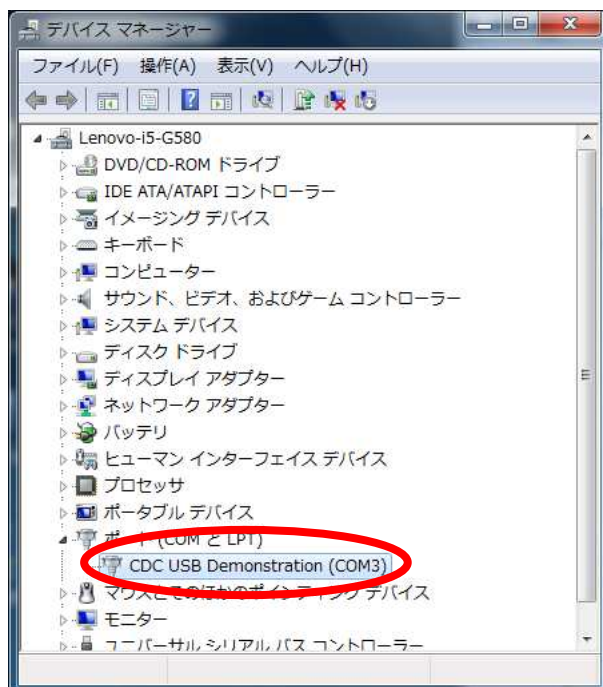
⑦下記のようなセキュリティーメッセージが現れることがありますが、下の【インストール】をクリックしてください。



⑧下記のような正常終了の画面が出れば、終了です。COM ポートの番号を覚えておいてください。この場合は、COM3 です。



⑨デバイスドライバーの画面でも、CDC USB Demonstration のポート番号が COM3 であることが確認できます。以上で USB ドライバのインストールは終了です。



5. ICS++システムの接続

5.1 ICS++システムの接続

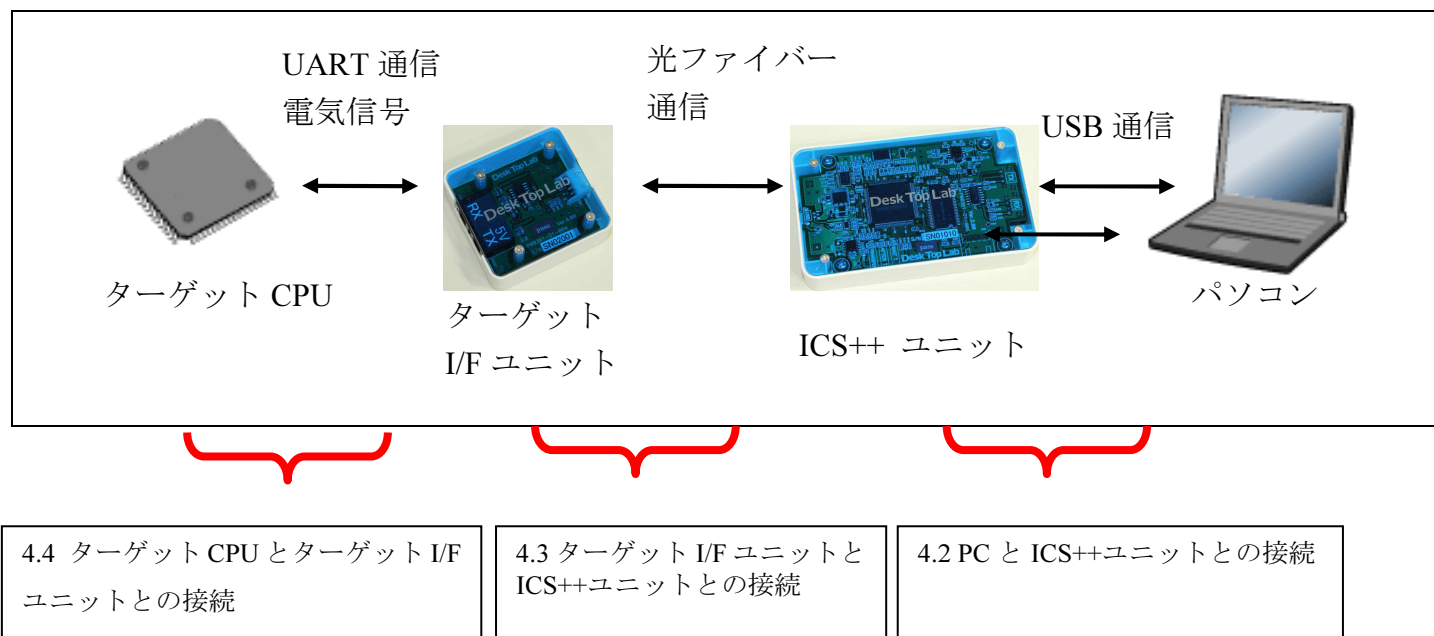


図 5-1 ハードウェア構成

5.2 PC と ICS++ユニットとの接続

ICS++ユニットは、ICS++システムを使用する上で中心となるユニットです。

PC と ICS++ユニットの間を付属の USB ケーブルで接続してください。

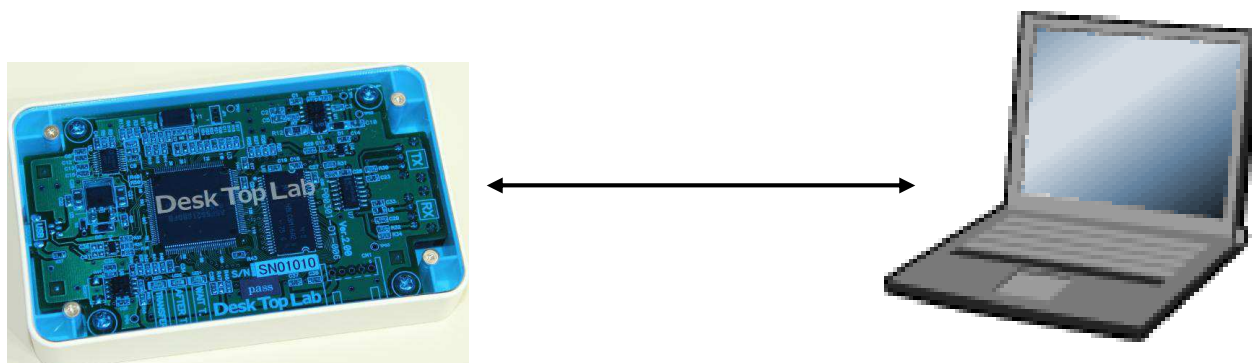


図 5-2 PC と ICS++ユニットとの接続

5.3 ターゲット I/F ユニットと ICS++ユニットとの接続

下記のようにターゲット I/F ユニットと ICS++ユニットとを光ファイバーで接続してください。

To PC (USB cable)



To Target CPU

図 5-3 ターゲット I/F ユニットと ICS++ユニットとの接続

光ファイバーは下記のように接続します。

ターゲット I/F ユニットの TX	←→	ICS++ユニット	RX
ターゲット I/F ユニットの RX	←→	ICS++ユニット	TX



図 5-4 ターゲット I/F ユニット



図 5-5 ターゲット I/F ユニット

5.4 ターゲット CPU とターゲット I/F ユニットとの接続

5.4.1 電氣的仕様概要

ターゲット I/F ユニットの 4 ピンコネクタからターゲット CPU へ接続します。5V I/O の CPU の場合には、5V 版のターゲット I/F ユニットの、3.3V I/O の CPU の場合には、3.3V 版のターゲット I/F ユニットをお使いください。



図 5-6 5V 版ターゲット I/F ユニット

表 5-1 接続端子

コネクタ	ピン番号	機能	備考
XH 4pin (JAE)	1	Vcc	
	2	CPU の TXD ポートへ	
	3	CPU の RXD ポートへ	
	4	GND	

表 5-2 5V 版 仕様

項目	MIN	Typ	MAX	備考
電源電圧(Vcc)	4.75V	5.00V	5.50V	
電源電流		17mA	40mA	
高レベル入力電圧	2.0V		Vcc	
低レベル入力電圧	0V		0.8V	
高レベル出力電圧	3.5V	-	-	Vcc=4.75V, Ioh=-150uA
低レベル出力電圧	-	-	0.4V	Vcc=4.75V IOL=2.0mA

表 5-3 3.3V 版 仕様

項目	MIN	Typ	MAX	備考
電源電圧(Vcc)	3.0V	3.3V	3.6V	
電源電流		17mA	40mA	
高レベル入力電圧	2.0V		Vcc	
低レベル入力電圧	0V		0.8V	
高レベル出力電圧	2.4V		-	Vcc=3.0V, Ioh=-2mA
低レベル出力電圧	-	-	0.55V	Vcc=3.0V IOL=2mA

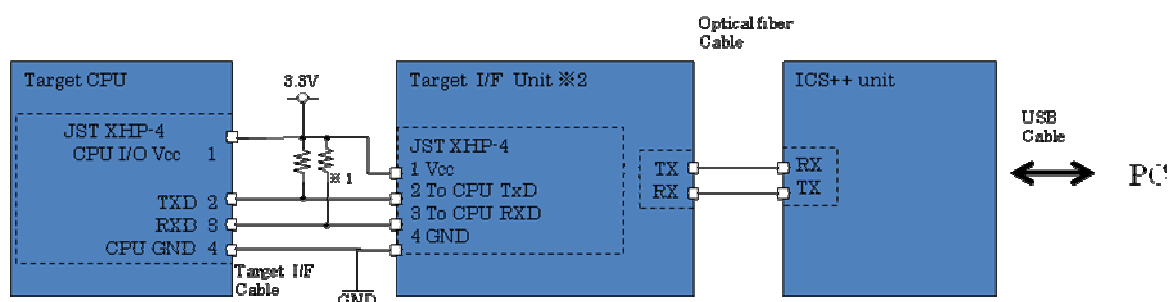
5.4.2 端子処理具体例

ICSに接続するユーザー側の TXD / RXD ピンは、 $10k\Omega \sim 47k\Omega$ 程度の抵抗で CPU の I/O 端子と同じ電位にプルアップしてください。

本来、ICS のターゲット I/F ユニットを接続している際には、ユーザー側の特別な端子処理は不要です。

ただし、ICS はデバッグツールであるため、何も接続しない場合があります。その際、RXD, TXD ピンがハイインピーダンスになる期間が存在するため、弊社ではプルアップを推奨しています。

5.5 ICS++ユニット、ターゲット IF ユニット、ターゲット CPU 接続例



※1: TXD, RXD は、 $10k\Omega \sim 47k\Omega$ でプルアップを推奨

※2: 3.3V I/O の CPU は、3.3V 対応のターゲット I/F ユニットを選択し、

5V I/O の CPU は、5V 対応のターゲット I/F ユニットを選択してください。

※3: ICS++には 2 種類のターゲットケーブルを添付しています。

片側に XHP-4 コネクタが圧着されているタイプ

両側に XHP-4 コネクタが圧着されているタイプ

弊社標準ボードでは、両側コネクタ付ターゲット I/F ケーブルを使用することにより、ハーネス加工なしに接続可能です。

6. 注意事項

6.1 ICS++のサポート CPU について

ICS++を使用する上での注意事項を以下に示します。

ICS++は、UART と DMA/DTC/FIFO を持つ 16bit 以上の殆ど全ての CPU で使用することが可能です。

2017 年 3 月 23 日時点で、ルネサス RL78G1G のみ不能使用であることを確認しております。

具体的な CPU についてご確認が必要な場合、デスクトップラボにお問い合わせください。

6.2 ICS++の通信クロック周波数計算方法

6.2.1 RX24T の場合

ICS++専用ライブラリでは、パラメータの設定により、0.5Mbps から最大 5Mbps の通信ができるようになります。

RX24T ICS ライブラリで使用する CPU のクロックを変更する場合、PC ソフトから

Setting ⇒ Communication Settings

で ICS ユニットに設定するクロックを変更する必要があります。ICS ライブラリでは、CLK=80MHz, PCLK=40MHz, ics2_init()で指定した SPEED=0 とした時、ICS ユニット上の通信クロックレートは 5Mbps となり、DTLScope へ設定するクロック周波数は 40MHz となっています。

ターゲット CPU 側で PCLK=40MHz, Speed パラメータ=0 以外の通信レートを使用する場合、

DTLScope に設定するクロック周波数は下記の式に従います。

クロック周波数 = $PCLK$

CLK	PCLK	ics2_init() Speed パラメータ	通信レート	Clock frequency
80MHz	40MHz	0	5Mbps	40MHz
80MHz	40MHz	1	2.5Mbps	20MHz
80MHz	40MHz	2	5/3 Mbps	13.333MHz
80MHz	40MHz	3	1.25Mbps	10MHz
80MHz	40MHz	4	1Mbps	8MHz

7. ホームページとサポート窓口

7.1 サポート

デスクトップラボ ホームページ

<http://desktoplab.co.jp/>

お問合せ先

saka@desktoplab.co.jp

8. 改訂記録

Rev.	発行日	対応 ICS++ PC software バージョン	改訂内容	
			Page	ポイント
1.00	2017/01/12	Ver.1.5.0.0		初版発行
1.01	2017/03/23	Ver.1.5.0.0	7	数値表示の最大チャンネル数を 48 から 24 に変更
			20	スケーリング部分の入力の定義について追記
			48	3.3V 版のターゲット I/F ボックスの電氣的仕様を修正
			50	ICS++の通信クロック周波数計算方法を RX62T から RX24T に変更
1.02	2017/5/16	Ver.1.5.0.0		

すべての商標および登録商標は、それぞれの所有者に帰属します。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れ、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。