

JobGen Plus の学習

本書では以下を学習します:

- 簡単な JobGen Plus アプリケーションの作成方法
- C 言語で複雑な JobGen Plus アプリケーションを作成する方法

目 次

第1章 簡単な JOBGEN PLUS アプリケーション	4
資産管理	4
第2章 複雑な JOBGEN PLUS アプリケーション	23
在庫管理	23

第1章 簡単な JobGen Plus アプリケーション

資産管理

このアプリケーションは各地にある現在の資産情報の収集を行うために設計されました。それぞれの資産は資産番号で表されます。ユーザはこのアプリケーションの入ったポータブル・データコレクタを持って各地にある資産番号を収集します。ユーザがこの JobGen Plus アプリケーションを設計する前に以下を決定する必要があります。

データのタイプ ポータブル・データコレクタの各データファイルは、複数のデータレコードで構成されています。各データレコードは同じデータフィールドの番号を含んでいます。二つのデータフィールドが資産管理の場所と資産番号に必要です。ユーザは各データフィールドのデータタイプを決めなければなりません。

プロセス ユーザは実行するために必要なプロセスの概略をまとめなければなりません。資産管理のアプリケーションは5つの別なプロセスが必要です。

1. 場所の情報を収集する。
2. 場所の情報を検証する。
3. 資産番号を収集する。
4. 資産番号を検証する。
5. データを編集する。

作業のフロー 資産管理の作業は以下の方法で行われます。

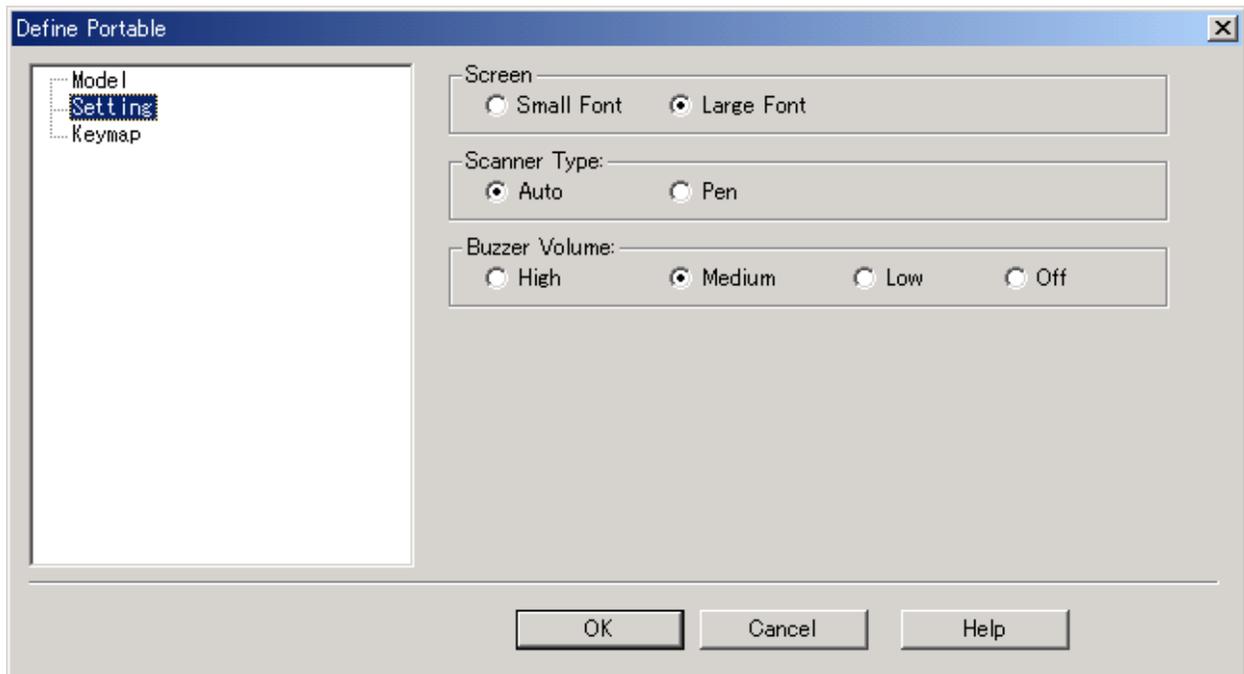
1. 場所の情報を収集します。あるいはポータブルターミナルに収集されたデータを編集する場合は”F2”キーを押します。
2. 場所の情報を検証します。
3. 場所の情報が指定された要求をパスした場合、ステップ 4 へ進みます。そうでなければステップ 1 に進みます。
4. 資産番号を収集します。
5. “F1” キーを押した場合、ステップ 1 へ進みます。そうでなければステップ 6 へ進みます。このステップは新しい場所の情報を入力するために”F1”キーを押すことができます。

6. 資産番号を検証します。
7. 資産番号が指定した要求をパスした場合、ステップ4へ進みます。このサンプルでは、それ以外もステップ4へ進みます。

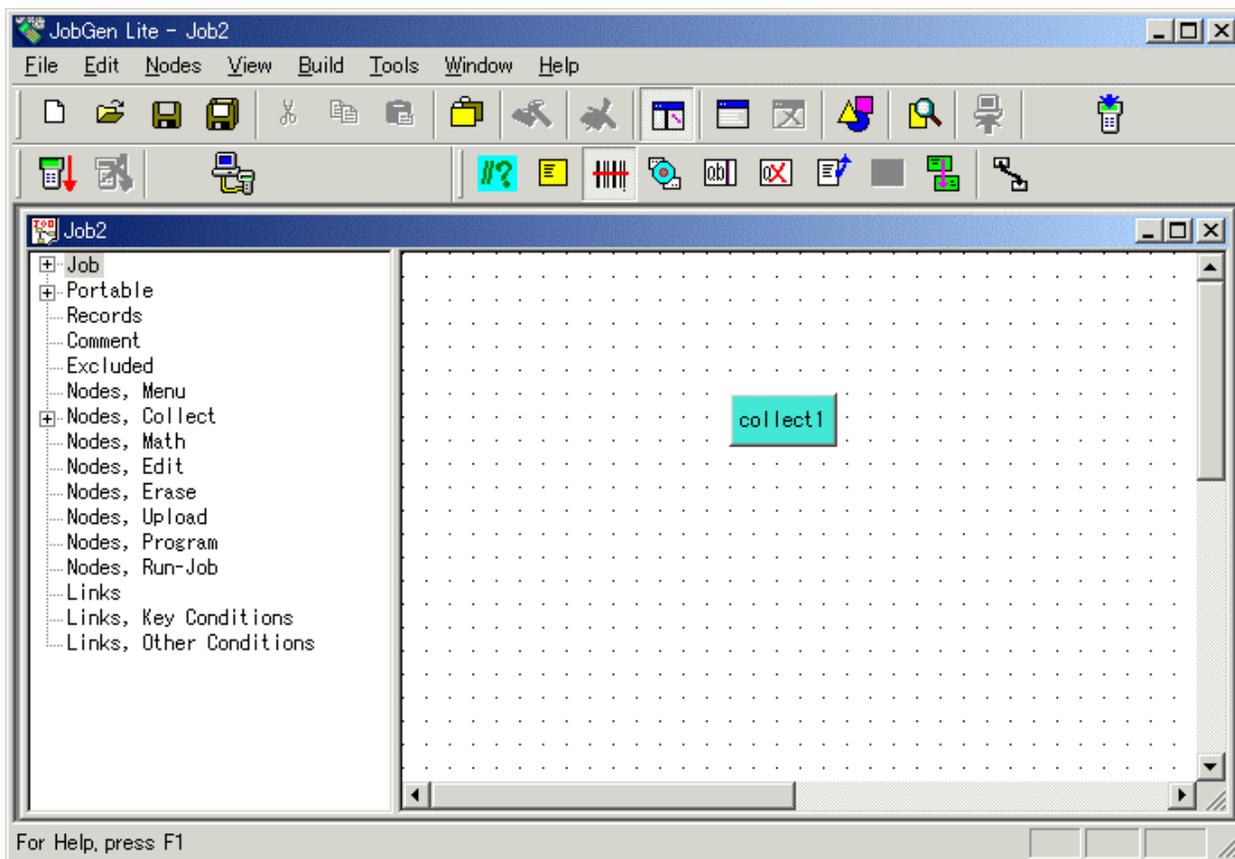
資産管理の作成

上記のリストに従って、資産管理のアプリケーションは二つのデータフィールドについて二つの収集(collect)ノード、そして収集したデータの表示と編集のために編集(Edit)ノードが必要です。また、いくつかのリンクがノード間の移動を実現するために必要です。以下のステップは資産管理のアプリケーションをどのようにして作成するかを示しています。

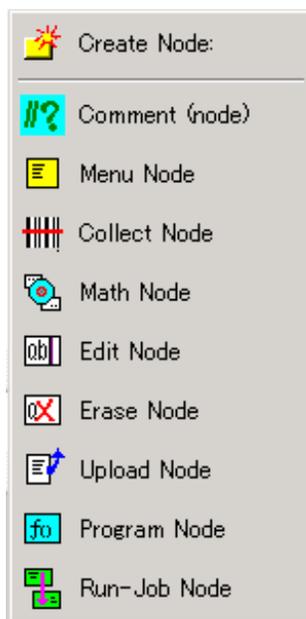
ヒント: 日本語メニューを作成する場合は、上記の作業をする前に Edit メニューから Define Portable を選択し、Setting > Screen > Large Font をチェックして下さい。



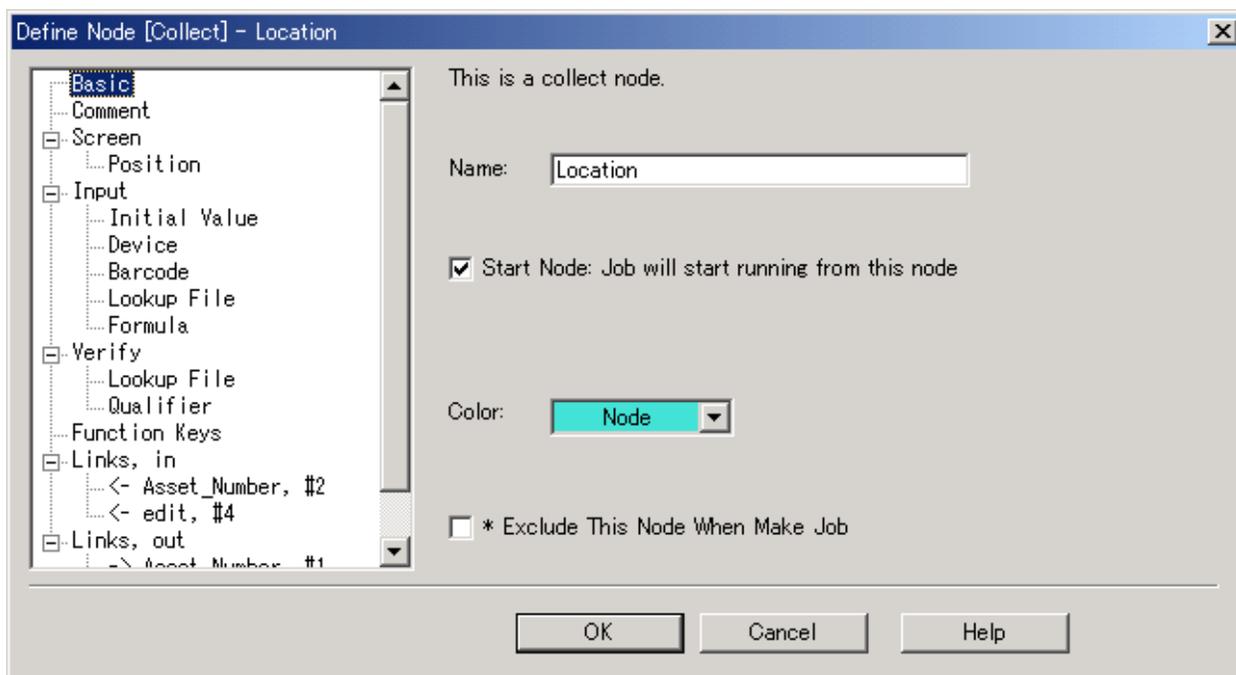
1. マウスポインタを Collect ノード ボタン  に移動して下さい。マウスポインタが Collect ノード ボタン  上にある時に左マウスボタンをクリックして下さい。そしてマウスポインタを Collect ノード が置かれる場所に移動して下さい。



左マウスボタンをダブルクリックして下さい。
 あるいは、ノードを置きたい場所で右クリックすると、コンテキストメニューがポップアップします。作成するために Collect ノード を選択して下さい。

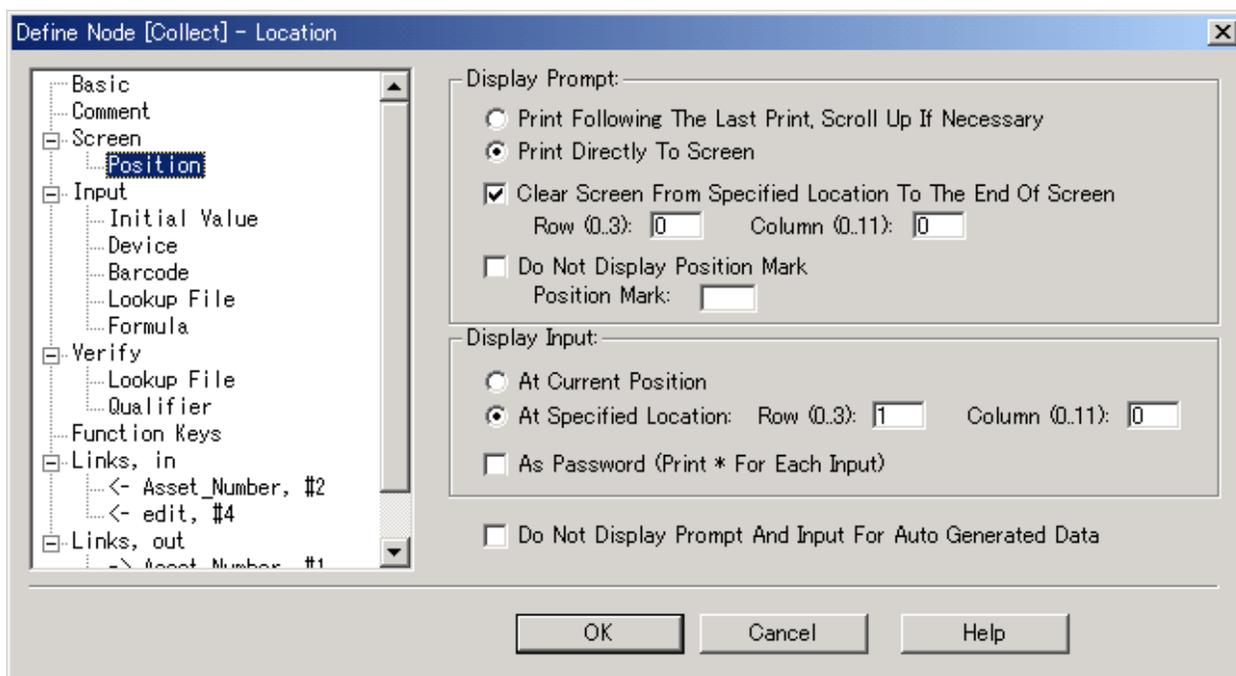


マウスボタンが Collect ノード にある間に左マウスボタンをダブルクリックして下さい。“Define Node [Collect]” ウィンドウがスクリーンに現れます。



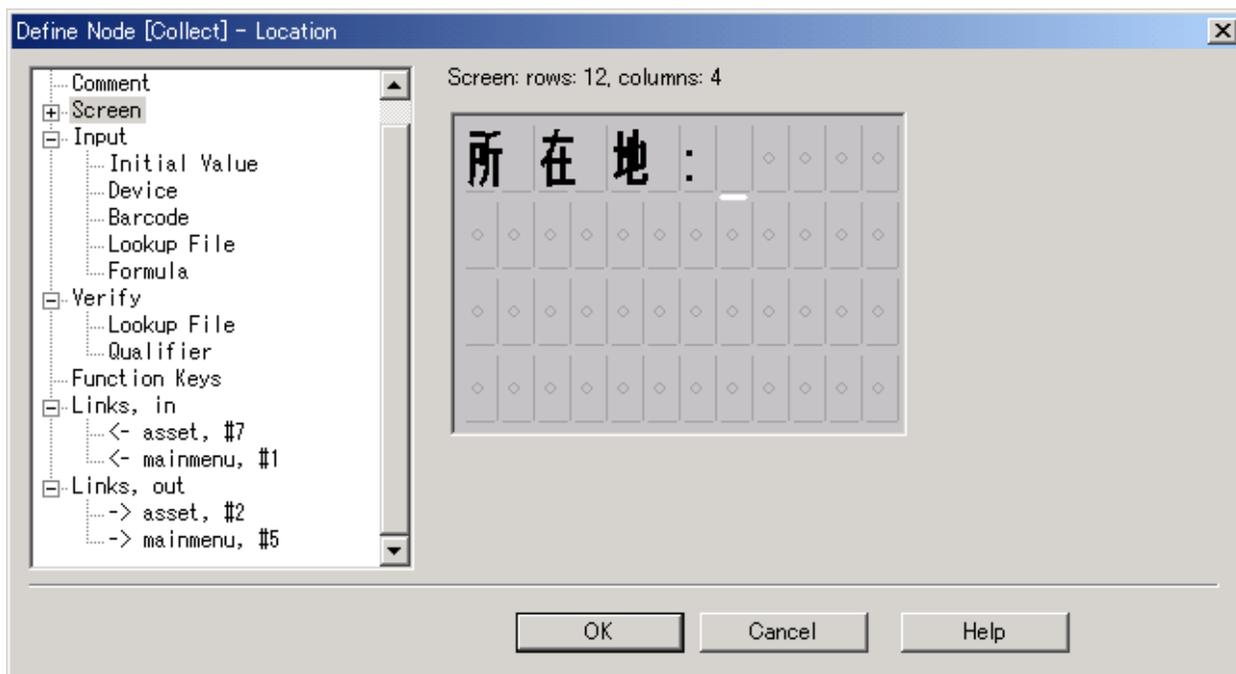
以下のオプションを変更します。

- (a) Basic プロパティで、“collect1” から “Location” に名前を変更します。
- (b) **Start Node** をチェックします。このノードは資産管理アプリケーションの始まりをマークします。
- (c) Screen > position プロパティで、“**Clear Screen From Specified Location To The End Of Screen**”(指定した場所からスクリーンの最後までをクリアする) オプションを選択し、“0” と “0” を row (行) と column (列) のボックスにタイプします。このオプションはターゲットデバイスのスクリーンを行 0 / 列 0 から最後の行 / 列まですべてクリアします。

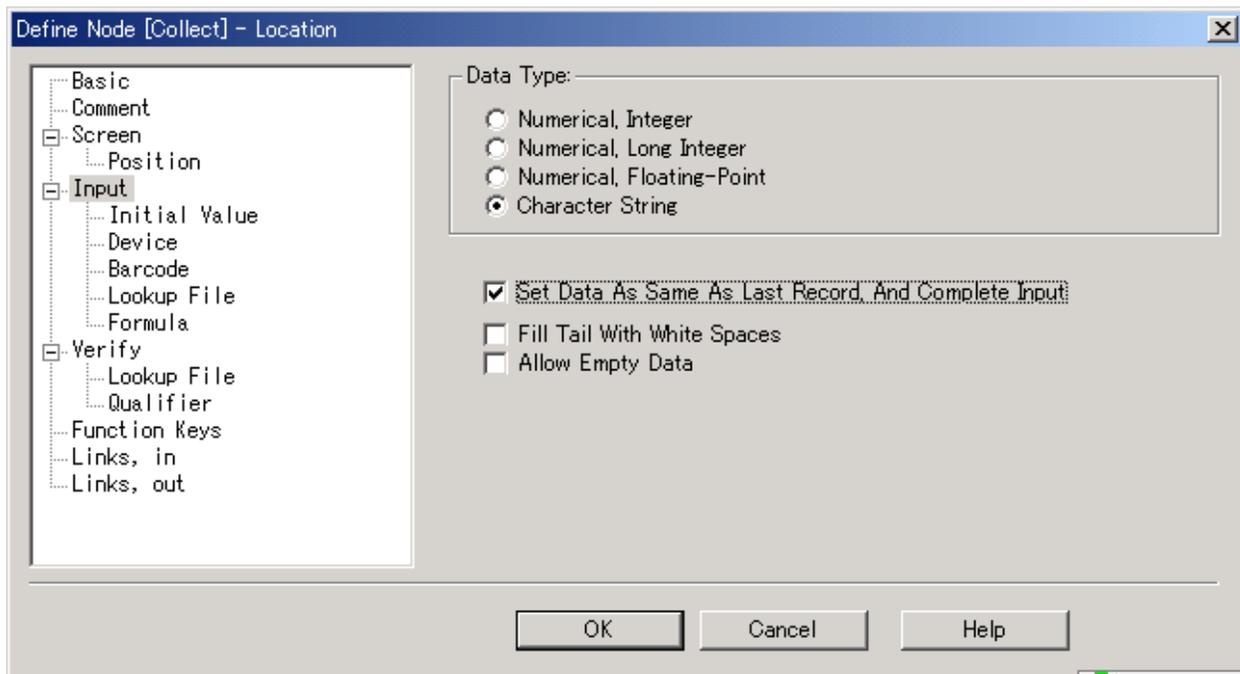


(d) “At Specified Location” オプションを選択して、Row と Column のボックスに”1”と”0”をタイプします。このオプションは行 1 と列 0 の位置から入力データ(ユーザによって入力される)を表示します。

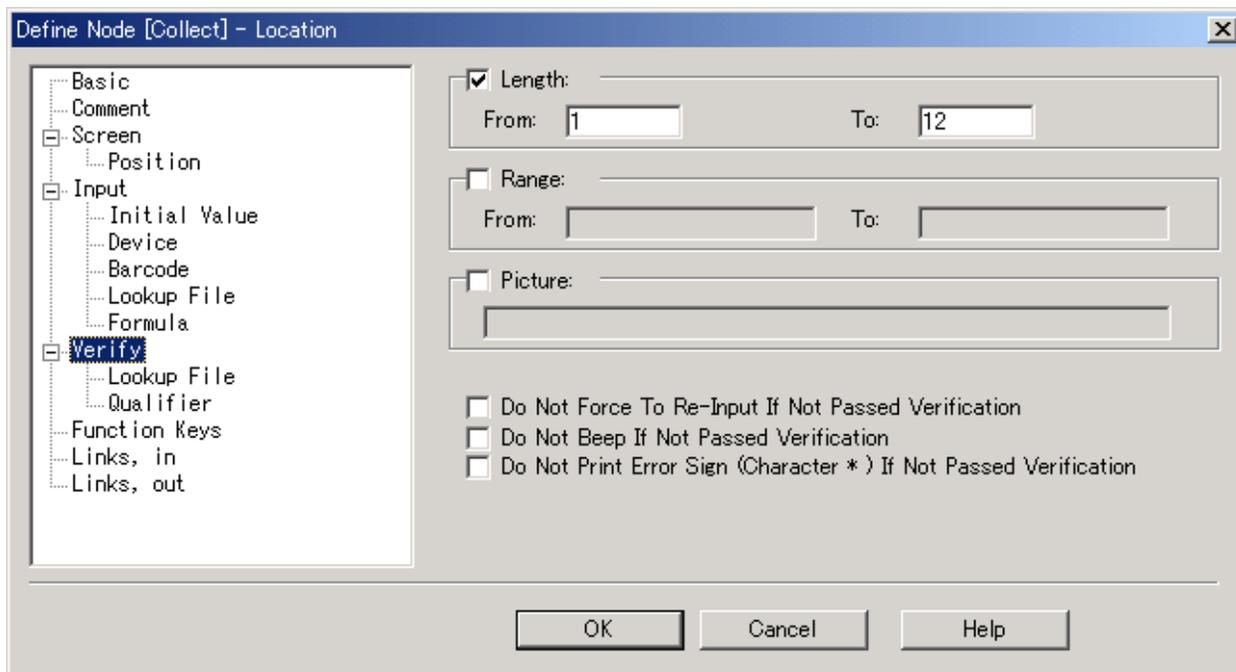
2. Screen プロパティを選択します。スクリーンのスペースに“所在地:”をタイプします。



3. Input プロパティを選択して下さい。“Define Collect” ウィンドウは以下のように変わります。



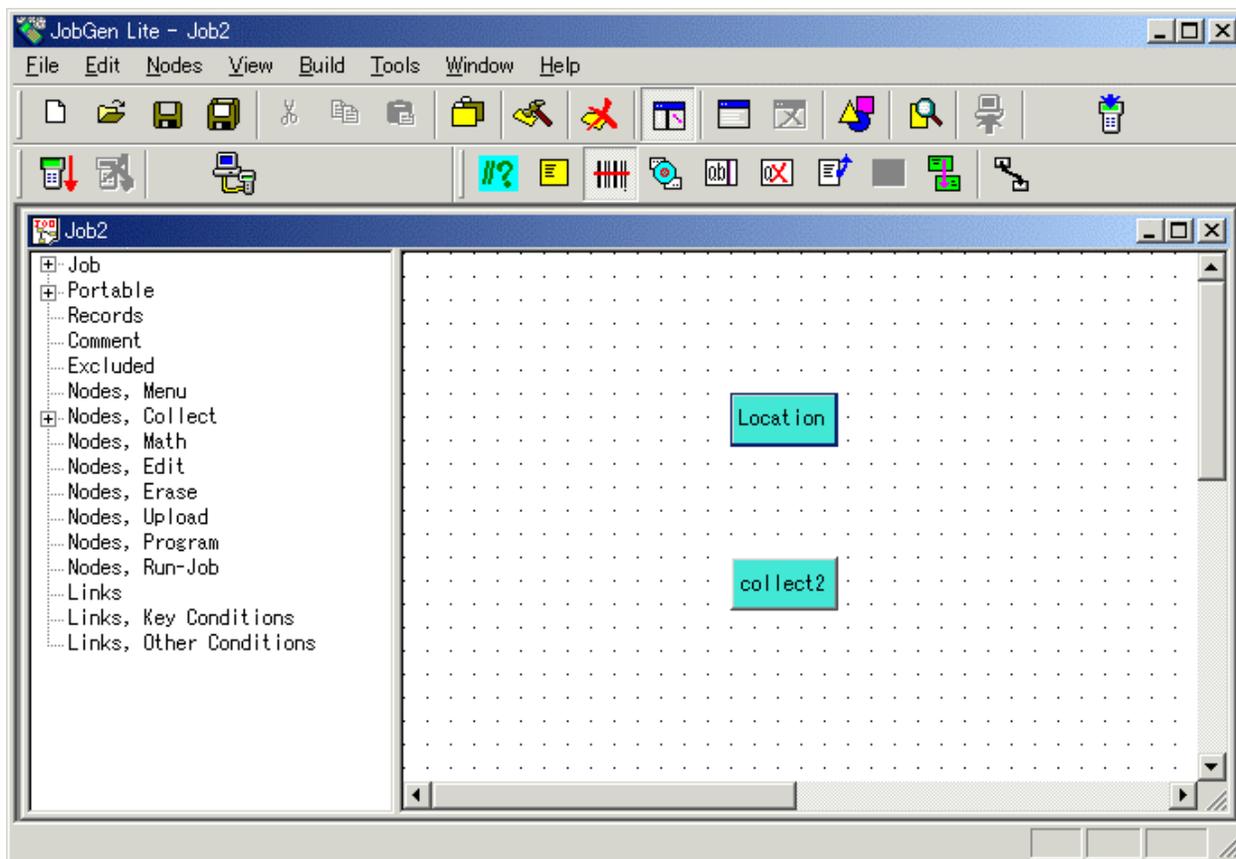
4. “Set Data As Same As Last Record, And Complete Input”をチェックします。これは“所在地”情報を一回だけ入力すればよろしく、そして”F1”キーを押すまで”資産番号”の入力を続けます。
5. Verify プロパティを選択して下さい。“Define Collect” ウィンドウは以下のように変わります:



6. **OK** をクリックして下さい。

次のステップは二番目の Collect ノード を作ります。

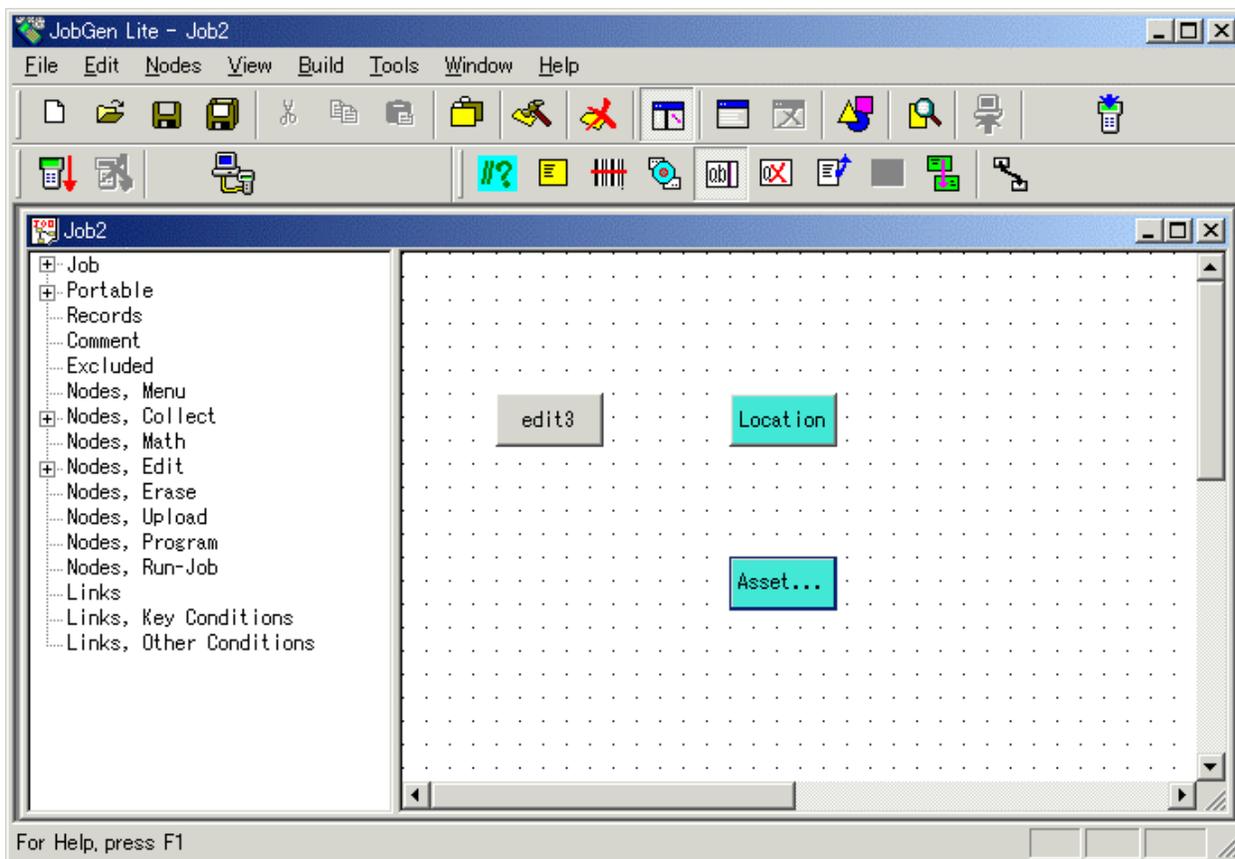
7. マウスカーソルを Collect ノード を置く場所に移動します。左マウスボタンをダブルクリックします。



8. マウスポインタが新しく作った Collect ノード にある間に左マウスボタンをダブルクリックして下さい。“Define Collect” ウィンドウがスクリーンに現れます。
9. 以下のオプションを変更します:
 - (a) “collect2” の名前を “Asset_Number”に変更します。
 - (b) Screen > Position プロパティで、“**Clear Screen From Specified Location To The End Of Screen**” を選択して、“1”と“0”を入力します。このオプションはターゲットデバイスのスクリーンを行 1 / 列 0 から最後の行と欄まですべてクリアします。
 - (c) “**At Specified Location**” を選択して、“1”と“0”をタイプします。このオプションは、行 1 と 列 0 の位置から入力データ(ユーザ入力データ)を表示します。
 - (d) Input > Initial Value のチェックは両方とも付けません。
10. Screen プロパティを選択します。“Define collect” ウィンドウは以下のように変わります。スクリーンのスペースに“資産#入力:”をタイプして下さい。

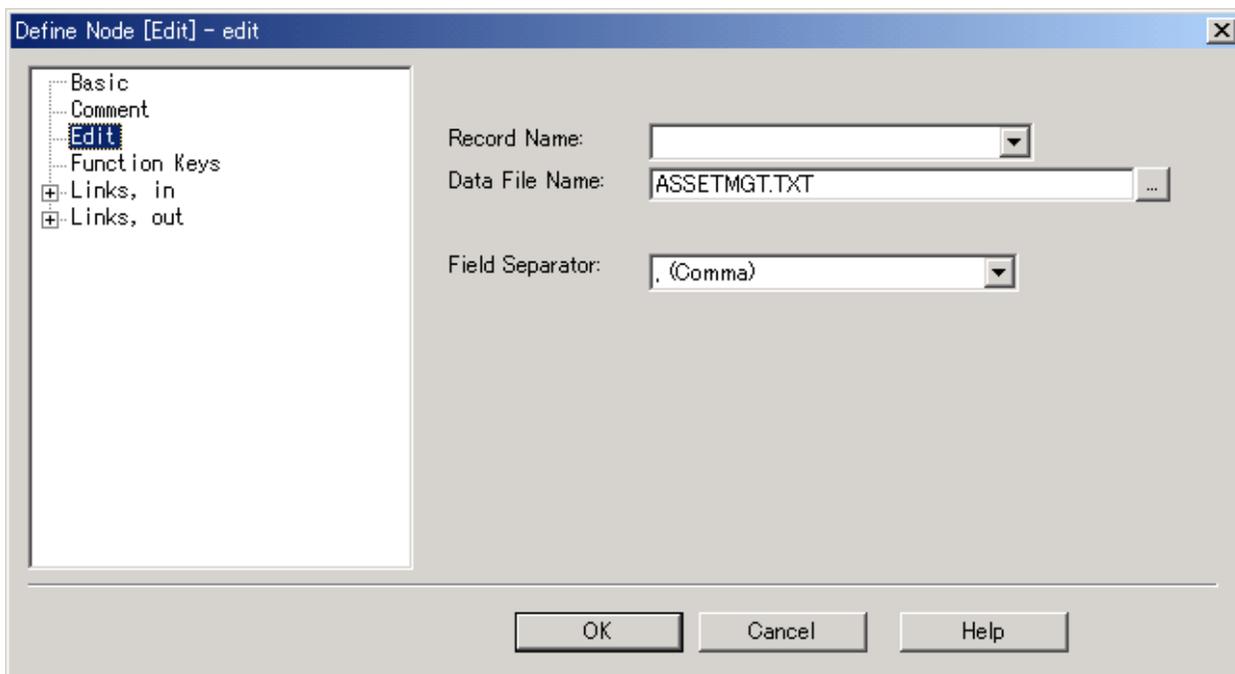


11. **OK** をクリックして下さい。
次のステップは Edit ノード を作ります。
12. マウスポインタを Edit ノード ボタン  に移動します。マウスポインタが Edit ノード ボタン  がある間に左マウスボタンをクリックします。次に、マウスポインタを Edit ノード を置く場所に移動します。



左マウスボタンをダブルクリックします。

13. マウスポインタが Edit ノード にある間に左マウスボタンをダブルクリックします。
“Define Edit” ウィンドウがスクリーンに現れます。



14. 以下のオプションを変更します。

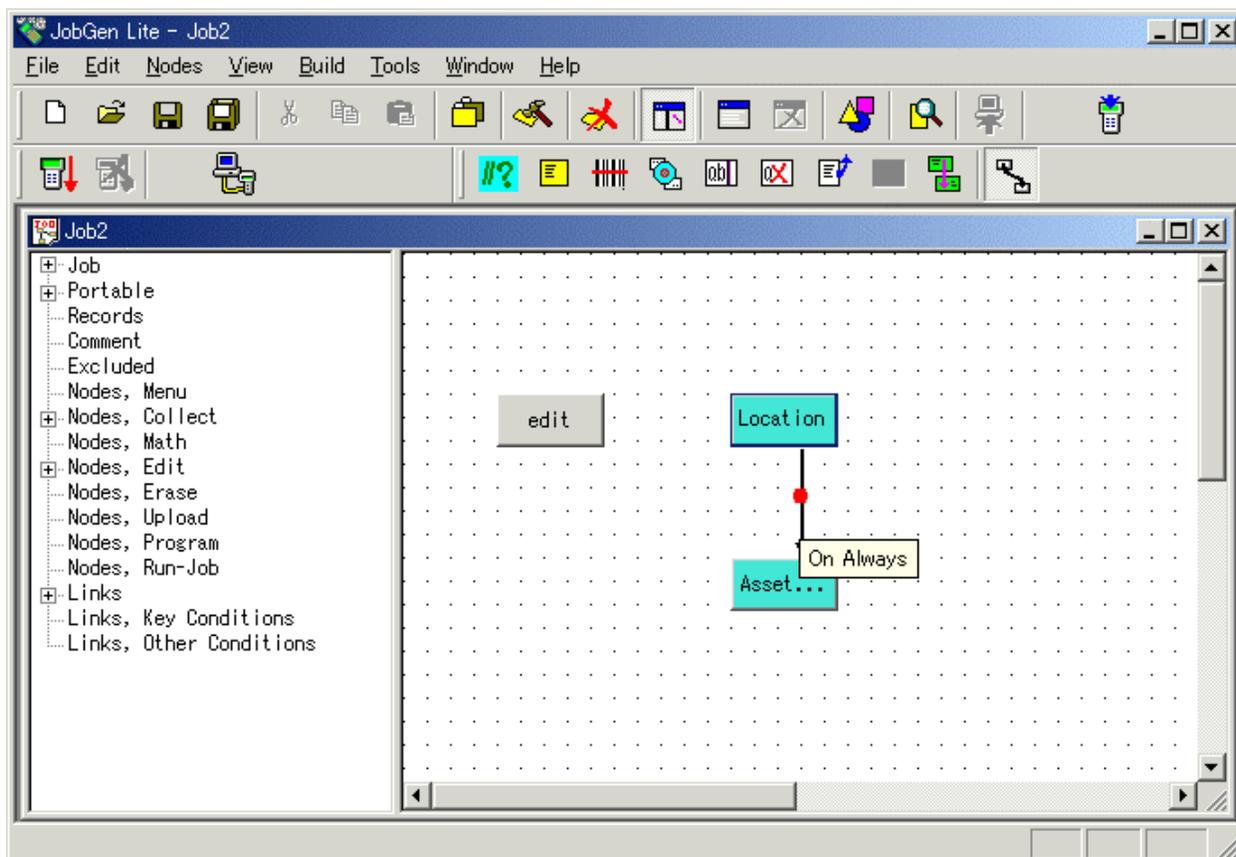
- (a) 名前を “Edit3” から “Edit” に変更します。
- (b) **Data File Name** オプションとして ASSETMGT.TXT をタイプします。
- (c) **Field Separator** について “, [Comma]” を選択します。

15. **OK** をクリックします。

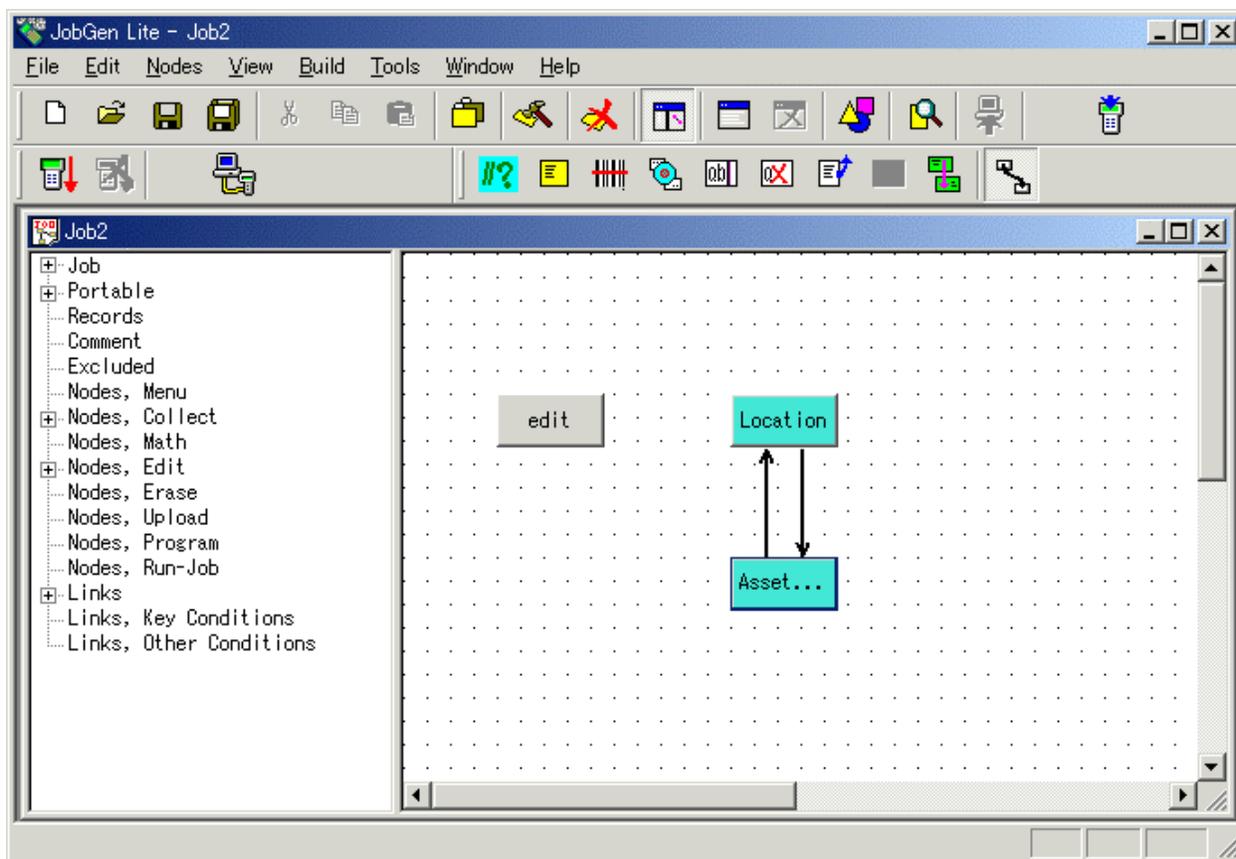
次のステップは最初の Collect ノード から二番目の Collect ノード へのリンク を作ります。

16. マウスポインタをリンク ボタン  に移動します。マウスポインタがリンク ボタンにある間に左マウスボタンをクリックします。そしてマウスポインタを “Location” ノード に移動します。左マウスボタンをクリックしたままにします。マウスポインタを “Asset_Number” ノード に移動し、左マウスボタンを離します。

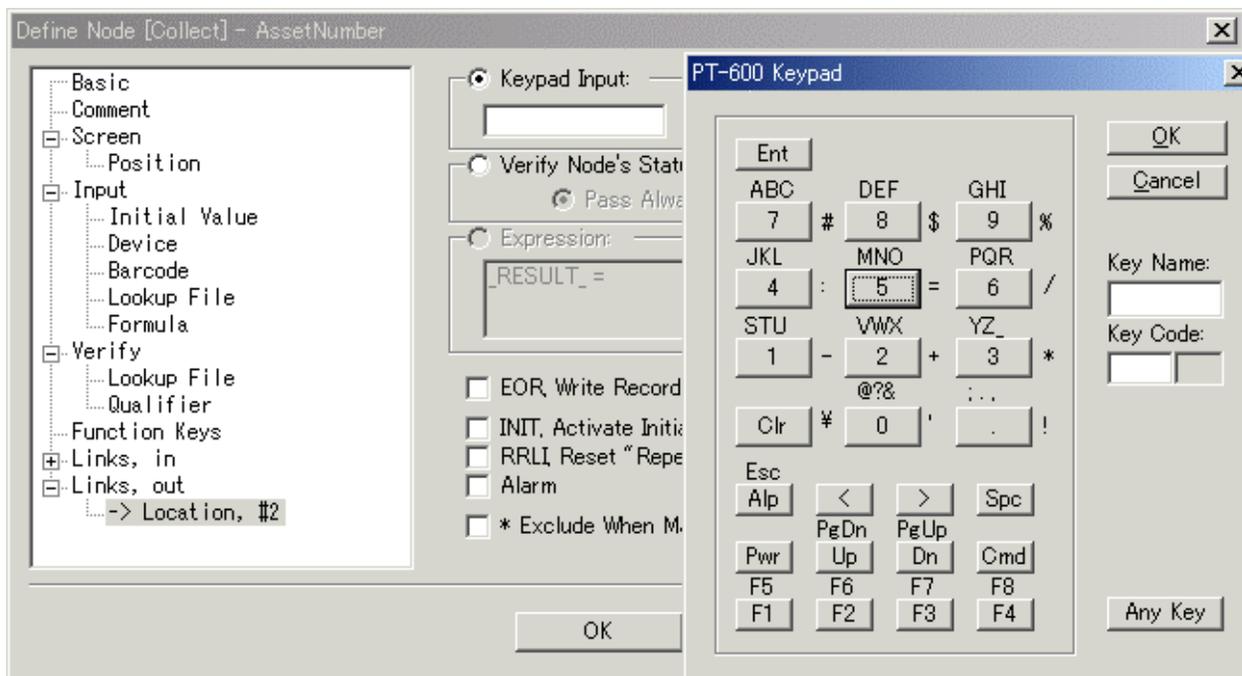
17. マウスポインタを新しい作成したリンク に移動し、そして、このリンク についての情報が現れます。リンク の標準設定は “On Always” です。これは “Location” ノード がデータ収集を終了したら、“Asset_Number” ノード に進むことを示しています。



18. 次のステップは “Asset_Number” ノード から “Location” ノードへの別なリンクを作ります。
19. マウスポインタを “Asset_Number” ノード に移動します。左マウスボタンをクリックしたままにします。マウスポインタを “Location” ノード に移動し、左マウスボタンを離します。



20. 新しく作ったリンク上にマウスポインタがある間に左のマウスボタンをダブルクリックします。 “Define Node” ウィンドウがスクリーンに現れます。
21. 以下のオプションを変更します。
 - (a) **Keypad Input** をチェックします。そしてマウスを空白の行に移動し、左マウスボタンをクリックします。 “PT-600 Keypad” ウィンドウがスクリーンに現れます。
マウスポインタを F1 キーに移動し、左マウスボタンをクリックします。 **OK** をクリックします。この選択はユーザがポータブル・データコレクタで “F1” キーを押した場合、 “Location” ノード が直ちに処理されることを示しています。



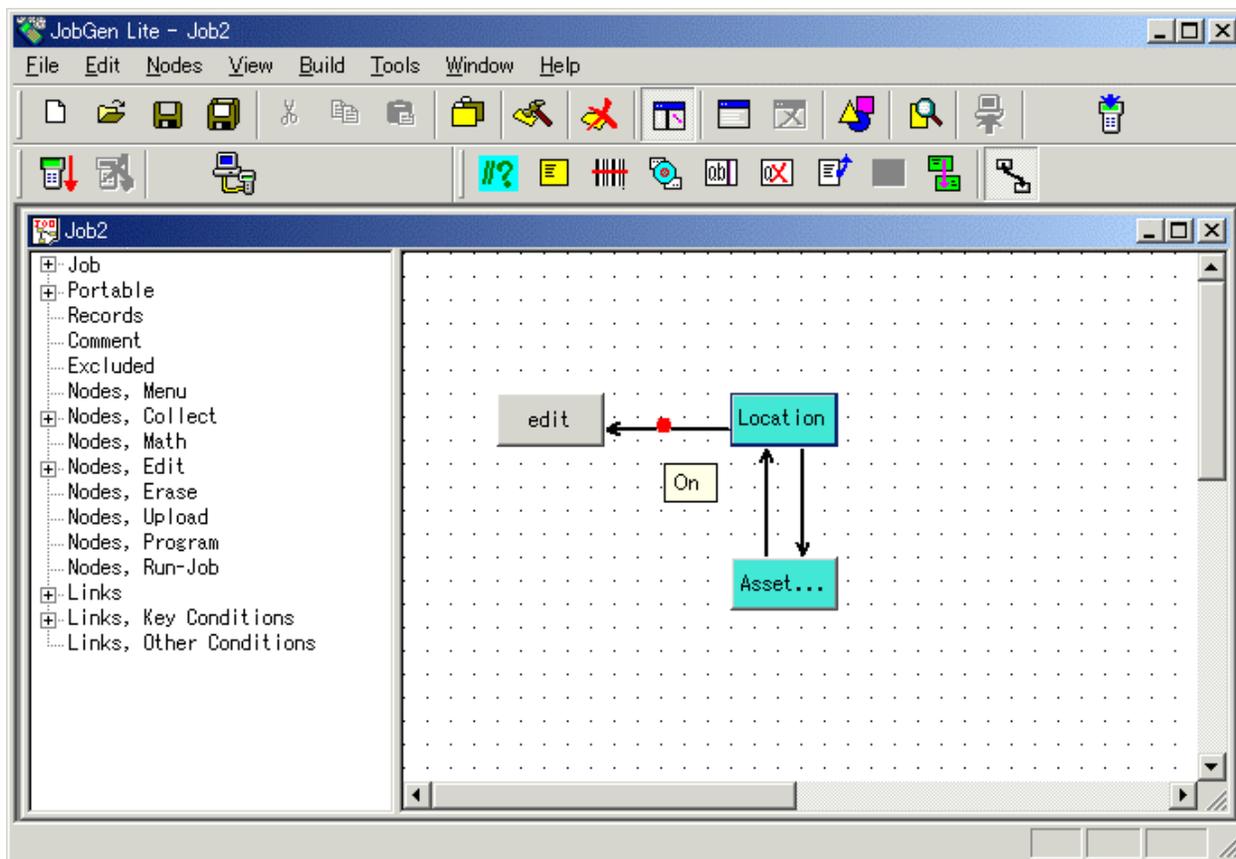
(b) “**RRLI, Reset ‘Repeat Last Input’ For All Nodes**”をチェックして下さい。この選択は“Location”ノードの“**Set Data As Same As Last Record, And Complete Input**”オプションに関係します。**RRLI**がチェックされているこのリンクがアクティブになったら、“Location”ノードは新しいデータをユーザが入力する必要があります。

22. **OK** をクリックします。

次のステップは、最初の Collect ノード から Edit ノード へのリンク を作成します。

23. マウスポインタを“Location”ノードに移動します。左マウスボタンをクリックしたままにします。マウスポインタを“Edit”ノードに移動し、左マウスボタンを離します。

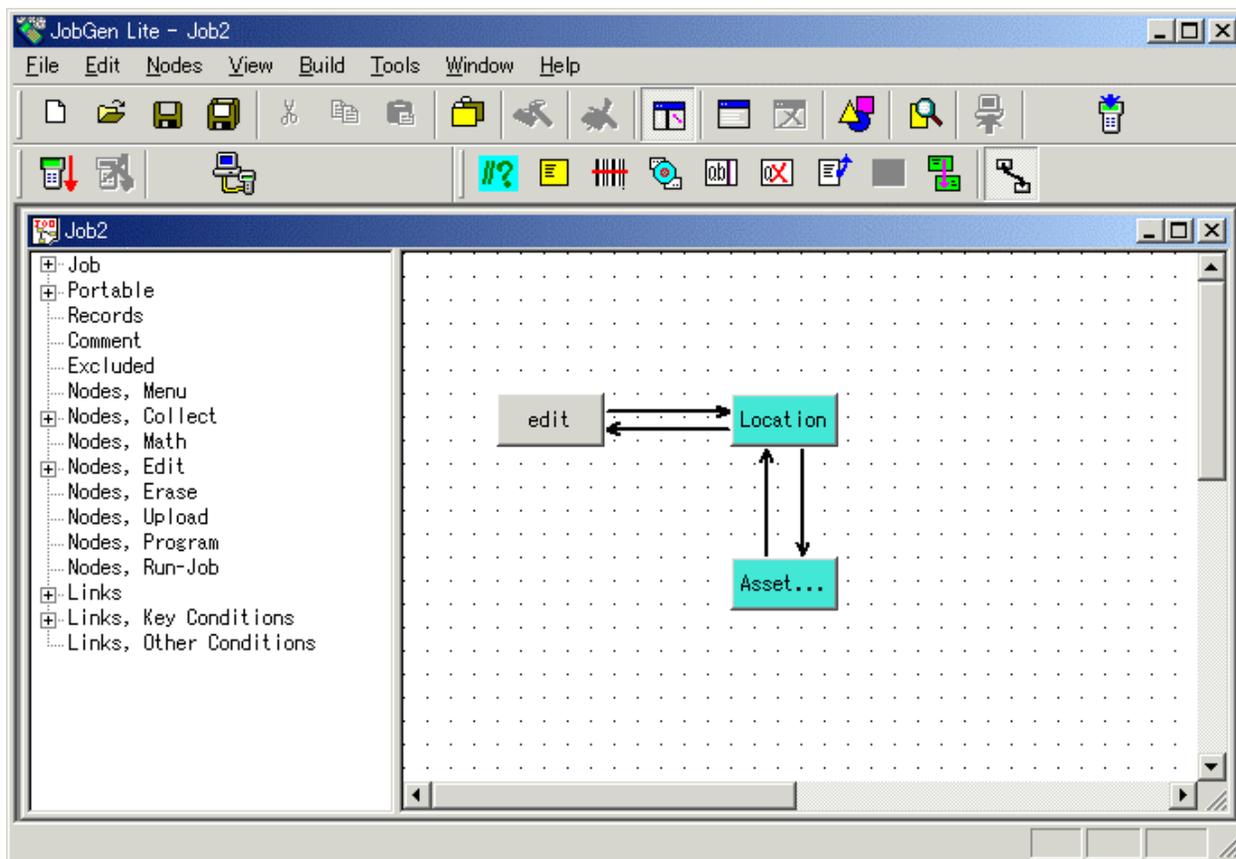
24. マウスポインタを新しく作成したリンクに移動し、そしてこのリンクについての情報がスクリーンに現れます。



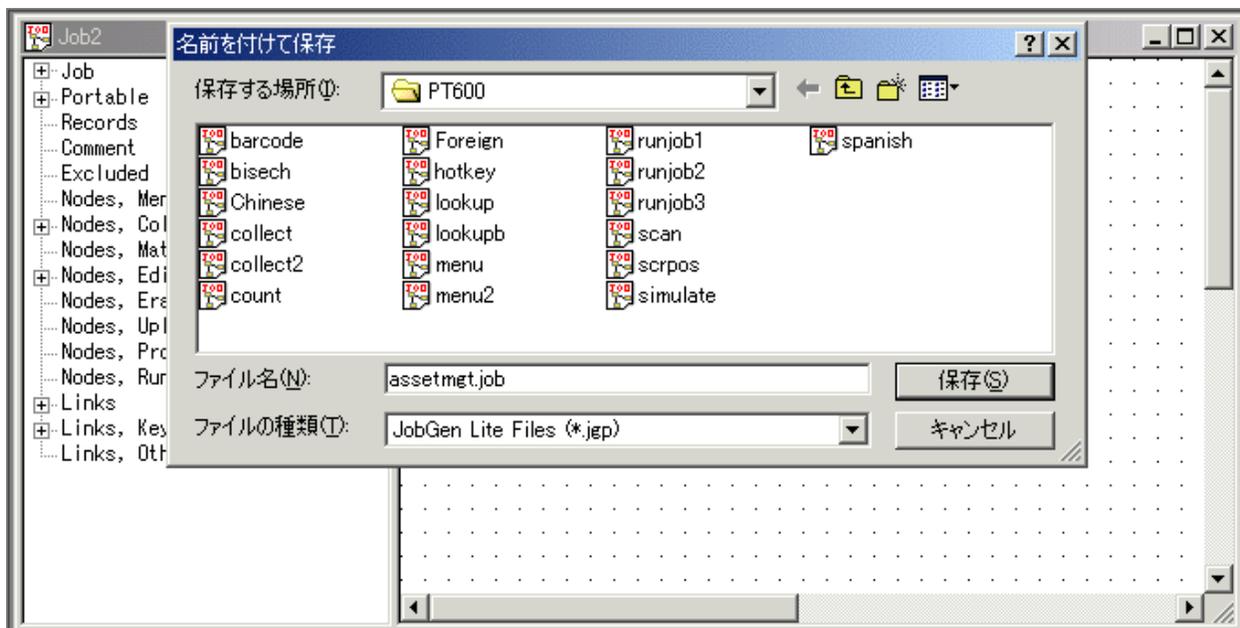
25. 以下のオプションを変更します:

- (a) **Keypad Input** をチェックします。そしてマウスを空白の行に移動して左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます (前のリンクと全く同じ)。
- (b) カーソルポインタを **F2** キーへ移動して、左マウスボタンをクリックします。**OK** をクリックします。この選択はユーザがポータブル・データコレクタの”F2”キーを押した場合、“Edit” ノード が直ちに処理されます。次のステップは Edit ノード から最初の Collect ノード へのリンク を作ります。

26. マウスポインタを “Edit” ノード へ移動します。左マウスボタンをクリックしたままにします。マウスポインタを “Location” ノード へ移動して、左のマウスボタンを離します。



- マウスポインタを新しく作成したリンクに移動し、そしてこのリンクについての情報がスクリーンに現れます。リンクの標準設定は“On Always”です。これは“Edit”ノードが終わった場合、“Location”ノードが処理されることを示しています。
アプリケーションを終了する前に、ジョブはまず保存されなければなりません。マウスポインタをメニューバーの **Build > Make Job** に移動します。マウスポインタが **Make Job** にある間に左マウスボタンをクリックします。そしてマウスポインタを **File > Save As...** に移動して、左マウスボタンをクリックします。
- ジョブのファイル名として、**File Name** スペース中に“assetmgt.job”を入力します。OKをクリックします。



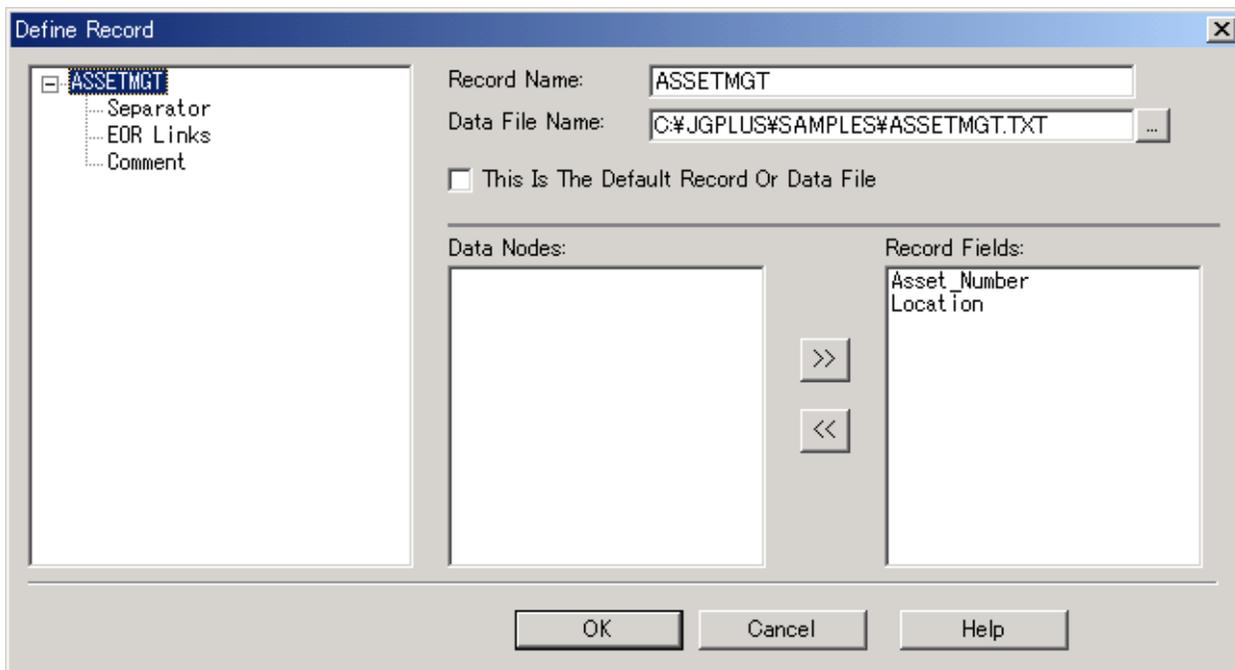
ジョブを保存した後、最後のステップはデータレコードの最後をマークする最後のリンクを別にする事です。

29. 新しいリンクの相手先として“Asset_Number”ノードを選択するループしたリンクを作成します。ノードメニューでリンクを選択するか、ツールバーのリンクアイコンをクリックします。マウスポインタを“Asset_Number”ノードに移動して、左マウスボタンをクリックしたまま、ノードの中を少し動かして左マウスボタンを離します。



はいをクリックします。

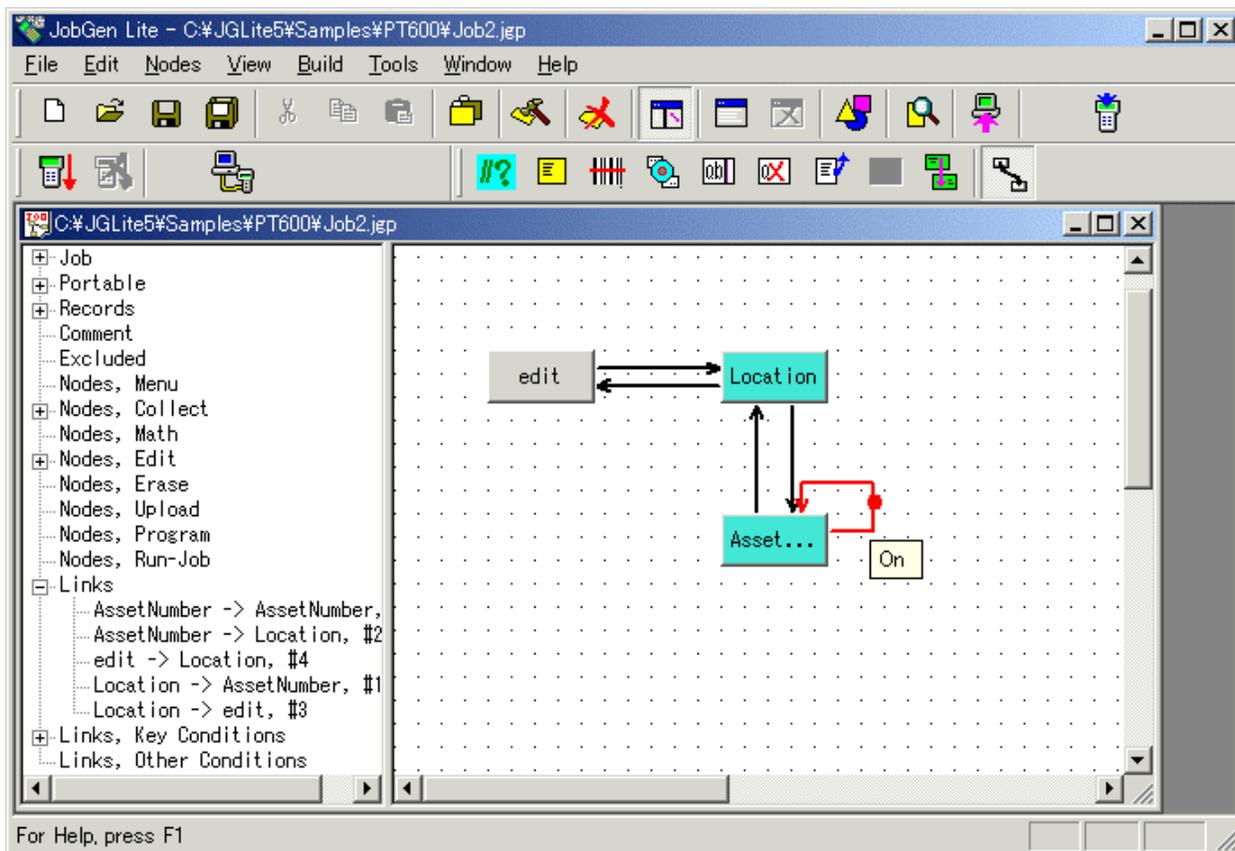
30. Edit > New Record で新しいレコードを作成し、次に Edit > Define Record でレコードを定義します。



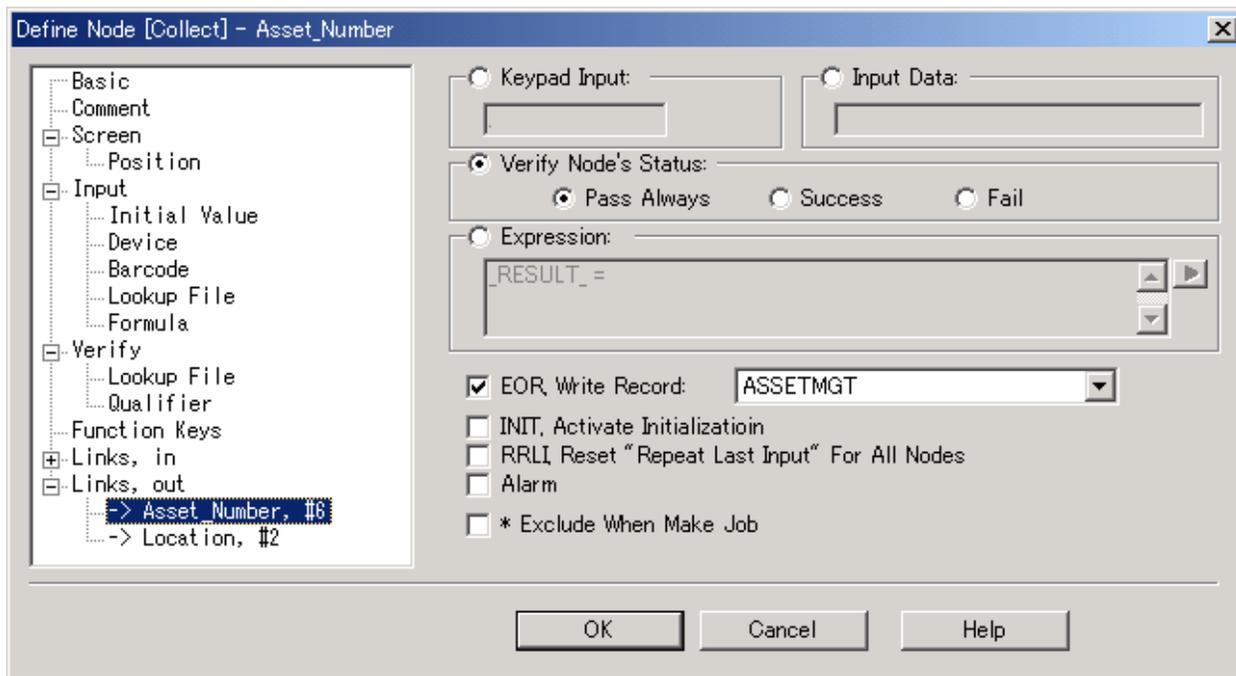
Record Name: を ASSETMGT とします。

Data Node の Asset_Number と Location を移すために、>> キーを使用します。

OK を押します。



31. 作成したリンク上をダブルクリックします。



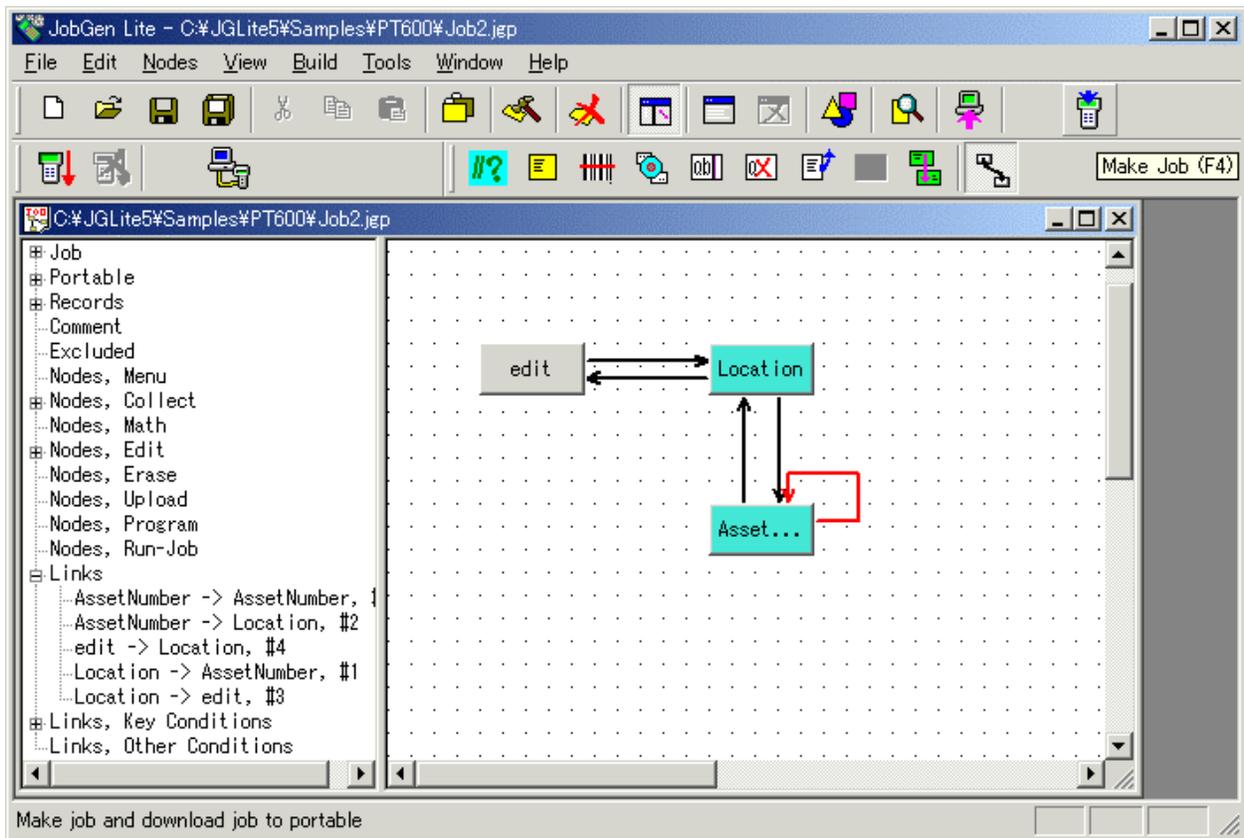
EOR をチェックします。この動作は“Asset_Number”がそのデータ収集を終了した後で、データレコードの収集も終了します。

32. Verify Node's Status をチェックし、**Pass Always** オプションを選択します。

これにより、F1 キーが押されるまでは、資産番号の収集を続けます。

33. **OK** をクリックし、ジョブを再度保存します。

34. ジョブを保存した後で、マウスのポインタをツールバーの Make JoB ボタンに置きます。資産管理のジョブをコンパイルするために左マウスボタンをクリックします。これにより実行可能な形式ファイル作成のプロセスをスタートします。



EXE ファイルを作った後で、JobGen Plus はポータブルターミナルにジョブの実行形式をダウンロードし、実行の準備ができました。

35. ポータブルターミナルを起動して、本プログラムを実行して下さい。

所在地情報を入力した後、F1 キーを押すまで資産番号の入力が続きます。

第2章 複雑な JobGen Plus アプリケーション

在庫管理

このアプリケーションは、現在の在庫情報を収集します。

在庫情報は三つの種類、部品番号のみ、部品番号と数量、そして部品番号とシリアル番号に分けることができます。このジョブを設計する前に、ユーザはこれらの基本的な要求のノートを作る必要があります:

データタイプ ポータブル・データコレクタの各データファイルは複数のデータレコードで作られています。三つのデータフィールド、Part Number(部品番号)、Quantity(数量)、Serial Number(シリアル番号)がこのジョブに必要です。ユーザは各データフィールドについてデータタイプ(文字列または整数)を決めなければなりません。このジョブは三つの違った情報タイプを含んでいるので、データレコードの内容は異なっています。部品番号だけの在庫情報については、Quantity フィールドと Serial Number フィールドは“0”にセットされます。部品番号と数量のある在庫情報については、Serial Number フィールドは“0”にセットされます。部品番号とシリアル番号を持つ在庫情報については Quantity フィールドは“1”にセットされます。

プロセス ユーザは実行する必要があるプロセスをまとめなければなりません。このジョブは 15 のプロセスを必要としています:

1. 在庫情報のタイプを選択
2. 部品番号を収集
3. 部品番号を検証
4. 数量のフィールドを 0 にセット (選択したタイプが部品番号のみの場合)
5. 数量のフィールドを 1 にセット (選択したタイプが部品番号とシリアル番号の場合)
6. シリアル番号フィールドを 0 にセット (選択したタイプが部品番号のみか、部品番号と数量の場合)
7. 数量を収集 (選択したタイプが部品番号と数量の場合)
8. 数量を検証 (選択したタイプが部品番号と数量の場合)
9. シリアル番号がバラバラか、連続かを選択 (選択したタイプが部品番号とシリアル番号の場合)

10. シリアル番号を収集 (選択したタイプが部品番号とシリアル番号の場合)
11. シリアル番号を検証 (選択したタイプが部品番号とシリアル番号の場合)
12. 最初のシリアル番号を収集 (選択したタイプが部品番号とシリアル番号で、シリアル番号が連続の場合)
13. 最初のシリアル番号を検証 (選択したタイプが部品番号とシリアル番号で、シリアル番号が連続の場合)
14. 在庫の全数量を収集 (選択したタイプが部品番号とシリアル番号、シリアル番号は連続の場合)
15. 連続したシリアル番号を自動的にデータファイルに書き込む (選択したタイプが部品番号とシリアル番号、シリアル番号は連続の場合)

在庫管理の作業フローは以下の方法で行われます:

1. ユーザは情報のタイプを選択します:
 - (a) 部品番号のみ
 - (b) 部品番号と数量
 - (c) 部品番号とシリアル番号
2. (a) を選択した場合、次のステップに進みます。 (b) を選択した場合、ステップ 7 に進みます。 (c) を選択した場合、ステップ 13 に進みます。
3. 部品番号(Part Number) 情報の収集
4. ユーザが“Exit” キー (CMD -> ESC) を押したら、ステップ 1 に進みます。
5. 部品番号情報を検証します。入力データが検証をパスしなかった場合、ステップ 3 へ進み、それ以外は次のステップへ進みます。
6. 数量(Quantity) フィールドとシリアル番号(Serial Number) の両方のフィールドに“0” をセットします。次の入力のためにステップ 3 へ進みます。
7. 部品番号(Part Number) 情報を収集します。
8. “Exit” キー (CMD -> ESC) が押されたら、ステップ 1 へ進みます。
9. 部品番号情報を検証します。入力データが検証をパスしなかった場合、ステップ 7 へ進みます。それ以外は、次のステップへ進みます。
10. 数量(Quantity)情報を収集します。

11. 数量(Quantity) 情報を検証します。入力データが検証をパスしなかった場合、ステップ 11 に進みます。それ以外は、次のステップへ進みます。
12. シリアル番号(Serial Number) フィールドに”0” を入れます。ステップ 7 に進みます。
13. 部品番号(Part Number) 情報を収集します。
14. “Exit” キー (CMD -> ESC) が押された場合、ステップ 1 に進みます。
15. 部品番号(Part Number) 情報を検証します。入力データが検証をパスしなかった場合、ステップ 13 に進みます。それ以外は、次のステップに進みます。
16. シリアル番号が連続か、連続でないかを選択します。
17. シリアル番号が連続の場合、次のステップに進みます。それ以外は、ステップ 25 に進みます。
18. 最初のシリアル番号を収集します。
19. シリアル番号(Serial Number) 情報を検証します。入力データが検証をパスしなかった場合、ステップ 18 に進みます。それ以外は、次のステップに進みます。
20. 連続する項目の番号を収集します。
21. 番号を検証します。入力データが検証をパスしなかった場合、ステップ 20 に進みます。それ以外は、次のステップに進みます。
22. カウンタをゼロにセットします。
23. 数量(Quantity) フィールドに”1” を入れ、データファイルに連続シリアル番号(Serial Number) を追加します。
24. カウンタに 1 を加えます。カウンタは連続する項目番号に等しい場合、ステップ 13 に進みます。それ以外はステップ 23 に進みます。
25. シリアル番号を収集します。
26. シリアル番号情報(Serial Number) 情報を検証します。入力データが検証をパスしなかった場合、ステップ 25 に進みます。それ以外は次のステップに進みます。
27. 数量(Quantity) フィールドに “1” を入れます。ステップ 13 に進みます。

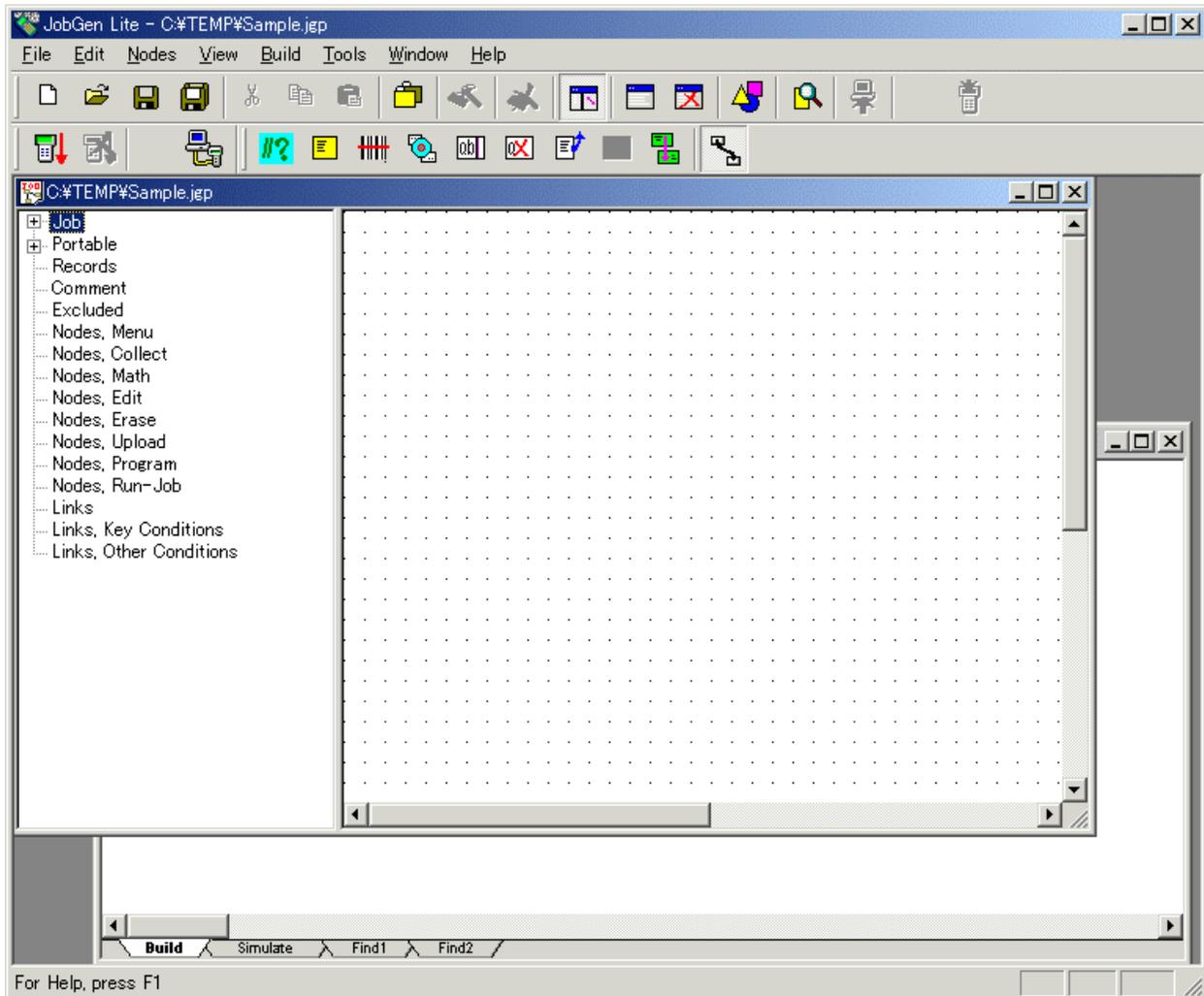
在庫管理の作成

複雑なので、このジョブの作成は作業フローに沿って進みます。各ステップは作業フローのプロセスとプロセスを実現するのに必要な JobGen Plus で対応する手順を示します。

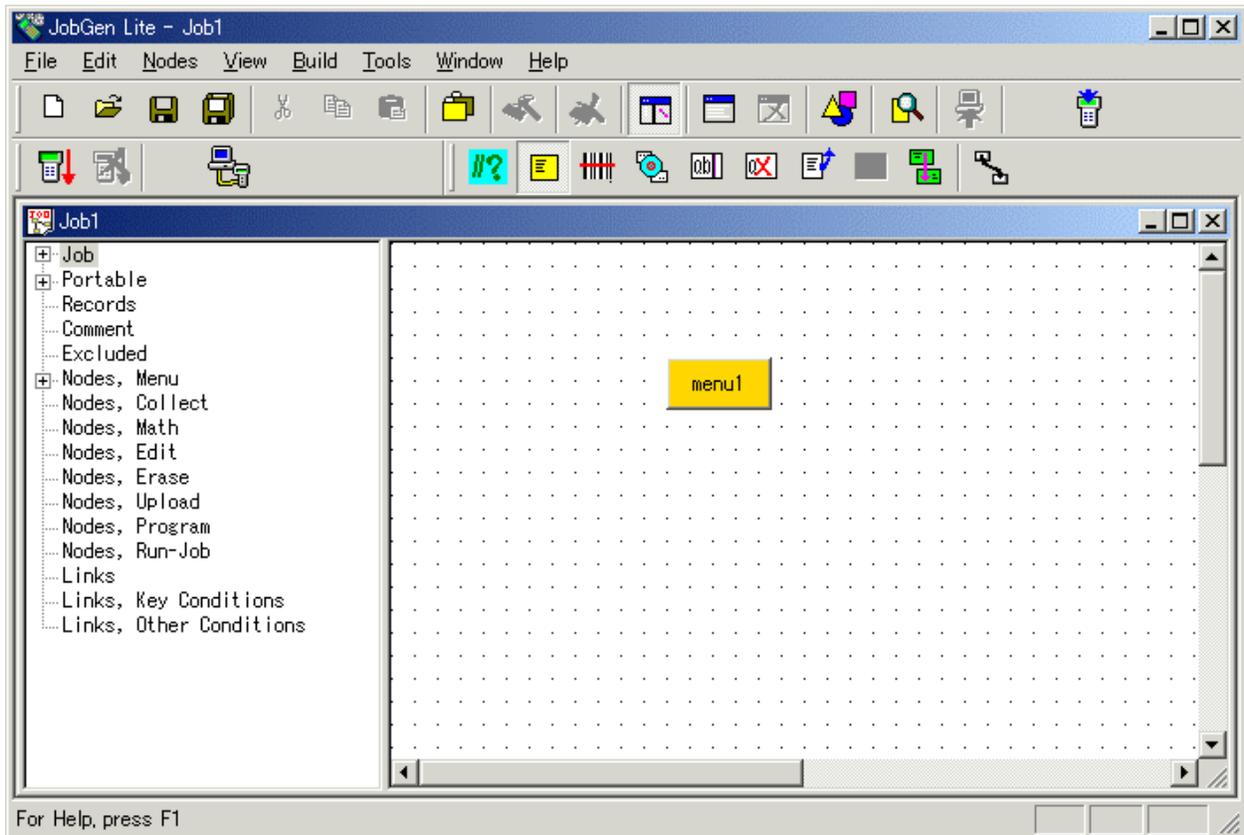
作業フロー

1. ユーザは情報のタイプを選択します:
 - (a) 部品番号のみ
 - (b) 部品番号と数量
 - (c) 部品番号とシリアル番号

JobGen Plus



1. マウスポインタを Menu ノード ボタンに移動します。
2. マウスポインタが Menu ノード ボタンにある時に左マウスボタンをクリックします。
3. マウスポインタを Menu ノード が置かれる場所に移動します。左マウスボタンをダブルクリックします。



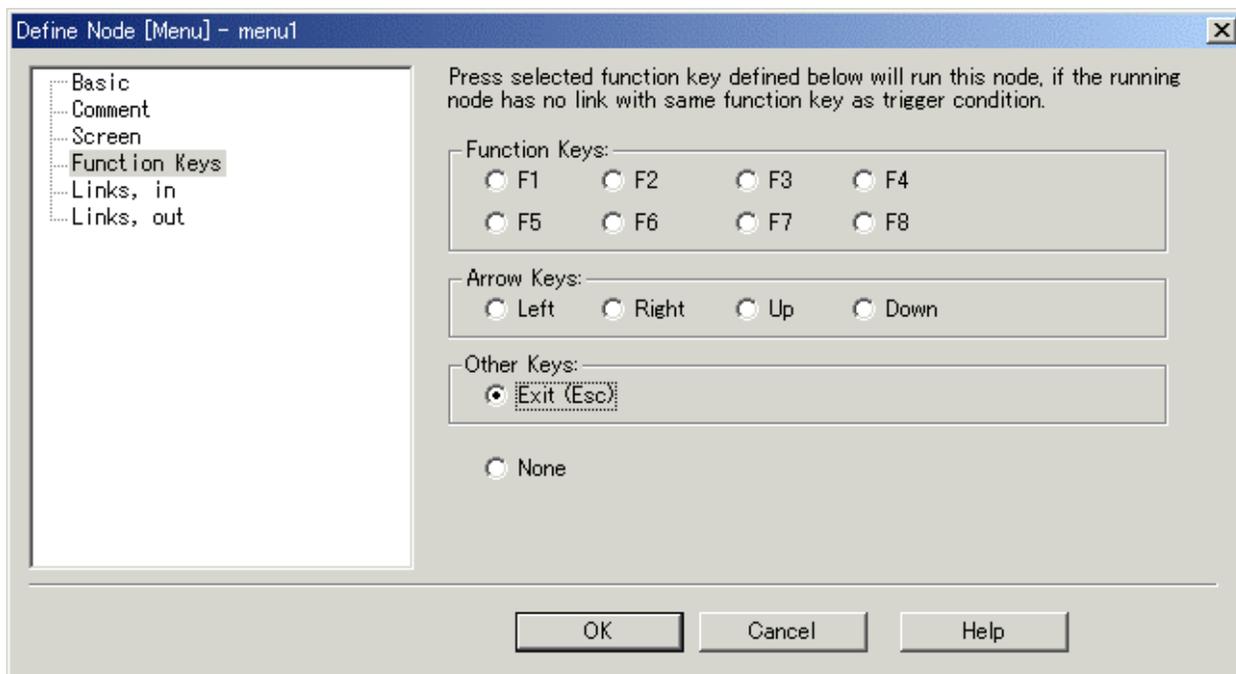
4. 新しい Menu ノード が作成されます。マウスポインタが Menu ノード がある時に左マウスボタンをダブルクリックします。“Define Menu” ウィンドウがスクリーンに現れます。

ヒント: 日本語メニューを作成する場合は、上記の作業をする前に Edit メニューから Define Portable を選択し、Setting > Screen > Large Font をチェックして下さい。



5. 以下の情報を入力します:

- (a) Basic プロパティで、名前を “mainMenu” に変更します。
- (b) Screen プロパティで、ポータブルターミナルのスクリーンに表示する内容をタイプします。
- (c) Basic プロパティで、**Start Node** オプションをチェックします。
- (d) Function Keys プロパティで、“Exit” をチェックします。”Exit”キー (CMD -> ESC) が押された場合、**mainMenu** ノード が処理されます。



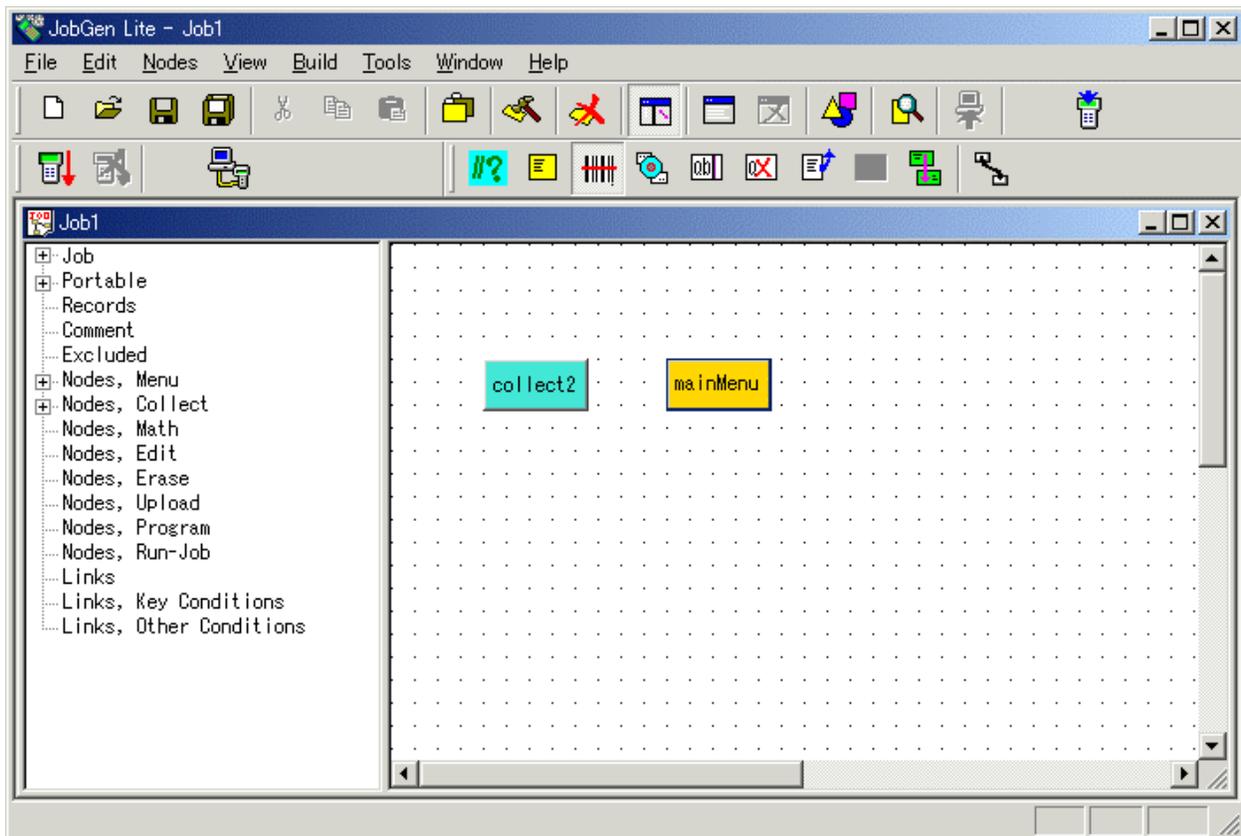
6. **OK** をクリックします。

作業フロー

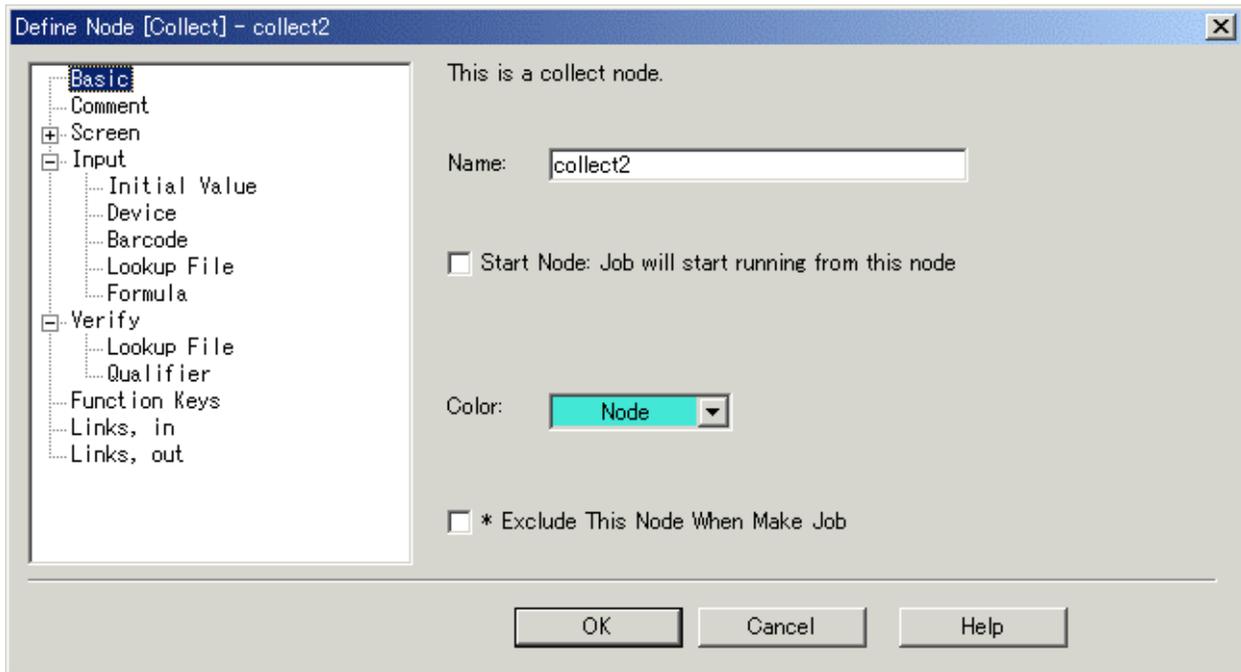
2. (a) を選択した場合、ステップ 3 に進みます。 (b) を選択した場合、ステップ 7 に進みます。 (c) を選択した場合、ステップ 13 に進みます。
3. 部品番号 (Part Number) 情報を収集します。
4. “Exit “ (CMD > ESC) キーが押されたら、ステップ 1 に進みます。
5. 部品番号 (Part Number) 情報を検証します。入力データが検証をパスしない場合、ステップ 3 に進みます。それ以外は、ステップ 6 に進みます。
6. 数量(Quantity) フィールドとシリアル番号(Serial Number) フィールドの両方に”0”を入れます。次の入力のためにステップ 3 に進みます。

JobGen Plus

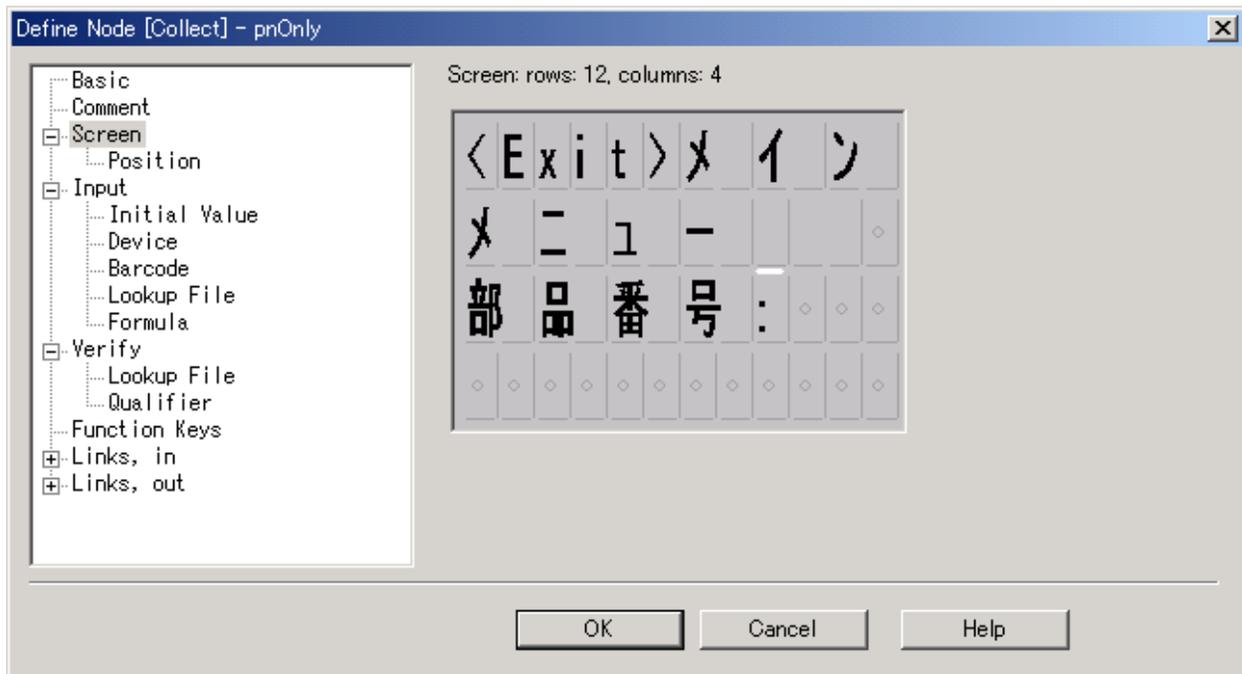
1. Collect(収集)ノードボタンにマウスポインタを移動することによって、部品番号のための収集ノードを作成します。マウスポインタが Collect(収集)ノードボタンにある時に左マウスボタンをクリックします。
2. Collect(収集)ノードが置かれる場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



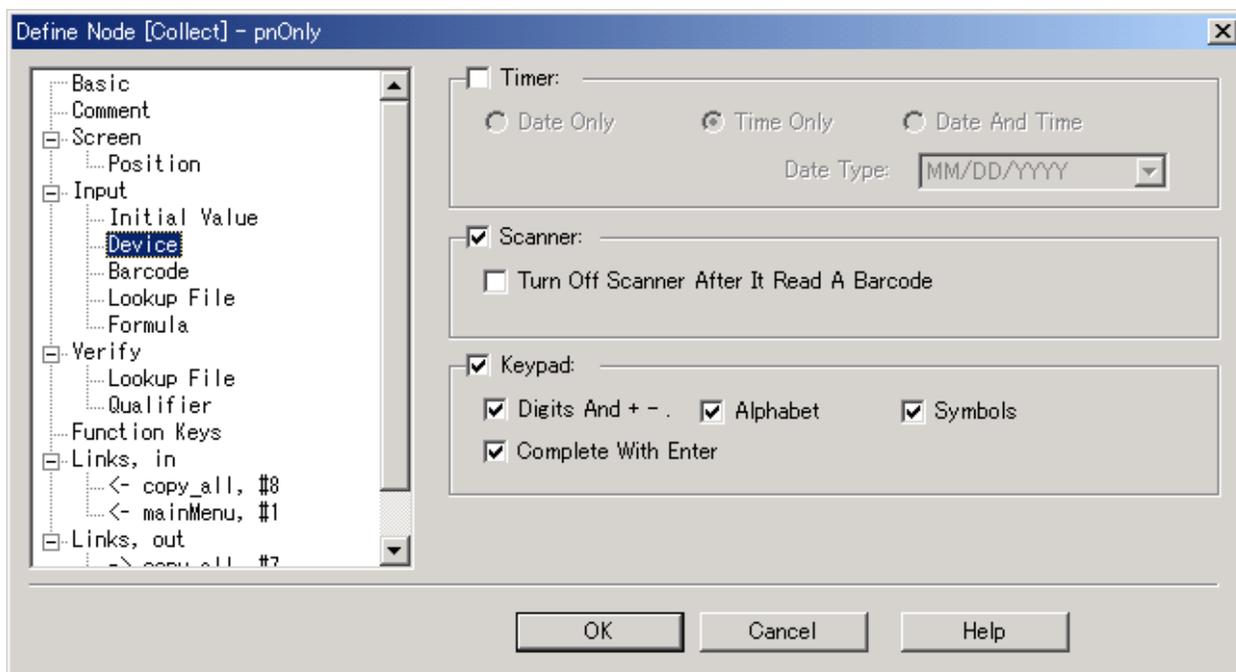
3. 新しい収集(collect) ノードが作成されました。マウスポインタが収集(Collect)ノードにある時に左マウスボタンをダブルクリックします。“Define Collect” ウィンドウがスクリーンに現れます。



4. 以下のオプションを変更します。
- (a) Basic プロパティで、名前を “pnOnly” に変更します。
 - (b) **Screen** プロパティをクリックします。

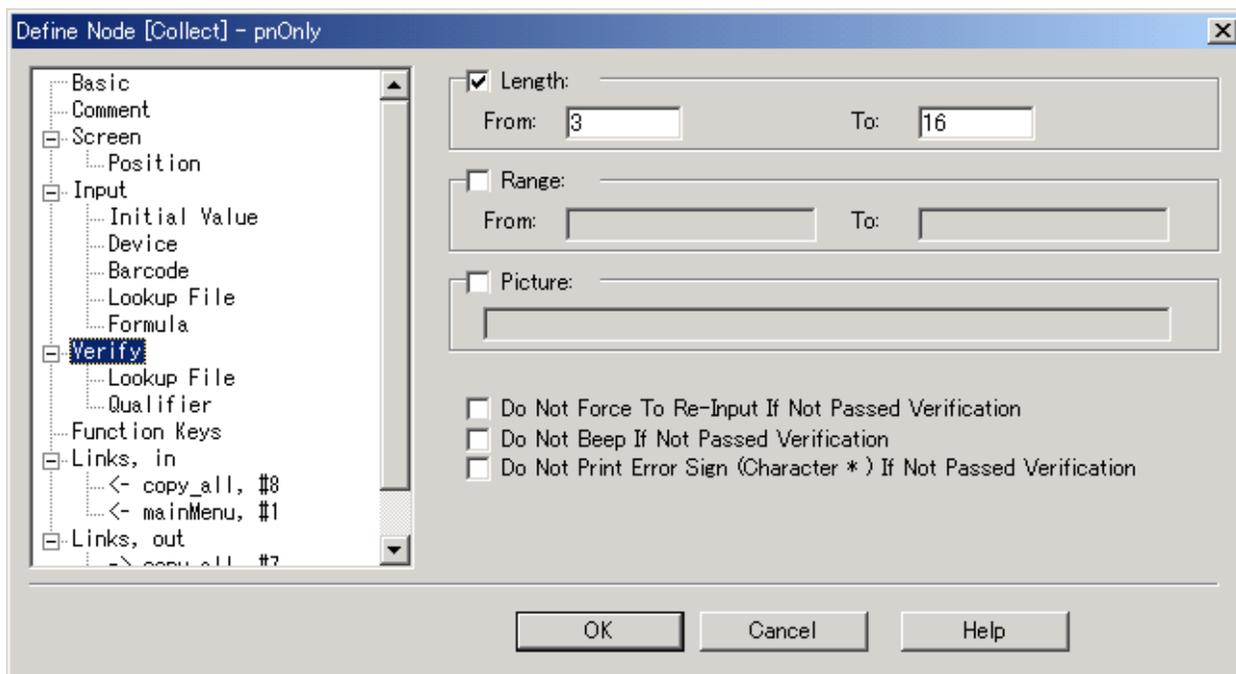


- (c) スクリーンにプロンプト(入力要求)をタイプします。
- (d) **Input > Device** プロパティをクリックします。



(e) **Keypad** と **Scanner** の両方のオプションが選択していることを確認して下さい。

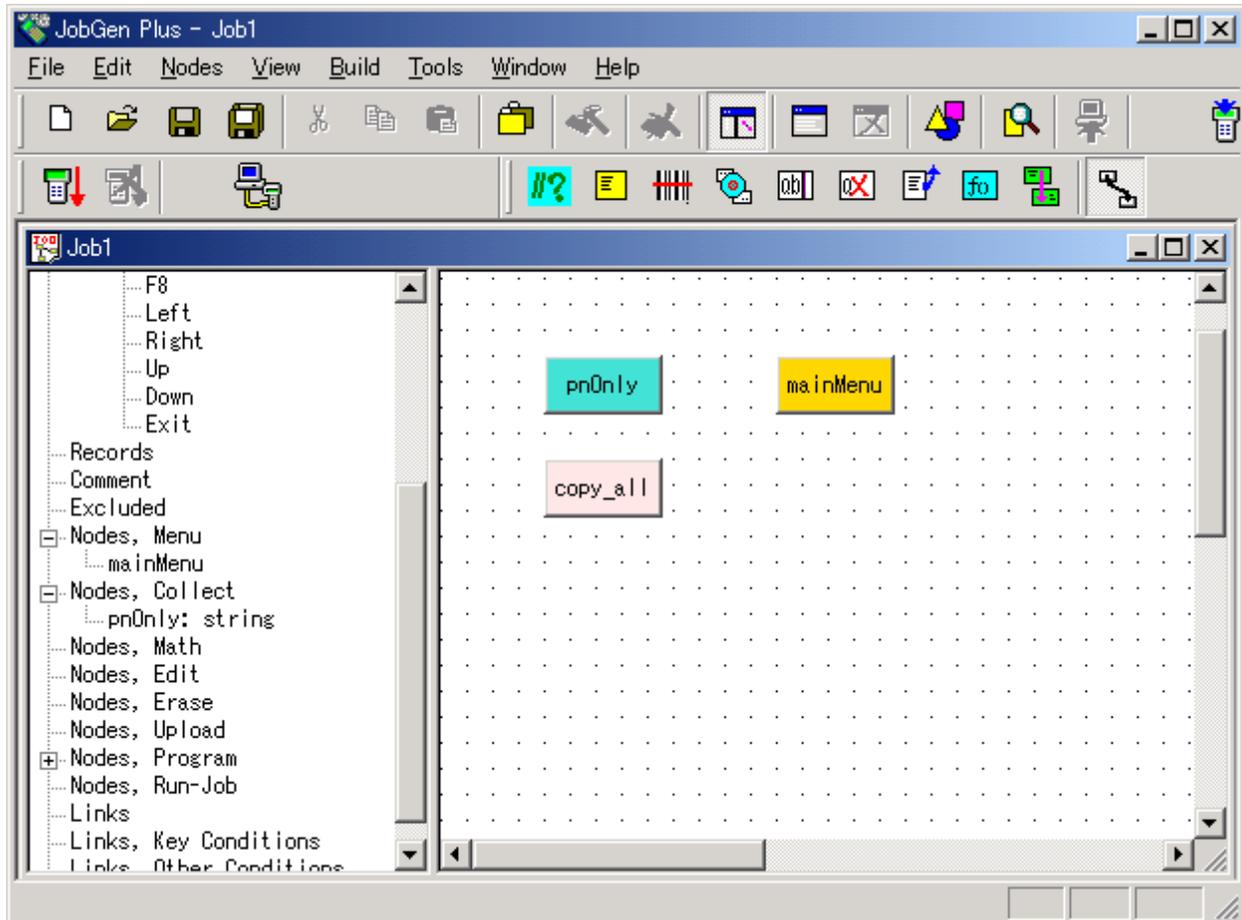
(f) **Verify** プロパティをクリックして下さい。



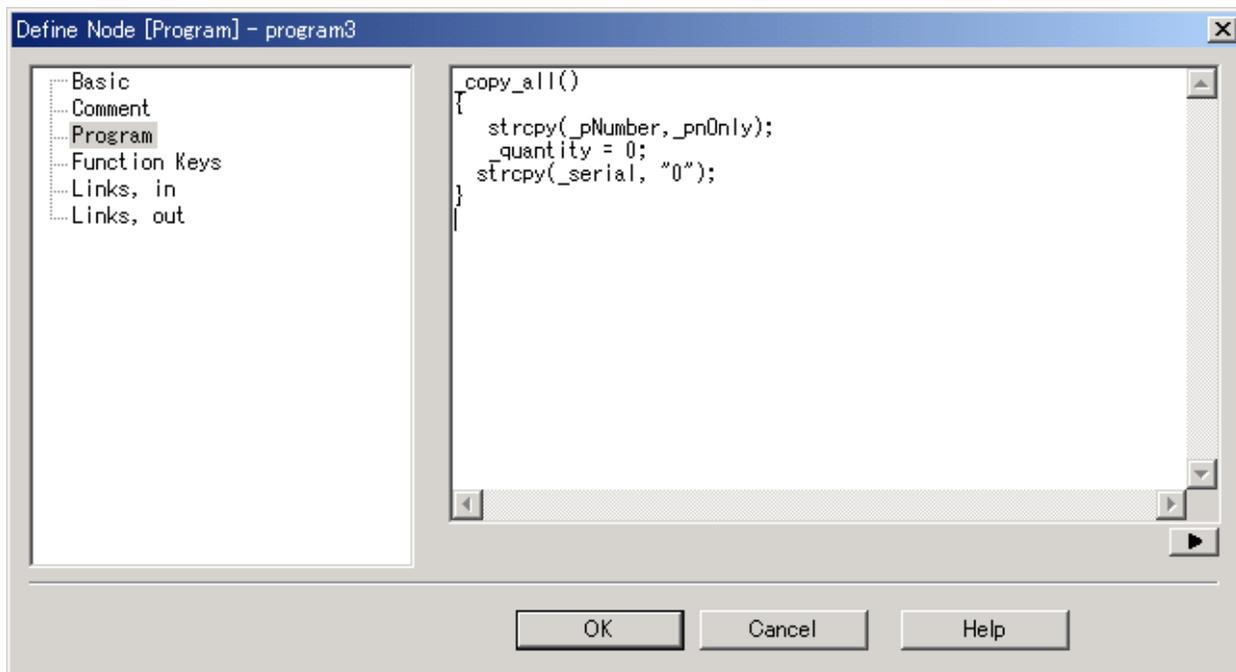
(g) **Length** オプションをチェックして、最小値 : 3 を From に、そして最大値 16 を To: にセットして下さい。

(j) **OK** をクリックして下さい。

5. Program ノード ボタンにマウスポインタを移動して、ポインタが Program ノードボタンにある間に左マウスボタンをクリックして下さい。
6. Program ノード を置く場所にマウスポインタを移動して下さい。左マウスボタンをダブルクリックして下さい。



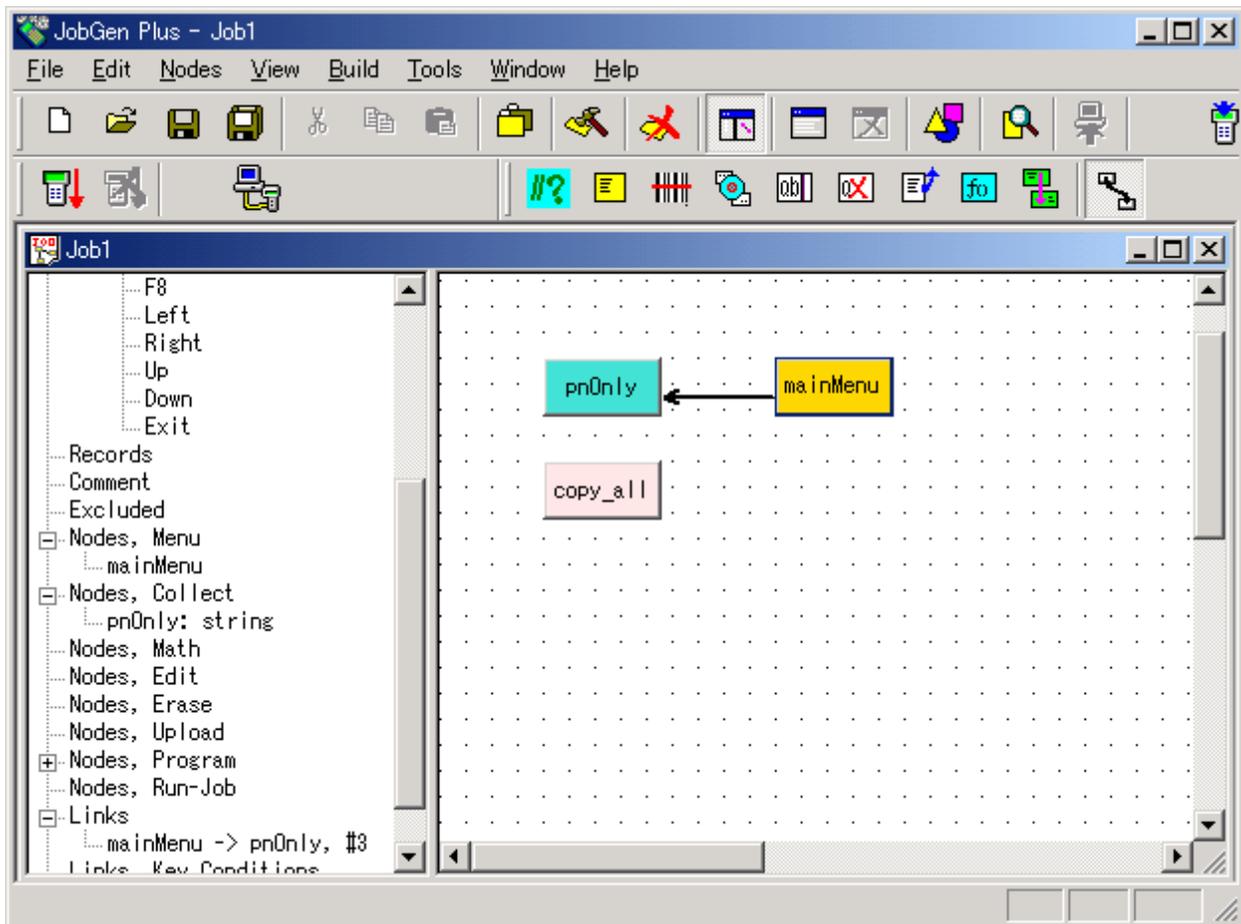
7. 新しく作成した Program ノード にマウスポインタがある間に左マウスボタンをダブルクリックして下さい。“Define Node[Program]” ウィンドウがスクリーンに現れます。



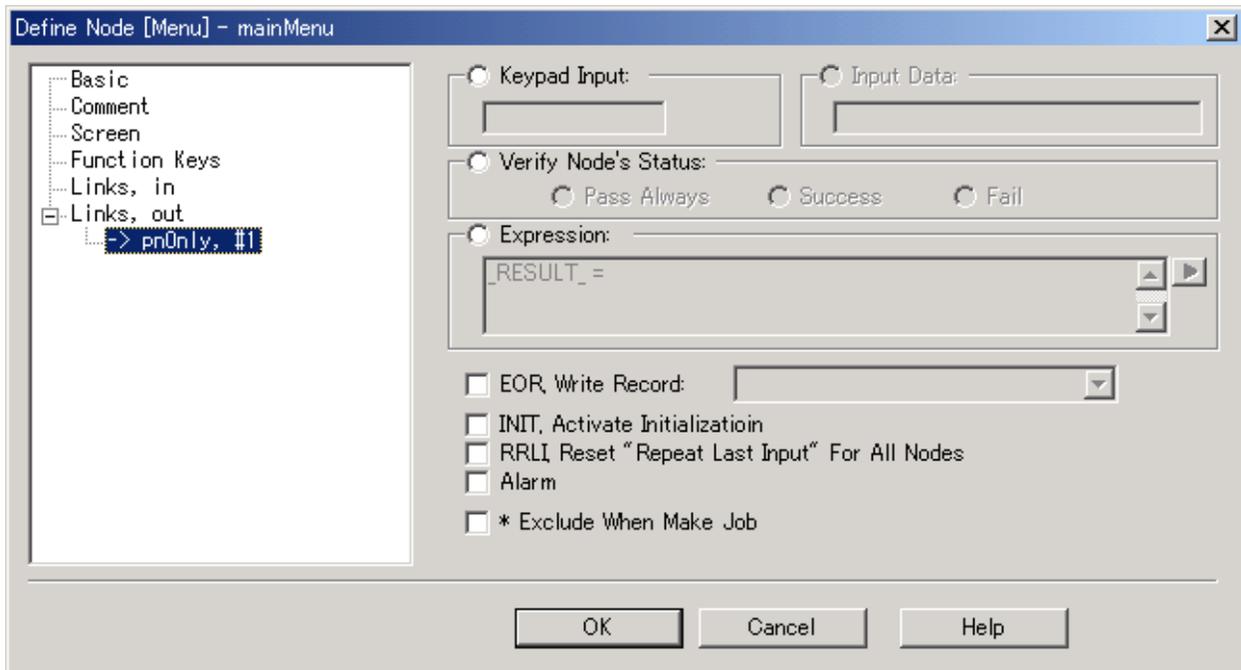
- この Program ノード は主にデータフィールドに情報を埋めるためのものです。このジョブには **pNumber**、**quantity**(数量) と **serial**(シリアル)の三つのデータフィールドがあります。Collect ノード **pnOnly** は **pNumber** についての情報を一時的に保存します。したがって、この Program ノード は **pnOnly** の情報を **pNumber** にコピーし、そして他の二つのフィールドを“0”で埋めます。この Program ノードを定義するために、ユーザはノードの名前を“copy_all”に変更する必要があります。そして、以下のCソースコードをタイプします:

```
_copy_all()
{
    strcpy(_pNumber, _pnOnly);
    _quantity = 0;
    strcpy(_serial, “0”);
}
```

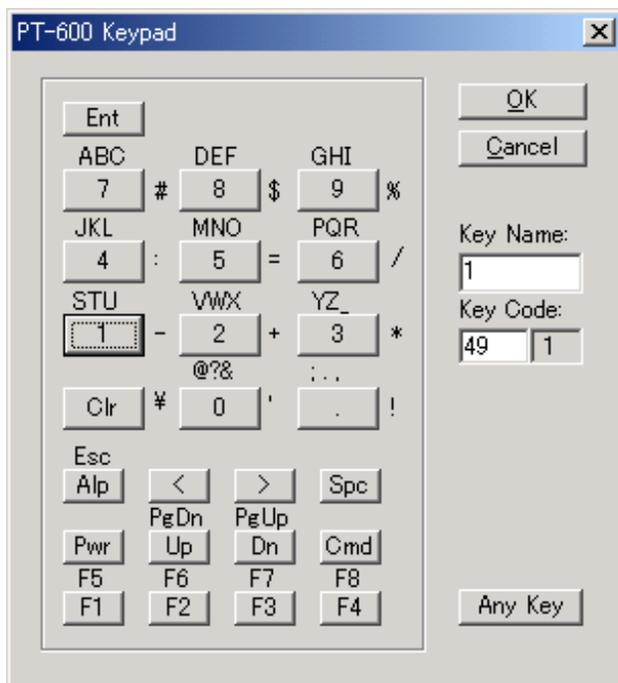
- これらのノード間のリンクをここで作成する必要があります。マウスポインタをリンクボタンに移動し、マウスポインタがリンクボタン上にある間に左マウスボタンをクリックします。
- マウスポインタを **mainMenu** ノードに移動し、左マウスボタンをクリックしたまま保持します。マウスポインタを **pnOnly** ノードに移動し、左マウスボタンを離します。



11. マウスポインタが新しく作成したリンクにある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。

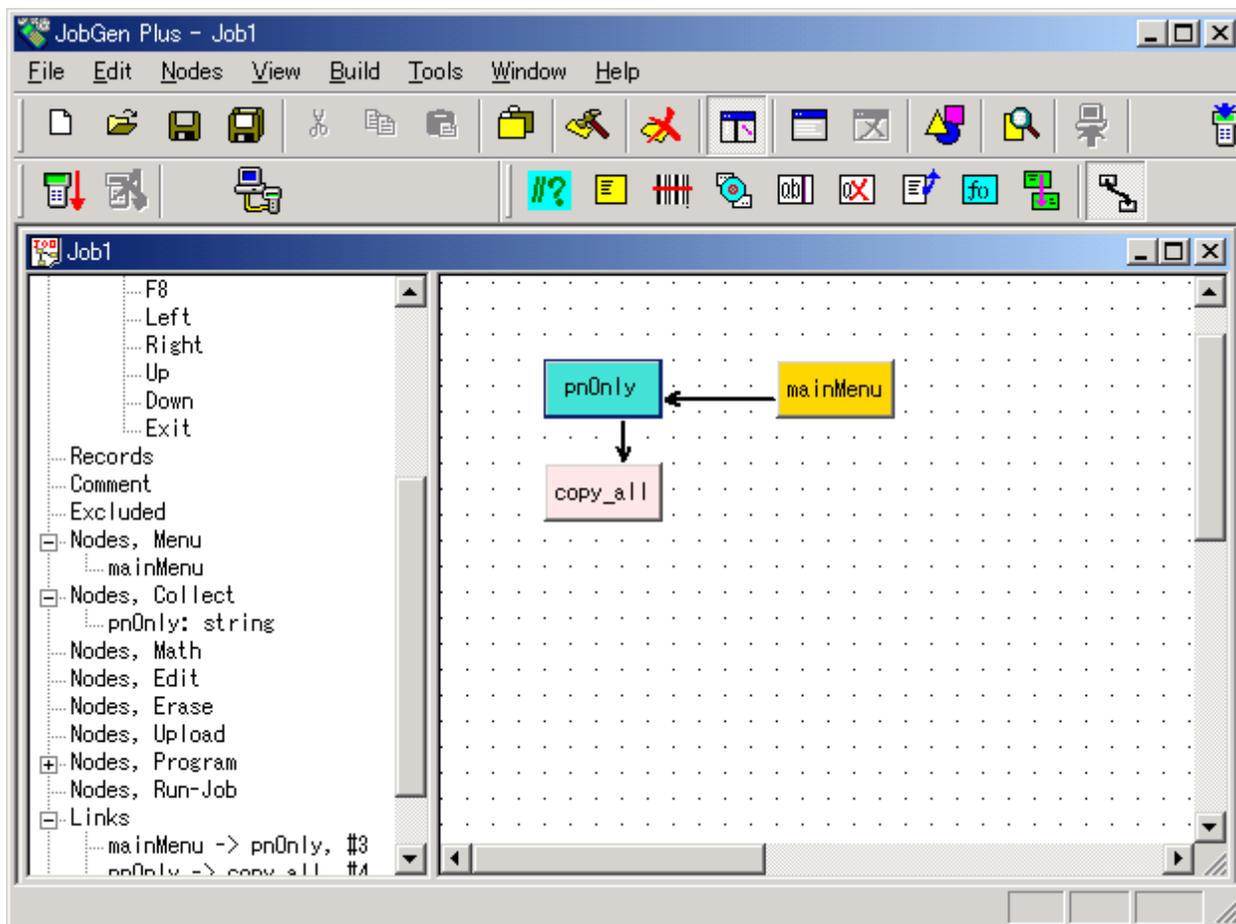


12. **Keypad Input** オプションをチェックし、そして **Keypad Input** オプションの下のスペースにマウスポインタがある間に左マウスボタンをクリックして下さい。“PT-600 Keypad” ウィンドウがスクリーンに現れます。
13. **1** ボタンをクリックします。これは”1”を押すと **mainMenu** ノード から **pnOnly** ノード にプロセスが移動することを示しています。“PT-600 Keypad” ウィンドウを閉じるために **OK** ボタンをクリックします。

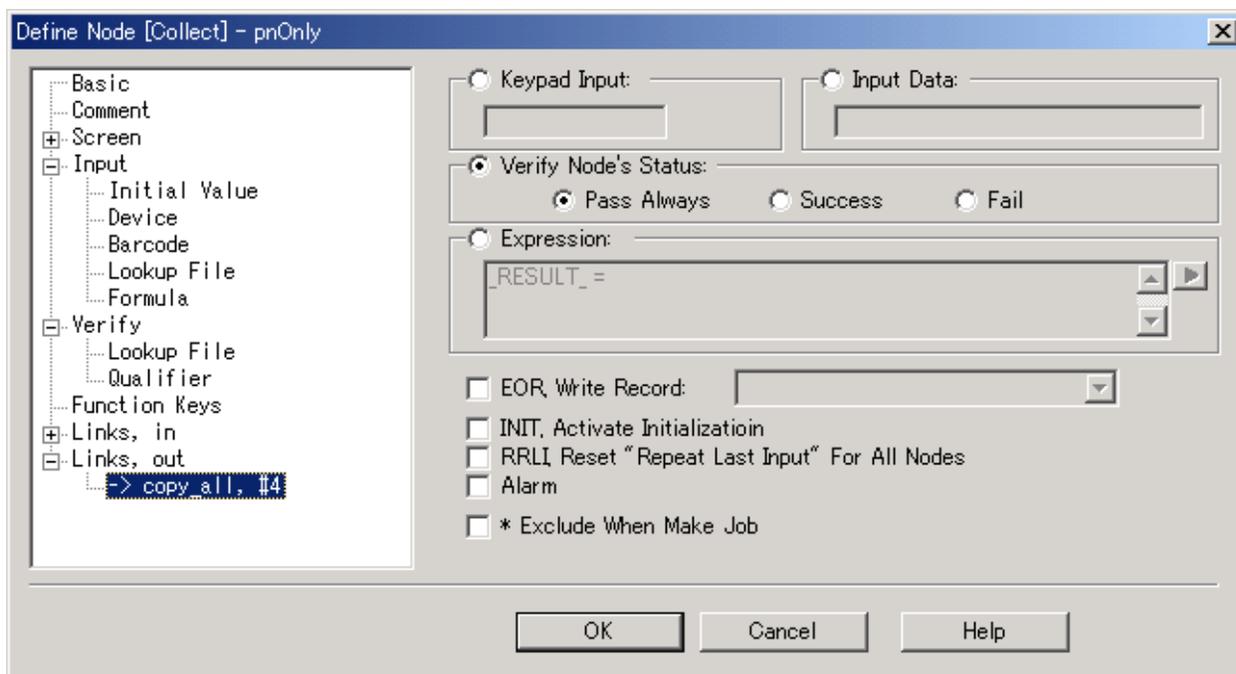


OK をクリックして下さい・

14. マウスポインタを **pnOnly** ノード に移動して、そして左マウスボタンをクリックしたままにします。マウスポインタを **copy_all** ノード に移動し、左マウスボタンを離します。



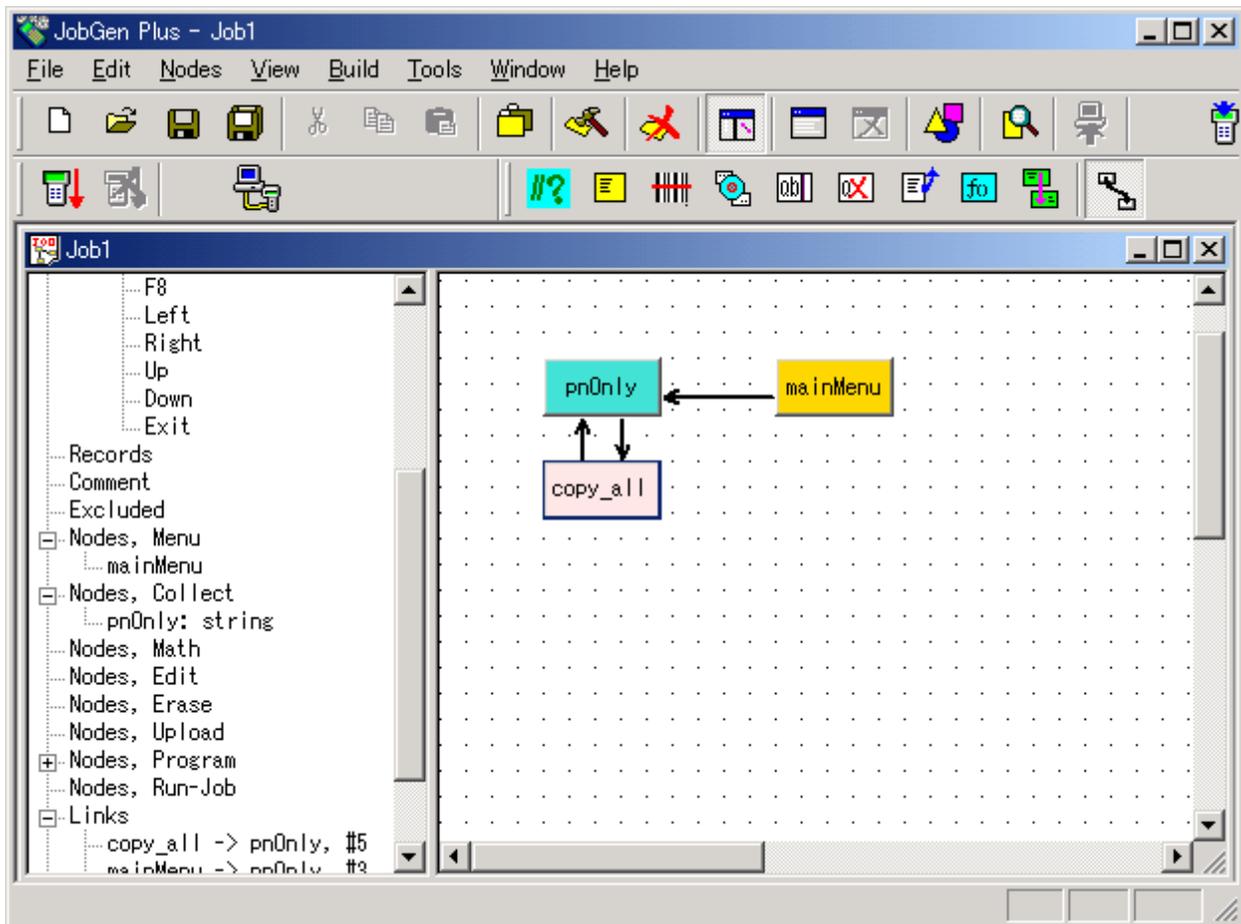
15. 新しく作成したリンクにマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



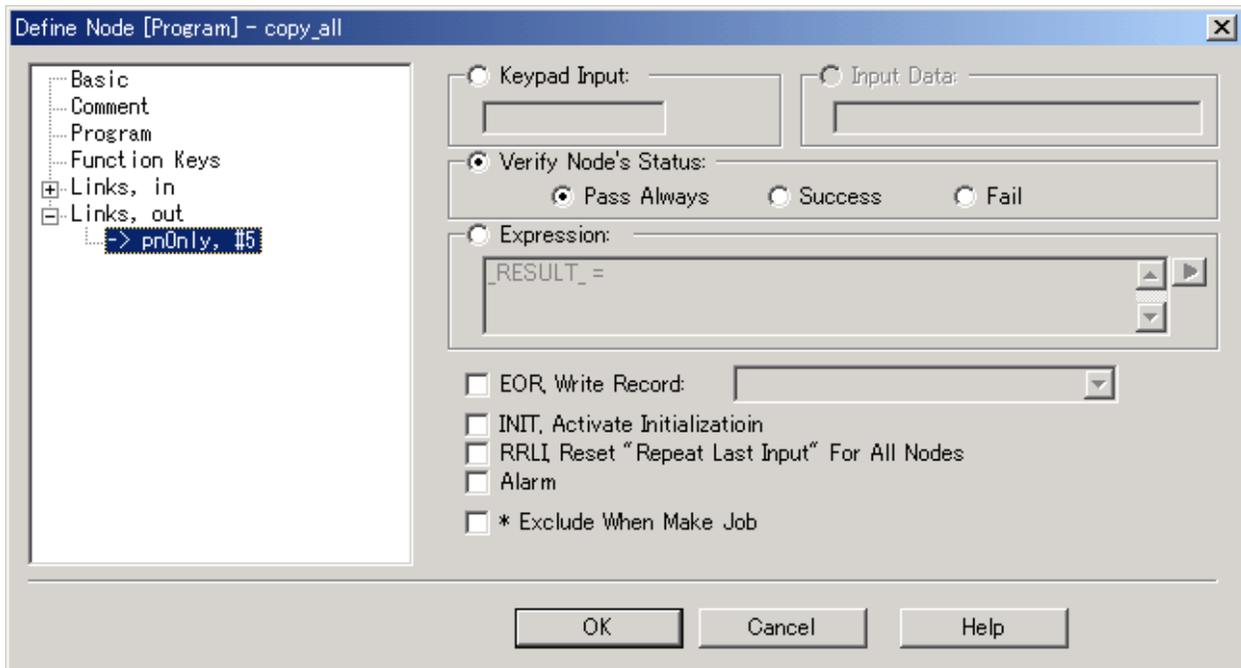
Pass Always オプションは ジョブが **pnOnly** ノードのプロセスを終えたら、**copy-all** ノード を処理することを示しています。

16. **OK** をクリックして下さい。

17. マウスポインタを **copy_all** ノード に移動し、そして左マウスボタンをクリックしたままにします。マウスポインタを **pnOnly** ノード に移動し、マウスボタンを離します。



18. マウスポインタが新しく作成したリンク 上にある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



Pass Always オプションはジョブが **copy_all** ノードの処理を終えたら、その後 **pnOnly** の処理を行うことを示しています。

19. **EOR** オプションをチェックして下さい。

21. **Ok** をクリックしてください。

作業フロー

7. Part Number 情報の収集。

8. “Exit” キーが押されたら、ステップ 1 へ進む。

9. Part Number 情報の検証。入力データが検証をパスしなかったら、ステップ 7 へ進む。
そうでなければ、ステップ 10 へ進む。

10. Quantity (数量)情報を収集。

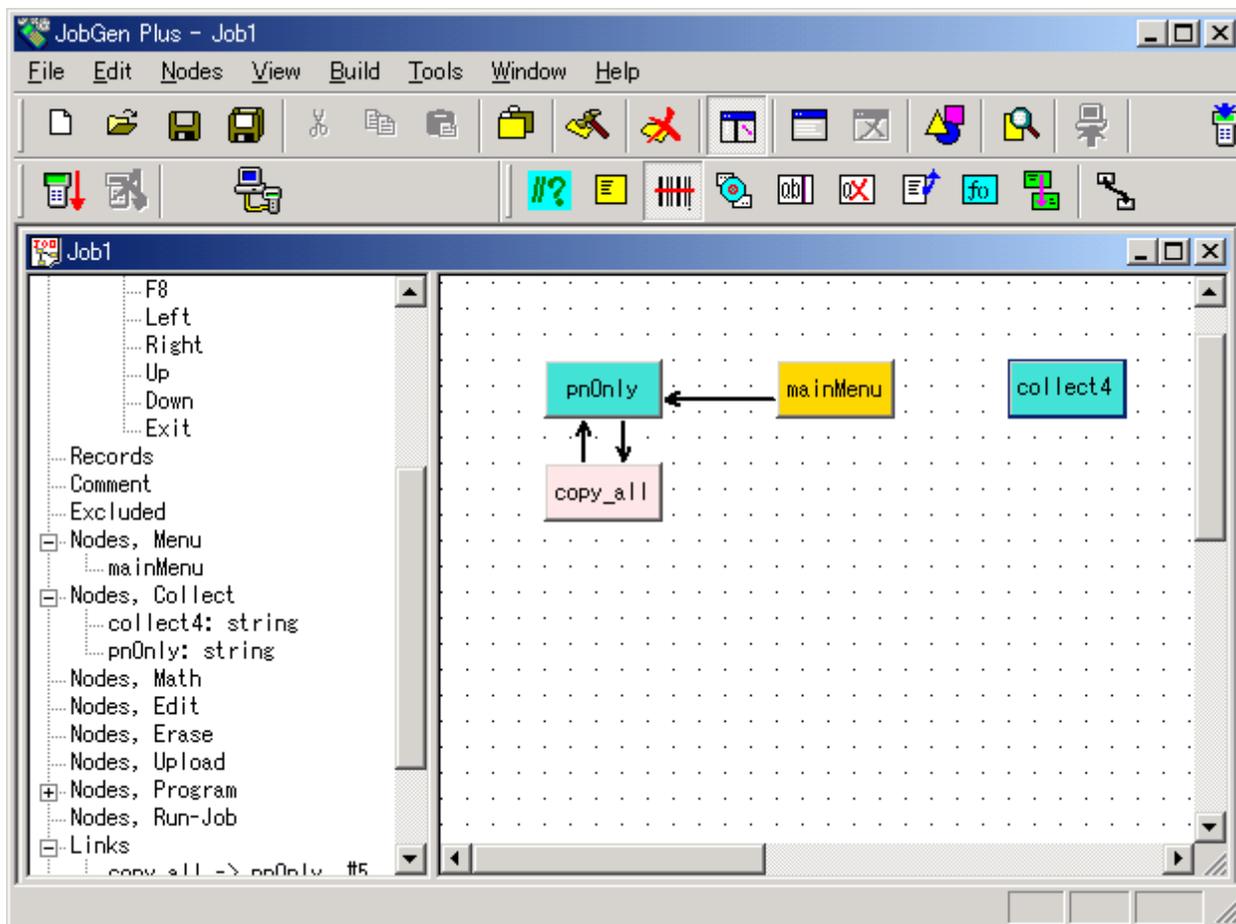
11. Quantity 情報を検証。入力データが検証をパスしなかった場合、ステップ 10 に進みます。
そうでなければ、ステップ 12 に進みます。

12. Serial Number フィールドに”0”を入れます。ステップ 7 に進みます。

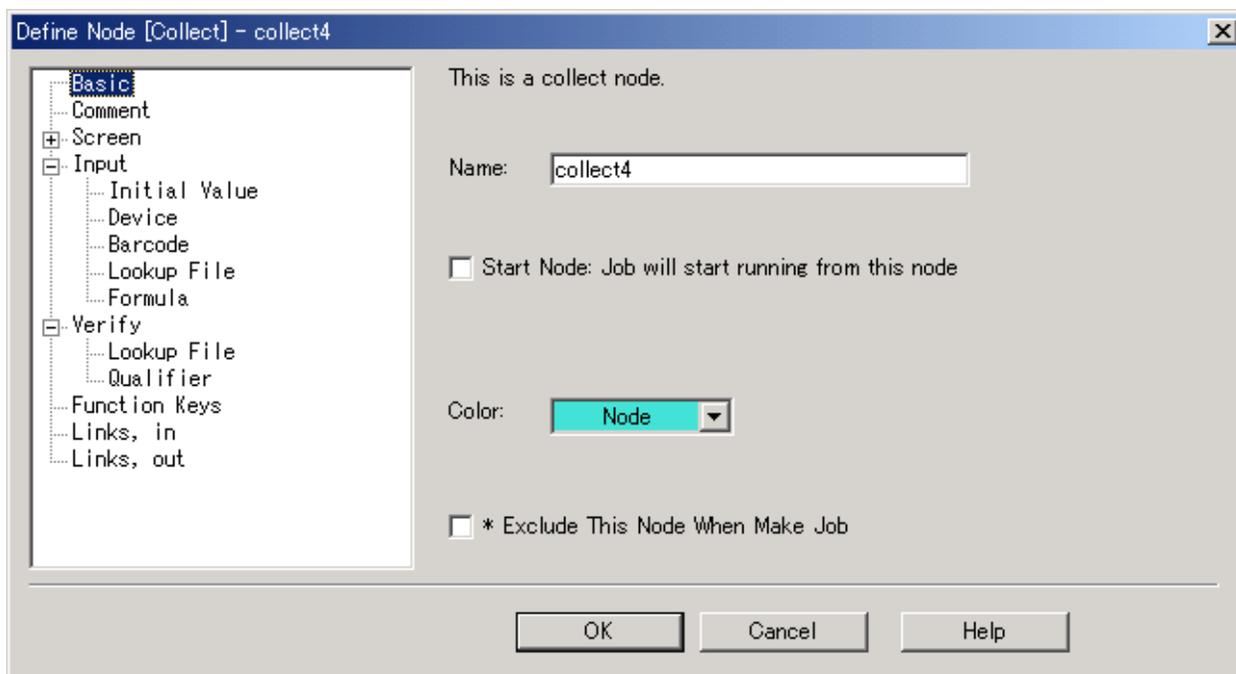
JobGen Plus

1. マウスポインタを Collect Node ボタンに移動することによって Part Number についての Collect ノードを作ります。マウスポインタが Collect ノード ボタンにある間に左マウスボタンをクリックします。

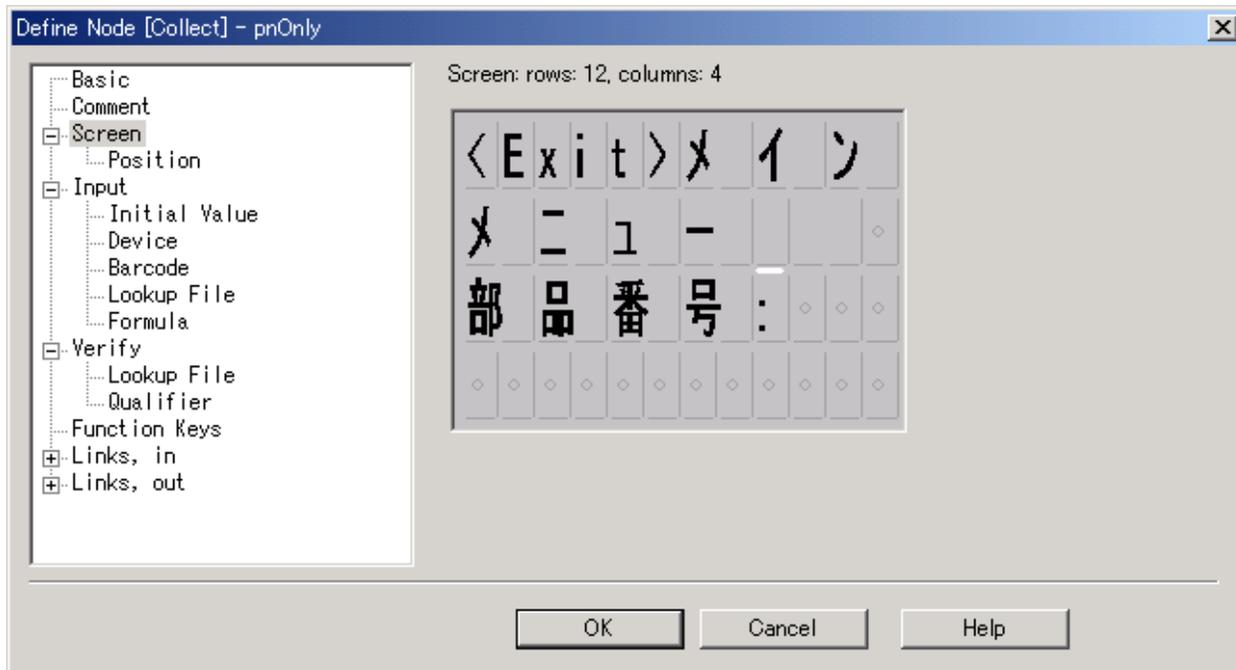
2. Collect ノード を置く場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



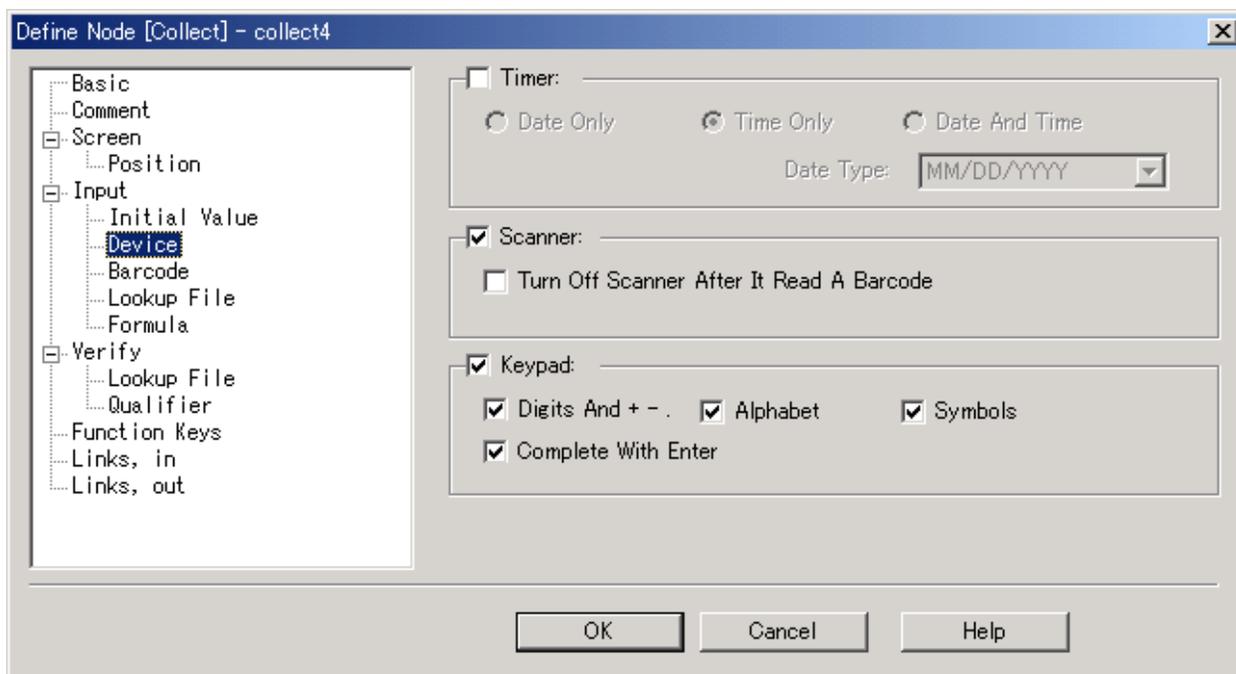
3. 新しい Collect ノード が作られました。マウスポインタが Collect ノード にある間に左マウスボタンをダブルクリックして下さい。“Define Node [Collect]” ウィンドウがスクリーンに現れます。



4. 以下のオプションを変更します。
 - (a) 名前を “pnQty” に変更します。
 - (b) **Screen** プロパティをクリックします。

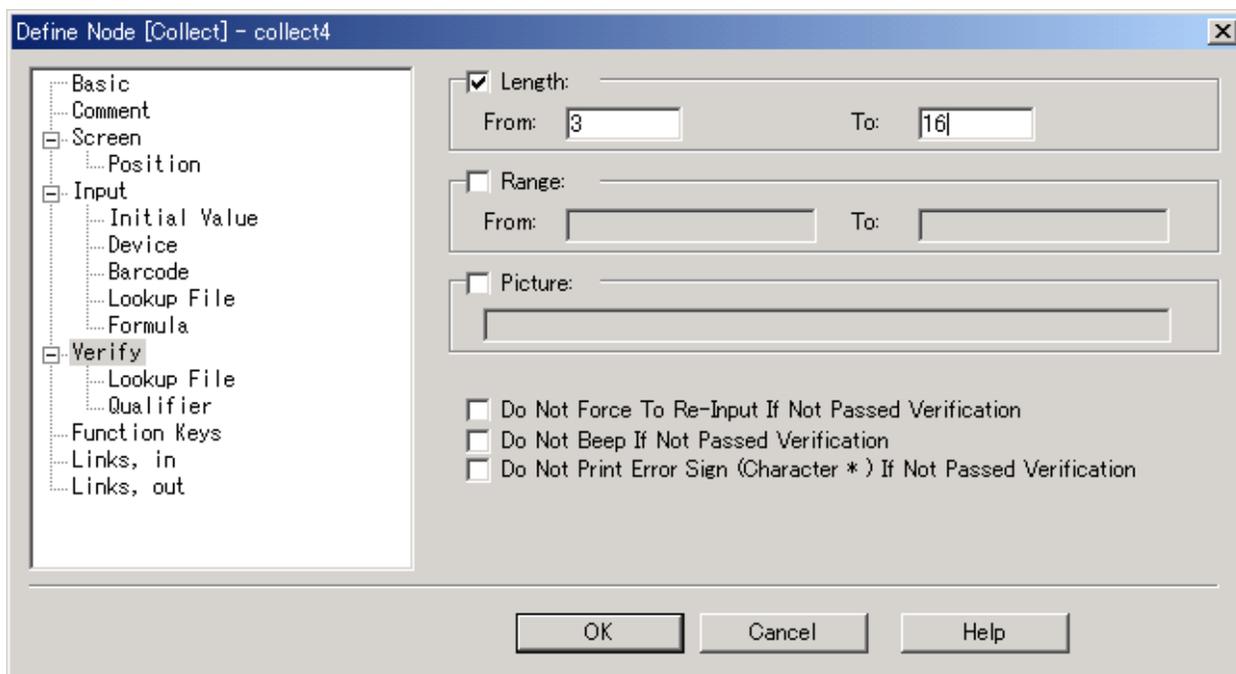


- (c) スクリーンのプロンプト(入力要求)をタイプします。
 - (d) **Input > Device** プロパティをクリックします。



(e) **Keypad** と **Scanner** オプションが選択されていることを確認して下さい。

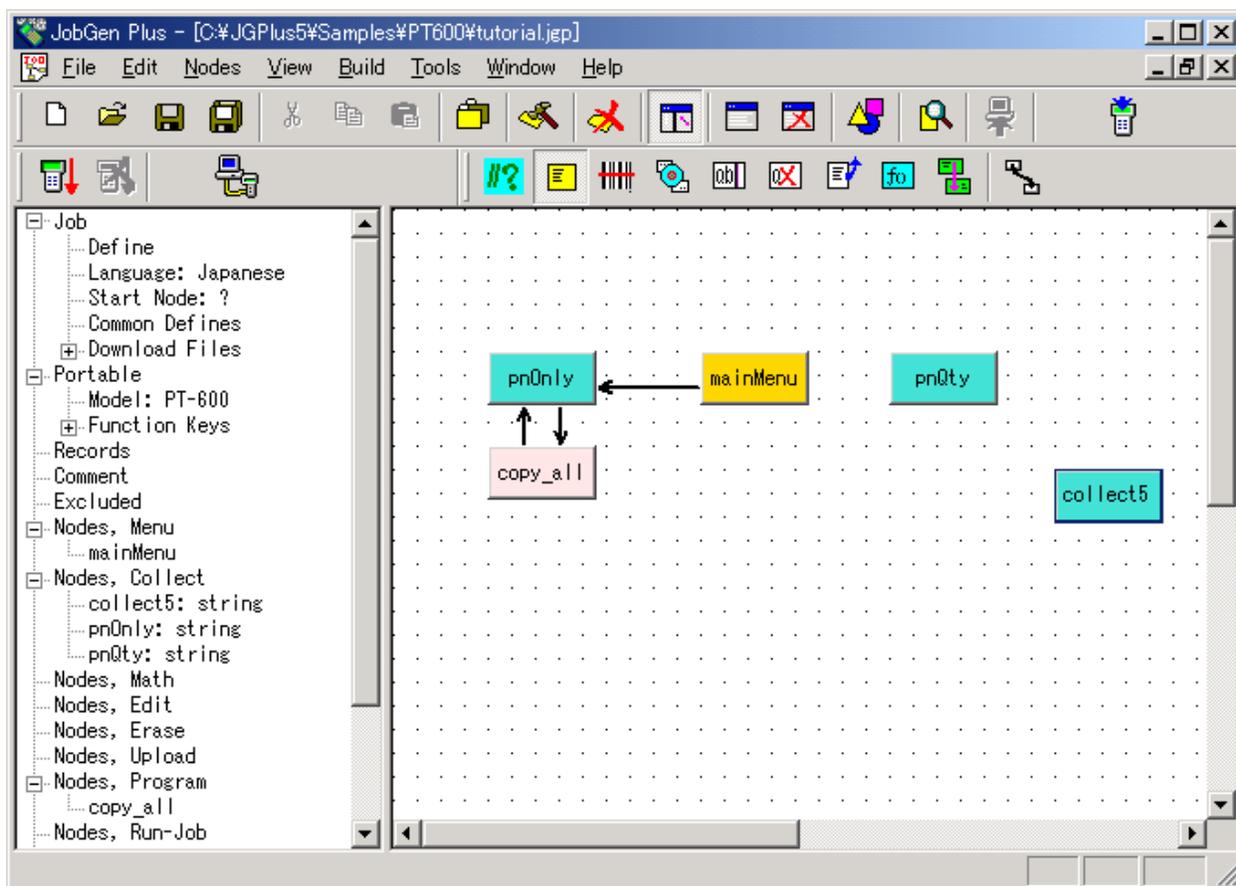
(f) **Verify** プロパティをクリックして下さい。



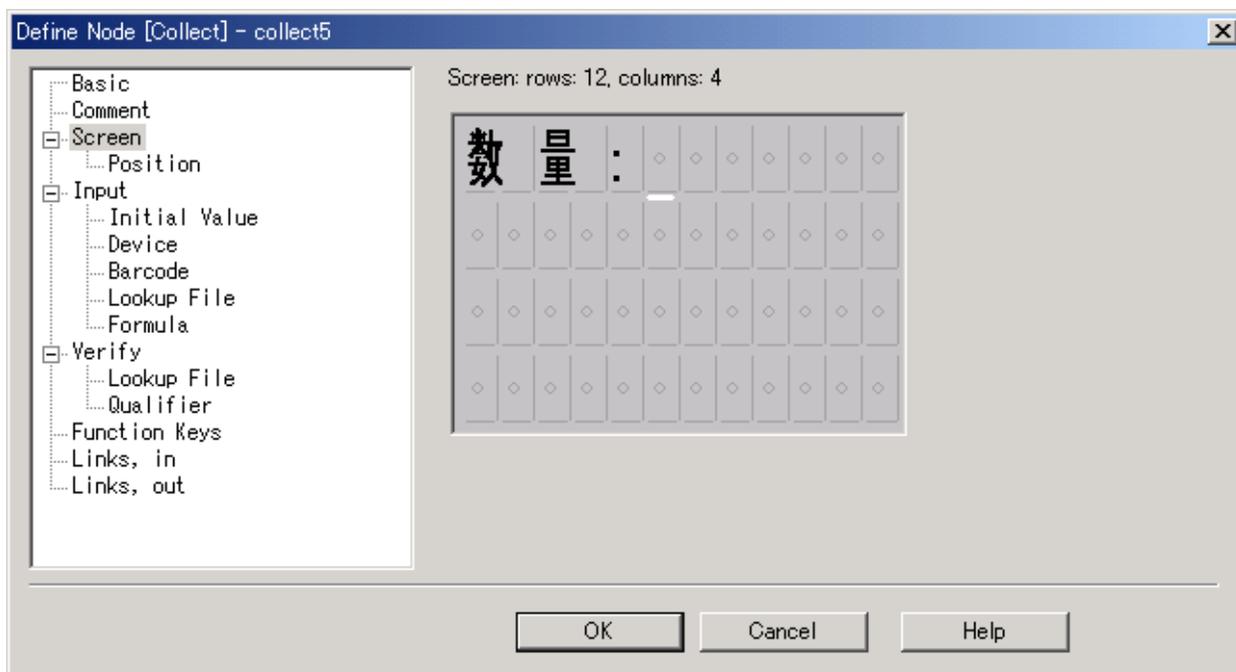
(g) **Length** オプションをチェックし、最小長 **3** を From に、最大長 **16** を To にを入れて下さい。

(j) **OK** をクリックして下さい。

5. Collect ノード が置かれる場所にマウスポインタを移動して下さい。左マウスボタンをダブルクリックして下さい。
6. マウスポインタが Collect Node にある間に左マウスボタンをダブルクリックして下さい。“Define Node [Collect]” ウィンドウがスクリーンに現れます。

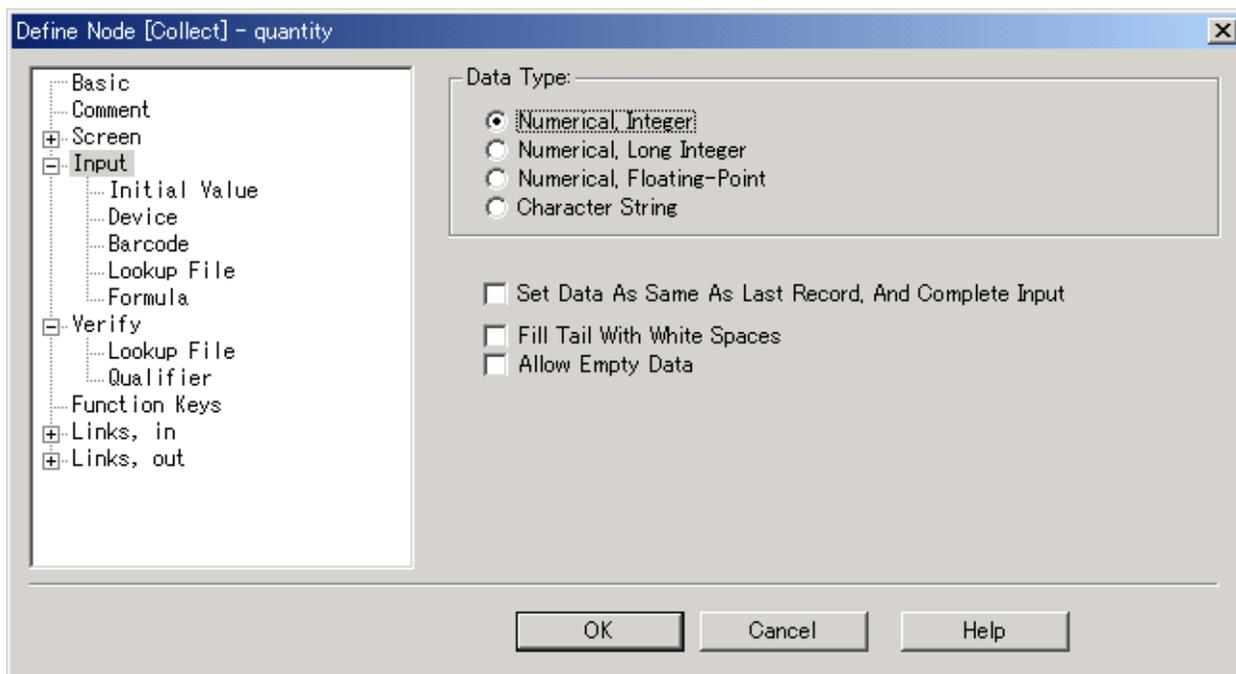


7. 以下のオプションを変更します。
 - (a) 名前を “quantity” に変更します。
 - (b) **Screen** プロパティをクリックします。



(c) プロンプト(入力要求)をタイプします。

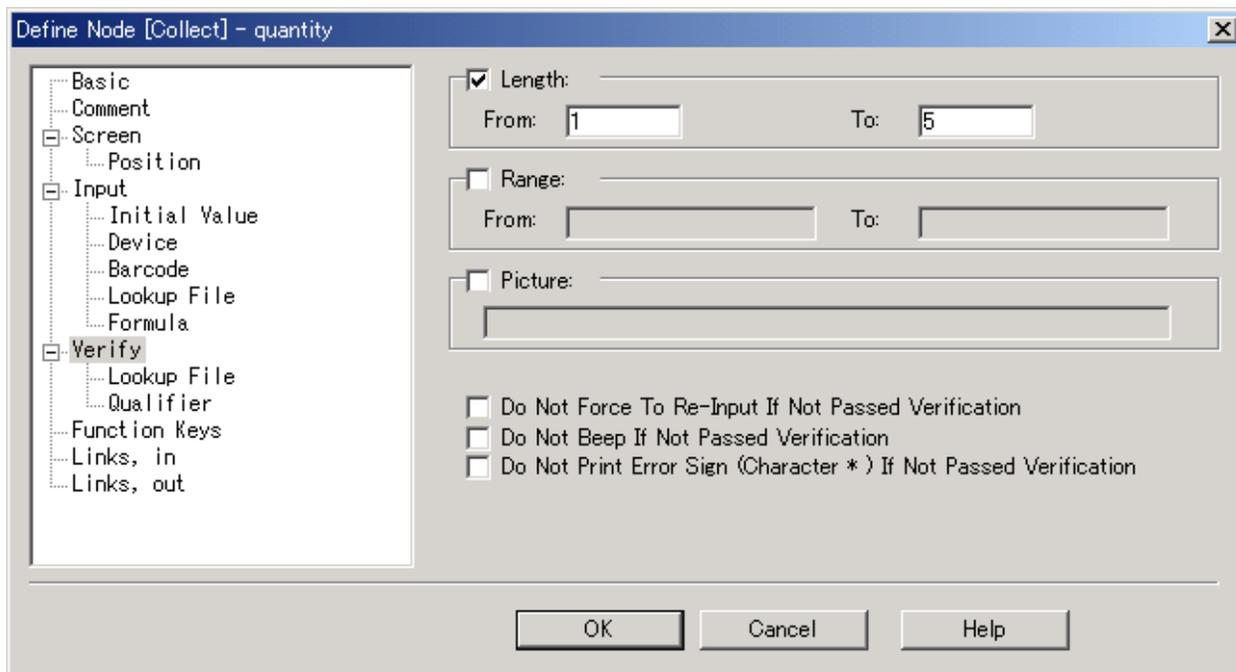
(d) **Input** プロパティをクリックします。Numerical Integer をチェックします。



(e) **Input > Device** プロパティをクリックします。

(f) **Keypad** と **Scanner** オプションの両方が選択されていることを確認して下さい。

(g) **Verify** プロパティをクリックして下さい。



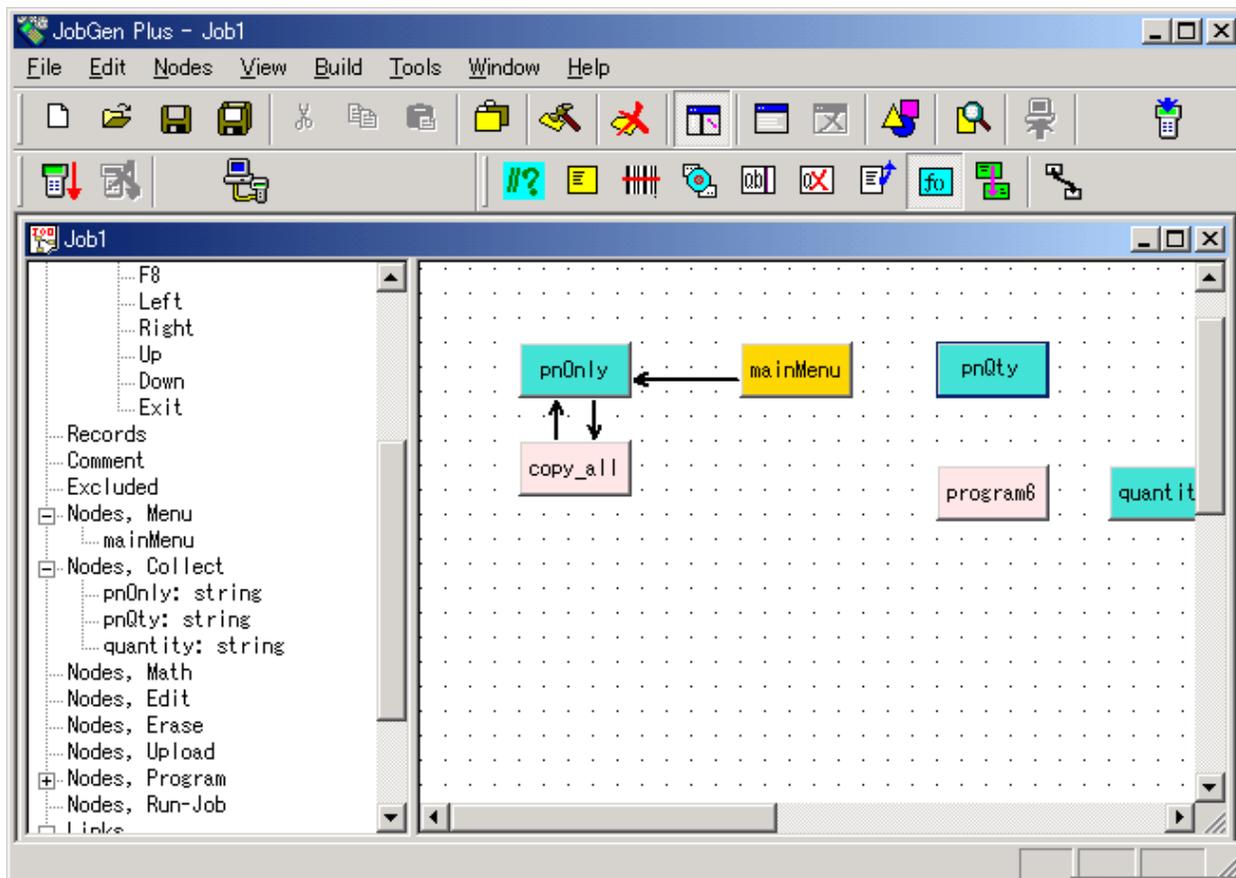
(h) **Length** オプションをチェックして、最少長 **1** を From に、そして最大長 **5** を To: に入れます。

(i) **Range** オプションをチェックし、最低限の値: **0** を From: に、そして最大の制限値 **32000** を To: にタイプします。

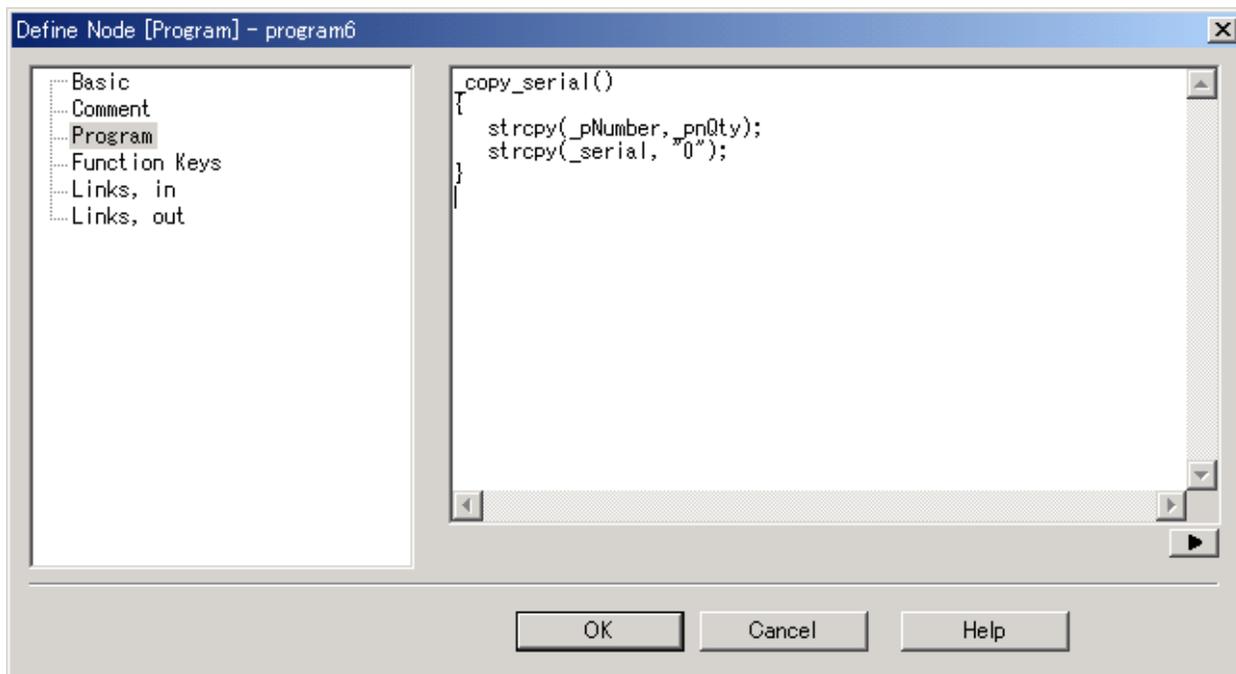
(k) **OK** をクリックします。

8. マウスポインタを Program ノード ボタンに移動します。ポインタが Program ノード ボタンにある間に左マウスボタンをクリックします。

9. Program ノード がおかれる場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



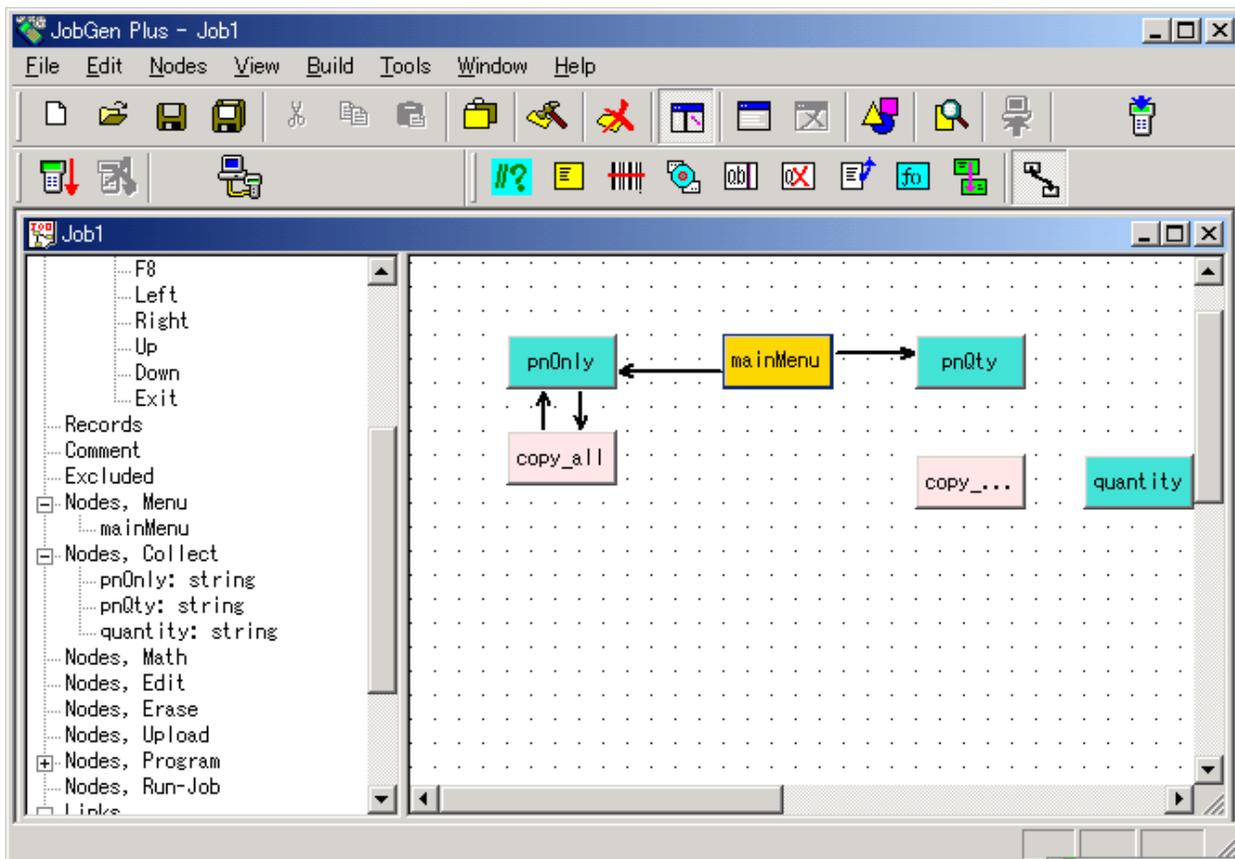
10.マウスポインタが新しく作成した Program ノード にある間に左マウスボタンをダブルクリックします。“Define Node [Program]” ウィンドウがスクリーンに現れます。



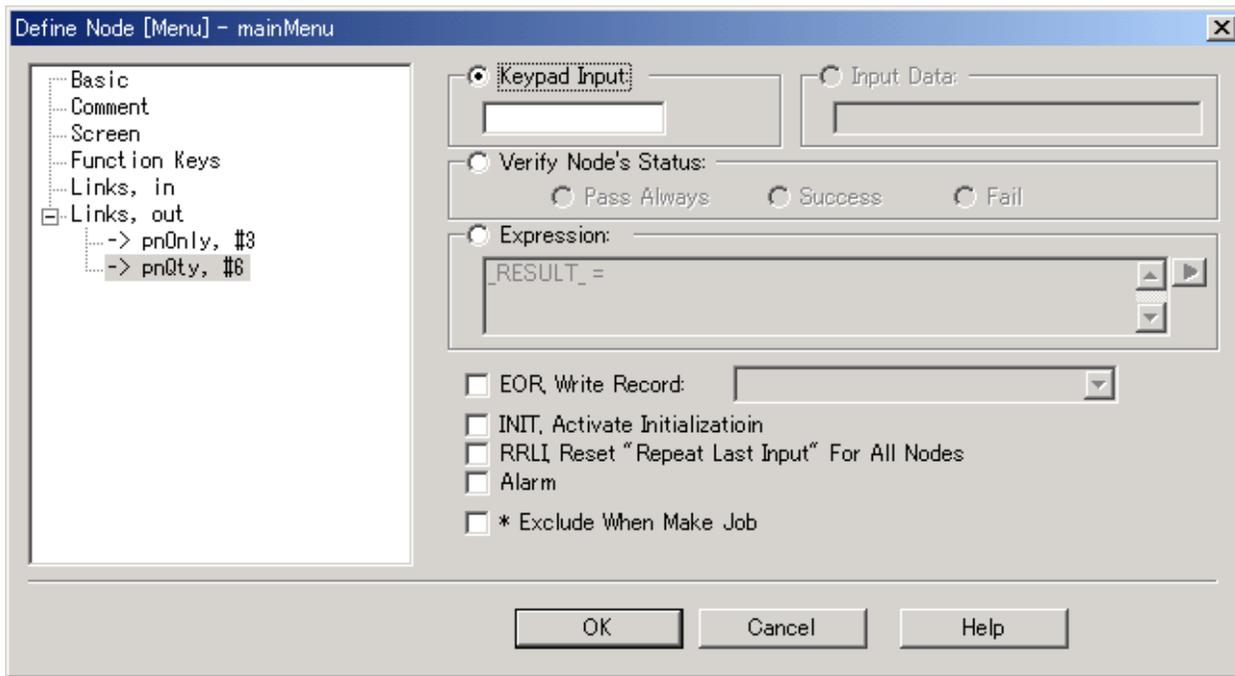
11. この Program ノード は主にデータフィールドに情報を埋めるためのものです。ジョブに、 **pNumber**、 **quantity** そして **serial** の三つのデータフィールドがあります。二番目の Collect ノード: **quantity** は **quantity** フィールドの情報を保存します。最初の Collect ノード: **pnQty** は **pNumber** について情報を一時的に保存します。したがって、 Program ノード は **pnQty** から **pNumber** へ情報をコピーし、そして **serial** のフィールドを “0” で埋めます。この Program ノード を定義するために、ユーザはノードの名前を “copy_serial” に変更し、そして以下の C ソースコードをタイプして下さい。

```
_copy_serial()
{
    strcpy(_pNumber, _pnQty);
    strcpy(_serial, “0”);
}
```

12. ここでこれら三つのノード間のリンク を作成する必要があります。マウスポインタをリンク ボタンに置いて、マウスポインタがリンク ボタンにある間に左マウスボタンをクリックします。
13. マウスポインタを **mainMenu** ノード に移動し、そして左マウスボタンをクリックしたままにします。マウスポインタを **pnQty** ノード に移動し、左マウスボタンを離します。



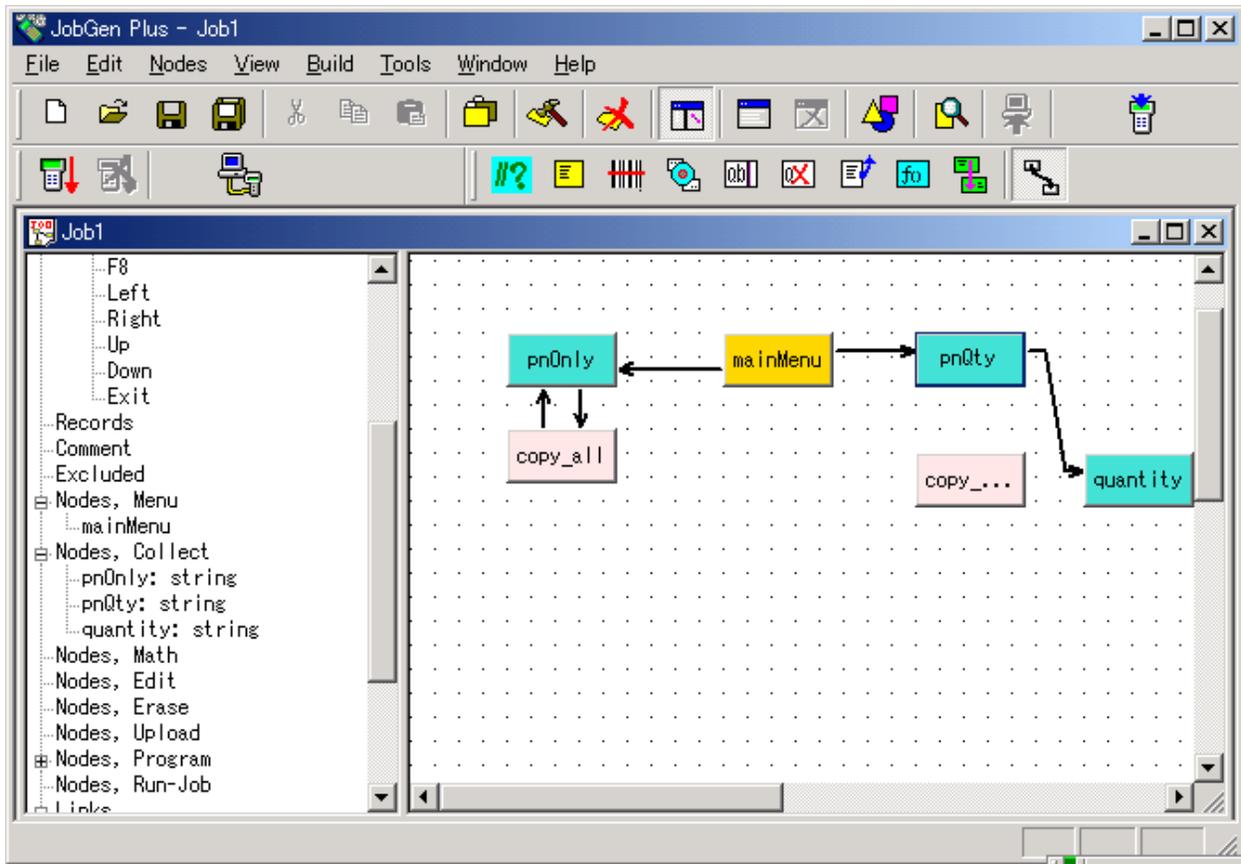
- マウスポインタが新しく作成したリンクにある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



15. **Key Input** オプションをチェックし、マウスポインタを **Key Input** の次のスペースに置いているときに左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます。

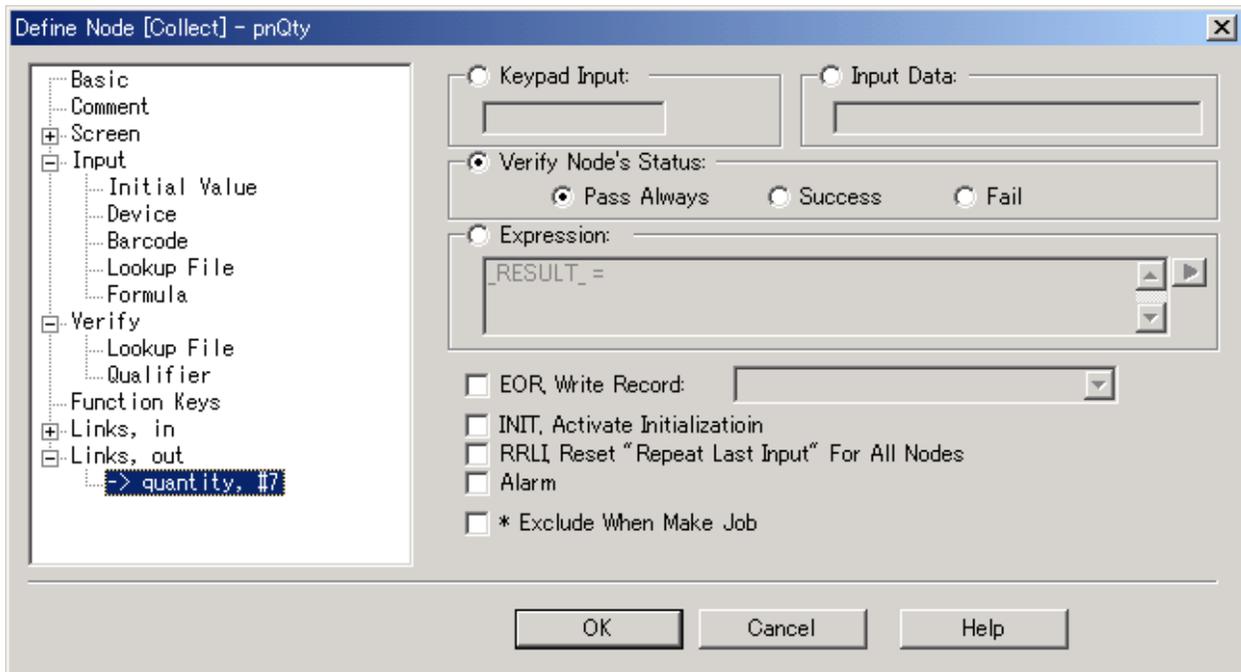


16. 2 のボタンをクリックします。これは、ユーザが”2”のキーを押した場合、プロセスは **mainMenu** ノード から **pnQty** ノード に進むことを示しています。“PT-600 Keypad” ウィンドウを閉じるために **OK** ボタンをクリックして下さい。
17. **OK** をクリックして下さい。
18. マウスポインタを **pnQty** ノード に移動し、左マウスボタンをクリックしたままにします。マウスポインタを **quantity** ノード に移動し、左マウスボタンを離します。



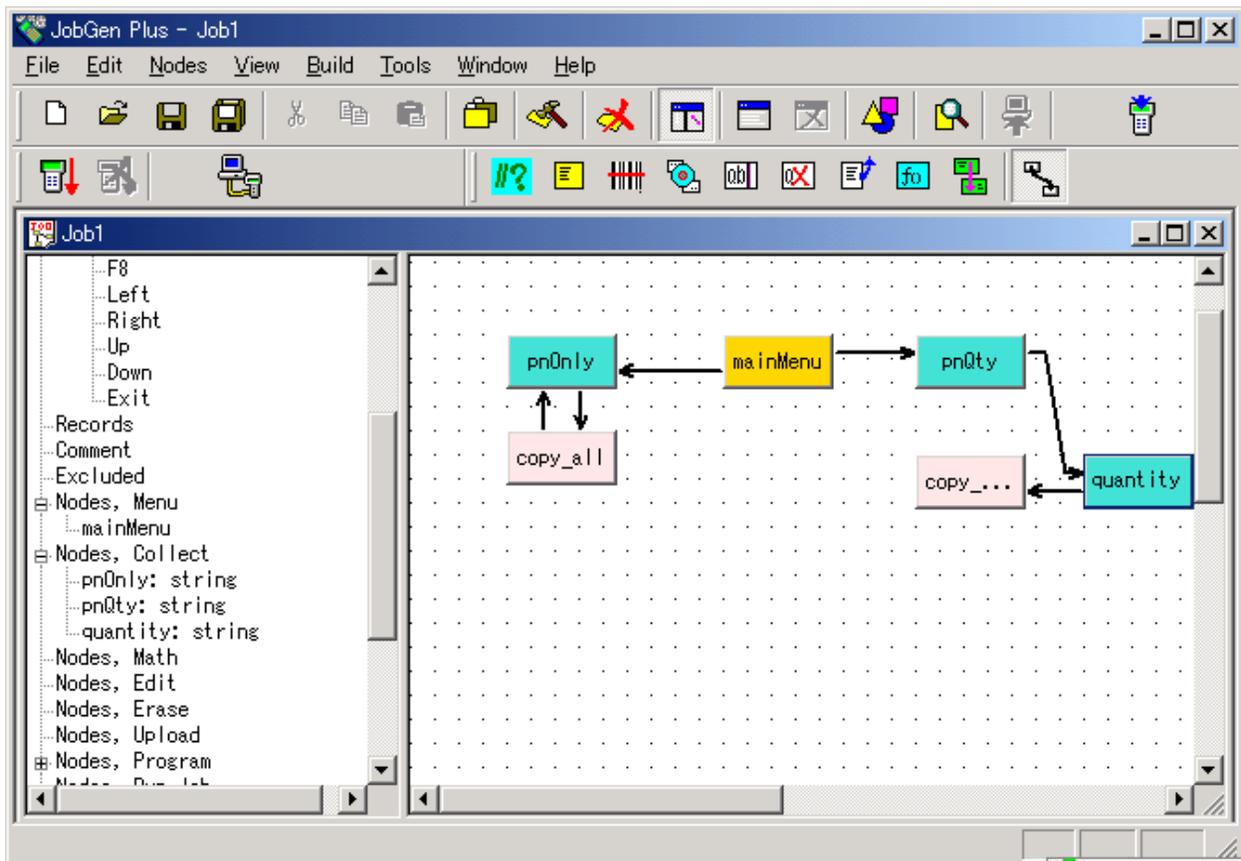
19. マウスポインタが新しく作成したリンク上にある間に左マウスボタンをダブルクリックします。

20. “Define Node” ウィンドウがスクリーンに現れます。

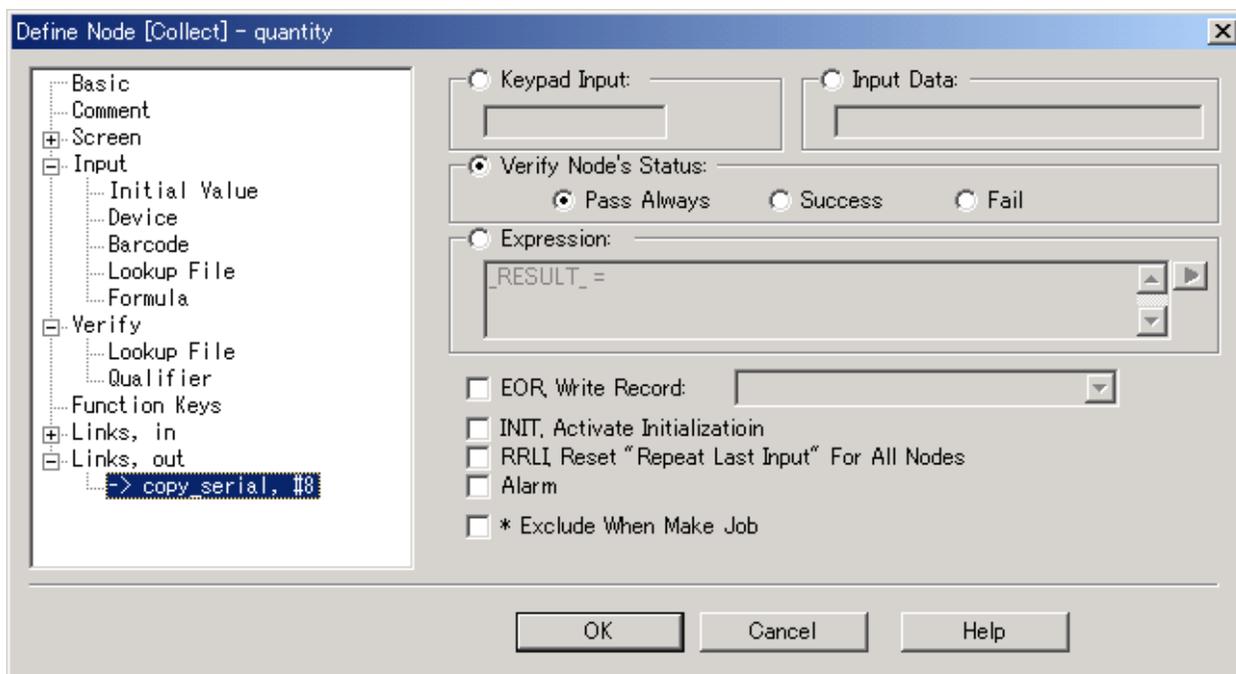


Pass Always オプションは、ジョブが pnQty ノードの処理を終えたら、quantity ノードの処理に進むことを示しています。

21. OK をクリックして下さい。
22. マウスポインタを quantity ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを copy_serial ノードに移動し、左マウスボタンを離します。



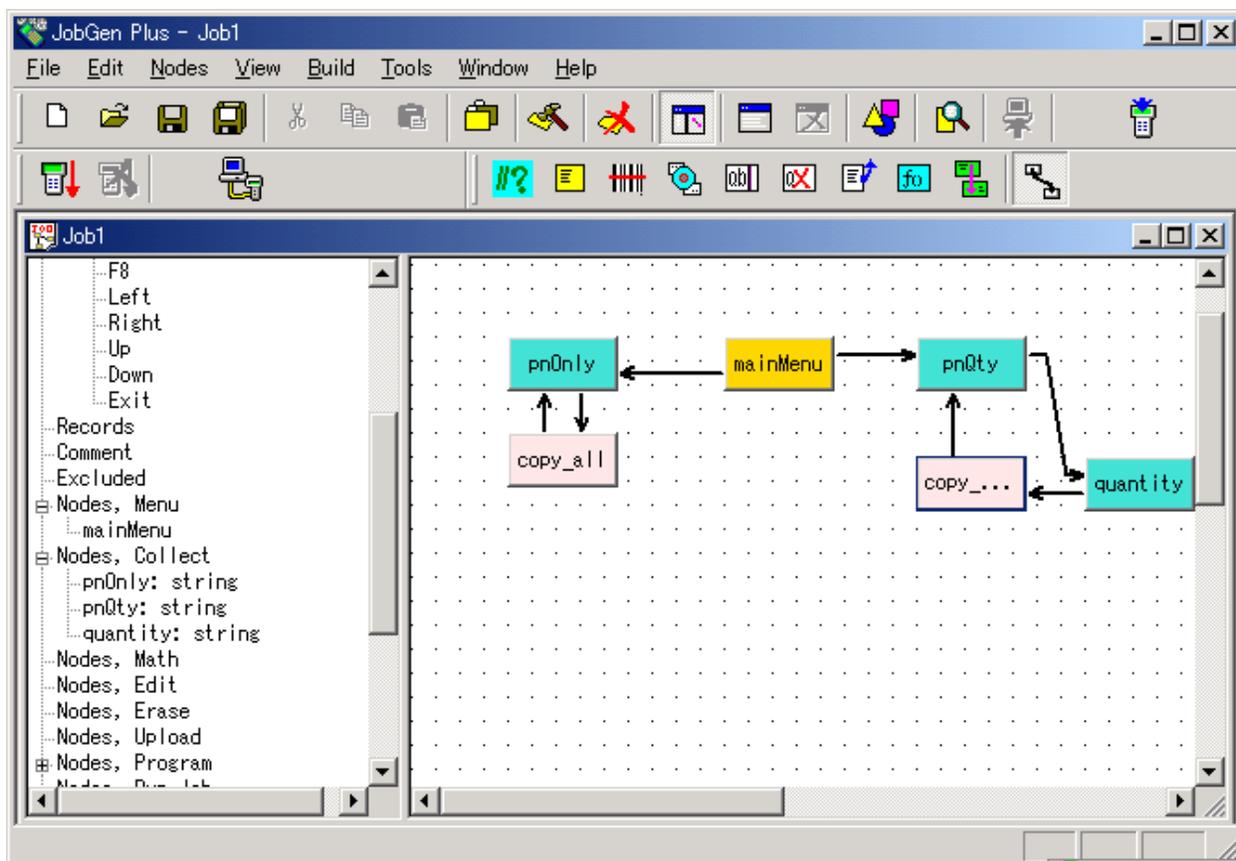
23. マウスポインタが新しく作成したリンク上にある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



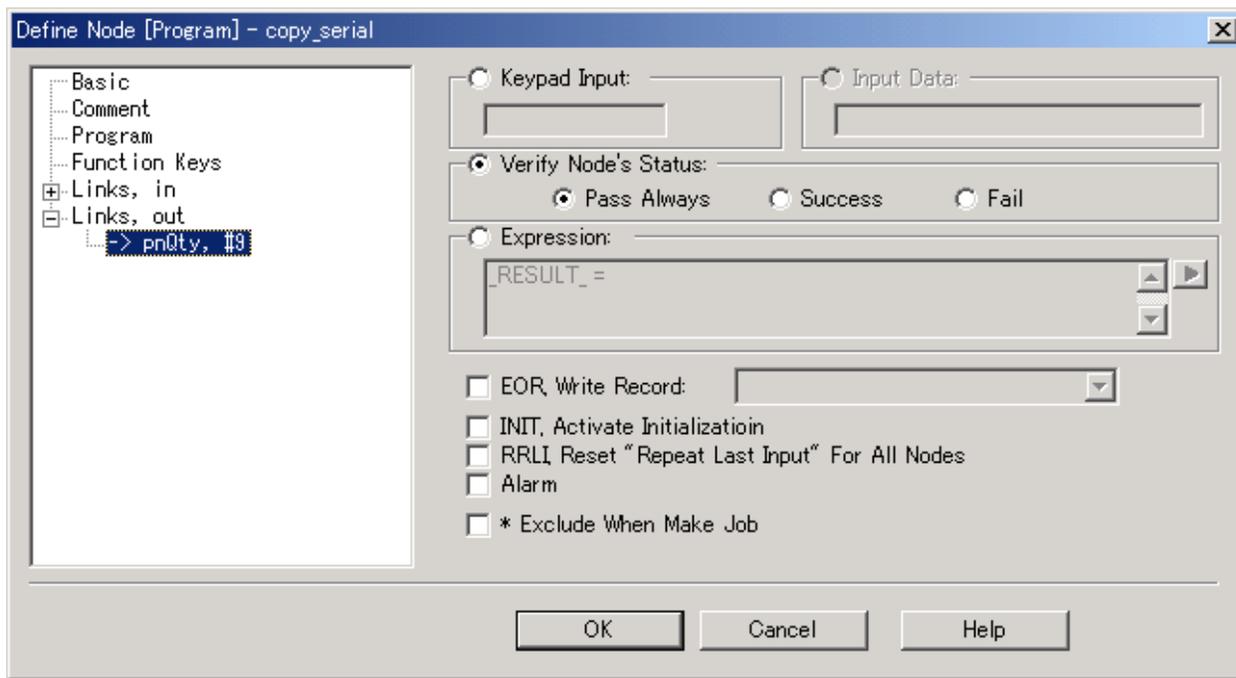
Pass Always オプションは、ジョブが **quantity** ノードの処理を終了したら、**copy_serial** ノードの処理を行うことを意味しています。

24. **Ok** をクリックして下さい。

25. マウスポインタを **copy_serial** ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを **pnQty** ノードに移動し、左マウスポインタを放します。



26. 新しく作成したリンク 上にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



Pass Always オプションは、ジョブが copy_serial ノード の処理を終えたら、 pnQty ノード の処理を行うことを示しています。

27. EOR オプションをチェックして下さい。

27. Ok をクリックして下さい。

作業フロー

13. Part Number 情報を収集。

14. “Exit” キーが押されたら、ステップ 1 へ進みます。

15. Part Number 情報を検証。入力データの検証がパスしなかったら、ステップ 13 に進みます。そうでなければ、次のステップに進みます。

16. Serial Number が連続か、連続でないかを選択します。

17. シリアル番号(Serial Number)が連続の場合、次のステップに進みます。そうでなければステップ 25 に進みます。

18. 最初の Serial Number を収集します。

19. Serial Number 情報を検証します。入力データが検証をパスしなかった場合、ステップ 18 に進みます。そうでなければ次のステップに進みます。

20. 連続した項目の番号を収集します。

21. 番号を検証します。入力データが検証をパスしなかった場合、ステップ 20 に進みます。そうでなければ、次のステップに進みます。

22. カウンタをゼロにセットします。

23. Quantity フィールドに”1”を入れ、データファイルに連続する Serial Number を加えます。

24. カウンタに 1 を加えます。カウンタが連続する項目の番号に等しい場合、ステップ 13 に進みます。そうでなければステップ 23 に進みます。

25. Serial Number を収集します。

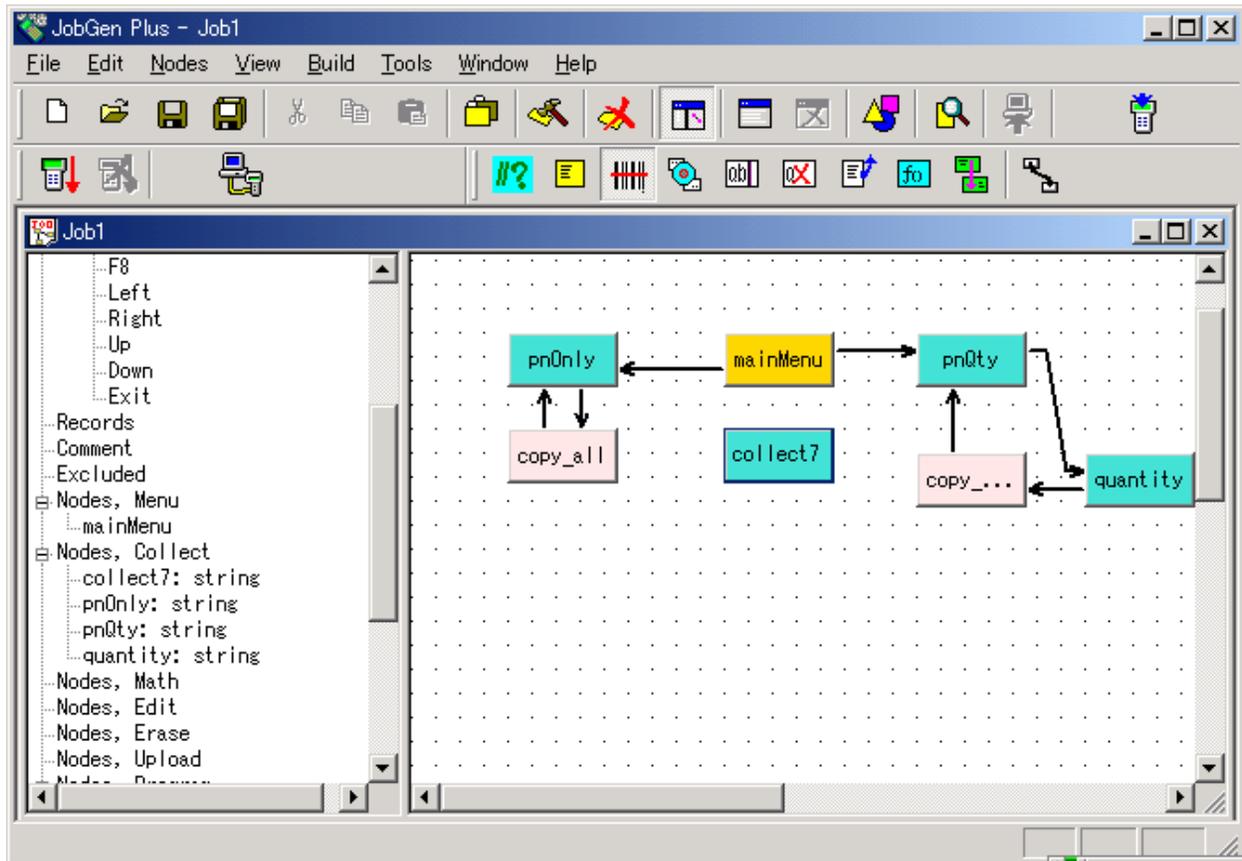
26. Serial Number 情報を検証します。入力データが検証をパスしない場合、ステップ 25 に進みます。そうでなければ、次のステップに進みます。

27. Quantity フィールドに”1”を入れます。ステップ 13 に進みます。

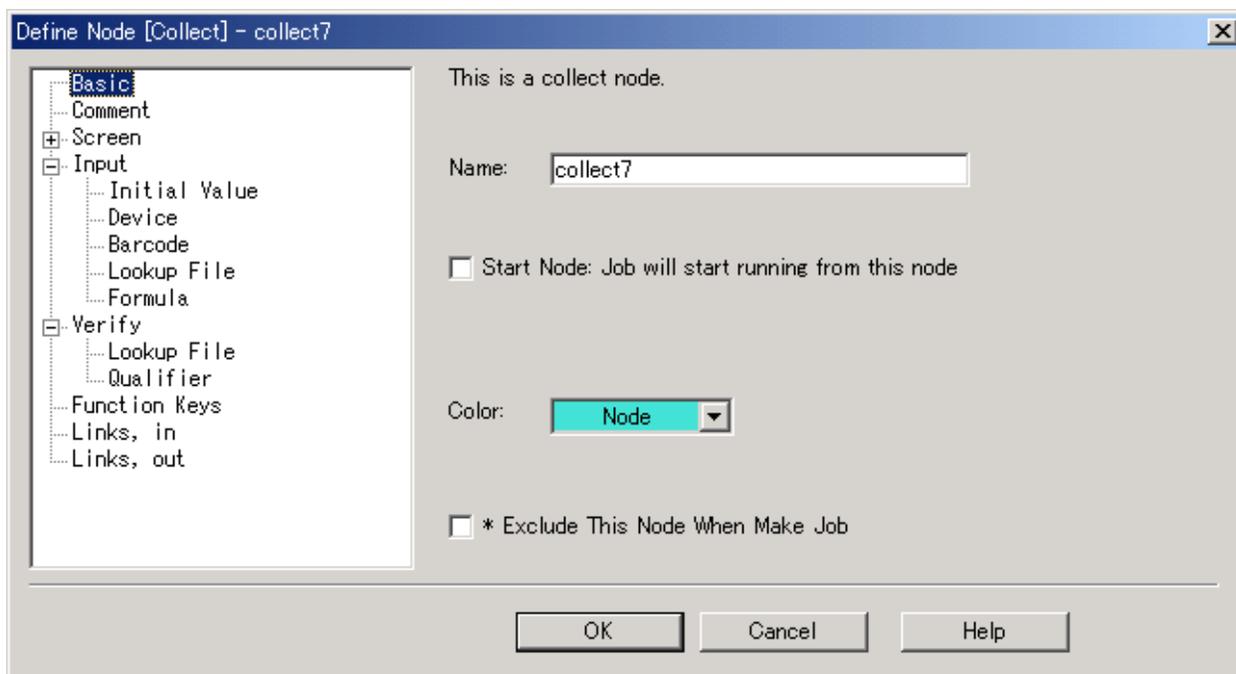
JobGen Plus

1. マウスポインタを Collect ノードボタンに移動することによって Part Number についての収集ノードを作成します。マウスポインタが Collect ノード ボタンにある間に左マウスボタンをクリックします。

2. Collect ノード が置かれる場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。

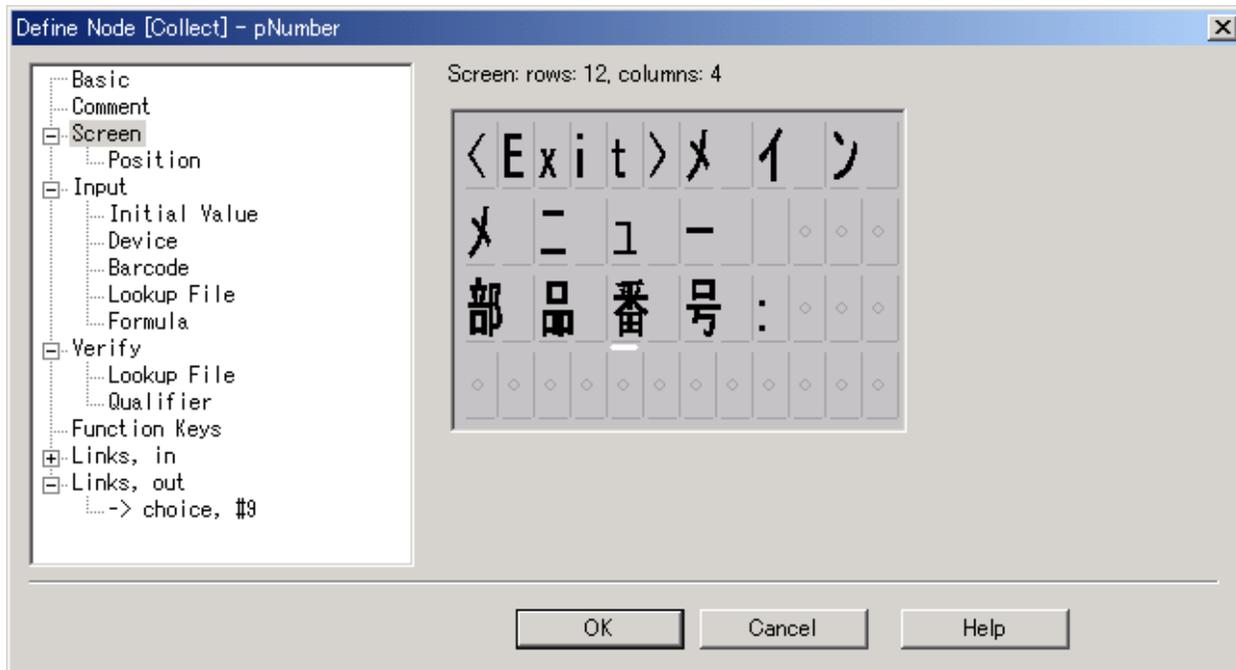


3. 新しい Collect ノード が作成されました。マウスポインタが Collect ノード がある間に、左マウスボタンをダブルクリックします。“Define Node [Collect]” ウィンドウ がスクリーンに現れます。

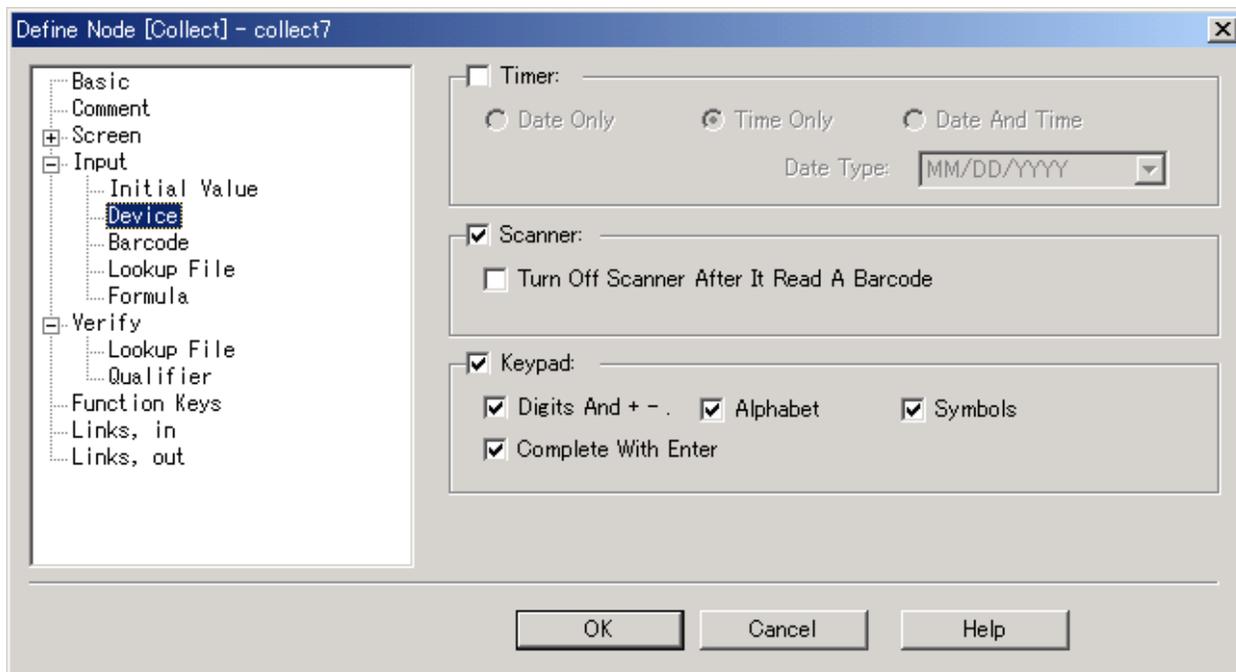


4. 以下のオプションを変更します:

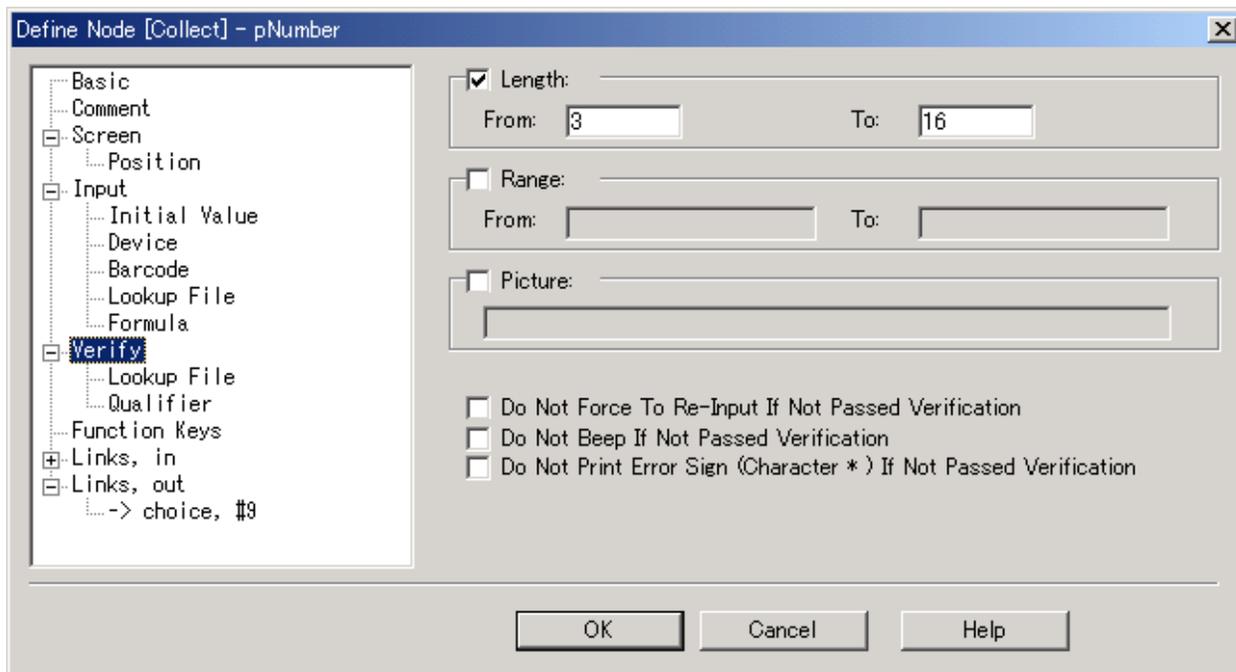
- (a) ノード名を“pNumber”に変更します。
- (b) **Screen** プロパティをクリックします。



- (c) プロンプト(入力要求)をタイプします。
- (d) **Input > Device** プロパティをクリックします。



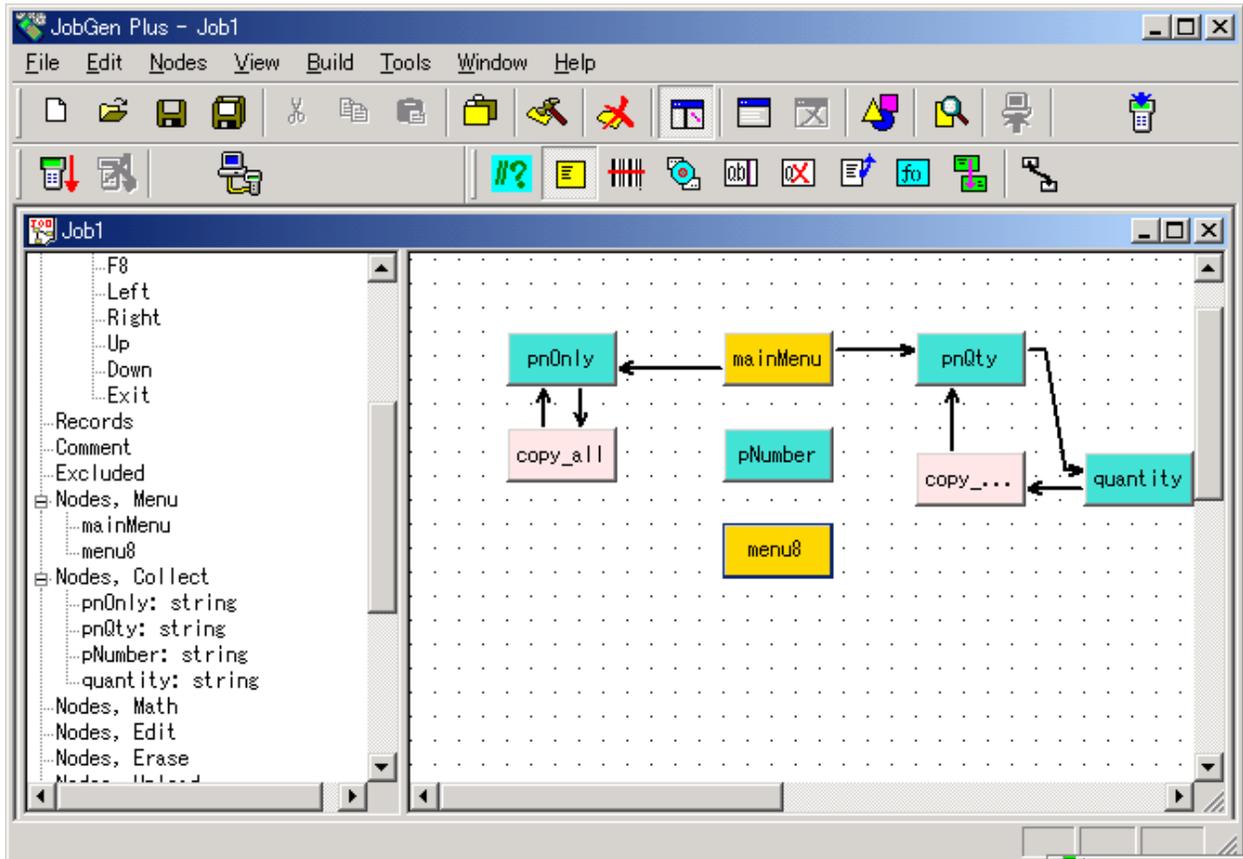
- (e) **Keypad** と **Scanner** オプションの両方が選択されていることを確認して下さい。
- (f) **Input** プロパティをクリックし、**Set Data As Last Record, And Complete Input** オプションをチェックします。
- (g) **Verify** プロパティをクリックします。



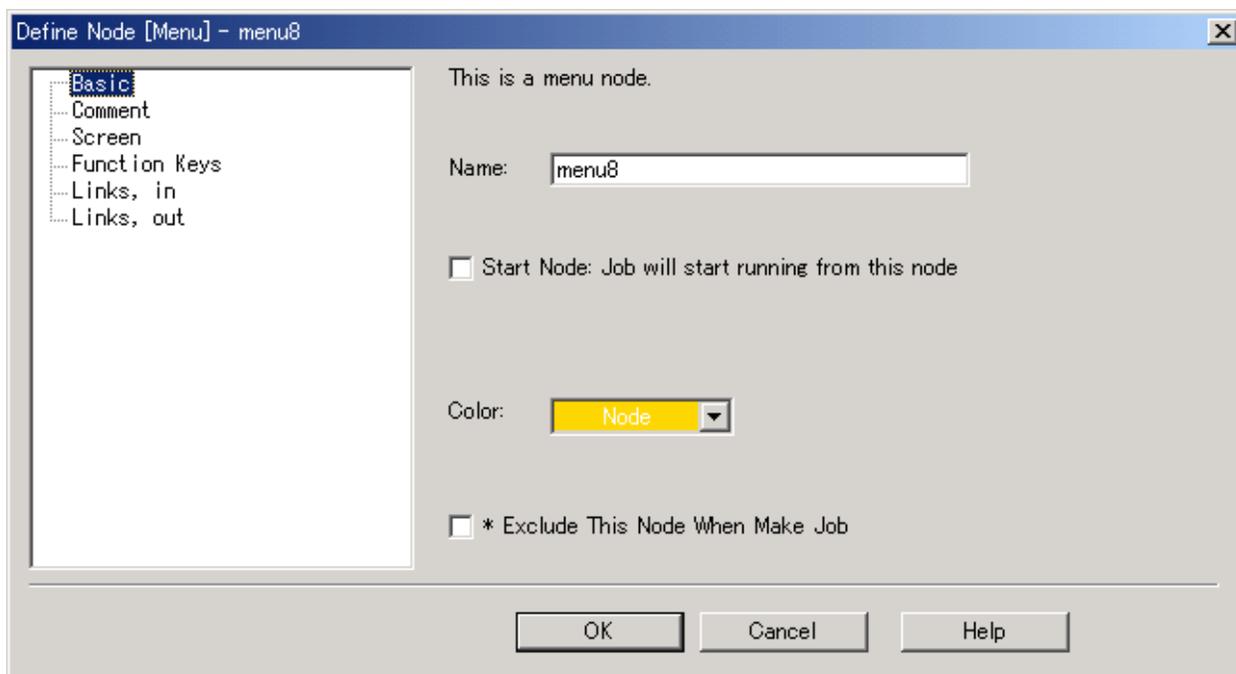
- (h) **Length** オプションをチェックし、最小値 **3** を From: に、最大値 **16** を To: に入れます。

(k) OK をクリックします。

5. Menu ノード ボタンをクリックします。
6. マウスポインタを Menu ノード が置かれる場所に移動します。左マウスボタンをダブルクリックします。



7. 新しい Menu ノード が作成されました。マウスポインタが Menu ノード にある間に左マウスボタンをダブルクリックして下さい。“Define Node[Menu]” ウィンドウがスクリーンに現れます。



8. 以下の情報を入力します:

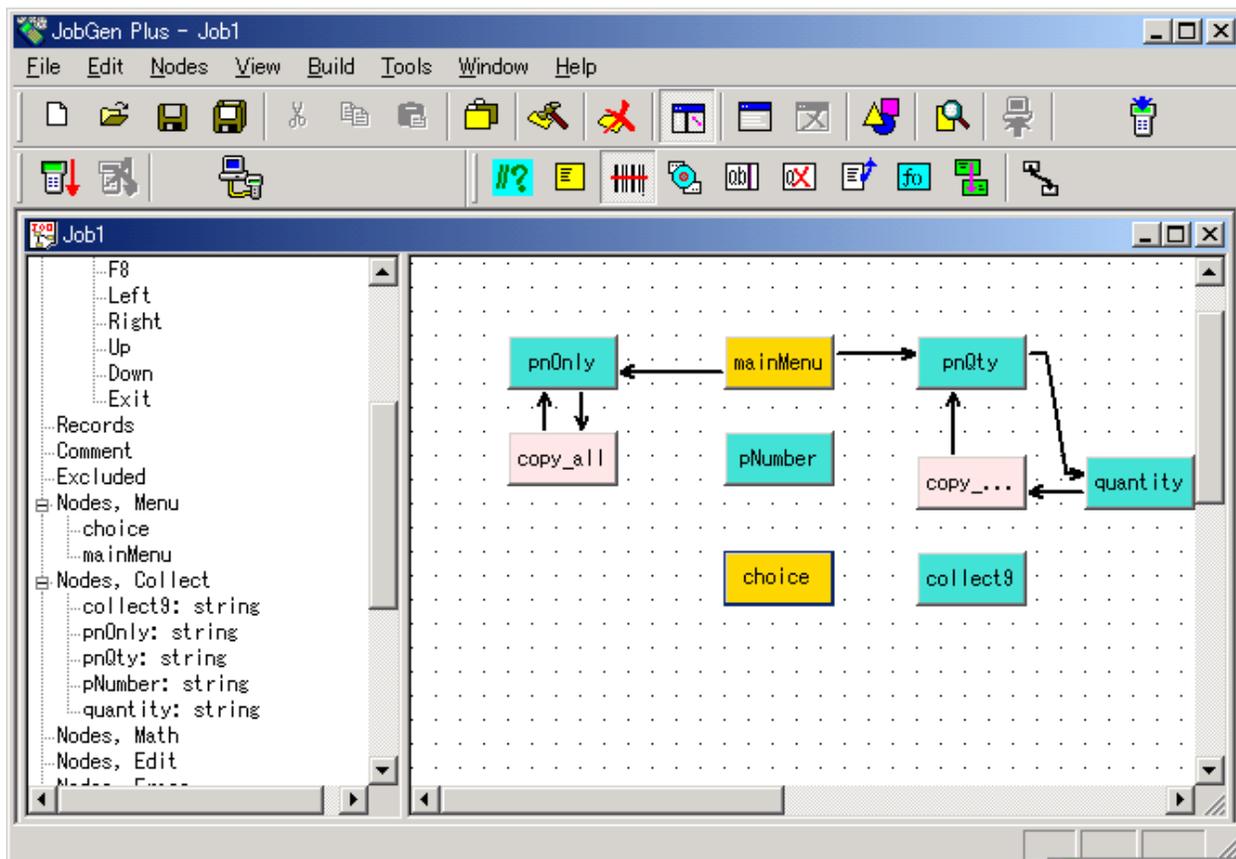
(a) 名前を“choice”に変更します。

(b) ポータブル・ターミナルのスクリーンに表示する内容をタイプします。

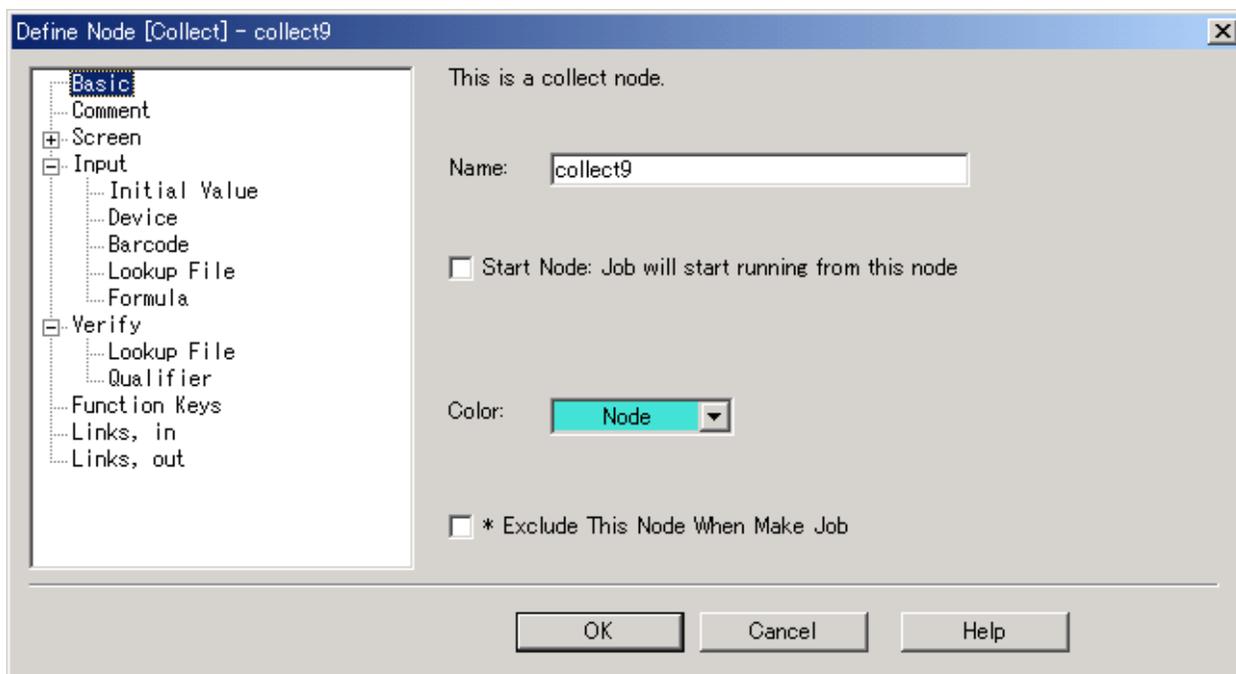


9. **OK** をクリックします。

10. マウスポインタを Collect ノード ボタンに移動することによってシリアル番号についての収集ノード を作成します。 マウスポインタが Collect ノード ボタンにある間に左マウスボタンをクリックします。
11. Collect ノード を置く場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



12. 新しい収集ノード(Collect ノード) が作成されました。マウスポインタが収集ノード (Collect ノード) にある間に左マウスボタンをダブルクリックします。 “Define Node [Collect]” ウィンドウがスクリーンに現れます。



以下のオプションをチェックします:

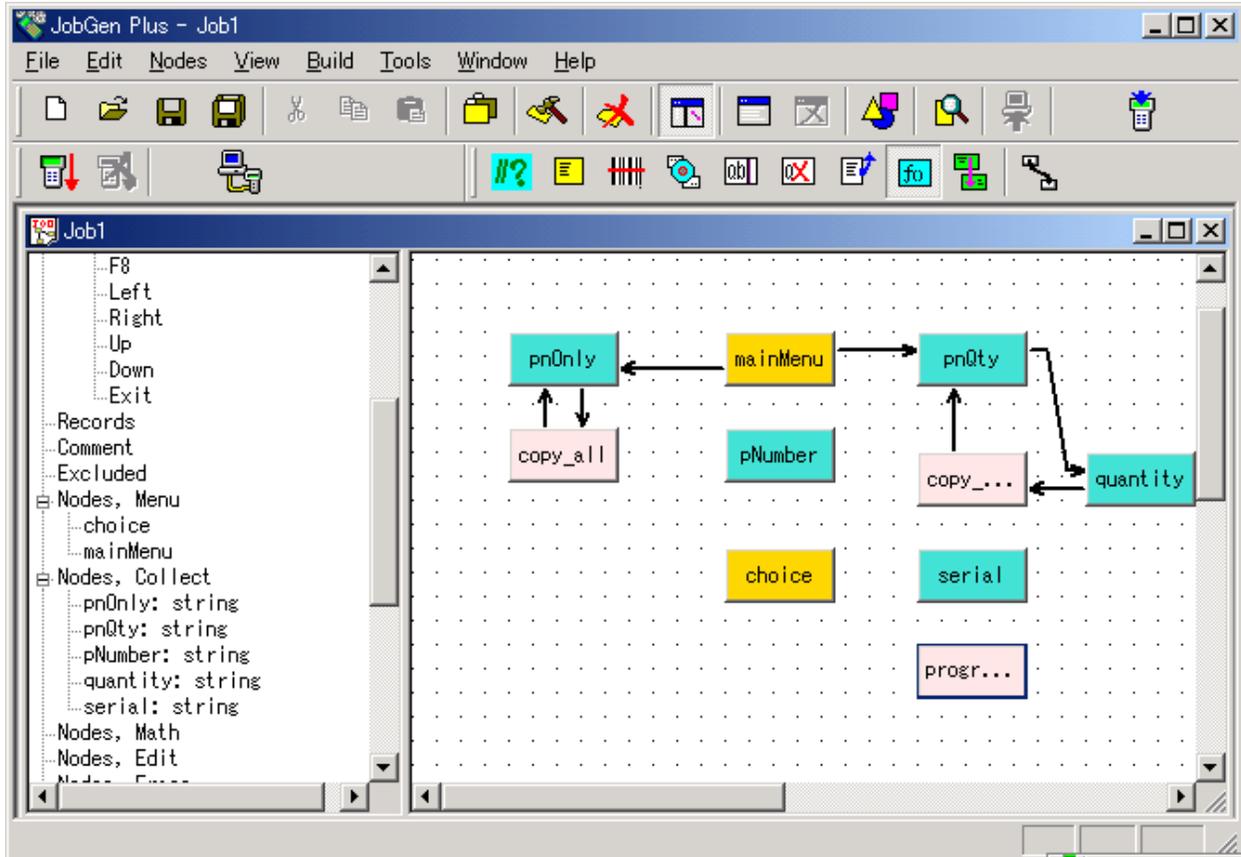
- (a) ノード の名前を “serial” に変更します。
- (b) **Screen** プロパティをクリックします。



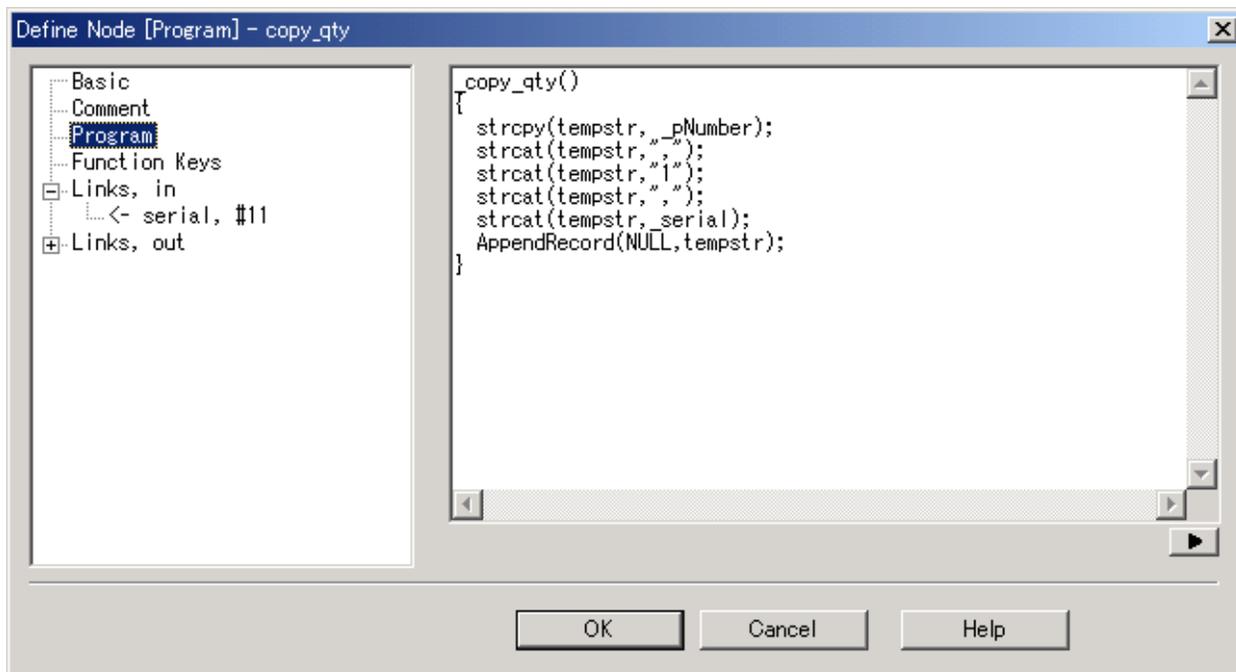
- (c) プロンプト(入力要求)をタイプします。
- (e) **OK** をクリックします。

Program ノード ボタンにマウスカーソルを移動します。Program ノード ボタンにポインタがある間に左マウスボタンをクリックします。

13. Program ノード を置く場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



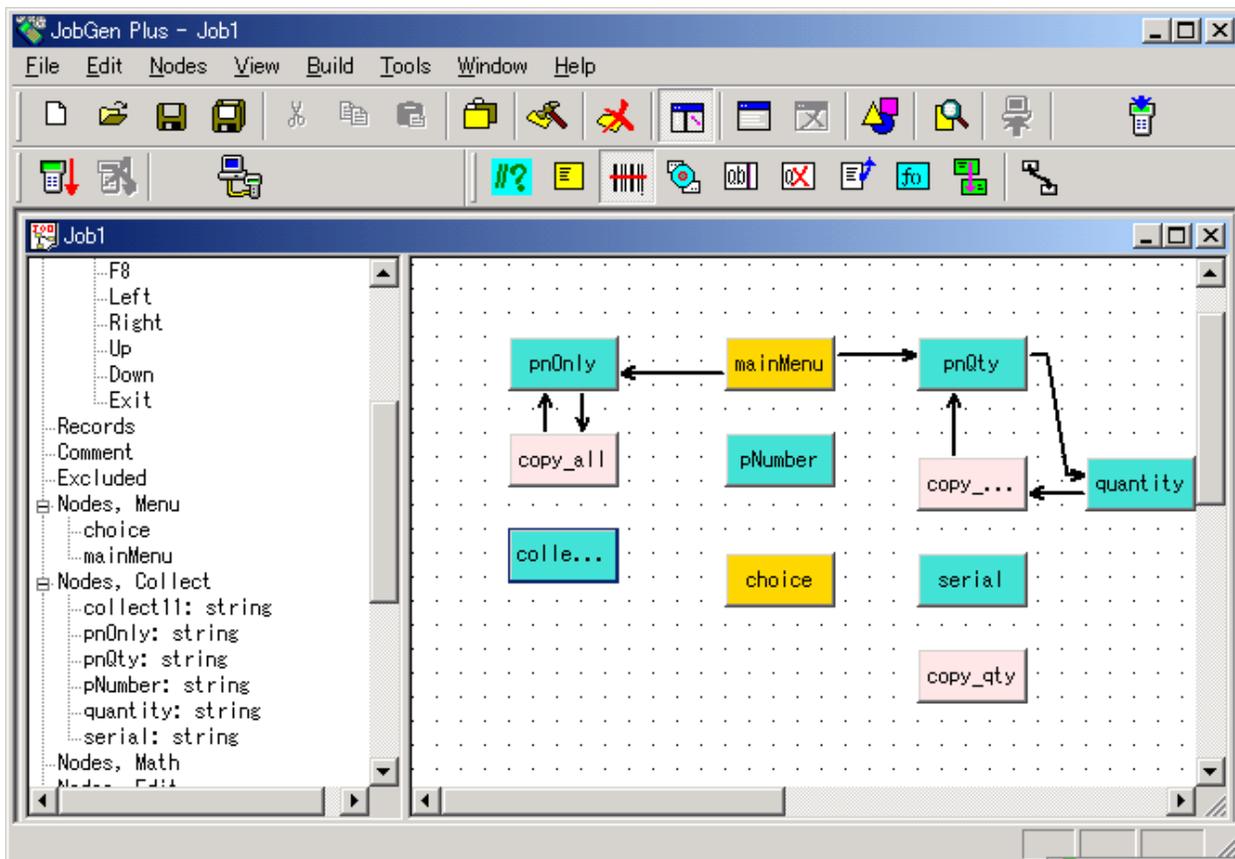
14. 新しく作成したプログラムノード (Program ノード) にマウスポインタがある間に左マウスボタンをダブルクリックします。"Define Node [Program]" ウィンドウがスクリーンに現れます。



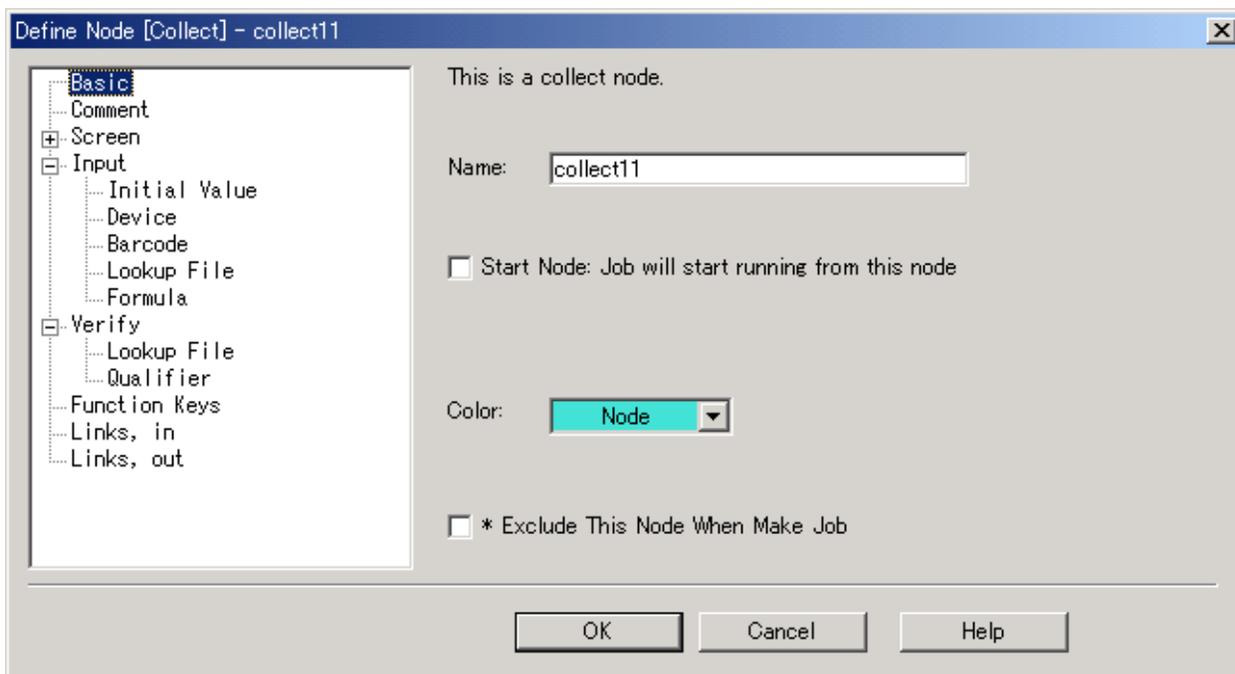
15. この Program ノード は主に **quantity** フィールドに情報を埋めます。 **pNumber** フィールドと **serial** フィールドにデータを入れたら、この Program ノード は **quantity** フィールドに”1”を入れます。 この Program ノード を定義するには、ユーザはノード名を “copy_qty”に変更し、以下の C ソースコードをタイプする必要があります。

```
_copy_qty()
{
    strcpy(tempstr, _pNumber);
    strcat(tempstr, ",");
    strcat(tempstr, "1");
    strcat(tempstr, ",");
    strcat(tempstr, serial);
    AppendRecord(NULL, tempstr);
}
```

16. OK をクリックして下さい。
17. マウスポインタを Collect ノード ボタンに移動します。マウスポインタが Collect ノード ボタンにある間に左マウスボタンをクリックします。
18. Collect ノード を置く場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



19. 新しい Collect ノード が作成されました。マウスポインタが Collect ノードにある間に左マウスボタンをダブルクリックします。“Define Node [Collect]” ウィンドウがスクリーンに現れます。

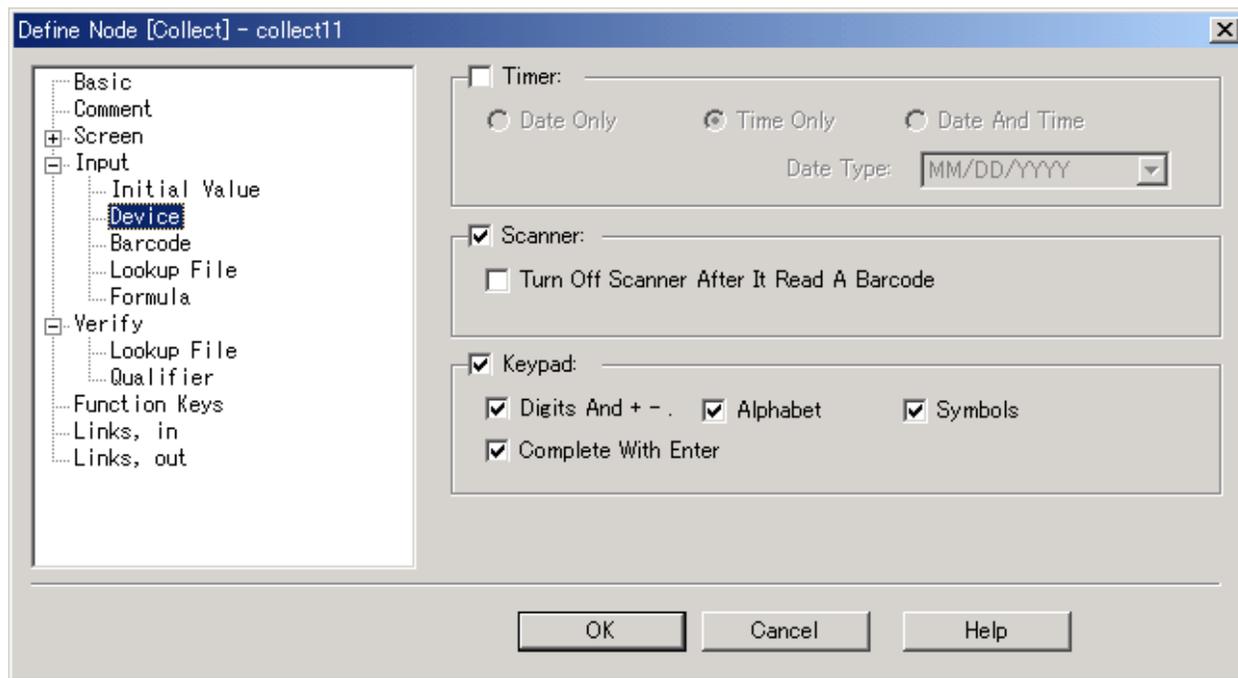


20. 以下のオプションを変更します。

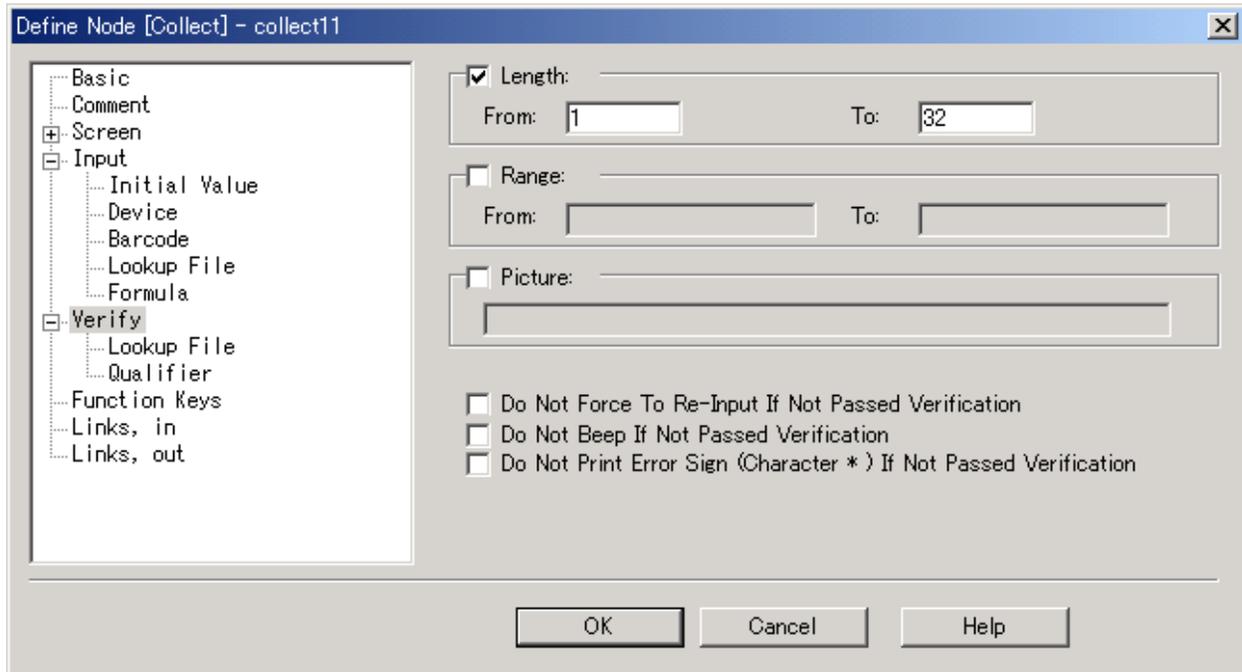
- (a) ノード名を “startSN”に変更します。
- (b) **Screen** プロパティをクリックします。



- (c) プロンプト(入力要求)をタイプします。
- (d) **Input > Device** プロパティをクリックします。

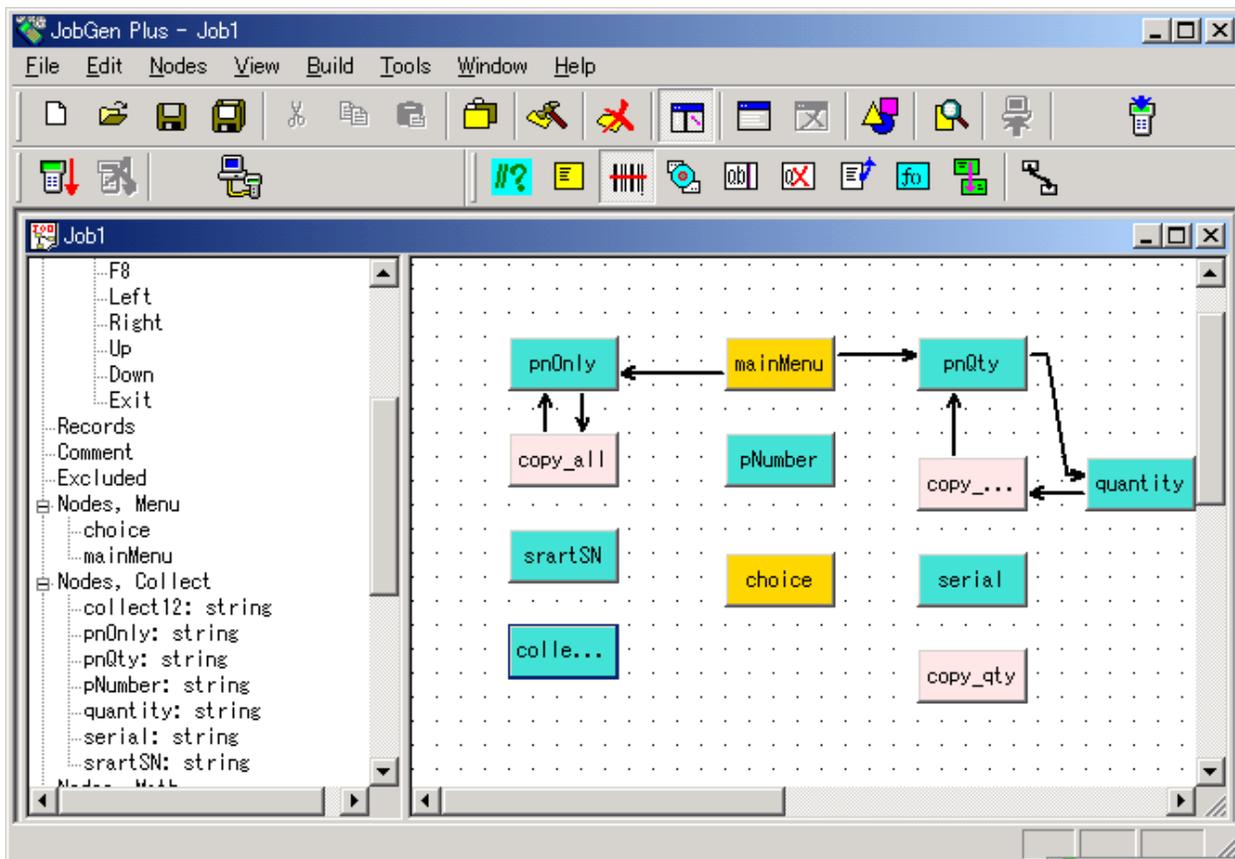


- (e) **Keypad** と **Scanner** オプションの両方が選択されていることを確認して下さい。
(f) **Verify** プロパティをクリックして下さい。

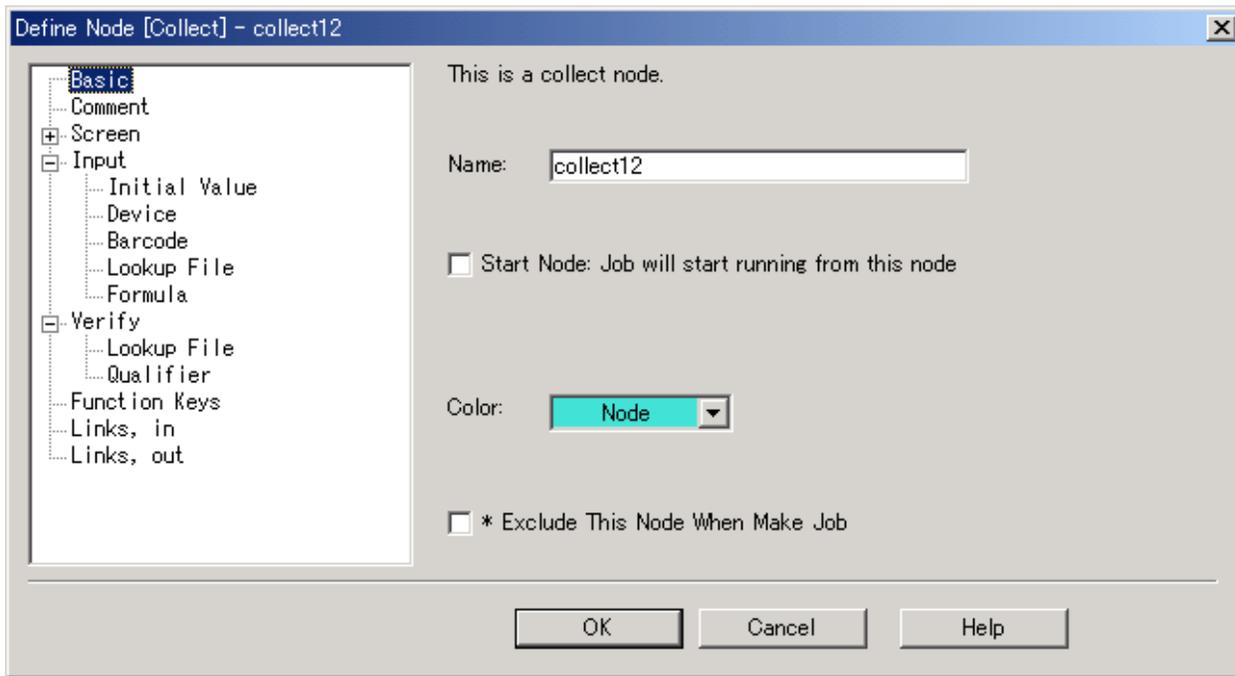


- (g) **Length** オプションをチェックして、最小値 **1** を From: に、そして最大値 **32** を To: に入れます。
(h) **OK** をクリックします。

21. マウスポインタを Collect ノード が置かれる場所に移動します。左マウスボタンをダブルクリックします。



22. マウスポインタが Collect ノード にある間に左マウスボタンをダブルクリックします。“Define Node [Collect]” ウィンドウがスクリーンに現れます。

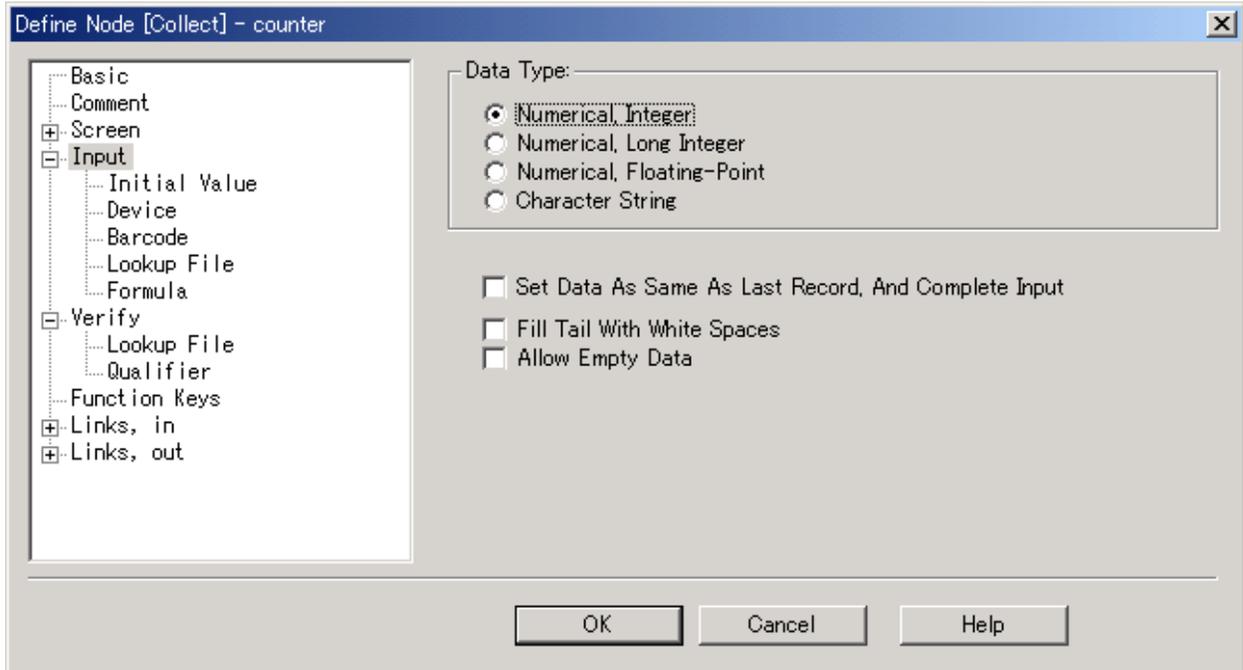


23. 以下のオプションを変更します。

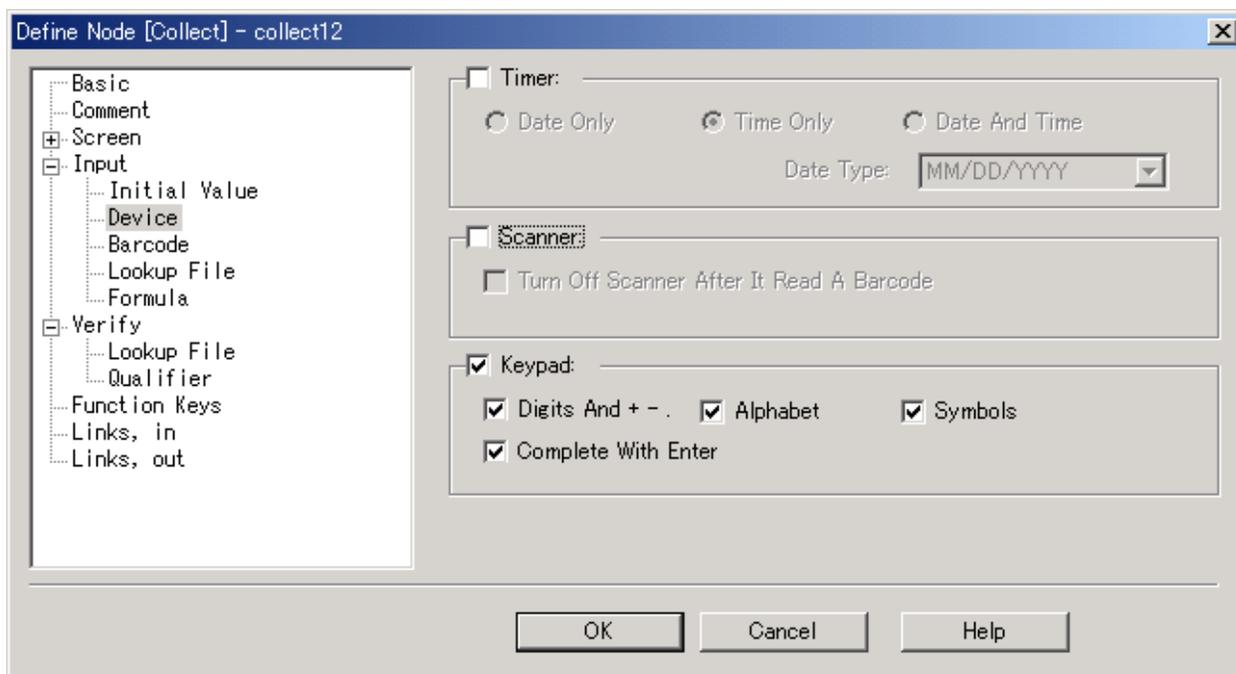
- (a) ノード名を “counter”に変更します。
- (b) **Screen** プロパティをクリックします。



- (c) プロンプト(入力要求)をタイプします。
- (d) **Input** プロパティをクリックします。Numerical Integer をチェックします。

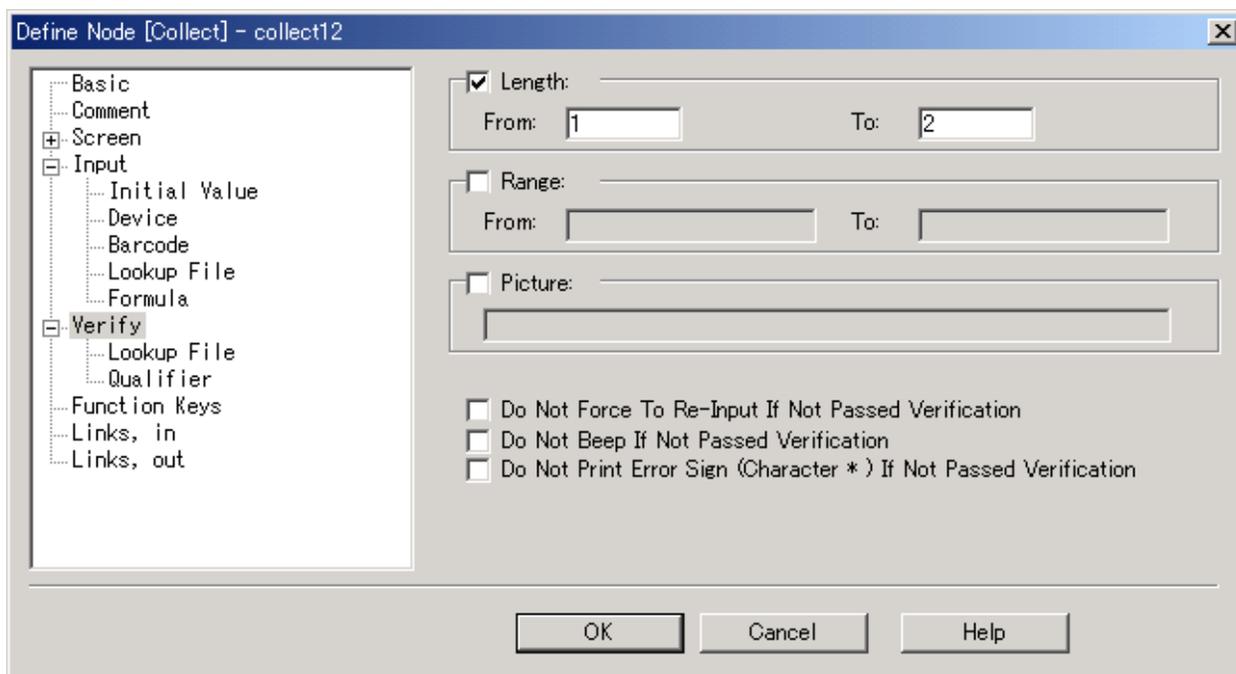


- (e) **Input > Device** プロパティをクリックします。



(f) **Keypad** オプションをチェックします。

(g) **Verify** プロパティをクリックします。



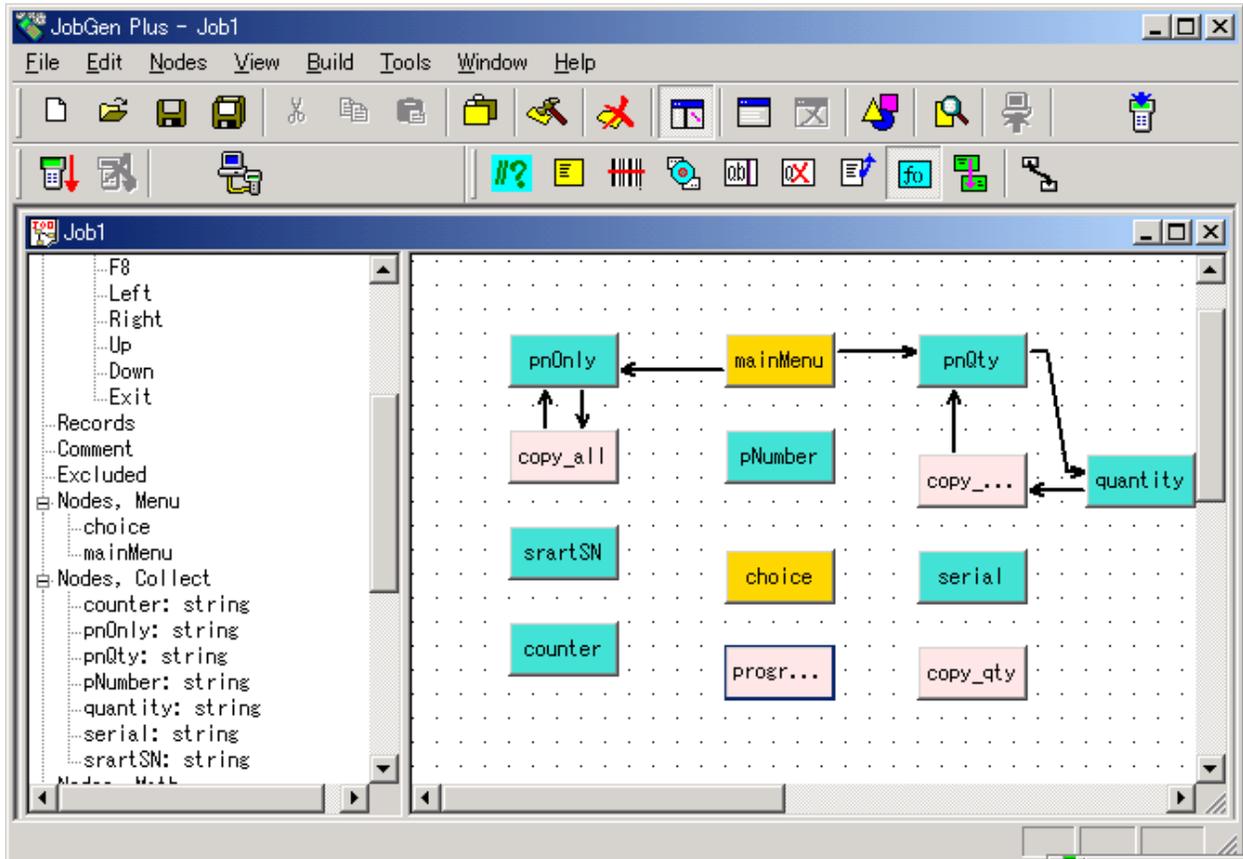
Length オプションをチェックし、最小値 1 を From: に、最大値 2 を To: に入れます。

(h) **Range** オプションをチェックし、From を 0、To: を 30 とします。

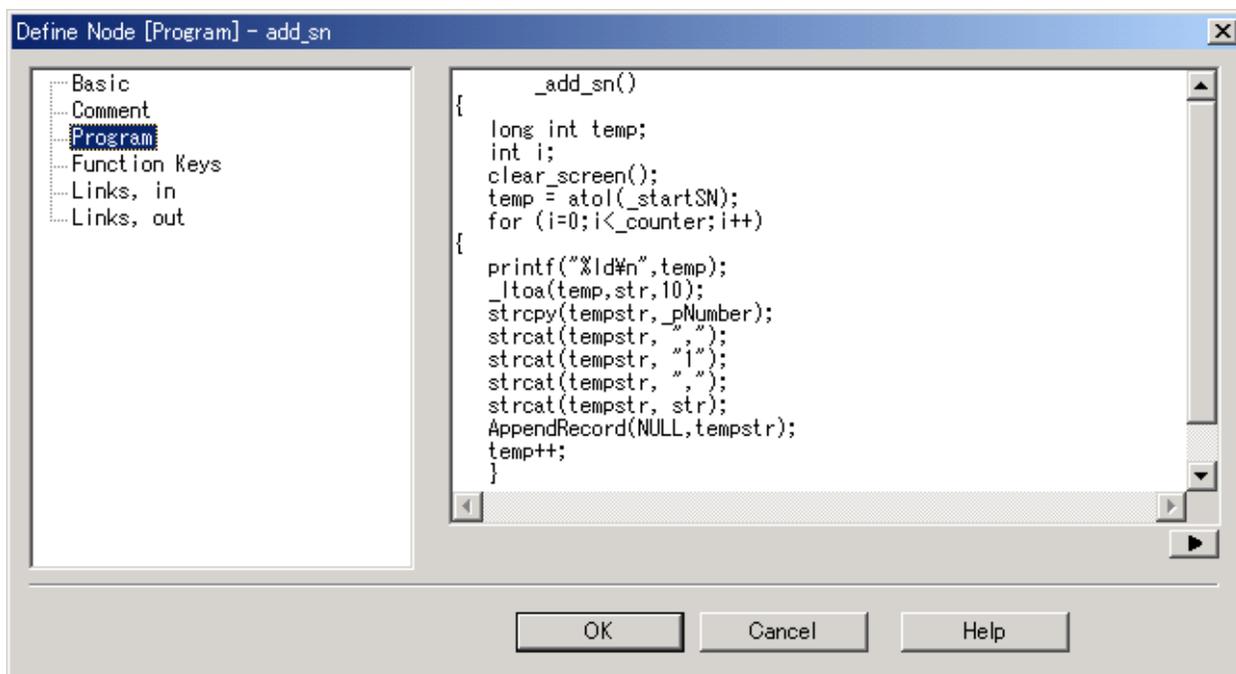
(k) **OK** をクリックします。

26. マウスポインタを Program ノード ボタンに移動します。ポインタが Program ノード ボタンにある間に左マウスボタンをクリックします。

27. Program ノード が置かれる場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。



28. 新しく作成した Program ノード にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node [Program]” ウィンドウがスクリーンに現れます。



29. この Program ノード は主にデータフィールドに情報を埋めます。ユーザははじめの Serial Number と保存する Serial Numbers 数を入力する必要があります。この Program ノード は **quantity** フィールドに”1”を入れ、そして生成した連続するシリアル番号を **serial** フィールドに入れます。この Program ノード を定義するには、ユーザはノード名を “add_sn”に変更し、以下の C ソースコードをタイプします。

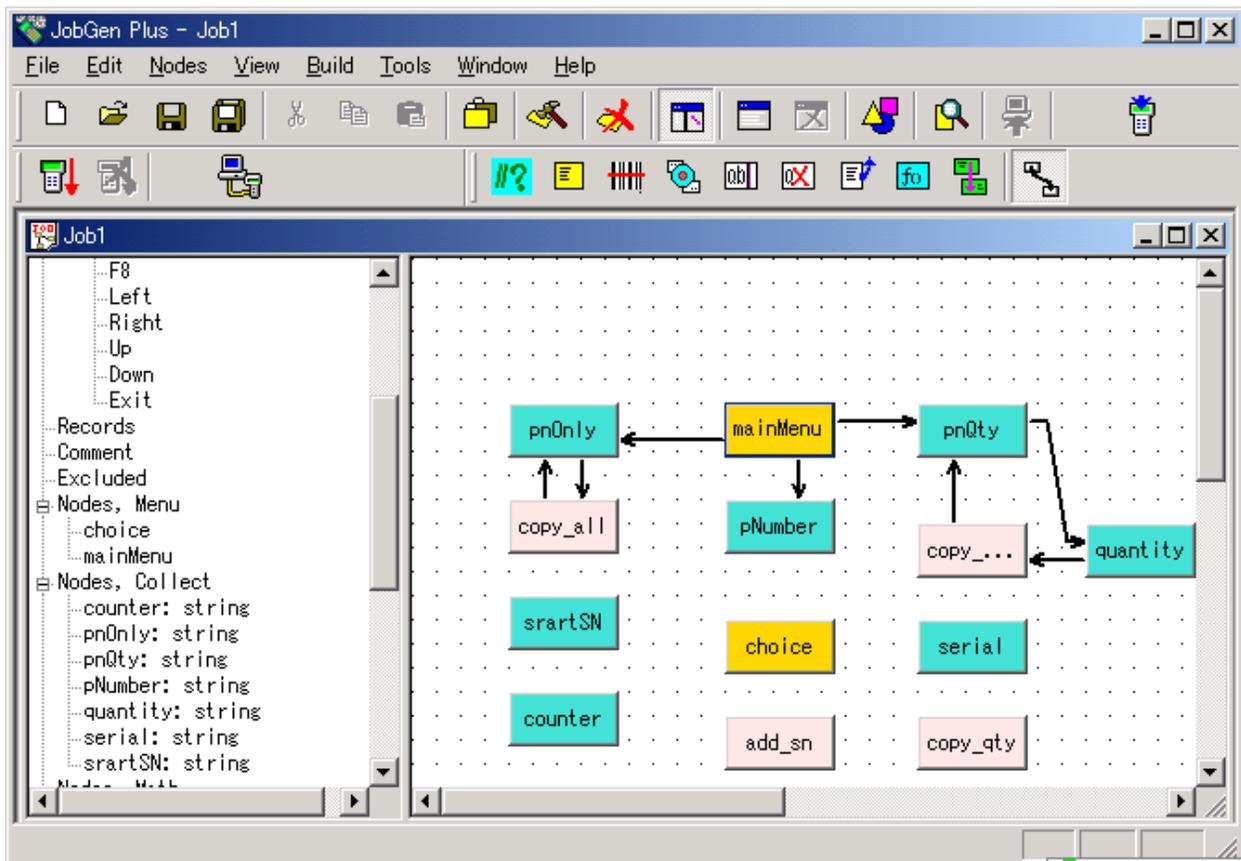
```
_add_sn()
{
    long int temp;
    int i;
    clear_screen();
    temp = atol(_startSN);
    for (i=0;i<_counter;i++)
    {
        printf(“%ld\n”,temp);
        _ltoa(temp,str,10);
        strcpy(tempstr,_pNumber);
        strcat(tempstr, “,“);
        strcat(tempstr, “1”);
        strcat(tempstr, “,“);
        strcat(tempstr, str);
    }
}
```

```

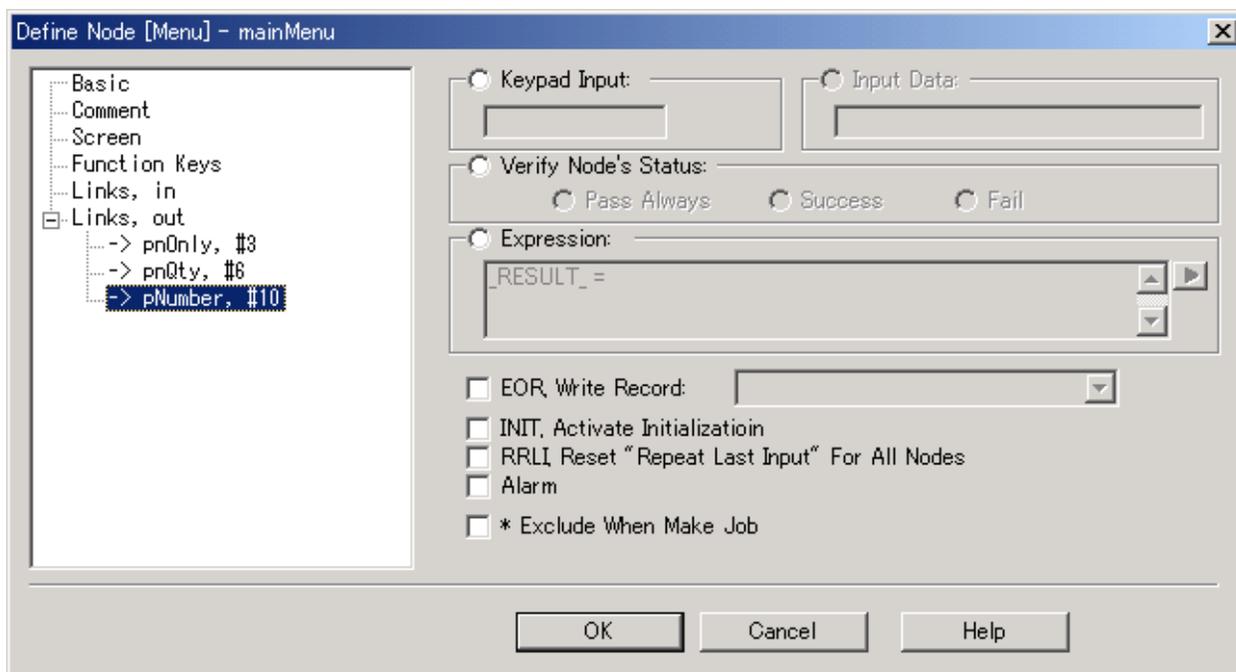
AppendRecord(NULL,tempstr);
temp++;
}
}

```

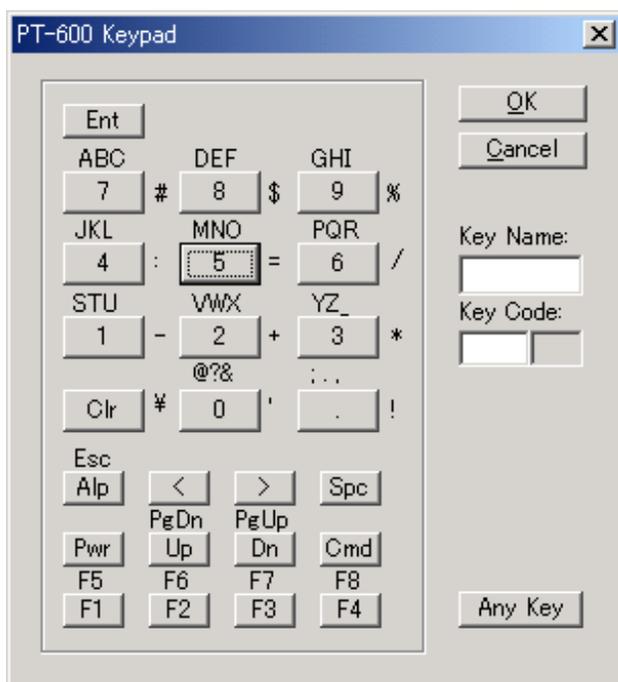
30. OK をクリックします。
31. これらの三つの ノード 間にリンクを作成する必要があります。マウスポインタをリンクボタンに移動し、マウスポインタがリンクボタンにある間に左マウスボタンをダブルクリックします。
32. マウスポインタを mainMenu ノード へ移動し、左マウスボタンをクリックしたままにします。マウスポインタを pNumber ノード へ移動し、左マウスボタンを離します。



33. マウスポインタが新しく作成したリンク上にある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



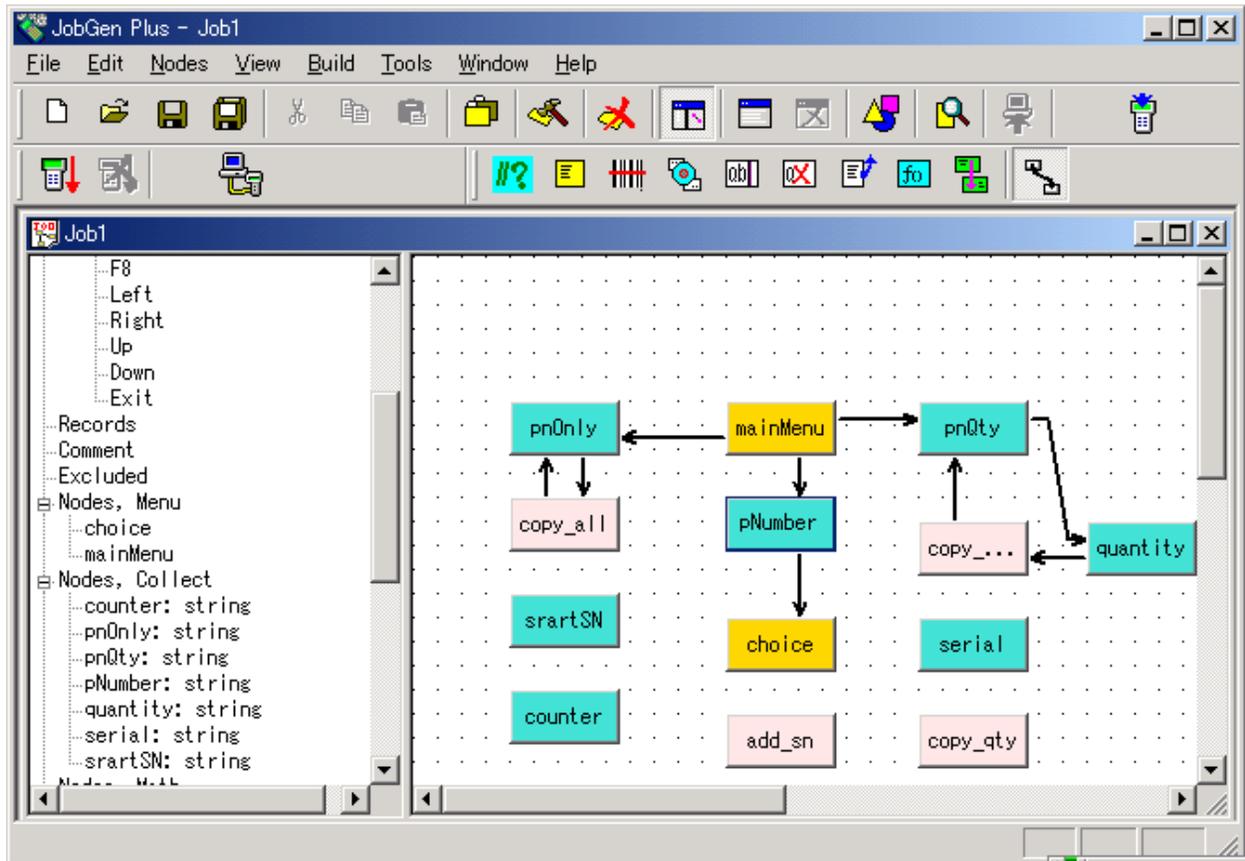
34. **Key Pad Input** オプションをチェックし、**Key Pad Input** オプションの次のスペース上にマウスポインタがある間に左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます。



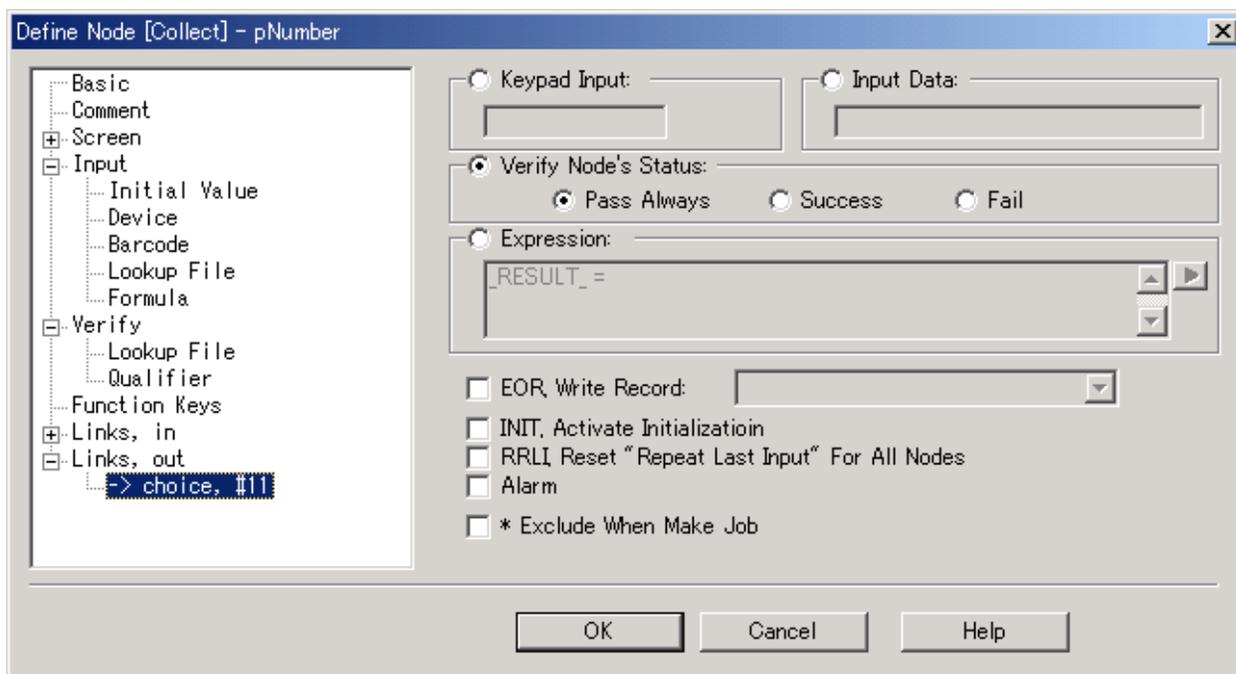
35. 3 のボタンをクリックします。これは、ユーザが”3”のキーを押した場合、プロセスが **mainMenu** ノードから **pNumber** ノードに進むことを意味します。“PT600 Keypad” ウィンドウを閉じるために **OK** ボタンを押します。

36. **OK** をクリックします。

37. マウスポインタを **pNumber** ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを **choice** ノードに移動し、左マウスボタンを離します。



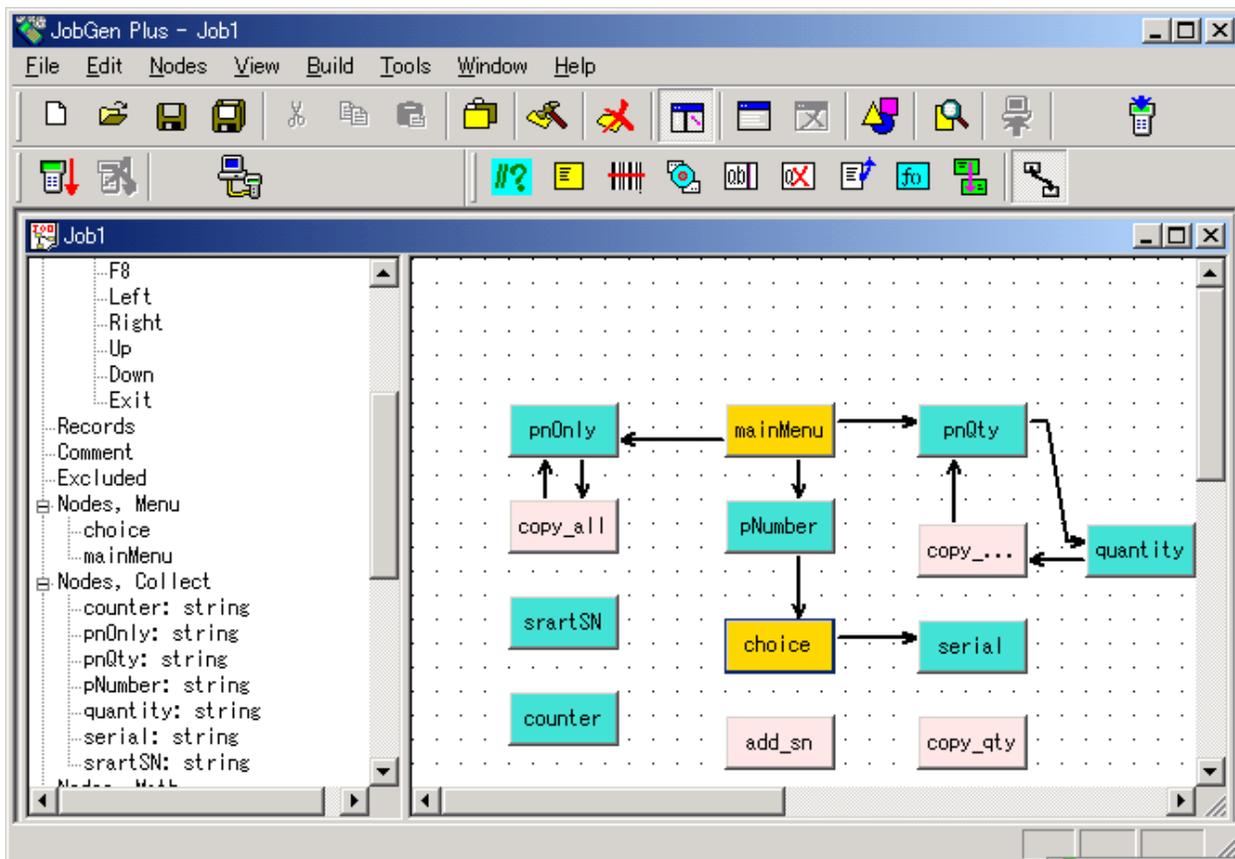
38. 新しく作成したリンク上にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



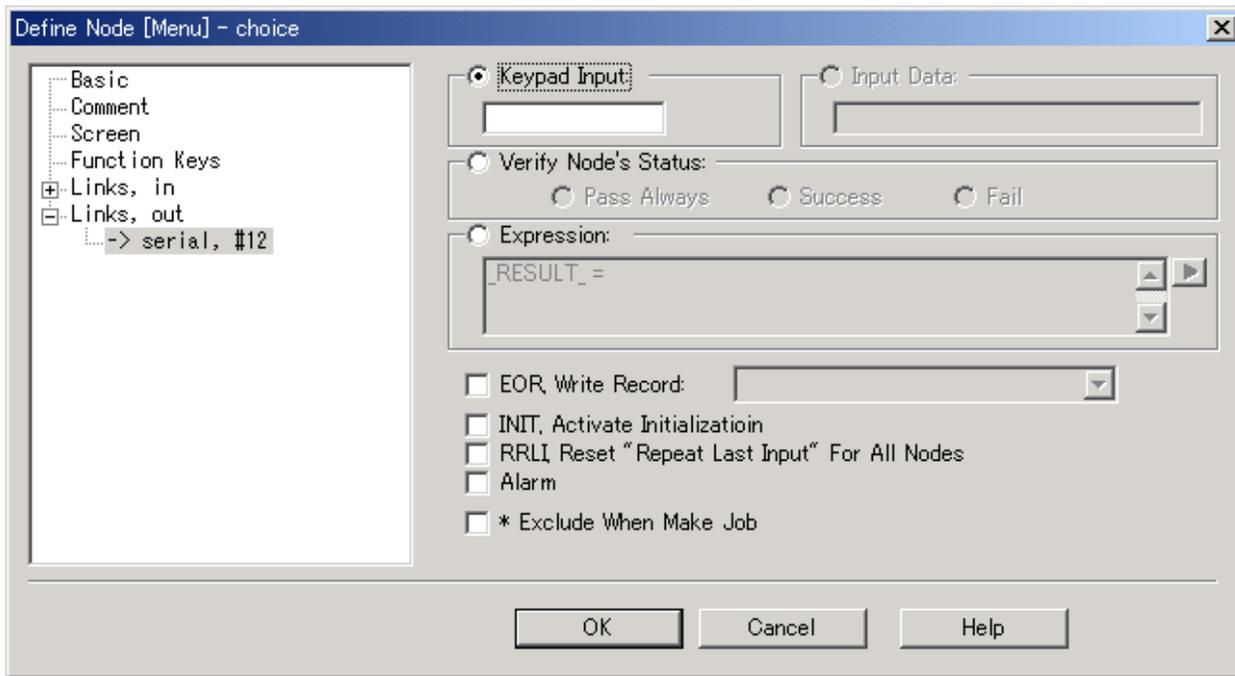
Pass Always オプションはジョブが pNumber ノードの処理を終えたら、choice ノードを処理することを示しています。

39. OK をクリックします。

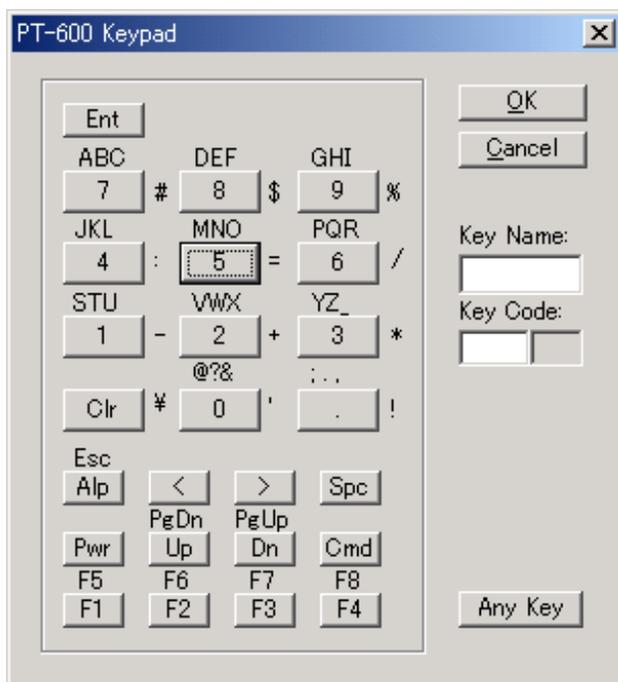
40. マウスポインタを choice ノード に移動し、左マウスボタンを押したままにします。マウスポインタを serial ノード に移動し、左マウスボタンを離します。



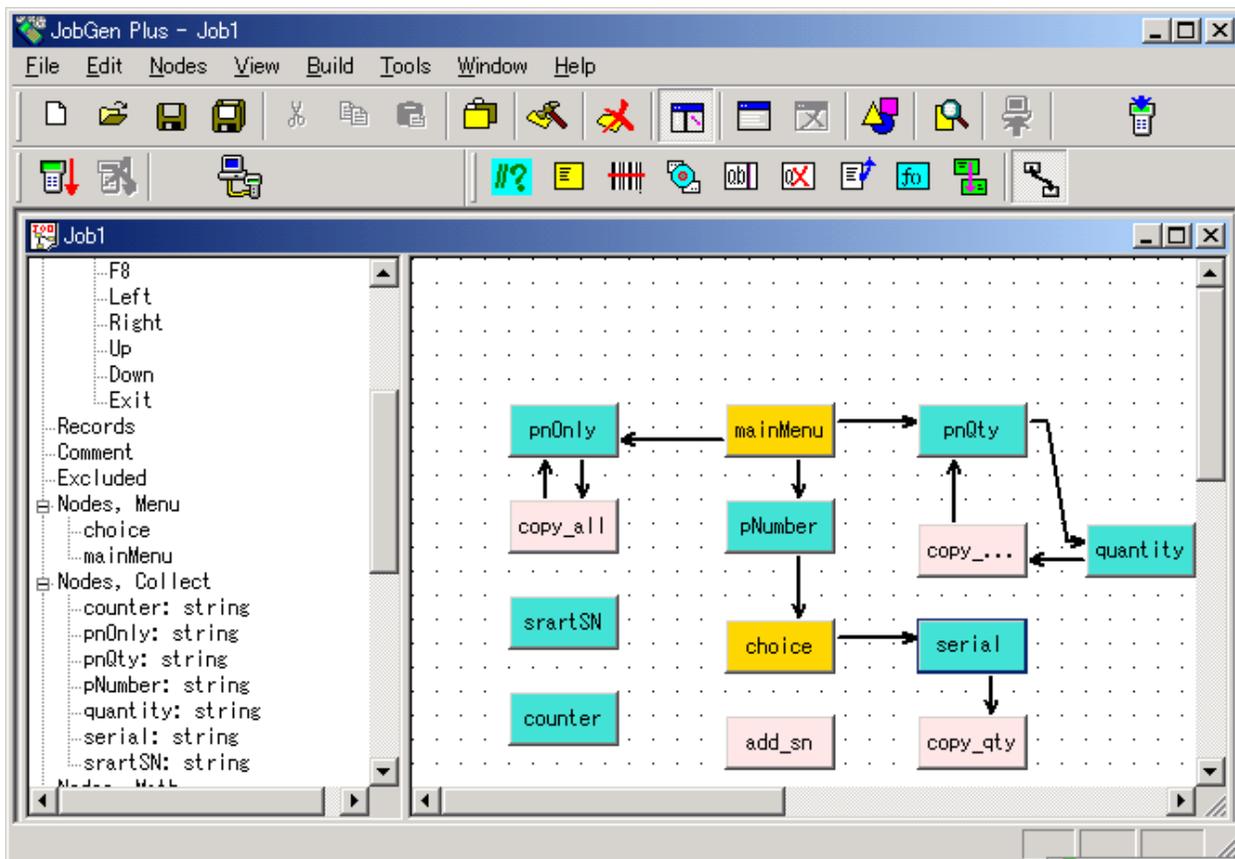
41. マウスポインタが新しく作成したリンク上にある間に左マウスボタン間をダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



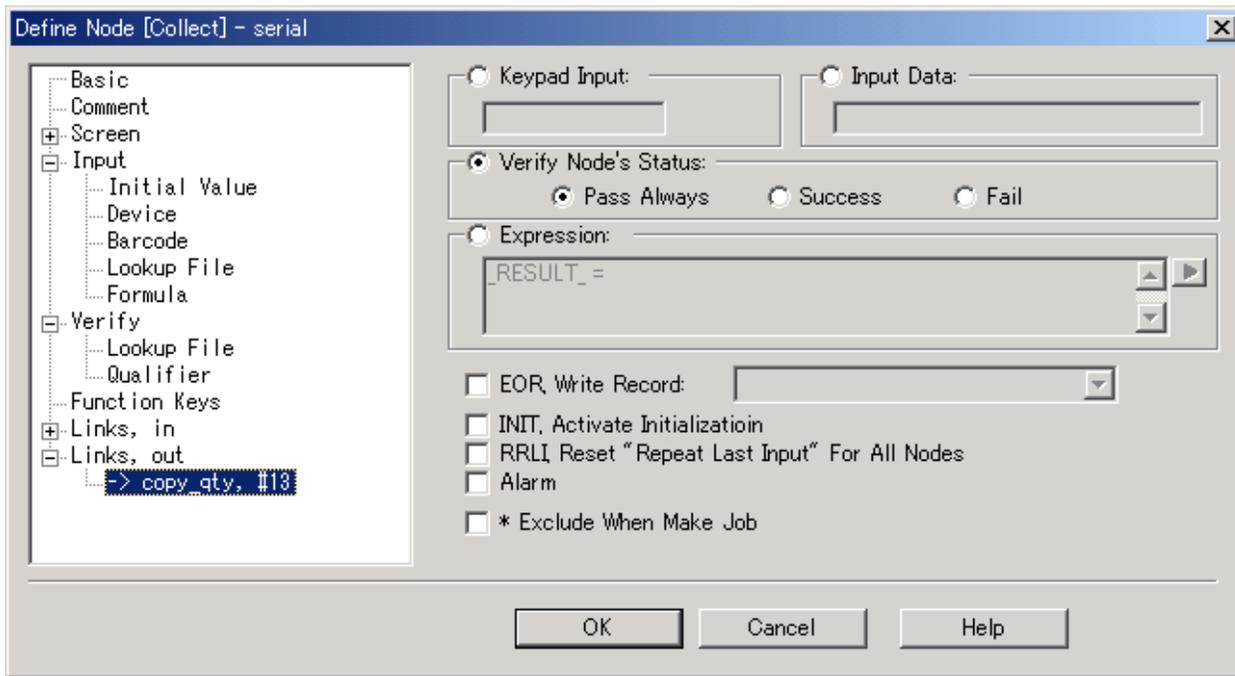
42. **Key Pad Input** オプションをチェックし、そしてマウスポインタが **Key Pad Input** オプションの次のスペース上にある時に左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます。



43. 2 のボタンをクリックします。これは”2”キーが押された場合、プロセスは **choice** ノードから **serial** ノードに移動することを示しています。“PT-600 Keypad” ウィンドウを閉じるために **OK** をクリックします。
44. **Ok** をクリックします。
45. マウスポインタを **serial** ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを **copy_qty** ノードに移動し、左マウスボタンを離します。

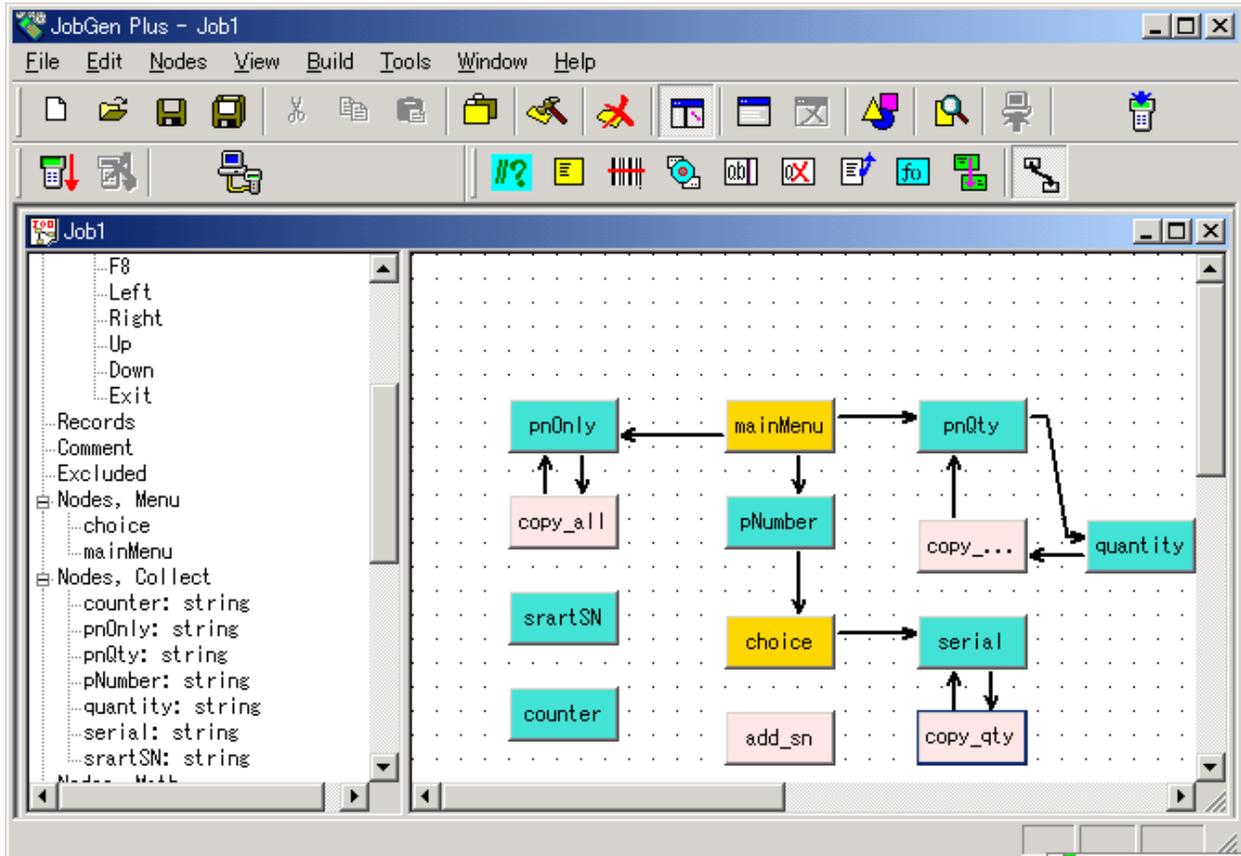


46. 新しく作成したリンク上にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。

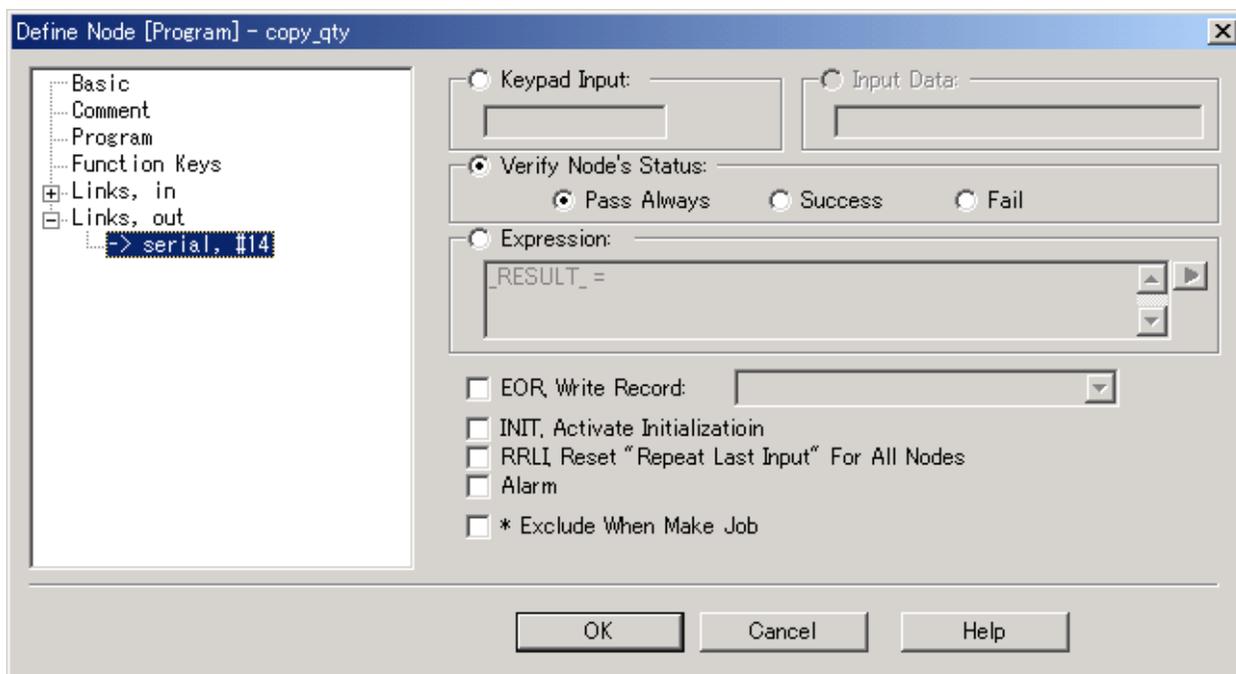


Pass Always オプションは、ジョブが serial ノードの処理を終えたら、copy_qty ノードの処理を行うことを示しています。

47. OK をクリックします。
48. マウスポインタを copy_qty ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを serial ノードに移動し、左マウスボタンを離します。

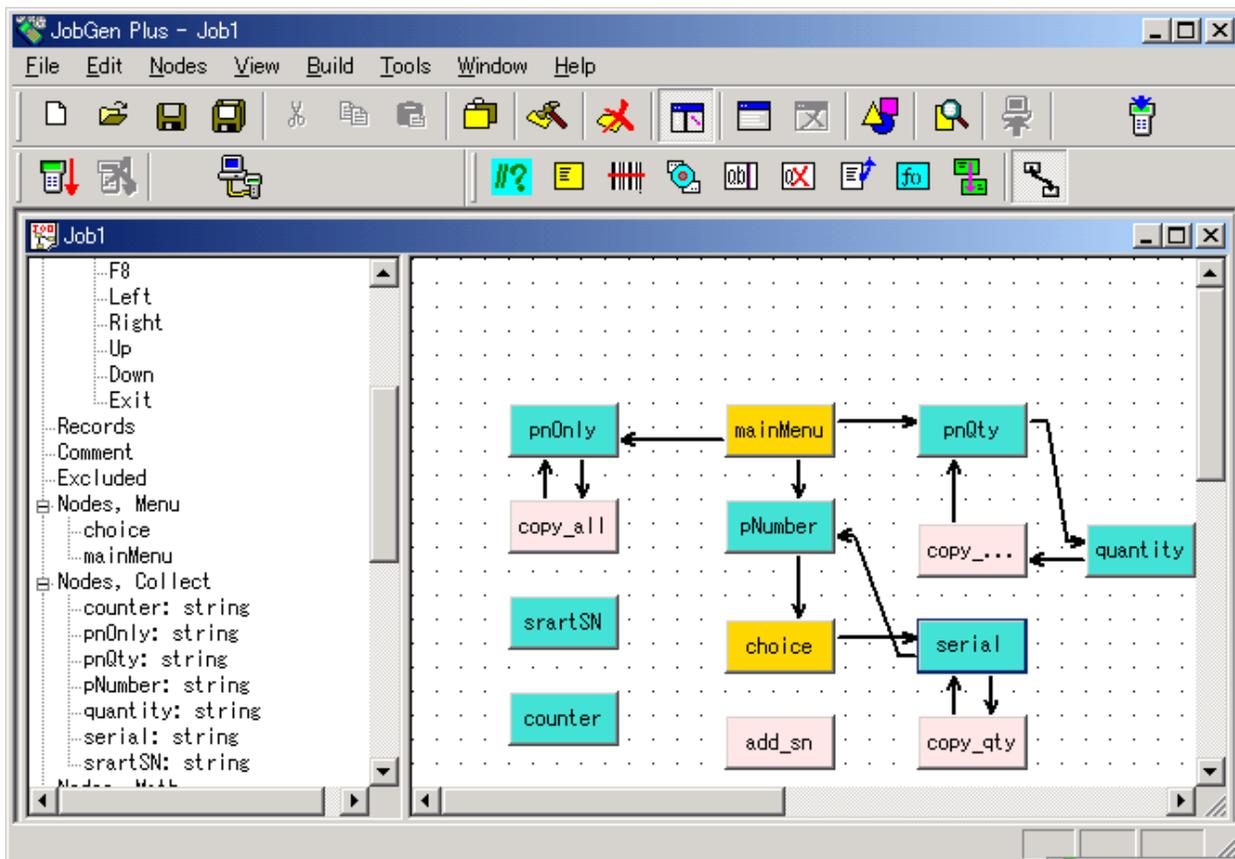


49. 新しく作成したリンク上にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。

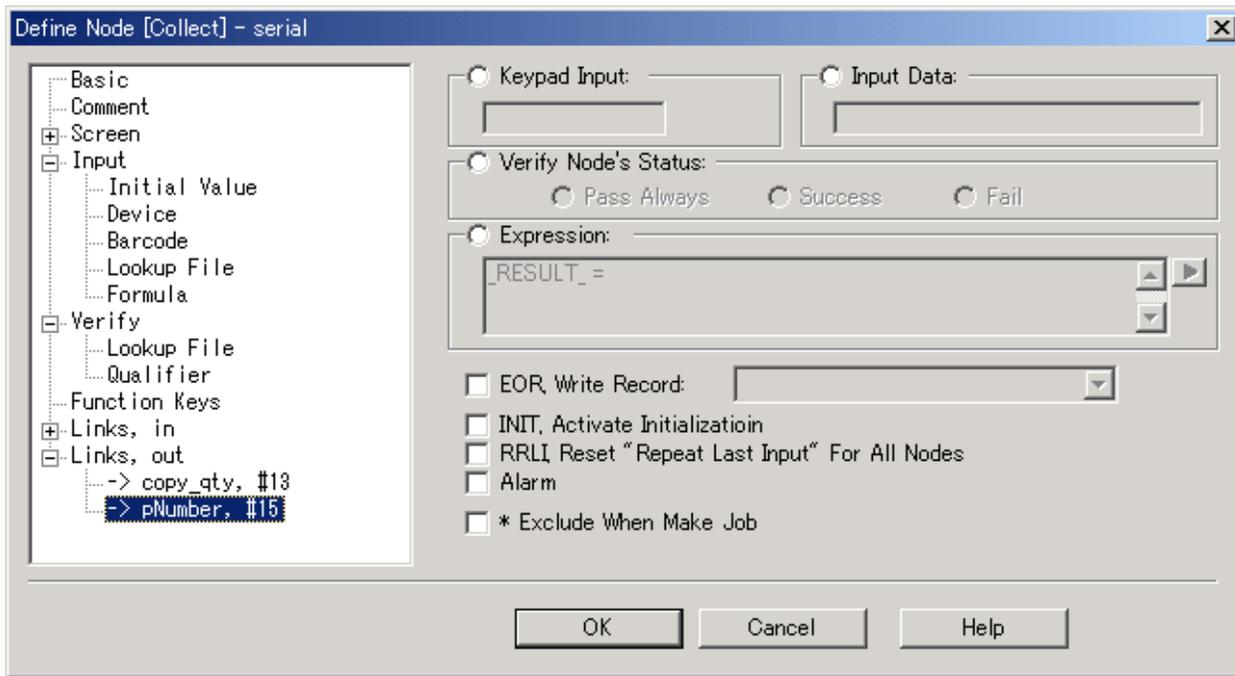


Pass Always オプションは、ジョブが **copy_qty** ノードの処理を終えたら、**serial** ノードの処理を行うことを示しています。

50. **EOR** オプションをチェックします。
51. **Ok** をクリックします。
52. マウスポインタを **serial** ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを **pNumber** ノードに移動して左マウスボタンを離します。



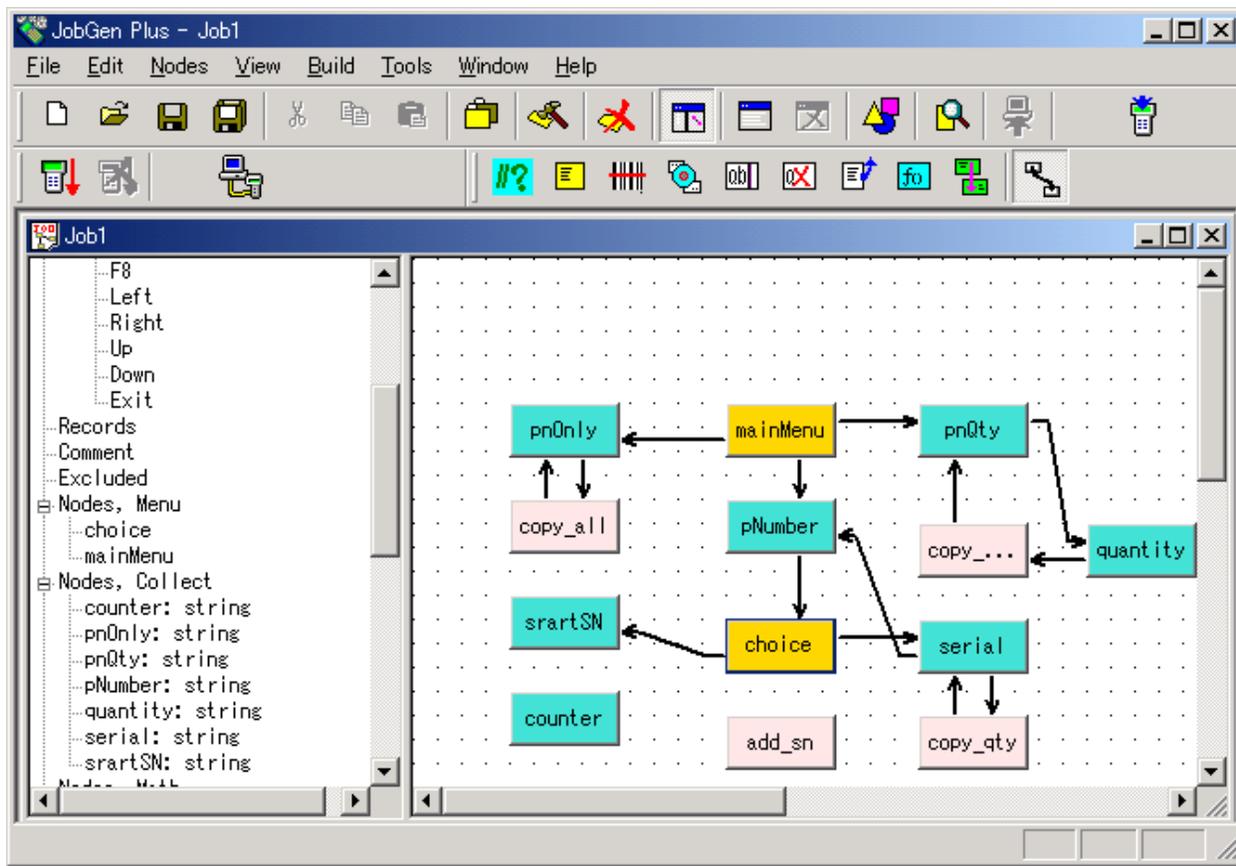
53. マウスポインタが新しく作成したリンク上にある間に、左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



54. **RRLI, Reset “Repeat Last Input” For All Nodes** オプションをチェックします。このオプションは、**pNumber** ノード に新しいデータを入力する場合に必要です。
Keypad Input オプションをチェックし、そしてマウスポインタが **Keypad Input** オプションの次のスペースにある間に左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます。

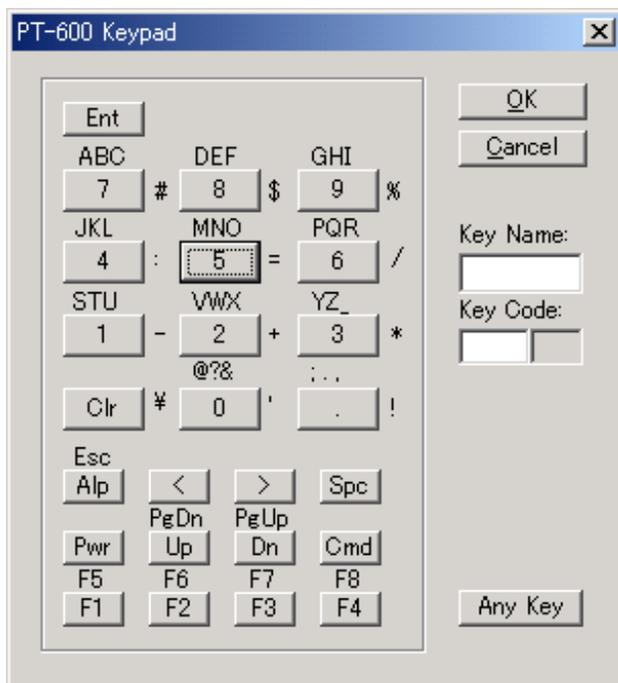


55. **F4** ボタンをクリックします。これはユーザが“**F4**”キーを押した場合に、プロセスは **serial** ノード から **pNumber** ノード に移動することを示します。“PT-600 Keypad” ウィンドウを閉じるために **OK** ボタンをクリックします。
56. **OK** をクリックします。
57. マウスポインタを **choice** ノード へ移動し、左マウスボタンをクリックしたままにします。マウスポインタを **startSN** ノード へ移動し、左マウスボタンを離します。

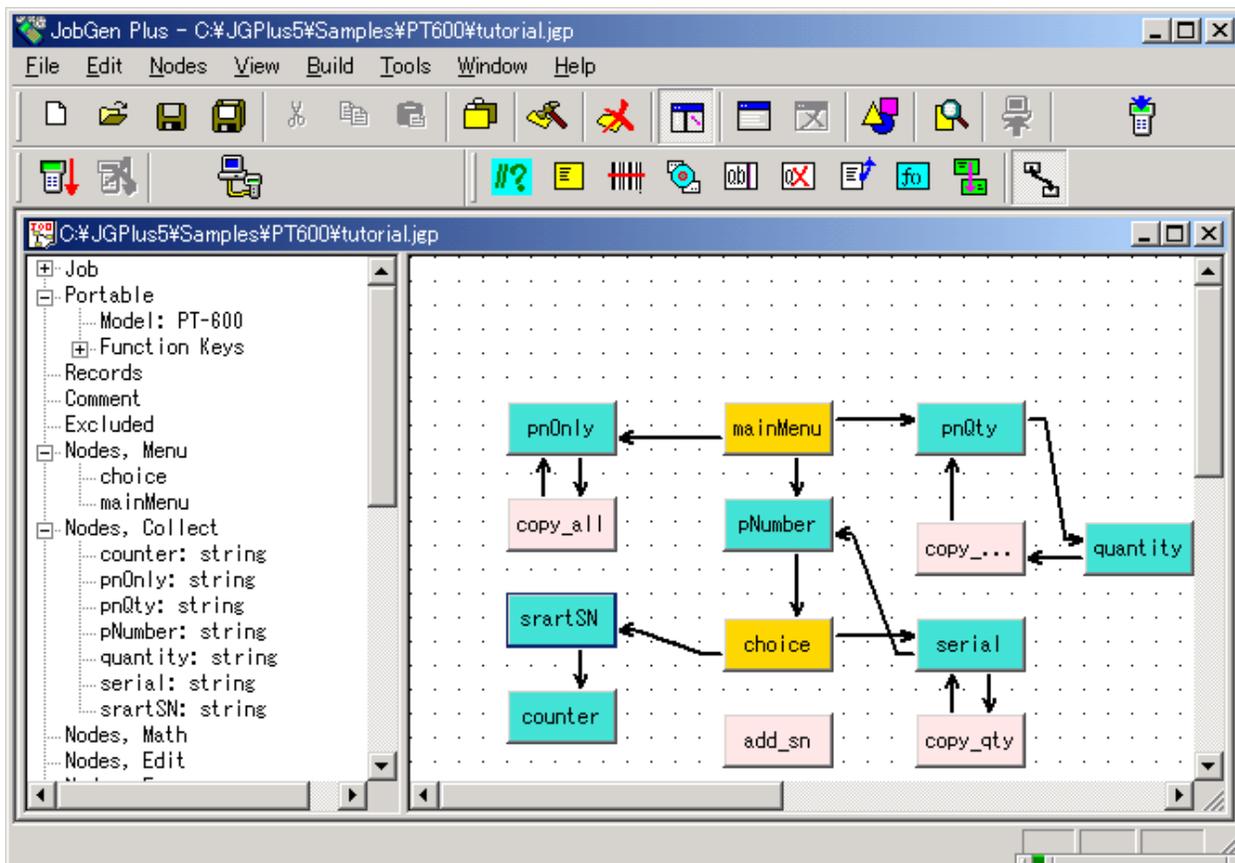


58. 新しく作成したリンクにマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。

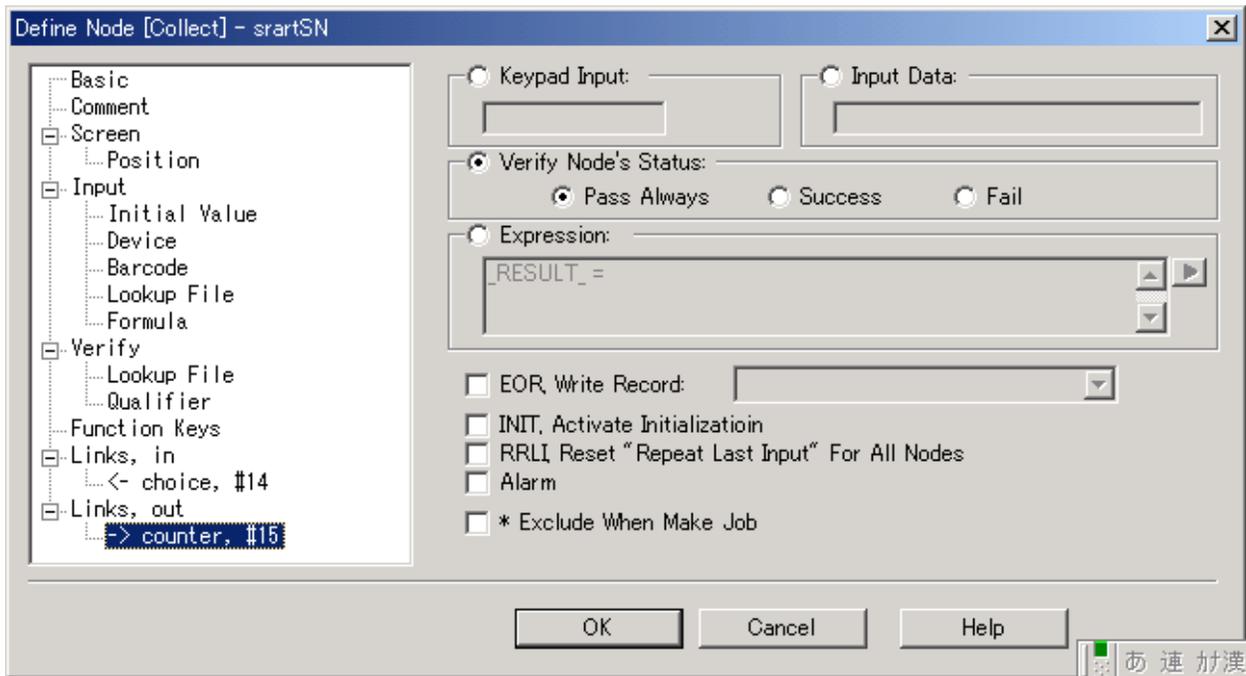
59. **Keypad Input** オプションをチェックし、**Keypad Input** オプションの次のスペースにマウスポインタがある間に左マウスボタンをクリックします。“PT-600 Keypad” ウィンドウがスクリーンに現れます。



60. 1 のボタンをクリックします。これはユーザが”1”のキーを押したら、プロセスはユーザが”1”のキーを押した場合、プロセスは **choice** ノード から **startSN** ノード へ移動することを示しています。“PT-600 Keypad” ウィンドウを閉じるために **OK** ボタンをクリックして下さい。
61. **OK** をクリックして下さい。
62. マウスポインタを **startSN** ノード に移動し、左マウスボタンをクリックします。マウスポインタを **counter** ノード に移動し、左マウスボタンを離します。



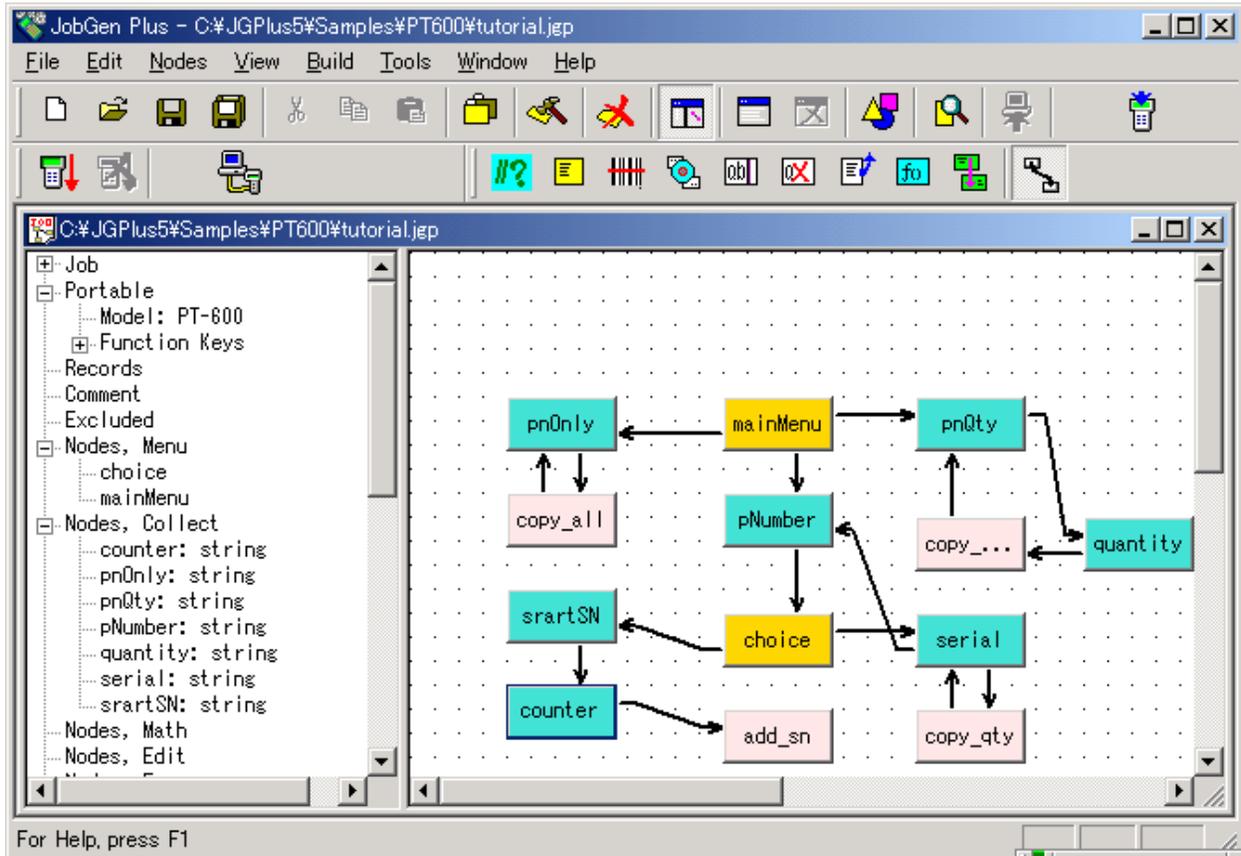
63. マウスポインタが新しく作成したリンク上にある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



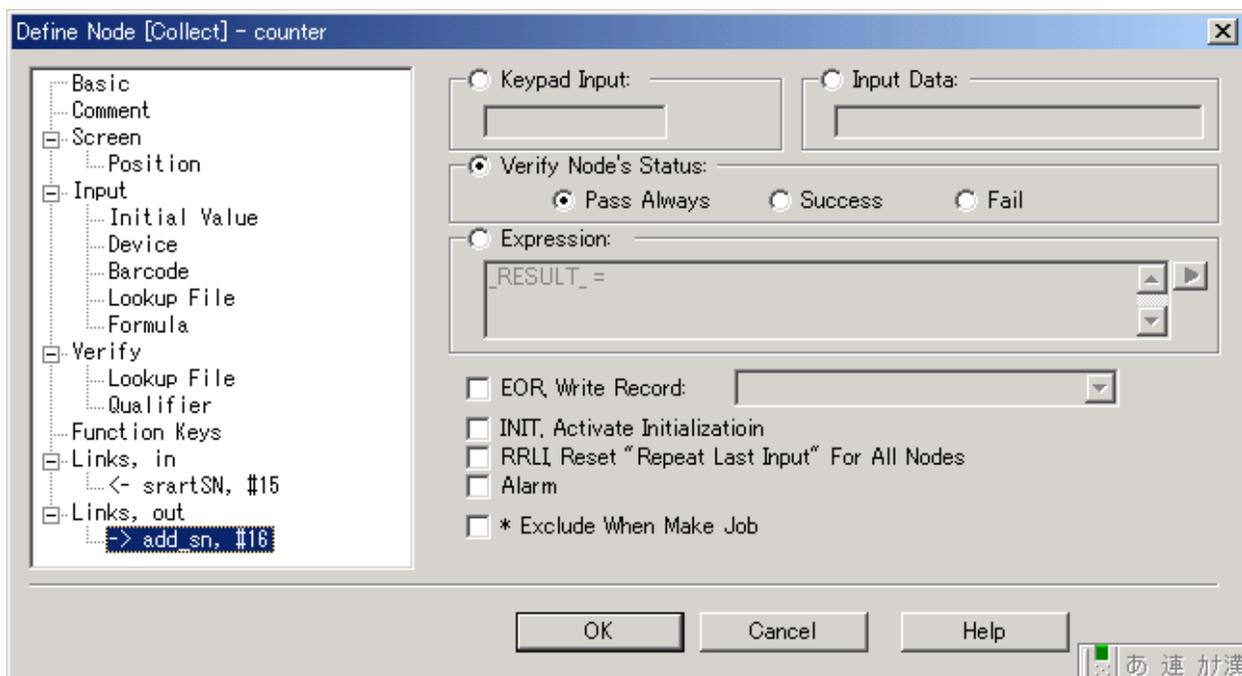
Pass Always オプションはジョブが startSN ノードの処理を完了したら、counter ノードの処理に進むことを示しています。

64. Ok をクリックして下さい。

65. マウスポインタを counter ノードに移動し、左マウスボタンをクリックしたままにします。マウスポインタを add_sn ノードに移動し、左マウスボタンを離します。



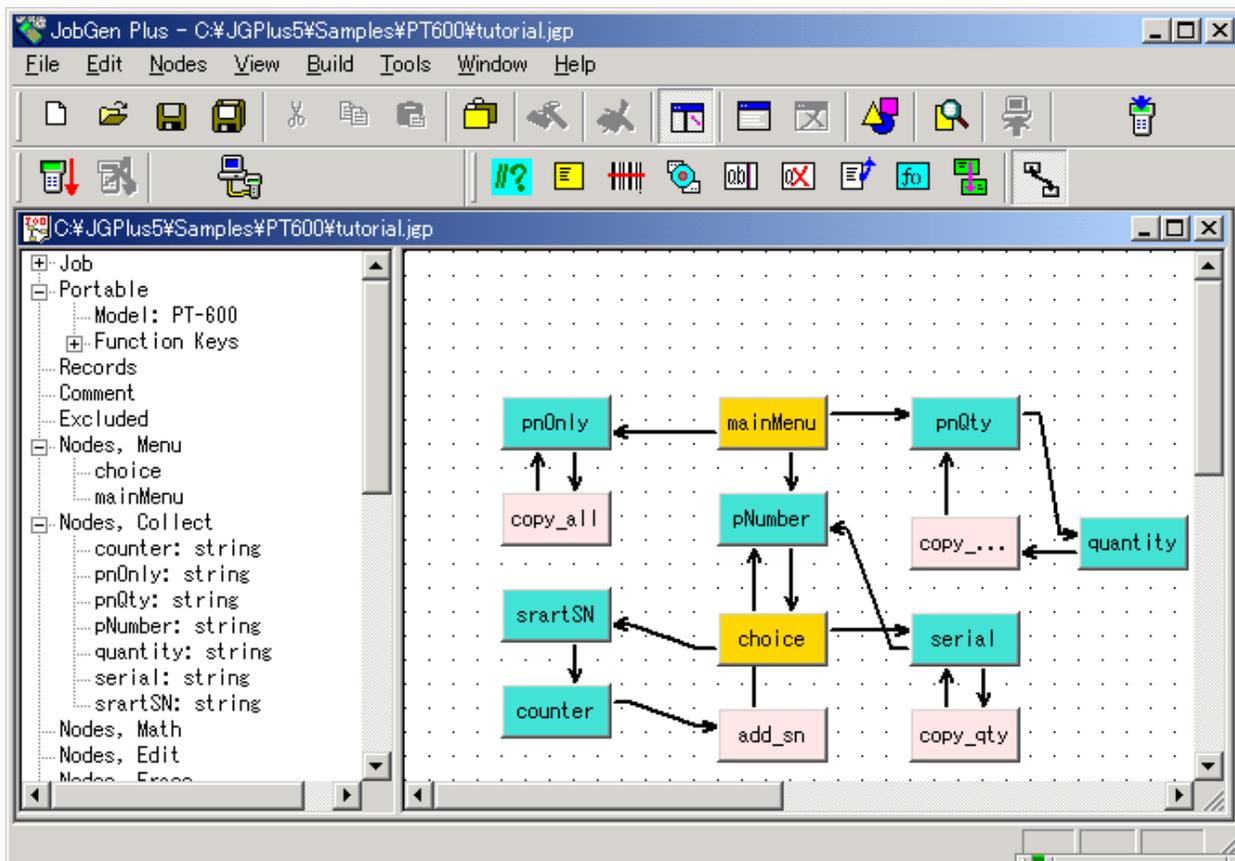
66. マウスポインタが新しく作成したリンク上にある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



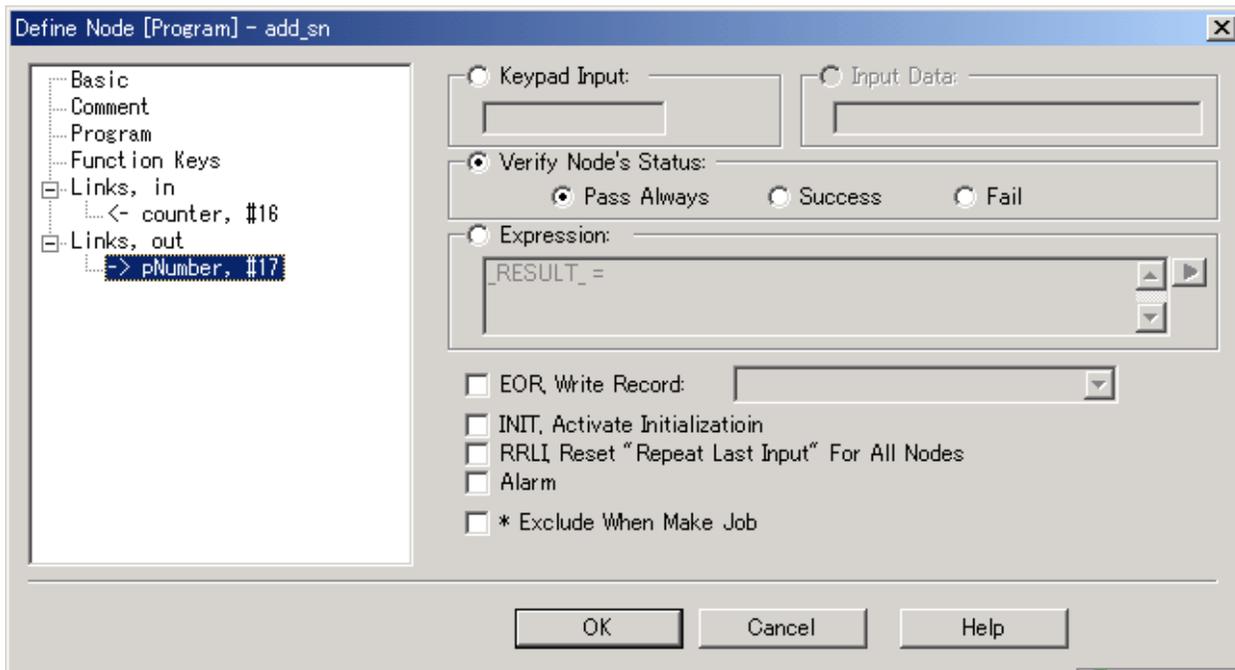
Pass Always オプションはジョブが counter ノードの処理を終えたら add_sn ノードの処理に進むことを示しています。

67. Ok をクリックします。

68. マウスカーソルを add_sn ノード に移動し、マウスボタンをクリックしたままにします。マウスポインタを pNumber ノード に移動し、左マウスボタンを離します。



69. 新しく作成したリンク上にマウスポインタがある間に左マウスボタンをダブルクリックします。“Define Node” ウィンドウがスクリーンに現れます。



Pass Always オプションはジョブが the **add_sn** ノード の処理を終えたら **pNumber** ノードの処理に進むことを示しています。

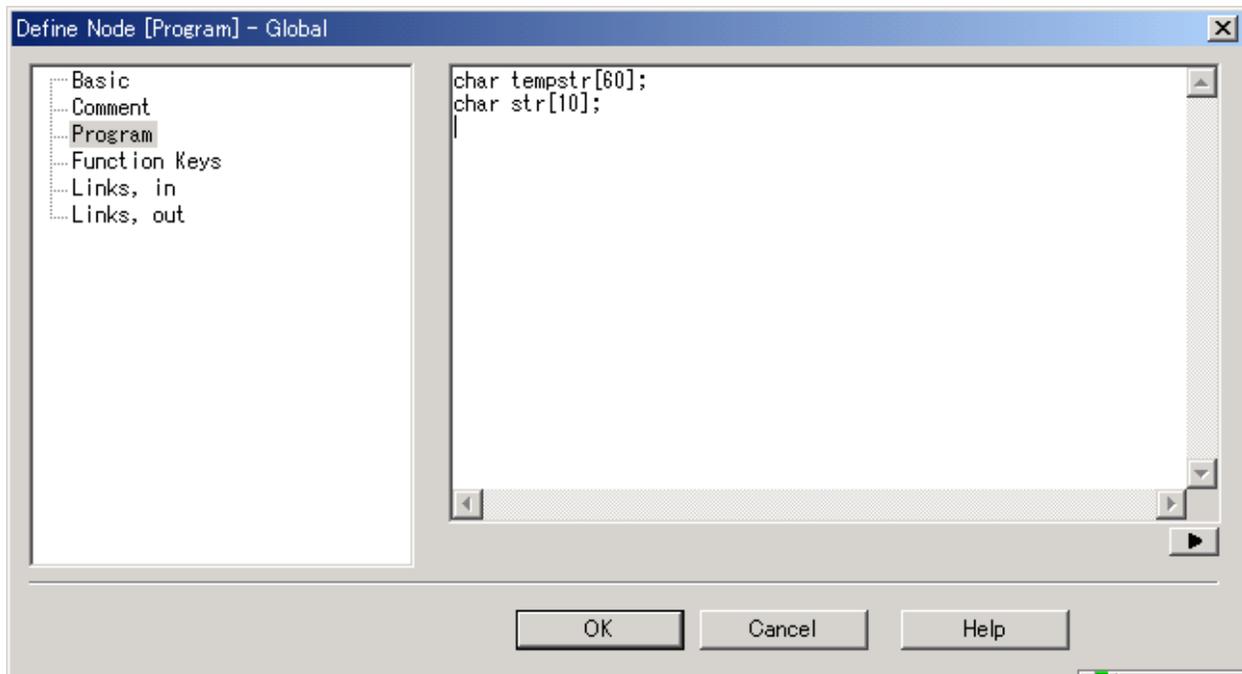
70. **RRLI** オプションをチェックします。

71. マウスポインタを **Program** ノード ボタンに移動します。ポインタが **Program** ノード ボタンにある間に左マウスボタンをクリックします。適当な場所にマウスポインタを移動します。左マウスボタンをダブルクリックします。 **Program** ノード が作られます。

Program ノードを左マウスボタンでダブルクリックします。 **Basic** をクリックし、名前を **Global** に変更します。 **Global Access** をチェックします。

(a) **Program** をクリックします。

(b) “**Define Node [Program] - Global**” がスクリーンに現れます。



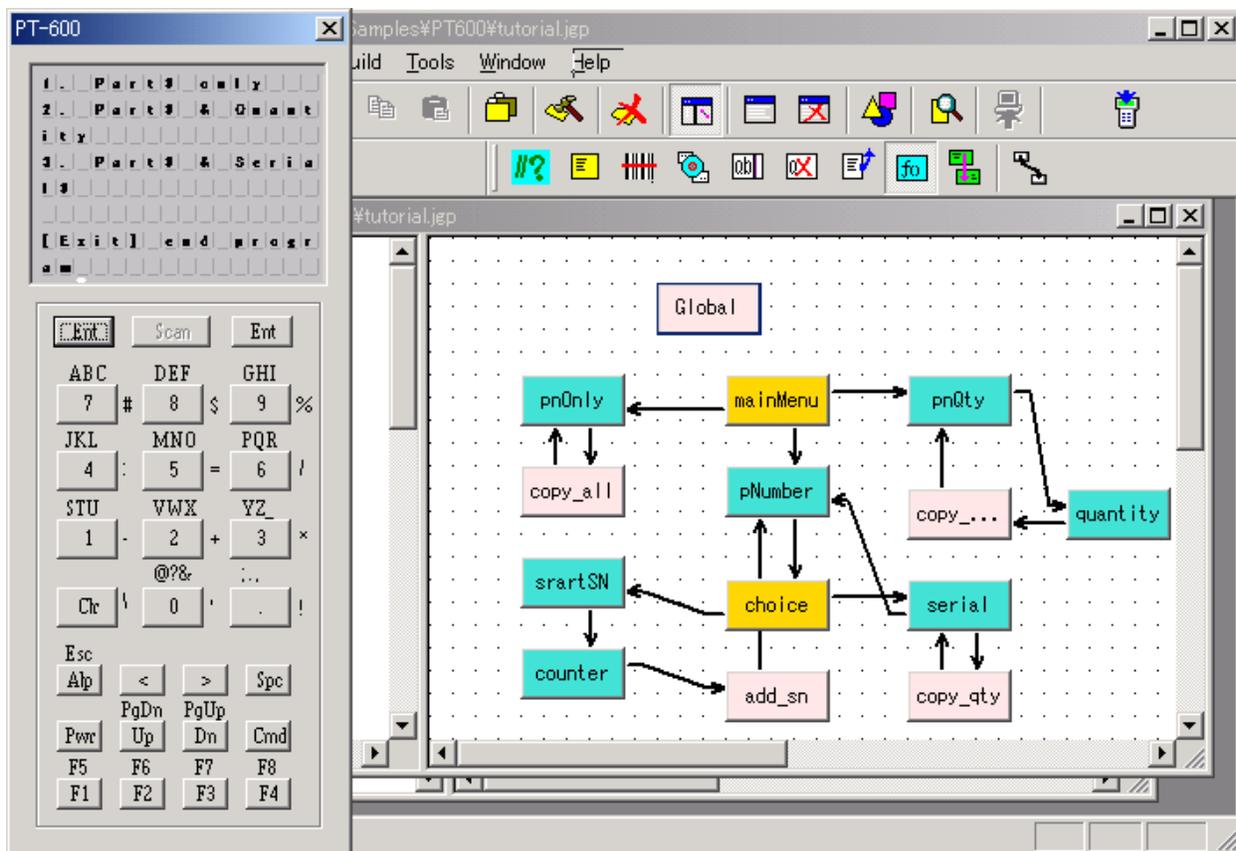
(c) 右のスペースの中にグローバル変数を指定します。

```
char tempstr[60];
```

```
char str[10];
```

72. **Ok** をクリックし、 **invent.jo** としてジョブを保存します。 .

73. ジョブを保存した後で、マウスポインタをツールバーの **Simulate Job** または **Make Job** ボタンに移動します。この在庫管理のジョブをコンパイルするために左マウスボタンをクリックします。実行可能なファイルを作るプロセスが始まります。



EXE ファイルができた後で、JobGen Plus は **Make Job** あるいは **Simulate Job** オプションのいずれを選択したかによって、ファイルをポータブル・ターミナルにダウンロードするか、あるいは PC のスクリーンに結果を表示するためにシミュレーションを行います。