

魚群探知機による魚礁効果調査に関する新たな手法の提案

森口朗彦・高木儀昌

Proposal of New Technique for Artificial Reef Investigation Using Fish Finder

Akihiko MORIGUCHI and Norimasa TAKAGI

Abstract: According to the conventional planning guideline, when fish shoal around an artificial reef is investigated using a fish finder, it is stipulated that data should be collected in the 8 radiating directions from the artificial reef. However, steering a boat in such a way is difficult in actual sea areas with strong wind or ocean current. In addition, it is difficult to estimate whole fish quantity from such radiating survey data.

When there are finely divided grid data, grasping whole fish quantity is easy by integrating these data. Even though a program software to convert random data to plane field data has appeared recently, when information volume is relatively large such as fish finding data, an apparatus with large amount of recording capacity is required.

The authors have established a new technique to input the data from a fish detector with a portable D-GPS to a note-type personal computer, and easily obtain grid data even on the sea using a system composing of a program language attached to a commercially available spread sheet software. The verification of the operation was carried out using the research boat "Takamaru" of our laboratory. The mobility a middle-sized research boat possesses, advanced electronic equipment of "Takamaru," and excellent steering technique have greatly contributed to the success in the verification work.

Keywords: *artificial reef, fish finder, grid data*

1 はじめに

魚群探知機（以下、魚探）で魚礁における魚群分布を調査する場合には、魚礁を中心に放射線状に航走する方法がよく用いられているが¹⁾、風や流れのある実海域で正確に操船するのは難しい。また、このような放射状のデータから全体量を推算するものまた難しい作業である。

データが細かな格子状にあれば、その積分によって全体量の把握は容易である。最近はランダムなデータから格子状データに変換するプログラムソフトもあるが、魚探データのように情報量が比較的大きい場合、データ取得時に大きな容量の記録機器が必要となる。またデータ取得を行いながら即時に3次元で図化し、状況を把握しながらさらに詳細あるいは広域な調査の実施が理想的な調査方法であり、そのためにも調査対象魚礁を中心とした格子状データを取得することが望ましい。

著者らは、携行型のD-GPS付き魚探からのデータをノートパソコンに取り込み、市販の表計算ソフトに添付されたプログラム言語を使用した解析システムの構築と、海上でも格子状のデータを容易に取得する手法の開発に取り組んでいる。最終的には解析結果の即時図化までを考えているが、本報では、現時点で完成している格子状データ取得について報告する。

もとより、移動する可能性のある魚群量を魚探からの時間差のある水深方向のみの分布データから推算するのは、たとえ魚礁周辺では移動量が小さいとはいえ、ある程度の誤差を生じることは言うまでもない。これは魚探を用いた魚礁調査ではどのような調査手法を用いたとしても生じうるものであり、当然ここで提案した手法でも例外ではない。この誤差をいかに小さくするかは調査手順の問題であり、今後データ取得作業を重ねることで明らかにする必要がある。

なお、格子状データの取得に関する動作の検証は当研究所

調査船「たか丸」で行った。調査船「たか丸」乗組員の優秀な操船技術と大型調査船にはない機動性及びARPAレーダーやカラープロッタなど高度な電子機器があつて可能となつた検証作業であつたことを付記する。

2 方法

2.1 基本設計

魚探データ取得対象海域（以下、「探索範囲」）は各辺がが正方位を向いた方形とする。また格子も等間隔に設定し、このため格子点を中心とした碁盤目状の区画（以下、「分割区画」）も方形となる。格子点数は魚礁位置を中心とできるよよう奇数個設定し、各々の格子点位置で魚探から得られる鉛直方向の魚群反応データを取得する。ただし、格子点位置直上を通過するように航走するのは困難であるため、格子点を中心としたある範囲（以下、「データ取得範囲」）を設定し、この範囲内に船が入ったとき取得した魚探データをその格子点でのデータとする。範囲内で複数回のデータが取得された場

合には格子点に近い距離で取得されたデータを採用することとする。

データが取得された分割区画はその時点で明示されるようにし、船は探索範囲を縦横に移動してマス目を塗りつぶしていくように調査を行う。全部あるいは必要十分な程度にマス目が塗りつぶされた時点で調査を終了する（参照，図1）。

2.2 機器構成

魚探は携行型のD-GPS付き魚探，記録器としてはノート型パソコンを用い，通常魚礁調査に用いる5トン程度の小型漁船でも使用できるようにした。今回用いた魚探は本多電子株式会社製HE-6721DGP(測深器として購入した場合の型式名はHFD-100)である。本体及びDGPSセンサー，ケーブル類はすべて中型の金属製アタッシュケース内に収まり，魚探センサー及び舷側への取り付け金具も小型のものである。10～40Vのバッテリーで駆動することができる。出力用コネクタよりデータが常時出力されるので，付属のRS232Cケーブルでパソコンのシリアル端子に接続した。パソコンはマイクロ

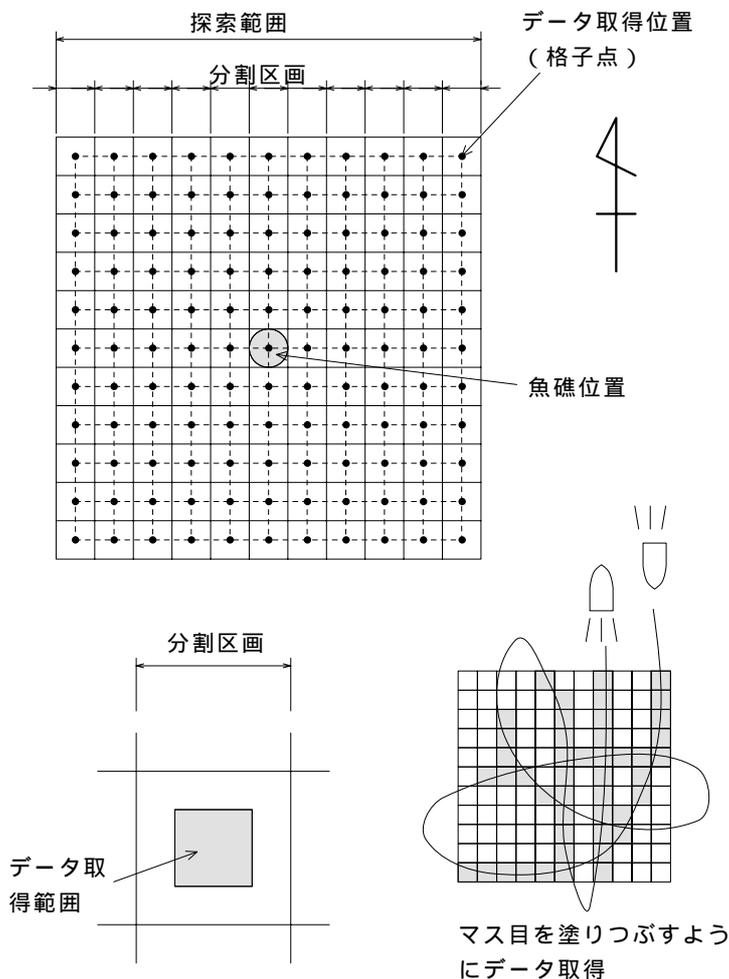


図1 調査海域の設定

ソフト社のエクセル2000以降（以下「エクセル」）が作動できる状態とし、その機能を用いてデータ取得と保存・管理をすることとした。

3 結果

3.1 作成

基本設計に基づいたデータの取得操作、データの読み込み、管理・保存はエクセルに付属するプログラミング言語であるVBAの初歩の機能を用い実現した。以下に用いるエクセルに関する用語は当該ソフト内の表現に準じている。

シリアルポートからの魚探データをエクセルにデータとして読み込む部分については文献2)の方法を用いた。

あらかじめ与える設定条件は 探査区画中心の緯度， 同経度， 格子点間距離， 1辺の格子数， 調査海域における緯度経度1分に対応する距離及び 分割区画とデータ取得範囲の辺長の比率 (< 1) である。一辺の格子数はエクセルの仕様から最大241個となるが，魚礁調査では十分であろう。一つのワークシートで設定条件を与えるようにした（参照，図2）。

魚探からは，表1に示した5種類の信号が出力される。このうちエコーデータ出力個数は300個と214個が選択できるようになっているが，エクセルの仕様の関係から後者に設定した。出力の周期は 位置に関する信号， 船速に関する信号が約4秒毎， 水深に関する信号， 水温に関する信号及び 魚探エコーに関する信号は水深が20m程度の場合で約0.8秒である。水深及びエコーデータは常に最新のものに更新しておき，緯度経度がデータ取得範囲に入ったとき，その直前のデータの必要な部分を選択しワークシートのセルに書き込むことにした。一組のデータセットの構成は 南北方向格子番号（中心が0，北方向が+，南方向が-） 東西方向格



図2 条件設定ワークシート

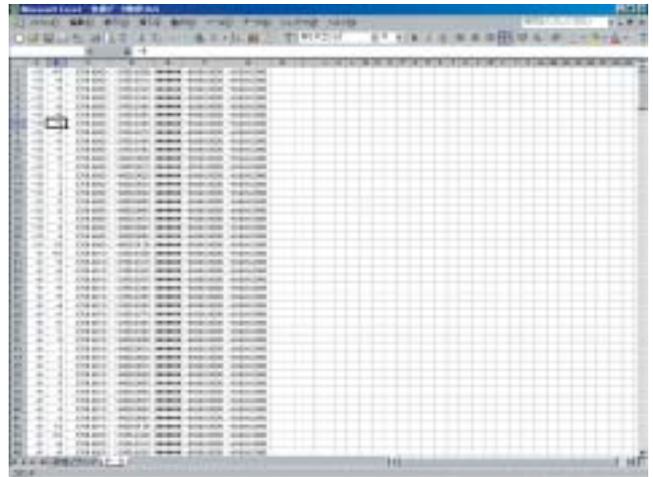


図3 データワークシート（初期状態）

表1 魚探出力信号一覧

| | |
|---|----------------------|
| 水深に関する信号 | |
| 出力形式：\$GPDBT,000,0,F,0123.4,M,0000.0F <CR> <LF> | → 水深データ（固定長）123.4m |
| 水温に関する信号 | |
| 出力形式：\$GPMTW,0023.4,C <CR> <LF> | |
| 位置に関する信号 | |
| 出力形式：\$GPGGA,013641,3442.9695,N,13726.9195,E,1,08,00.85,000000.0,M,0000.0,M <CR> <LF> | |
| | → 経度データ 137° 26.9195 |
| | → 緯度データ 34° 42.9695 |
| 船速に関する信号 | |
| 出力形式：\$GPDBT,000,0,F,0123.4,M,0000.0F <CR> <LF> | |
| 魚探エコーに関する信号 | |
| 出力形式：\$ECHO,0000,0008,m,FFFEDC.....00568000 <CR> <LF> | |
| | → エコーデータ |
| | → ENDスケール |
| | → STARTスケール |

子番号（中心が0，東方向が+，西方向が-），格子位置の緯度，格子位置の経度，データ取得時間，データ取得位置の緯度，データ取得位置の経度，水深，STARTスケール，ENDスケール，以降がエコーデータとした。このうち～の値及びには初期値として"####"を書き込んでおき，以前に同格子点でデータ取得が行われたかどうかをを読み込んで判断するようにした。が数値でなければ初めてデータ取得がなされる格子であるので以降のデータをそのまま書き込むが，が数値であった場合，及びを読み込み格子点位置の緯度経度からの距離を計算して，前回より近ければ更新する。データのワークシートは条件設定のものとは別に設けた（参照，図3）。

さらにもう一枚ワークシートを設け，探索範囲及び分割区画を図示し，これらと調査船舶の位置関係及び各区画でデータ取得が行われたかどうかを即時に示すようにした。まず，分割区画数に対応した数のセル群を方形になる変形してセル内を水色にし，碁盤目状になったセル群の上下に東西方向格子番号，左右に南北方向格子番号，その外側には探索範囲の中心及び東西南北端の緯度経度を，さらに最上部には船舶の現在位置を表示するようにした（参照，図4）。探索範囲と船舶位置との関係はセルを黄色にすることで示した。探索範囲内であれば各分割区画を示すセルが黄色なり，その区画に船舶が入っていることを表す。探索範囲外であればその外縁のセルを黄色になり，船舶がどの方向にずれているのかを示す。例えば東西方向があっけていて北方向にずれていれ探索範囲を示す船舶の位置する南北線上セル群の上側のセルが黄色になり，そのまま南進すれば目的とする分割区画に進入できることを示す。東西方向，南北方向ともにはずれている場合

は外縁セルの角部が黄色となる。データ取得が行われた分割区画は，対応するセル色を緑色に変えることで示した。

また，これとは別に現在位置の緯度経度を設定画面上に示すプログラムも作成した。

文末にリストを掲載した。多大な修正・改良の余地はあるが，とりあえず設計どおり動作することは次節に示すように確認した。

3.2 動作検証

検証作業は当所調査船「たか丸」で実施した。

本船は，総トン数61トン，全長29.5m，幅5.2m，喫水2.0m，主機関出力1,000PS，航海速度12ノットの中型調査船である。

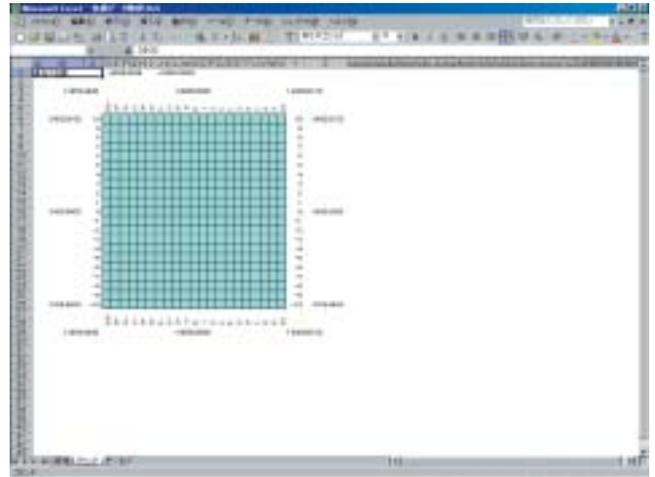


図4 表示ワークシート（初期状態）

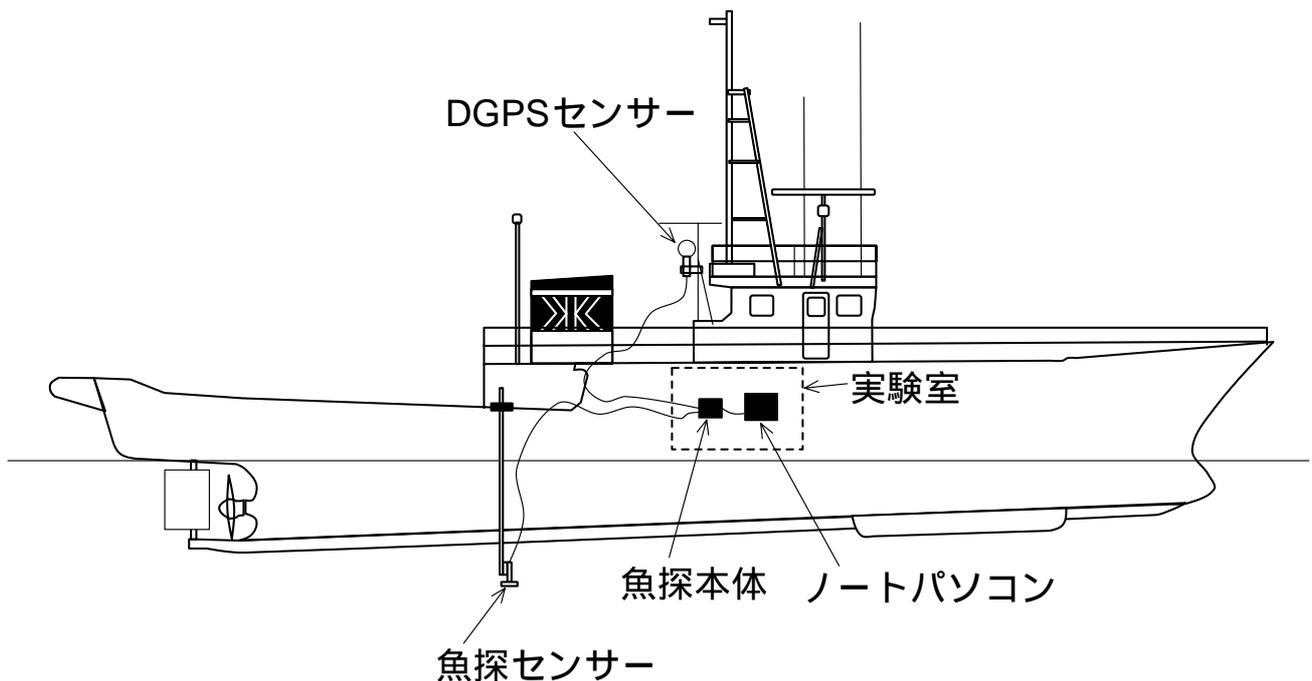


図5 調査船「たか丸」への機器設置位置

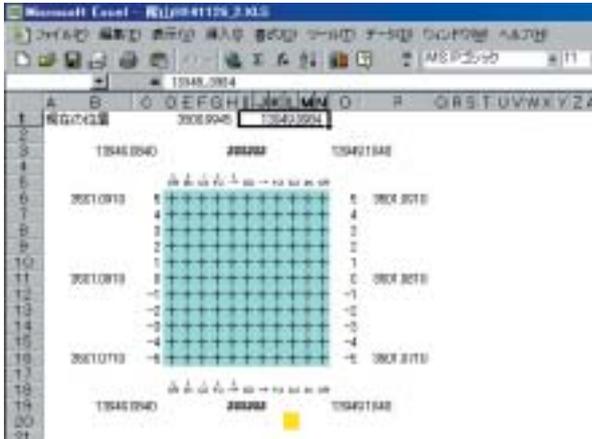


図6 表示ワークシート稼働状況1

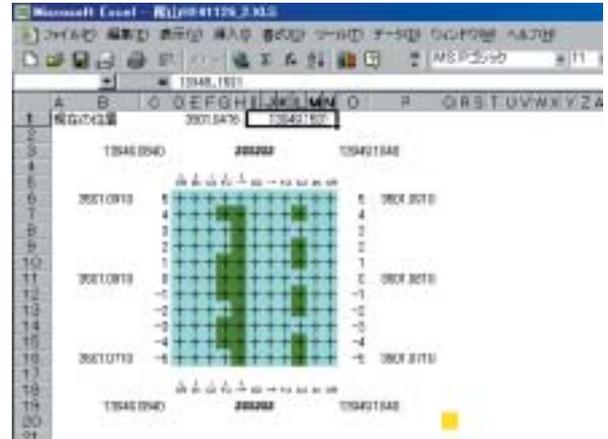


図8 表示ワークシート稼働状況3

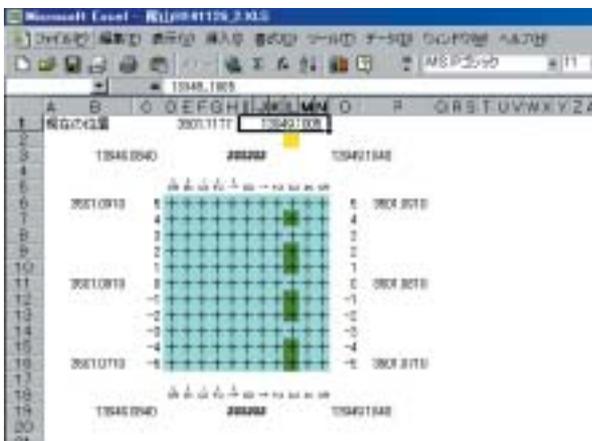


図7 表示ワークシート稼働状況2

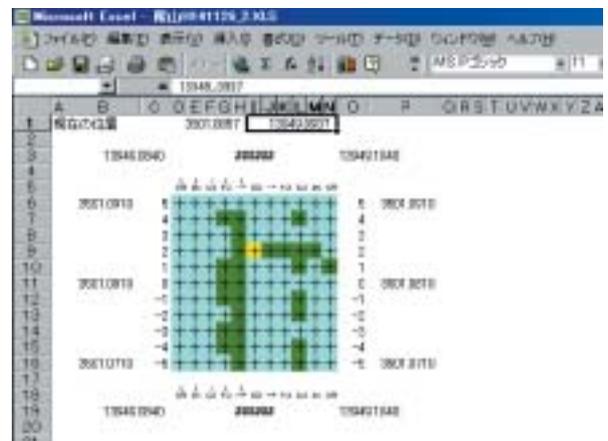


図9 表示ワークシート稼働状況4

魚探操作部及びパソコンは船内の実験室に設置し、DGPSセンサーは操船室後ろのアンテナ支柱に、魚探センサーはデッキ側壁に固定した(参照, 図5)。操船室と実験室の連絡には船内放送施設を用いた。実験室からは東西方向あるいは南北方向の位置を指示し、その後は船の航法機器を用いて航走する。

実施海域は千葉県館山市地先の天然礁海域で、まず天然礁位置の緯度経度を測定し、中心位置を設定した。分割区画数は東西及び南北方向に各13、分割区画の一边の長さは0.005分、データ取得範囲との比率は大きめに0.8とした。したがって探索範囲はおおむね100m×100m、分割区画の一边は8.3m、データ取得範囲のそれは6.6mとなる

第1回目は探索は、位置を探索範囲の南側に移動し、東西方向を第3測線に合わせて最小船速で北上した。(参照, 図7, 8)。図6が探索範囲に入る前の表示で図7が通過後のものである。緑に変わっているセルが示す分割区画でデータの取得ができたことを示している。データが取得できなかったいくつかのものはデータ取得範囲をはずれたまま分割区画を通過してしまったものである。当日は波・風ともにやや強く海流もあり、この条件下で先に示した諸元の中型船を幅6

mの範囲でこれだけの区画内を直進させたことは、たか丸の操船技術の優秀さを示すものである。南北方向の探索を3回繰り返した後、船を東西方向、南北方向ともにはずれている位置に移動し(参照, 図8)、南北方向を第3測線に合わせて西進させた(参照, 図9)。南北方向に流れが強かったため、探索範囲に入ってまもなく第2測線にずれてしまったので、そのまま第2測線を維持して進めた。図9は探索範囲に船が入っている状態の時のものである。その後、天候がさらに悪化して途中で作業を打ち切らざるを得ない状況となり、全区画のデータをとるには至らなかった。しかしながら、今後、図化等を検討するために使用する十分なデータは得られたと考えている(参照, 図10)。

4 今後の課題

検証作業を通じて、調査を円滑に行うために必要ないくつかの改良すべき点が見出された。それらを列挙すると、船が探索範囲外にあるとき、双方の距離関係がエクセルの表示では分かりにくい。同様に探索範囲外にあり探索範囲に進入しようとするとき、データ取得範囲に向かっていくかどうか分かりにく

The image shows a screenshot of a Microsoft Excel spreadsheet. The spreadsheet contains a grid of data. The columns are labeled with letters A through Z, and the rows are numbered 1 through 25. The data appears to be organized into several columns, with some cells containing numerical values and others containing text or symbols. The spreadsheet is titled 'Microsoft Excel - 魚山(0112)3.xls'. The status bar at the bottom indicates 'シート1: 魚山(0112)3.xls'.

図10 取得されたデータ

い。

すべての分割区画でデータを取得するまでプログラムが終了しない。

一度プログラムを終了して再度同じ位置で探索を開始しようとすると、データをすべてクリアしてしまい、追加ができない。

これらの改良点は、すべて若干の修正で対処できると考えている。例えば については探索範囲外にいることを示す外側の黄色になるセルを、船から探索範囲外縁までの距離に応じてある段階で色を変える、あるいは点滅する時間間隔で示すなどが考えられる。後者の方が感覚的に良いかもしれない。

については を点滅方式にした場合は正方位に直進すればデータ取得範囲に入る状況の時に色を変える、そうでない場合はセルに「 」などの記号を書き込むといった方法がある。 に関しては、ある程度の距離だけ探索範囲から離れたらプログラムを終了するような方法を考えている。 は設定ワークシートのどこかにその条件での探索の履歴を書き込んでおき、それを参照しながらプログラム開始時点でデータを追加するかクリアするかを訪ねるようにすればよい。ただし設定条件を変えれば探索開始に当たって当然データはクリアしなければならず、その判断をどこでするか、多少の工夫が必要となる。

なお、「今後の大きな課題」はデータを取得しながらそれを3次元に図化する手法の開発である。これには相当の時間

と労力が必要であろう。

5 おわりに

魚礁の効果調査がなかなか行われない理由の一つとして、必要な機材や船の準備手配等が多面で面倒くさいという点がある。また魚探調査を行ってもその後のデータ整理が煩雑であったことも一員と考えられる。今回提案した手法は、5トン未満の小型漁船で対応でき、機材構成もすべて携行型のものよりなっている。また、データもすでに表計算ソフトのデータとなっているので、解析、図下等の作業は容易である。

本手法が普及して魚礁効果調査が積極的に行われることで魚礁の効果が明確になり、魚礁漁場造成が推進され、水産物供給が増大してほしいと願うものである。

参考文献

- 1) 沿岸漁場整備開発事業人工漁礁漁場造成計画指針平成12年度版，(社)全国沿岸漁業振興開発協会，2000
- 2) 木下隆：Easy Comm for Excelの制作と応用，トランジスタ技術2000年6月号，CQ出版株，2001,5

プログラムリスト

```

Sub 魚探データ取得()
Dim answerN As Integer
Dim dataF As String, dataECHO As String
Dim centerID0 As Double, centerKED As Double, pointID0 As Double, pointKED As Double
Dim newID0 As Double, newKED As Double, beforeID0 As Double, beforeKED As Double
Dim inputID0 As Double, inputKED As Double
Dim newDEPTH As Single
Dim numNS As Integer, numEW As Integer, pointNS As Integer, pointEW As Integer
Dim countNOW As Long, numGETDATA As Long
Dim lengNS As Double, lengEW As Double, lengONBdeg As Double
Dim ratioNS As Double, ratioEW As Double
Dim shiftColumn As Integer, shiftRow As Integer
Dim scaleSTR As Long, scaleEND As Long
Dim dummyD As Double
Dim dummyS As String
Dim figDOKED As Boolean, figEHO As Boolean, figDEPTH As Boolean
Dim timeSPEED1 As Single, timeSPEED2 As Single, timeMOVE1 As Single, timeMOVE2 As Single
Dim speedNS As Double, speedEW As Double
Dim indRow As Long, indCol As Long, indRow2 As Long, indCol2 As Long, figRowCol2 As Boolean, colRowCol2 As Variant
Dim i As Integer, j As Integer
Dim numColumn As Long, numRows As Long
'メモ: Cells[RowIndex,ColumnIndex] のRow は行 (縦順) で普通に数字, Column は列 (横順) でAから順に1~256
'開始の確認
answerN = MsgBox("魚探データの取得を始めますか?" & Chr(13) & "設定シートを確認してください。", vbOKCancel + vbQuestion, "開始の確認")
If answerN = 1 Then
'設定データの取得
With Worksheets("設定")
centerID0 = .Range("D4").Value
centerKED = .Range("D5").Value
lengNS = .Range("D6").Value
lengEW = .Range("D6").Value
lengONBdeg = .Range("D8").Value
numNS = .Range("D7").Value - 1
numEW = .Range("D7").Value - 1
ratioNS = .Range("D9").Value
ratioEW = .Range("D9").Value
End With
'グリッド表示シートの準備
shiftColumn = 3
shiftRow = 5
'フォント変更
With Worksheets("グリッド")
With .Cells
.Clear
.Delete Shift:=xlUp
End With
With .Cells.Font
.Name = "MS Pゴシック"
.Size = 8
End With
'行列幅変更
.Columns(shiftColumn - 2).ColumnWidth = 1.125
.Columns(numEW + shiftColumn + 4).ColumnWidth = 1.125
.Columns(shiftColumn).ColumnWidth = 3
.Columns(numEW + shiftColumn + 2).ColumnWidth = 3
.Columns(shiftColumn - 1).ColumnWidth = 8
.Columns(numEW + shiftColumn + 3).ColumnWidth = 8
.Range(.Columns(shiftColumn + 1), .Columns(numEW + shiftColumn + 1)).ColumnWidth = 1.125
.Range(.Rows(shiftRow - 3), .Rows(shiftRow + numNS + 5)).RowHeight = 11.25
.Range(.Columns(shiftColumn + numEW + 5), .Columns(256)).ColumnWidth = 1.125
'セル色変更
'番号入力セル書式設定と入力
With .Range(.Cells(shiftRow + 1, shiftColumn + 1), .Cells(shiftRow + numNS + 1, numEW + shiftColumn + 1))
.Interior.ColorIndex = 8
.Borders(xlDiagonalDown).LineStyle = xlNone
.Borders(xlDiagonalUp).LineStyle = xlNone
With .Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With .Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With

```

```

With .Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With .Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With .Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With .Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
End With
For i = 0 To numBW
    With .Range(.Cells(shiftRow, shiftColumn + 1 + i), .Cells(shiftRow - 1, shiftColumn + 1 + i))
        .VerticalAlignment = xlBottom
        .Orientation = -90
        .MergeCells = True
        .Value = i - numBW / 2
    End With
Next i
For i = 0 To numBW
    With .Range(.Cells(shiftRow + numNS + 2, shiftColumn + 1 + i), .Cells(shiftRow + numNS + 3, shiftColumn + 1 +
1)
        .VerticalAlignment = xlBottom
        .Orientation = -90
        .MergeCells = True
        .Value = i - numBW / 2
    End With
Next i
For i = 0 To numNS
    .Cells(shiftRow + i + 1, shiftColumn).Value = numNS / 2 - i
Next i
For j = 0 To numNS
    .Cells(shiftRow + i + 1, shiftColumn + numBW + 2).Value = numNS / 2 - j
Next j
'緯度経度入力
'経度最西の北側
With .Range(.Cells(shiftRow - 2, shiftColumn - 1), .Cells(shiftRow - 2, shiftColumn))
    .NumberFormatLocal = "0.0000_"
    .HorizontalAlignment = xlRight
    .MergeCells = True
End With
.Cells(shiftRow - 2, shiftColumn - 1).Value = tas(DOKED)centerKED, -lengBW * numBW / 2)
'経度最西の南側
With .Range(.Cells(shiftRow + numNS + 1 + 3, shiftColumn - 1), .Cells(shiftRow + numNS + 1 + 3, shiftColumn))
    .NumberFormatLocal = "0.0000_"
    .HorizontalAlignment = xlRight
    .MergeCells = True
End With
.Cells(shiftRow + numNS + 1 + 3, shiftColumn - 1).Value = tas(DOKED)(centerKED, -lengBW * numBW / 2)
'経度中央北側
2) With .Range(.Cells(shiftRow - 2, shiftColumn + numBW / 2 + 1 - 2), .Cells(shiftRow - 2, shiftColumn + numBW / 2 + 1 +
    .NumberFormatLocal = "0.0000_"
    .HorizontalAlignment = xlCenter
    .MergeCells = True
End With
.Cells(shiftRow - 2, shiftColumn + numBW / 2 + 1 - 2).Value = centerKED
'経度中央南側
With .Range(.Cells(shiftRow + numNS + 1 + 3, shiftColumn + numBW / 2 + 1 - 2), .Cells(shiftRow + numNS + 1 + 3, shift
Column + numBW / 2 + 1 + 2))
    .NumberFormatLocal = "0.0000_"
    .HorizontalAlignment = xlCenter
    .MergeCells = True
End With
.Cells(shiftRow + numNS + 1 + 3, shiftColumn + numBW / 2 + 1 - 2).Value = centerKED
'経度最東北側
With .Range(.Cells(shiftRow - 2, shiftColumn + numBW + 2), .Cells(shiftRow - 2, shiftColumn + numBW + 1 + 2))
    .NumberFormatLocal = "0.0000_"
    .HorizontalAlignment = xlLeft
    .MergeCells = True

```

```

End With
.Cells(shiftRow - 2, shiftColumn + numEW + 2).Value = tasIDOKED(centerKED, lengEW * numEW / 2)
'経度最東南側
With Range(.Cells(shiftRow + numNS + 1 + 3, shiftColumn + numEW + 2), .Cells(shiftRow + numNS + 1 + 3, shiftColumn
+ numEW + 1 + 2))
.NumberFormatLocal = "0.0000_"
.HorizontalAlignment = xlLeft
.MergeCells = True
End With
.Cells(shiftRow + numNS + 1 + 3, shiftColumn + numEW + 2).Value = tasIDOKED(centerKED, lengEW * numEW / 2)
'緯度はいいど
With .Cells(shiftRow + 1, shiftColumn - 1)
.NumberFormatLocal = "0.0000_"
.Value = centerID0 + lengNS * numNS / 2
End With

With .Cells(shiftRow + numNS / 2 + 1, shiftColumn - 1)
.NumberFormatLocal = "0.0000_"
.Value = centerID0
End With
With .Cells(shiftRow + numNS + 1, shiftColumn - 1)
.NumberFormatLocal = "0.0000_"
.Value = tasIDOKED(centerID0, -lengNS * numNS / 2)
End With
With .Cells(shiftRow + 1, shiftColumn + numEW + 3)
.NumberFormatLocal = "0.0000_"
.Value = tasIDOKED(centerID0, lengNS * numNS / 2)
End With
With .Cells(shiftRow + numNS / 2 + 1, shiftColumn + numEW + 3)
.NumberFormatLocal = "0.0000_"
.Value = centerID0
End With
With .Cells(shiftRow + numNS + 1, shiftColumn + numEW + 3)
.NumberFormatLocal = "0.0000_"
.Value = tasIDOKED(centerID0, -lengNS * numNS / 2)
End With
'現在の緯度経度を書く場所を確保
With .Range("A1:C1")
.MergeCells = True
.FormulaR1C1 = "現在の位置"
.Characters(1, 2).PhoneticCharacters = "ゲンサイ"
.Characters(4, 2).PhoneticCharacters = "イチ"
End With
With .Range("D1:H1")
.NumberFormatLocal = "0.0000_"
.MergeCells = True
.FormulaR1C1 = "-9999.9999"
End With
With .Range("I1:N1")
.NumberFormatLocal = "0.0000_"
.MergeCells = True
End With
.Range("I1:N1").FormulaR1C1 = "-19999.9999"

Range("A1").Select
End With

'データシートの準備
With Worksheets("データ")
With .Cells
.Clear
.Delete Shift:=xlUp
End With
With .Range(.Columns(1), .Columns(2))
.ColumnWidth = 4.75
.NumberFormatLocal = "0_"
.HorizontalAlignment = xlRight
End With
With .Range(.Columns(3), .Columns(4))
.ColumnWidth = 11
.NumberFormatLocal = "0.0000_"
.HorizontalAlignment = xlRight
End With
With .Columns(5)
.ColumnWidth = 7.25
.NumberFormatLocal = "h:mm:ss"
.HorizontalAlignment = xlCenter
End With
With .Range(.Columns(6), .Columns(7))
.ColumnWidth = 11

```

```

        .NumberFormatLocal = "0.0000_ "
        .HorizontalAlignment = xlRight
    End With
    With .Columns(8)
        .ColumnWidth = 5.75
        .NumberFormatLocal = "0.0_ "
        .HorizontalAlignment = xlRight
    End With
    With .Range(.Columns(9), .Columns(10))
        .ColumnWidth = 4.75
        .NumberFormatLocal = "0_ "
        .HorizontalAlignment = xlRight
    End With
    With .Range(.Columns(11), .Columns(256))
        .ColumnWidth = 2.75
        .NumberFormatLocal = "0_ "
        .HorizontalAlignment = xlRight
    End With
    For i = 0 To numNS
        For j = 0 To numEW
            .Cells((numEW + 1) * i + j + 1, 1).Value = i - numNS / 2
            .Cells((numEW + 1) * i + j + 1, 2).Value = j - numEW / 2
            .Cells((numEW + 1) * i + j + 1, 3).Value = tasIDOKED(centerID0, lengNS * (i - numNS / 2))
            .Cells((numEW + 1) * i + j + 1, 4).Value = tasIDOKED(centerKED, lengEW * (j - numEW / 2))
            .Cells((numEW + 1) * i + j + 1, 5).Value = -0.001
            .Cells((numEW + 1) * i + j + 1, 6).Value = -99999
            .Cells((numEW + 1) * i + j + 1, 7).Value = -99999
        Next j
    Next i
End With
'観測開始
Worksheets("グリッド").Select
'CIM1 ポート設定
ec.DOWN = 1
ec.Setting = "baud=4800 parity=n data=8 stop=1"
ec.Delimiter = ec.DELIMS.CrLf
figDOKED = False
figDEPTH = False
'初期データ取得
dataF = ec.ASCIIline
Do
    dataF = ec.ASCIIline
    dummyS = Left(dataF, InStr(1, dataF, ",") - 1)
    Select Case dummyS
        Case Is = "$GPRM?"
            figDEPTH = True
            For i = 1 To 4
                dataF = Right(dataF, Len(dataF) - InStr(dataF, ","))
            Next i
            nowDEPTH = Val(Left(dataF, InStr(dataF, ",") - 1))
        Case Is = "$GPGGA"
            figDOKED = True
            For i = 1 To 2
                dataF = Right(dataF, Len(dataF) - InStr(dataF, ","))
            Next i
            nowID0 = Val(Left(dataF, InStr(dataF, ",") - 1))
            beforID0 = nowID0
            For i = 1 To 2
                dataF = Right(dataF, Len(dataF) - InStr(dataF, ","))
            Next i
            nowKED = Val(Left(dataF, InStr(dataF, ",") - 1))
            beforKED = nowKED
        Case Else
            End Select
    Loop Until figDEPTH = True And figDOKED = True
    indRow2 = 1
    indCol2 = 1
    colRowCol2 = Cells(indRow2, indCol2).Interior.ColorIndex
    figRowCol2 = False
    '全部のデータを取り終わるまで繰り返し
    Do
        figRHD = False
        figIDOKED = False
        Worksheets("グリッド").Select
        With Worksheets("グリッド")
            .Range("D1").Value = nowID0
            .Range("I1").Value = nowKED
        End With
        'データ取得
    Do

```

```

dataP = ec.AsciiLine
dummyS = Left(dataP, InStr(1, dataP, ",") - 1)
Select Case dummyS
Case Is = "$CPDBT"
  For i = 1 To 3
    dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  Next i
  nowDEPTH = Val(Left(dataP, InStr(dataP, ",") - 1))
Case Is = "$CPGGA"
  figDOKED = True
  For i = 1 To 2
    dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  Next i
  nowIDO = Val(Left(dataP, InStr(dataP, ",") - 1))
  For i = 1 To 2
    dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  Next i
  nowKED = Val(Left(dataP, InStr(dataP, ",") - 1))
Case Is = "$SCHD"
  figBHD = True
  dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  scaleSTR = Val(Left(dataP, InStr(dataP, ",") - 1))
  dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  scaleEND = Val(Left(dataP, InStr(dataP, ",") - 1))
  For i = 1 To 2
    dataP = Right(dataP, Len(dataP) - InStr(dataP, ","))
  Next i
  dataSCHD = dataP
Case Else
End Select

Loop Until ((figBHD = True) And (figDOKED = True))
'前回グリッド表示の復元観測された場合は緑に
Worksheets("グリッド").Select
Range(indRowCol2).Select
If figRowCol = True Then
  Selection.Interior.ColorIndex = 10
Else
  Selection.Interior.ColorIndex = colRowCol2
End If
'現在位置が探索範囲にあるか
pointNS = Round((hikFUN(nowIDO, centerIDO) / lengNS, 0)
pointEW = Round((hikFUN(nowKED, centerKED) / lengEW, 0)
If Abs(pointNS) <= numNS / 2 And Abs(pointEW) <= numEW / 2 Then
  '対応するグリッドを選択してセルの色を取得
  indRow = -pointNS + numNS / 2 + shiftRow + 1
  indCol = pointEW + numEW / 2 + shiftColumn + 1
  colRowCol2 = Worksheets("グリッド").Cells(indRow, indCol).Interior.ColorIndex
  pointIDO = tasIDOKED[centerIDO, lengNS * pointNS]
  pointKED = tasIDOKED[centerKED, lengEW * pointEW]
  '現在位置が区画内のデータ取得区画にあるか
  If Abs(nowIDO - pointIDO) < lengNS * ratioNS And Abs(nowKED - pointKED) < lengEW * ratioEW Then
    'グリッドの位置に赤表示
    Worksheets("グリッド").Cells(indRow, indCol).Interior.ColorIndex = 6
    indRowCol2 = indRowCol
    figRowCol = True
    'データシートへの書き込み
    With Worksheets("データ")
      .Select
      numRows = (pointNS + numNS / 2) * (numEW + 1) + (pointEW + numEW / 2 + 1)
      inputIDO = .Cells(numRow, 6).Value
      inputKED = .Cells(numRow, 7).Value
    End With
    dummyD = (hikFUN(pointIDO, nowIDO) ^ 2 + hikFUN(pointKED, nowKED) ^ 2) - (hikFUN(pointIDO, inputIDO) ^ 2 +
hikFUN(pointKED, inputKED) ^ 2)
    If inputIDO < 0 Or dummyD < 0 Then
      numGETDATA = numGETDATA + 1
      With Worksheets("データ")
        .Cells(numRow, 5).Value = Time
        .Cells(numRow, 6).Value = nowIDO
        .Cells(numRow, 7).Value = nowKED
        .Cells(numRow, 8).Value = nowDEPTH
        .Cells(numRow, 9).Value = scaleSTR
        .Cells(numRow, 10).Value = scaleEND
        For i = 1 To 214
          .Cells(numRow, 10 + i).Value = Val("&H" & Mid(dataSCHD, i, 1))
        Next i
      End With
    End If
  End If
Else

```

```

Worksheets("グリッド").Cells(indRow, indCol).Interior.ColorIndex = 6
indRow2 = indRow
indCol2 = indCol
figRowCol = False
End If
Else
If pointNS > numNS / 2 And pointEW < -numEW / 2 Then '北西はずれ
    indRow = shiftRow - 3
    indCol = shiftColumn - 2
Elseif pointNS > numNS / 2 And pointEW > numEW / 2 Then '北東はずれ
    indRow = shiftRow - 3
    indCol = shiftColumn + numEW + 4
Elseif pointNS < -numNS / 2 And pointEW < -numEW / 2 Then '南西はずれ
    indRow = shiftRow + numNS + 5
    indCol = shiftColumn - 2
Elseif pointNS < -numNS / 2 And pointEW > numEW / 2 Then '南東はずれ
    indRow = shiftRow + numNS + 5
    indCol = shiftColumn + numEW + 4
Elseif pointNS > numNS / 2 And (pointEW >= -numEW / 2 And pointEW <= numEW / 2) Then '北はずれ
    indRow = shiftRow - 3
    indCol = shiftColumn + pointEW + numEW / 2 + 1
Elseif pointNS < -numNS / 2 And (pointEW >= -numEW / 2 And pointEW <= numEW / 2) Then '南はずれ
    indRow = shiftRow + numNS + 5
    indCol = shiftColumn + pointEW + numEW / 2 + 2
Elseif (pointNS >= -numNS / 2 And pointNS <= numNS / 2) And pointEW < -numEW / 2 Then '西はずれ
    indRow = shiftRow - pointNS + numNS / 2 + 1
    indCol = shiftColumn - 2
Elseif (pointNS >= -numNS / 2 And pointNS <= numNS / 2) And pointEW > numEW / 2 Then '東はずれ
    indRow = shiftRow - pointNS + numNS / 2 + 1
    indCol = shiftColumn + numEW + 4
End If
With Worksheets("グリッド").Cells[indRow, indCol]
    colRowCol2 = .Interior.ColorIndex
    .Interior.ColorIndex = 6
End With
indRow2 = indRow
indCol2 = indCol
figRowCol = False
End If
Loop Until numGETDATA > (numNS + 1) * (numEW + 1)
ec.DOWN = 0
Else
    i = MsgBox("実行しませんでした。","処理終了")
End If
End Sub
'緯度経度の足し算を緯度経度で返す(引き算の時は2番目の引数を負にする)
Function tasIDOKED(dataIDOKED1 As Double, dataIDOKED2 As Double) As Double
    Dim dummyFUN1 As Double, dummyFUN2 As Double, ansFUN As Double
    dummyFUN1 = Fix(dataIDOKED1 / 100) * 60 + dataIDOKED1 - Fix(dataIDOKED1 / 100) * 100
    dummyFUN2 = Fix(dataIDOKED2 / 100) * 60 + dataIDOKED2 - Fix(dataIDOKED2 / 100) * 100
    ansFUN = dummyFUN1 + dummyFUN2
    tasIDOKED = Fix(ansFUN / 60) * 100 + ansFUN - Fix(ansFUN / 60) * 60
End Function
'緯度経度の引き算を分けて返す
Function hikFUN(dataIDOKED1 As Double, dataIDOKED2 As Double) As Double
    Dim dummyFUN1 As Double, dummyFUN2 As Double, ansFUN As Double
    dummyFUN1 = Fix(dataIDOKED1 / 100) * 60 + dataIDOKED1 - Fix(dataIDOKED1 / 100) * 100
    dummyFUN2 = Fix(dataIDOKED2 / 100) * 60 + dataIDOKED2 - Fix(dataIDOKED2 / 100) * 100
    hikFUN = Round(dummyFUN1 - dummyFUN2, 4)
End Function
'緯度経度の表示
Sub 緯度経度10回観測()
    Dim dataF As String, dummyS As String
    Dim lngCount As Integer
    Dim nowID0 As Double, nowKED As Double
    lngCount = 0
    ec.DOWN = 1
    ec.Setting = "baud=4800 parity=n data=8 stop=1"
    ec.Delimiter = ec.DELIMS.CrLf
    Worksheets("設定").Select
    Cells(32, 3).Value = "現在の位置"
    Cells(34, 5).Value = " / 3 個"
    lngCount = 0
    Cells(32, 4).Value = "未観測"
    Cells(33, 4).Value = "未観測"
    Cells(34, 4).Value = lngCount
    Do
        dataF = ec.AsciiLine
        If Left(dataF, 1) = "$" Then
            dummyS = Left(dataF, InStr(1, dataF, "$") - 1)

```

```

    If dummyS = "$PCGA" Then
        lngCount = lngCount + 1
        For i = 1 To 2
            dataP = Right(dataF, Len(dataF) - InStr(dataF, "."))
        Next i
        nowIDO = Val(Left(dataF, InStr(dataF, ".") - 1))
        For i = 1 To 2
            dataP = Right(dataF, Len(dataF) - InStr(dataF, "."))
        Next i
        nowKED = Val(Left(dataF, InStr(dataF, ".") - 1))
        Cells(32, 4).Value = nowIDO
        Cells(33, 4).Value = nowKED
        Cells(34, 4).Value = lngCount
    End If
End If
Loop Until lngCount >= 10
ec.COMn = 0
Cells(29, 3).Value = "最終の位置"

End Sub
'緯度経度の表示
Sub 緯度経度3回観測()
    Dim dataF As String, dummyS As String
    Dim lngCount As Integer
    Dim nowIDO As Double, nowKED As Double
    lngCount = 0
    ec.COMn = 1
    ec.Setting = "baud=4800 parity=n data=8 stop=1"
    ec.Delimiter = ec.DELIMn.CrLf
    Worksheets("設定").Select
    Cells(32, 3).Value = "現在の位置"
    Cells(34, 5).Value = " / 3観"
    lngCount = 0
    Cells(32, 4).Value = "未観測"
    Cells(33, 4).Value = "未観測"
    Cells(34, 4).Value = lngCount
Do
    dataF = ec.AsciiLine
    If Left(dataF, 1) = "$" Then
        dummyS = Left(dataF, InStr(1, dataF, ".") - 1)
        If dummyS = "$PCGA" Then
            lngCount = lngCount + 1
            For i = 1 To 2
                dataP = Right(dataF, Len(dataF) - InStr(dataF, "."))
            Next i
            nowIDO = Val(Left(dataF, InStr(dataF, ".") - 1))
            For i = 1 To 2
                dataP = Right(dataF, Len(dataF) - InStr(dataF, "."))
            Next i
            nowKED = Val(Left(dataF, InStr(dataF, ".") - 1))
            Cells(32, 4).Value = nowIDO
            Cells(33, 4).Value = nowKED
            Cells(34, 4).Value = lngCount
        End If
    End If
Loop Until lngCount >= 3
ec.COMn = 0
Cells(29, 3).Value = "最終の位置"

End Sub

```