# はじめに

## ADS-B

ADS-Bは、Automatic Dependent Surveillance-Broadcastの略です。衛星による監視システムです。航空機の位置、速度および識別情報は、Mode-S Extended Squitter（1090 MHz）を介して送信されます。 現在の航空機の大半は常にADS-Bメッセージを送信しています。受信機とアンテナを設定して、それらの信号をタップする方法はたくさんあります。 （DVB-T usbスティック、ModeSBeast、Raspberry Pi、RadarScapeなど）。  
ADS-Bメッセージの長さは112ビットです。次に例を示します。

BIN形式： 100011010100100001000000001101011000100000001011001100001101110001110000110010110011100000010101110110000010011000 HEX形式：  
8D4840D6202CC371C32CE0576098

次の表に、メッセージの主なビット形式を示します。

|  |  |  |
| --- | --- | --- |
| **bit位置** | **略称** | **名前** |
| 1~5 | DF | DownLink Format |
| 6~8 | CA | Messagae Subtype |
| 9~32 | modeS address | ICAO aircraft address |
| 33~88 | DATA | DATA Frame |
| 89~112 | PC | Parity Check |

メッセージ形式は、ダウンリンクフォーマット（DF）、ビット1〜5をチェックすることによって識別することができる。ADS-Bメッセージは、DF = 17（10進数）または10001（2進数）のときだけである。  
DATA Frame内では、タイプコードにより、DATA Frameの内容が区別されます。タイプコードは、ビット33から37（5ビット）に位置します。

|  |  |  |
| --- | --- | --- |
| **bit位置** | **略称** | **名前** |
| 33~37 | TC | TypeCode |

## ADS-Bメッセージタイプ

DFとTCを見ることで、データフレームにどのような情報が含まれているかを素早く理解できます。関係は次のようにリストされます。

|  |  |  |
| --- | --- | --- |
| **DF** | **TC** | **内容** |
| 17 | 1~4 | コールサイン |
| 17 | 5~8 | 地上機器位置 |
| 17 | 9~18 | 航空機位置(高度含む) |
| 17 | 19 | 航空機速度・向き |
| 17 | 20~22 | 航空機位置(GNSS) |
| 17 | 23 | テストメッセージ |
| 17 | 24 | 地上機器状態 |
| 17 | 25~27 | 予約済み |
| 17 | 28 | 拡張スキッターAC状態 |
| 17 | 29 | ターゲット状態 |
| 17 | 30 | 予約済み |
| 17 | 31 | 航空機の運航状態 |

異なるタイプのメッセージ内では、データフレーム内のビットの構成が異なることに注意してください。 次の章では、これらについて詳しく説明します。

## ADS-Bチェックサム

ADS-Bは、巡回冗長検査を使用して、受信したメッセージの正当性を検証します。 パリティチェックビットは、最後の6バイト（ビット89〜112）

# コールサイン

DF：17、TC：1〜4であれば、コールサインメッセージになります。

例えば：

 8D4840D6202CC371C32CE0576098

メッセージの構造は次のとおりです。

|  |  |  |  |
| --- | --- | --- | --- |
|  | **modeS address** | **Data** | **CRC** |
| 8D | 4840D6 | 202CC371C32CE0 | 576098 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **DF** | **CA** | **modeS address** | **TC** |  | **Data** | **CRC** |
| 10001 | 101 | 010010000100000011010110 | 00100 | 000 | 001011001100001101110001110000110010110011100000 | 010101110110000010011000 |

TCはDATAフレームの内側にあることに注意してください。 DFとTCは下記のとおり簡単に計算できます。

DF：10001 - > 17

TC：0010 - > 4

これらの2つの値は、メッセージがコールサインをデコードするのに適していることを確認します。

次に、航空機のコールサインを含むデータフレームをデコードします。コールサインを取得するには、インデックス番号を文字にマッピングするためのルックアップテーブルが必要です。

#ABCDEFGHIJKLMNOPQRSTUVWXYZ ############## 0123456789 ######

メッセージデータフレームでは、次のようにデコードするのは簡単です。

HEX: 202CC371C32CE0

BIN: 00100 000 | 001011 001100 001101 110001 110000 110010 110011 100000

DEC: | 11 12 13 49 48 50 51 32

LTR: | K L M 1 0 2 3 \_

*これより、コールサインはKLM1023*だと分かります。

# 航空機位置

DF：17、TC：9〜18であれば、航空機位置メッセージになります。

メッセージは次のように構成されます。

|  |  |  |  |
| --- | --- | --- | --- |
| **bit位置** | **ビット数** | **略称** | **内容** |
| 1~5 | 5 | DF | DownLink Format |
| 33~37 | 5 | TC | TypeCode |
| 38~39 | 2 | SS | 監視ステータス |
| 40 | 1 | NICsb | NIC supplement-B |
| 41~52 | 12 | ALT | 高度 |
| 53 | 1 | T | 時間 |
| 54 | 1 | F | CPR Even/Oddフレームフラグ |
| 55~71 | 17 | LAT-CPR | CPR形式の緯度 |
| 72~88 | 17 | LON-CPR | CPR形式の経度 |

航空機位置をデコードするのは少し複雑です。

航空機のLAT(緯度)とLON(経度)を抽出するには、2種類の位置メッセージ（EvenフレームとOddフレーム）が必要です。 その位置は、Compact Positioning Reporting（CPR）形式で記述されています。 CPRの利点は、少ないビットでより高い解像度で位置を符号化できることです。 しかし、これは復号プロセスの複雑さをもたらす。

## EvenフレームとOddとフレームとは

各フレームにて、54ビット目により、そのフレームが「Even(偶数)フレーム」であるか「Odd(奇数)フレーム」であるかを判定します。

 0 - >偶数フレーム

 1 - >奇数フレーム

例えば、次の2つのメッセージ

8D40621D58C382D690C8AC2863A7

8D40621D58C386435CC412692AD6

|  |  |  |  |
| --- | --- | --- | --- |
|  | **modeS address** | **データ** | **CRC** |
| 8D | 40621D | 58C382D690C8AC | 2863A7 |
| 8D | 40621D | 58C386435CC412 | 692AD6 |

両方のメッセージをバイナリ文字列に変換すると

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DF** | **CA** | | | **modeS address** | | | | | |
| 10001 | 101 | | | 010000000110001000011101 | | | | | |
| 10001 | 101 | | | 010000000110001000011101 | | | | | |
| **Data** | | | | | | | | | | | |
| **TC** | | **SS** | **NICsb** | | **ALT** | | **T** | **F** | **LAT-CPR** | | **LON-CPR** |
| 01011 | | 00 | 0 | | 110000111000 | | 0 | 0 | 10110101101001000 | | 01100100010101100 |
| 01011 | | 00 | 0 | | 110000111000 | | 0 | 1 | 10010000110101110 | | 01100010000010010 |
| **CRC** | | | | | |
| 001010000110001110100111 | | | | | |
| 011010010010101011010110 | | | | | |

両方のメッセージで、同じmodeSアドレス40621Dを使用して、DF = 17とTC = 11を見つけることができます。したがって、この2つのフレームは、この航空機の位置を解読するのに有効です。

## CPRのパラメータと機能

まず、ここではデコード処理で使用されるいくつかのパラメータと共通関数を示します。

### NZ

赤道と極の間の地理的緯度帯の数。 Mode-S CPRエンコーディングの場合、NZ = 15に設定されます。

### floor（x）

floor関数floor（x）は、 となるような最大整数値kを算出します。

floor（5.6）= 5

floor（-5.6）= -6

### mod（x、y）

モジュラス関数mod（x、y）は次のように返します。

yには0は入らない。

### NL(緯度)

緯度latを考慮して、「経度ゾーンの数」関数を示します。 返される整数値は[1~59]以内に制限され、次のように計算されます。

赤道または極に近い緯度の場合、次の値が返されます。

lat = 0→NL = 59  
lat = + 87→NL = 2  
lat = -87→NL = 2  
lat> + 87 - > NL = 1  
lat <-87 - > NL = 1

## 緯度経度の計算

CPRの背後にある数学を詳細に説明するいくつかの技術文書があります。例えば、Eurocontrolの文書です。

最初に両方のメッセージのCPRの緯度と経度のビットを区切りましょう。その後の手順で、航空機のLAT(緯度)/LON(経度)を計算します。

例

|  |  |  |  |
| --- | --- | --- | --- |
| **F** | **lat緯度(CPR形式)** | **lon経度(CPR形式)** | **最新** |
| 0 | 10110101101001000 | 01100100010101100 | new |
| 1 | 10010000110101110 | 01100010000010010 |  |

### ステップ1：バイナリを10進値に変換

例

LAT\_CPR\_EVEN：93000/131072 - > 0.7095  
LON\_CPR\_EVEN：51372/131072 - > 0.3919  
LAT\_CPR\_ODD：74158/131072 - > 0.5658  
LON\_CPR\_ODD：50194/131072 - > 0.3829

CPRの緯度と経度は17ビットで符号化されるため、131072（217）が最大値です。計算結果の値は、その最大値のパーセンテージを表します。

### ステップ2：緯度番号jを導く

緯度番号jは、位置情報が「3.2.4NL(緯度)」で導いた「地理的経度ゾーンの数」のうち、何番目に該当するかを示した値である。 次の式を使用します。

例 j=8

### ステップ3：緯度

まず、2つの定数 と が使用されます。

次に、次の式を使用して相対緯度を計算することができます。

南半球では、値は270度から360度になってしまいます。 よって、緯度が範囲内(-90~90)にあることを確認する必要があります。

Even、Oddのうち、最新のものが使用します。

### ステップ4：チェック

　と を計算により求めます。それらが同一でない場合、2つの位置が異なる緯度ゾーンに位置します。その場合は経度の計算は不可能です。計算を終了し、新しいメッセージを待ちます。

2つの値が同じ場合、経度の計算に進みます。

### ステップ5：経度

Even(偶数)フレームが最新の場合、

Odd(奇数)フレームが最新の場合、

例 Lon: 3.91937

## 高度計算

高度計算は、ずっと簡単です。 高度フィールド（Even(偶数)フレームまたはOdd(奇数)フレーム）のビットは次のとおりです。

1100001 1 1000  
　　　　^  
　　　Qビット

このQbit（48bit目）は、高度が25または100フィートの倍数（0：100フィート、1：25フィート）でエンコードされているかどうかを示します。

Q = 1の場合、次のように標高を計算できます。

まず、Qビットを削除する。

N = 1100001 1000 => 1560（10進数）

最終的な高度値は次のようになります。

Alt = N \* 25 - 1000(ft.)

この例では、航空機が飛行している高度は次のとおりです。

1560 \* 25 - 1000 = 38000フィート

Qビットが1の場合、高度は+/- 25フィートの精度を持ち、値は-1000〜+50175フィートの高度を表すことができます。

## 最終位置

最後に、航空機位置の3つのコンポーネント（緯度/経度/高度）がすべてあります。

LAT：52.25720（度N）  
LON：3.91937（度E）  
ALT：38000フィート

# 対地速度

航空機の速度メッセージはDF：17、TC：19.、サブタイプコードはビット38〜40で表されます。ここでそれらのメッセージをデコードできます。データブロックは以下の通り。



例として以下のようなデータを受信したとする。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TC** | **ST** | **IC** | | **RESV\_A** | | **NAC** | **S-EW** | | **V-EW** | | **S-NS** | | **V-NS** |
| 10011 | 001 | 0 | | 1 | | 000 | 1 | | 0000001001 | | 1 | | 0010100000 |
| **VrSrc** | **S-Vrc** | | **Vr** | | **RESV\_B** | | | **S-Dif** | | **Dif** | |
| 0 | 1 | | 000001110 | | 00 | | | 0 | | 0010111 | |

## 水平方向の速度

どの方角でどのくらいの速さかを調べるためには4つの要素がいる。

東―西軸方向の速さを表すV-EW、 航空機が東―西軸方向のどちらに向かって進むのかを表すS-EW、北―南軸方向の速さを表すV-NS、 航空機が北―南軸方向のどちらに向かって進むのかを表すS-NS、の4つである。以下に説明を記す。

S-NS:

1 -> 北から南へ飛行している

0 -> 南から北へ飛行している

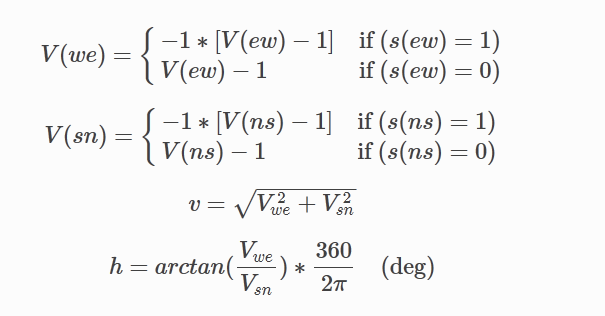
S-EW:

1 -> 東から西へ飛行している

0 -> 西から東へ飛行している

V-NS、V-EW共に単位はknである。

以下に航空機の速度を求める式を記す。



もしhの値がマイナスになってしまった場合は

h = h + 360 (if h<0)

とする。

今回の例の場合以下のようになる。

V-EW: 0000001001 -> 9

S-EW: 1

V-NS: 0010100000 -> 160

S-NS: 1

V(we) = -(9 - 1) = -8

V(sn) = - (160 - 1) = -159

v = 159.20 (kn)

h = 182.88 (deg)

よってこの航空機は159.20 (kn)の速さで飛行し、182.88 (deg)の方向を向き、飛行しているということが分かる。

**垂直方向の速度**

航空機の垂直方向の動きは69bit目のS-Vrを読めばわかる。

0 -> UP

1 -> Down

どのくらいの速度で上下しているかはVrを参照すればよい。(ft/min)

今回の例では以下のようになる。

Vr: 000001110 => 14

S-Vr: 0 => Down / Descending

よって14ft/minで航空機が降下していることが分かる。