**ALLOCATION JV PROCESSING**

**PROGRAM ID** **PRG\_FIN\_RECT\_ALLOC\_JV\_PROCESS**

**PROGRAM BRIF DESC.** This program is to read the Individual Receipt table **TBFN\_RECEPT\_FILE**  and analyses the Receipt amount to what goes to Insured Savings and Risk Premium . It should also check for Insured Contributions from which Allocation charges should be deducted.

It should generate General Ledger AllocationTransactions into a Table (**TBIL\_FIN\_RECT\_GL\_TRANS)**

**TABLES**

**Input**

Receipt Table **TBFN\_RECEPT\_FILE**

Proposal/Policy Premium Information **TBIL\_POLICY\_PREM\_INFO**

Policy Premium Details **TBIL\_POLICY\_PREM\_DETAILS**

Claims Reported File **TBIL\_CLAIM\_REPTED**

Chart Of Account Table **TBFN\_ACCT\_CODES**

Admin charge Rate Table **TBIL\_ADMIN\_CHG**

**Input/Output**

Insured Premium/Allocation Table **TBFN\_ALLOC\_DETAIL**

General Ledger Allocation Trans (**TBIL\_FIN\_RECT\_GL\_TRANS)**

**PROCESSING**

Accept the Range of the date of Transactions to be extracted and allocated from the Receipt file.

Alloc\_Start\_Date and Alloc\_End\_date

**Read Individual Receipt Table TBFN\_RECEPT\_FILE**

Read the Receipt Table.

Records to be processed must meet the following :

- The Entry Date **(TBFN\_ACCT\_ENTRY\_DATE )** should be within the

Alloc\_start\_date and Alloc\_end\_date.

* The Record read must have been approved. (TBFN\_ACCT\_APPRV\_YN) must be Ý’

**Check for Receipt Type and how to Process them**

**PREMIUM1 = 0**

**SAVINGS1 = 0**

**PREMIUM2 = 0**

**SAVINGS2 = 0**

**LUMP1 = 0**

**CHG\_PREM = 0**

1. If Receipt Type not relating to Insurance Premium or Claims (i.e. **TBFN\_ACCT\_RECP\_TYP not**  = D – Premium Deposit, P- Regular Premium

U – Lump Sum , F – FAC C – Co.Ins Y – Claims)

**Perform Generate Bank DR Transaction**

**Perform Generate Other CRTransaction**

**Exit**

1. If Receipt Type for Claims (i.e. Y – Claims)

Use the Claims Number from the Receipt Table **TBFN\_ACCT\_REF\_NO**

To Read the Claims Reported File. **TBIL\_CLAIM\_REPTED**

**If not Found on Claims File then there is Error.**

Get the following from Claims Reported File

**Product\_Code1** = **TBIL\_CLM\_RPTD\_PRDCT\_CD**

**Policy Type**  **= “Y’**

Business Type (New or Renewal) To get this, check the

Policy Start Date (**TBIL\_CLM\_RPTD\_POLY\_FROM\_DT**). If Policy

Start Date Less or Equal one from Current Date Then **Business1** is

**NEW** else **RENEWAL**. **(N for New R for Renewal)**

**CLAIMS1 = TBFN\_ACCT\_AMT\_LC**

**Perform Get Account Number**

**Perform Generate Bank DRTransaction**

**Perform Generate Claims CR Transaction**

**Exit**

1. If Receipt Type for Premium = D – Premium Deposit, P- Regular Premium

U – Lump Sum , F – FAC C – Co.Ins )

If Receipt Type is for Premium = D – Premium Deposit Use the **Proposal**

**Number else Use Policy Number** from the Receipt Table

**(TBFN\_ACCT\_REF\_NO)** To Read the Proposal/Policy Premium Info

**TBIL\_POLICY\_PREM\_INFO**

**There is Error If not Found on Premium Info File**

**End if**

Get the following from Proposal/Policy Premium Info File

**Product\_ Code1** =**TBIL\_POL\_PRM\_PRDCT\_CD**

**Policy Type**  **= “P’**

Business Type (New or Renewal) To get this, check the

Policy Start Date (**TBIL\_POL\_PRM\_FROM**). If Policy

Start Date Less or Equal one from Current Date Then **Business1** is

**NEW** else **RENEWAL**. **(N for New R for Renewal)**

Calculate Year (**Year1)** Policy in existence (Policy start date to Current Date-

Year value only)

**Get Allocation Applicable code (TBIL\_POL\_PREM\_ALLOC\_CD)**

**ALLOC\_CD1 = TBIL\_POL\_PREM\_ALLOC\_CD**

**End if**

If Receipt Type is for Premium = D – Premium Deposit Use the **Proposal**

**Number else Use Policy Number** from the Receipt Table

**(TBFN\_ACCT\_REF\_NO)** To Read the Policy Premium Details

**TBIL\_POLICY\_PREM\_DETAILS**

**There is Error If not Found on Policy Premium Details File.**

Get MOP Premium **TBIL\_POL\_PRM\_DTL\_MOP\_PRM\_LC**

Get MOP Contribution **TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC**

End if

If Receipt Type is for Premium = U – Lump Sum

Then **PREMIUM1 = 0**

**SAVINGS1 = 0**

**LUMP1 =** Receipt Amount **TBFN\_ACCT\_AMT\_LC**

Go to **Calculate Alloc\_Prem**i**um**

End if

**If TBIL\_POL\_PRM\_DTL\_MOP\_PRM\_LC =**

**TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC**

**Then PREMIUM1 = 0**

**SAVINGS1 = TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC**

Go to **Calculate Alloc\_Prem**

End if

Check if MOP Contribution **TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC**

Equal Receipt Amount **TBFN\_ACCT\_AMT\_LC**

If equal then **PREMIUM1** = **TBIL\_POL\_PRM\_DTL\_MOP\_PRM\_LC**

**SAVINGS1** = **TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC**

Less **TBIL\_POL\_PRM\_DTL\_MOP\_PRM\_LC**

**Else**

**PREMIUM1 = TBIL\_POL\_PRM\_DTL\_MOP\_PRM\_LC** Divided by

**TBIL\_POL\_PRM\_DTL\_MOP\_CONTRB\_LC** Multiply by **TBFN\_ACCT\_AMT\_LC.**

**SAVINGS1 = TBFN\_ACCT\_AMT\_LC less PREMIUM1**

End if

**Calculate Alloc\_Prem**

**Get the Percent for Allocation**

If **Receipt Type** is for LumpSum Premium = U – Lump Sum

**Go to Generate Required Records**

**Use Product Code Product\_ Code1** to Read theAdmin charge Rate Table (**TBIL\_ADMIN\_CHG)**

Policy Existence Tenor **(YEAR1)**  Should be within Admin charge Table

**TBIL\_ ADMIN\_CHG\_START\_YR** and **TBIL\_ ADMIN\_CHG\_END\_YR**.

If Record does not exist then

**RATE1** = 0  **PREMIUM2 = PREMIUM1**

**SAVINGS2 = SAVINGS1**

**Go to Generate Required Records**

**End if**

Get the Charge Rate (**TBIL\_ ADMIN\_CHG \_RATE**) **RATE1** and Rate Per

(**TBIL\_ ADMIN\_CHG \_RATE\_PER**)  **RATEPER1**

Calculate **CHG\_PREM** as

**SAVINGS1** multiplied by **RATE1** divided by **RATEPER1**

Calculate Savings Balance **SAVINGS2 = SAVINGS1** less **GHG\_PREM**

**PREMIUM2 = PREMIUM1**

**Generate Required Records**

**Perform Generate Bank DRTransactin**

**If PREMIUM2 > 0**

**Perform Get Account Number**

**Perform Generate Premium CR Transaction**

**If CHG\_PREM > 0**

**Perform Get Account Number**

**Perform Generate Charge CR Transaction**

**If SAVINGS2 > 0**

**Perform Get Account Number**

**Perform Generate Savings CR Transaction**

**Move ‘S’ to SAVE\_TYPE**

**Perform Generate Insured Savings Transaction Record**

**If LUMP1 > 0**

**Perform Get Account Number**

**Perform Generate LumpSum CR Transaction**

**Move ‘L’ to SAVE\_TYPE**

**Perform Generate Insured Savings Transaction Record**

**Exit**

**Get Account Number**

Read the Chart of Account File to Get the General Ledger Accounts to be credited.

**For Claims Amount**

Move **PRODUCT\_CODE1** to TBFN\_ACCT\_PRDCT\_CD

Move **Business1** to TBFN\_ACCT\_BUS\_TYPE

Move **Policy\_Type** (Y) to TBFN\_ACCT\_POL\_TYPE

**For Savings Amount**

Move **PRODUCT\_CODE1** to TBFN\_ACCT\_PRDCT\_CD

Move **Business1** to TBFN\_ACCT\_BUS\_TYPE

Move **Policy\_Type** (PS) to TBFN\_ACCT\_POL\_TYPE

**For Lump Sum Amount**

Move **PRODUCT\_CODE1** to TBFN\_ACCT\_PRDCT\_CD

Move **Business1** to TBFN\_ACCT\_BUS\_TYPE

Move **Policy\_Type** (PS) to TBFN\_ACCT\_POL\_TYPE

**For Premium Amount**

Move **PRODUCT\_CODE1** to TBFN\_ACCT\_PRDCT\_CD

Move **Business1** to TBFN\_ACCT\_BUS\_TYPE

Move **Policy\_Type** (PP) to TBFN\_ACCT\_POL\_TYPE

**For Charge Premium Amount**

Move **PRODUCT\_CODE1** to TBFN\_ACCT\_PRDCT\_CD

Move **Business1** to TBFN\_ACCT\_BUS\_TYPE

Move **Policy\_Type** (PP) to TBFN\_ACCT\_POL\_TYPE

Move Account Number retrieved

TBFN\_ACCT\_MAIN\_CD to **MAIN\_ACCT**

TBFN\_ACCT\_SUB\_CD to **SUB\_ACCT**

EXIT

**Generate Bank DR Transaction Routine**

This section should generate the Debit transaction. This is the Record that debits the Bank with the Receipt.

Initialize Record **TBIL\_FIN\_RECT\_GL\_TRANS**

**Perform Move Basic Data Routine**

**Move ‘D’ to TBFN\_ALLC\_DRCR\_CD**

**Move TBFN\_ACCT\_DR\_MAIN to TBFN\_ALLC\_MAIN\_ACCT**

**Move TBFN\_ACCT\_DR\_SUB to TBFN\_ALLC\_SUB\_ACCT**

Move TBFN\_ACCT\_AMT\_LC to TBFN\_ALLC\_AMT\_LC

Move TBFN\_ACCT\_AMT\_FC to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate Other CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move TBFN\_ACCT\_CR\_MAIN to TBFN\_ALLC\_MAIN\_ACCT**

**Move TBFN\_ACCT\_CR\_SUB to TBFN\_ALLC\_SUB\_ACCT**

Move TBFN\_ACCT\_AMT\_LC to TBFN\_ALLC\_AMT\_LC

Move TBFN\_ACCT\_AMT\_FC to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate Claims CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move MAIN\_ACCT to TBFN\_ALLC\_MAIN\_ACCT**

**Move SUB\_ACCT to TBFN\_ALLC\_SUB\_ACCT**

Move TBFN\_ACCT\_AMT\_LC to TBFN\_ALLC\_AMT\_LC

Move TBFN\_ACCT\_AMT\_FC to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate Premium CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move MAIN\_ACCT to TBFN\_ALLC\_MAIN\_ACCT**

**Move SUB\_ACCT to TBFN\_ALLC\_SUB\_ACCT**

Move  **PREMIUM2** to TBFN\_ALLC\_AMT\_LC

Move **PREMIUM2** to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate Savings CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move MAIN\_ACCT to TBFN\_ALLC\_MAIN\_ACCT**

**Move SUB\_ACCT to TBFN\_ALLC\_SUB\_ACCT**

Move  **SAVINGS2** to TBFN\_ALLC\_AMT\_LC

Move **SAVINGS2** to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate LumpSum CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move MAIN\_ACCT to TBFN\_ALLC\_MAIN\_ACCT**

**Move SUB\_ACCT to TBFN\_ALLC\_SUB\_ACCT**

Move  **LUMP1** to TBFN\_ALLC\_AMT\_LC

Move **LUMP1** to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Generate Charge CR Transaction Routine**

**Perform Move Basic Data Routine**

**Move ‘C’ to TBFN\_ALLC\_DRCR\_CD**

**Move MAIN\_ACCT to TBFN\_ALLC\_MAIN\_ACCT**

**Move SUB\_ACCT to TBFN\_ALLC\_SUB\_ACCT**

Move  **CHG\_PREM** to TBFN\_ALLC\_AMT\_LC

Move  **CHG\_PREM** to TBFN\_ALLC\_AMT\_FC

**Write Record**

**Move Basic Data Routine**

Initialize Record for File (**TBIL\_FIN\_RECT\_GL\_TRANS)**

Move TBFN\_ACCT\_TRANS\_COMP TO TBFN\_ALLC\_TRANS\_COMP

Move TBFN\_ACCT\_BATCH\_NO TO TBFN\_ALLC\_BATCH\_NO

Move TBFN\_ACCT\_SER\_NO TO TBFN\_ALLC\_SER\_NO

Move TBFN\_ACCT\_TRANS\_TYP TO TBFN\_ALLC\_TRANS\_TYP

Move TBFN\_ACCT\_TEMP\_TRANS\_NO TO TBFN\_ALLC\_TEMP\_TRANS\_NO

Move TBFN\_ACCT\_DOC\_**DATE** TO TBFN\_ALLC\_DOC\_**DATE**

Move TBFN\_ACCT\_ENTRY\_DATE TO TBFN\_ALLC\_ENTRY\_DATE

Move TBFN\_ACCT\_TRANS\_MODE TO TBFN\_ALLC\_TRANS\_MODE

Move TBFN\_ACCT\_RECP\_TYP TO TBFN\_ALLC\_RECP\_TYP

Move TBFN\_ACCT\_REF\_NO TO TBFN\_ALLC\_REF\_NO

Move TBFN\_ACCT\_DOC\_NO TO TBFN\_ALLC\_DOC\_NO

Move TBFN\_ACCT\_CURRCY\_TYPE TO TBFN\_ALLC\_CURRCY\_TYPE

Move TBFN\_ACCT\_CHQ\_TELLER\_NO TO TBFN\_ALLC\_CHQ\_TELLER\_NO

Move TBFN\_ACCT\_CHQ\_INWARD\_NO TO TBFN\_ALLC\_CHQ\_INWARD\_NO

Move TBFN\_ACCT\_CHQ\_INWARD\_DATE TO TBFN\_ALLC\_CHQ\_INWARD\_DATE

Move TBFN\_ACCT\_PAYER\_PAYEE\_NAME TO

TBFN\_ALLC\_PAYER\_PAYEE\_NAME

Move TBFN\_ACCT\_TRANS\_DESC 1 TO TBFN\_ALLC\_TRANS\_DESC 1

Move TBFN\_ACCT\_TRANS\_DESC2 TO TBFN\_ALLC\_TRANS\_DESC 1

Move TBFN\_ACCT\_BRANCH\_CD TO TBFN\_ALLC\_BRANCH\_CD

Move TBFN\_ACCT\_BANK\_CD TO TBFN\_ALLC\_BANK\_CD

Move TBFN\_ACCT\_AMT\_LC TO TBFN\_ALLC\_AMT\_LC

Move TBFN\_ACCT\_AMT\_FC TO TBFN\_ALLC\_AMT\_FC

Move TBFN\_ACCT\_INS\_CODE TO TBFN\_ALLC\_INS\_CODE

Move TBFN\_ACCT\_AGENT\_CODE TO TBFN\_ALLC\_AGENT\_CODE

Move TBFN\_ACCT\_COMM\_YN TO TBFN\_ALLC\_COMM\_YN

Move TBFN\_ACCT\_POLY\_CONTRIB TO TBFN\_ALLC\_POLY\_CONTRIB

Move TBFN\_ACCT\_POLY\_MOP TO TBFN\_ALLC\_POLY\_MOP

Move TBFN\_ACCT\_DRCR\_CD TO TBFN\_ALLC\_DRCR\_CD

Move TBFN\_ACCT\_MAIN\_ACCT TO TBFN\_ALLC\_MAIN\_ACCT

Move TBFN\_ACCT\_SUB\_ACCT TO TBFN\_ALLC\_SUB\_ACCT

Move TBFN\_ACCT\_LEDG\_TYP\_CR TO TBFN\_ALLC\_LEDG\_TYP

Move TBFN\_ACCT\_POSTED\_YN TO TBFN\_ALLC\_POSTED\_YN

Move TBFN\_ACCT\_APPRV\_YN TO TBFN\_ALLC\_APPRV\_YN

**Exit**

**Generate Insured Savings Transaction Record Routine**

Initialize Record . INSURED PREMIUM/ALLOCATION TABLE

**(TBFN\_ALLOC\_DETAIL)**

Move TBFN\_ ACCT \_COMP to TBFN\_TRANS\_COMP

Move TBFN\_ ACCT \_YR to TBFN\_TRANS\_YR

Move TBFN\_ ACCT \_PRDCT\_CD to TBFN\_ TRANS \_PRDCT\_CD

Move TBFN\_ACCT\_REF\_NO to TBFN\_ TRANS \_POLY\_NO

Move TBFN\_ACCT\_DOC\_**DATE** to TBFN\_ TRANS \_DATE

Move TBFN\_ACCT\_PRDCT\_CD to TBFN\_ TRANS \_PRDCT\_CD

Move TBFN\_ACCT\_INS\_CODE to TBFN\_ TRANS \_INS\_CODE

Move TBFN\_ACCT\_TRANS\_DESC2 to TBFN\_ TRANS \_DESC

If **SAVE\_TYPE = ‘L’**

Move ‘L’ to TBFN\_ TRANS \_TYPE

If **SAVE\_TYPE = ‘S’**

Move ‘**N**’ to TBFN\_ TRANS \_TYPE

Move TBFN\_ACCT\_DOC\_NO to TBFN\_ TRANS \_NO

TBFN\_ TRANS \_TOT\_AMT = **LUMP1 + SAVINGS2**

Move **SAVINGS2** to TBFN\_ TRANS \_ALLOC\_AMT

TBFN\_ TRANS \_PREM\_AMT = **PREMIUM2 + CHG\_PREM**

Move  **LUMP1** to TBFN\_ TRANS \_LUMP\_AMT

Move 0 to TBFN\_ TRANS \_PREM\_RECOVER

Move TBFN\_ACCT\_AGENT\_CODE TO TBFN\_TRANS\_AGENT\_CODE

Move TBFN\_ ACCT \_ENTRY\_DATE to TBFN\_ TRANS \_ENTRY\_DATE

Move TBFN\_ ACCT \_STATUS\_FLAG to TBFN\_ TRANS \_STATUS\_FLAG

Move TBFN\_ ACCT \_OPERID to TBFN\_ TRANS \_OPERID

**Write Record**

**EXTRACT G/LEDGER TRANS FOR GL MASTER UPDATE**

**PROGRAM ID** **PRG\_FIN\_TRANS\_EXTRACT**

**PROGRAM BRIF DESC.** This program Reads the Allocated Individual Receipt table(**TBIL\_FIN\_RECT\_GL\_TRANS).** It also Reads the General Ledger Transaction File which comprises of Other Receipts, Journals and Payment Vouchers. **TBFN\_TRANS\_FILE**

It Reads both Files and extract Transactions for specified periods and format them into The Trans Extract File **TBFN\_GL\_DETAILS\_EXTRACT** which will in a later Program be used for the General Ledger Masters Update.

It should also ReadtheCreditors Transaction Table **TBFN\_CRED\_MASTER** and format the record for extract to the Extract Table

Transactions for the following will be summarized to one record when extracted and will be posted as one record to General Ledger during update.

Individual Receipts

* Debtors Receipts (for each type of Debtors eg Agents, Brokers etc)

Other Receipts, Journals and Payment Vouchers

* Debtors Receipts (for each type of Debtors eg Agents, Brokers etc)
* Creditors Journals (for each type of Creditor)

Creditors Transaction (Payment Vouchers)

* Creditors Payments (for each type of Creditor)

For other transactions each record from Input files will be extracted to the output file.

**TABLES**

**Input**

Allocated Individual Receipt table(**TBIL\_FIN\_RECT\_GL\_TRANS**

Receipts, Journals and Payment Voucher Trans. **TBFN\_TRANS\_FILE**

Creditors Transaction Table **TBFN\_CRED\_MASTER**

**Input/Output**

G/Ledger Trans Extract File **TBFN\_GL\_DETAILS\_EXTRACT**

**PROCESSING**

Accept the Range of the date of Transactions to be extracted for G/Ledger Update

**Extract\_Start\_Date** and **Extract\_End\_date**

**Check General Business Software for other parameters that can be used for the extract**

**Read the Allocated Individual Receipt table** (**TBIL\_FIN\_RECT\_GL\_TRANS**

Read the Allocated Individual Receipt table

Records to be processed must meet the following :

- The Entry Date **(TBFN\_ALLC\_ENTRY\_DATE )** should be within the

**Extract\_start\_date** and **Extract\_end\_date.**

Perform **Move Allocated Individual Receipt to Extract Trans**

**Read Receipts, Journals and Payment Voucher Trans.** **TBFN\_TRANS\_FILE**

Read the Receipts, Journals and Payment Voucher Trans

Records to be processed must meet the following :

- The Entry Date (**TBFN\_GL\_DOC\_DATE)** should be within the

**Extract\_start\_date** and **Extract\_end\_date.**

Perform **Move Receipts, Journals and Payment Voucher Trans**

**Read Creditors Transaction Table** **TBFN\_CRED\_MASTER**

Read the Creditors Transaction Table

Records to be processed must meet the following :

- The Entry Date (**TBFN\_CRD\_TRANS\_DATE)** should be within the

**Extract\_start\_date** and **Extract\_end\_date.**

Perform **Move Creditors Transaction Table**

**Move Allocated Individual Receipt to Extract Trans**

There are two steps involved here. The first is for Accounts that has to be summarized before writing to Extract file. This will be in case of Debtors Accounts.

The second is for other Accounts which needs to be written individually.

1. Read Allocated Individual Receipt table (**TBIL\_FIN\_RECT\_GL\_TRANS**

Should be sorted by Ledger Type and Main Account Code.

Process only for Records with Ledger Code “**T**” (Debtors)

**For each Main Account Code**

Sum All debit records **(TBFN\_ALLC\_DRCR\_CD) = ‘D’ into variable**

**Debtor\_Debit**

Sum All debit records **(TBFN\_ALLC\_DRCR\_CD) = ‘C’ into variable**

**Debtor\_Credit**

At end of each Main Account **Perform Write Receipt Debtors To Extract**

At end of file go to **(b)**

**(b )** Read Allocated Individual Receipt table (**TBIL\_FIN\_RECT\_GL\_TRANS**

Should be sorted by Ledger Type and Main Account Code.

Do not Process for Records with Ledger Code “**T**” (Debtors)

**For each Record Read**

**Perform Write Receipt Records To Extract**

At end of file exit File Read

**Move Receipts, Journals and Payment Voucher Trans**

There are also two steps involved here. The first is for Accounts that has to be summarized before writing to Extract file. This will be in case of Debtors or Creditors Accounts.

The second is for other Accounts which needs to be written individually.

1. Read Receipts, Journals and Payment Voucher Trans**.** **TBFN\_TRANS\_FILE**

Should be sorted by Ledger Type and Main Account Code.

Process only for Records with Ledger Code “**T**” ( Debtors or Creditors)

**For each Main Account Code**

Sum All debit records **(TBFN\_GL\_DR\_CR\_CD ) = ‘D’ into variable**

**Debtor\_Debit**

Sum All debit records **(TBFN\_GL\_DR\_CR\_CD** **) = ‘C’ into variable**

**Debtor\_Credit**

At end of each Main Account

**Perform Write Receipts, Journals and Payment Voucher Summary To Extract**

At end of file go to **(b)**

**(b )** Read Receipts, Journals and Payment Voucher Trans**.** **TBFN\_TRANS\_FILE**

Should be sorted by Ledger Type and Main Account Code.

Do not Process for Records with Ledger Code “**T**” (Debtors and Creditors)

**For each Record Read**

**Perform Receipts, Journals and Payment Vouchers Details** **To Extract**

At end of file exit File Read

**Move Creditors Transaction Table**

There are also two steps involved here. The first is for Accounts that has to be summarized before writing to Extract file. This will be in case of Creditors Accounts.

The second is for other Accounts which needs to be written individually.

1. Read Creditors Transaction Table **TBFN\_CRED\_MASTER**

**For each Main Account Code**

Sum All Amounts for the Creditor Acct **(TBFN\_CRD\_ACCT\_NO) into variable**

**Debtor\_Credit**

At end of File

**Perform Write CreditorsTrans Summary To Extract**

**(b )** Read Creditors Transaction Table **TBFN\_CRED\_MASTER**

**For each Record Read**

**Perform Write Creditors Debit Trans Details To Extract**

At end of file exit File Read