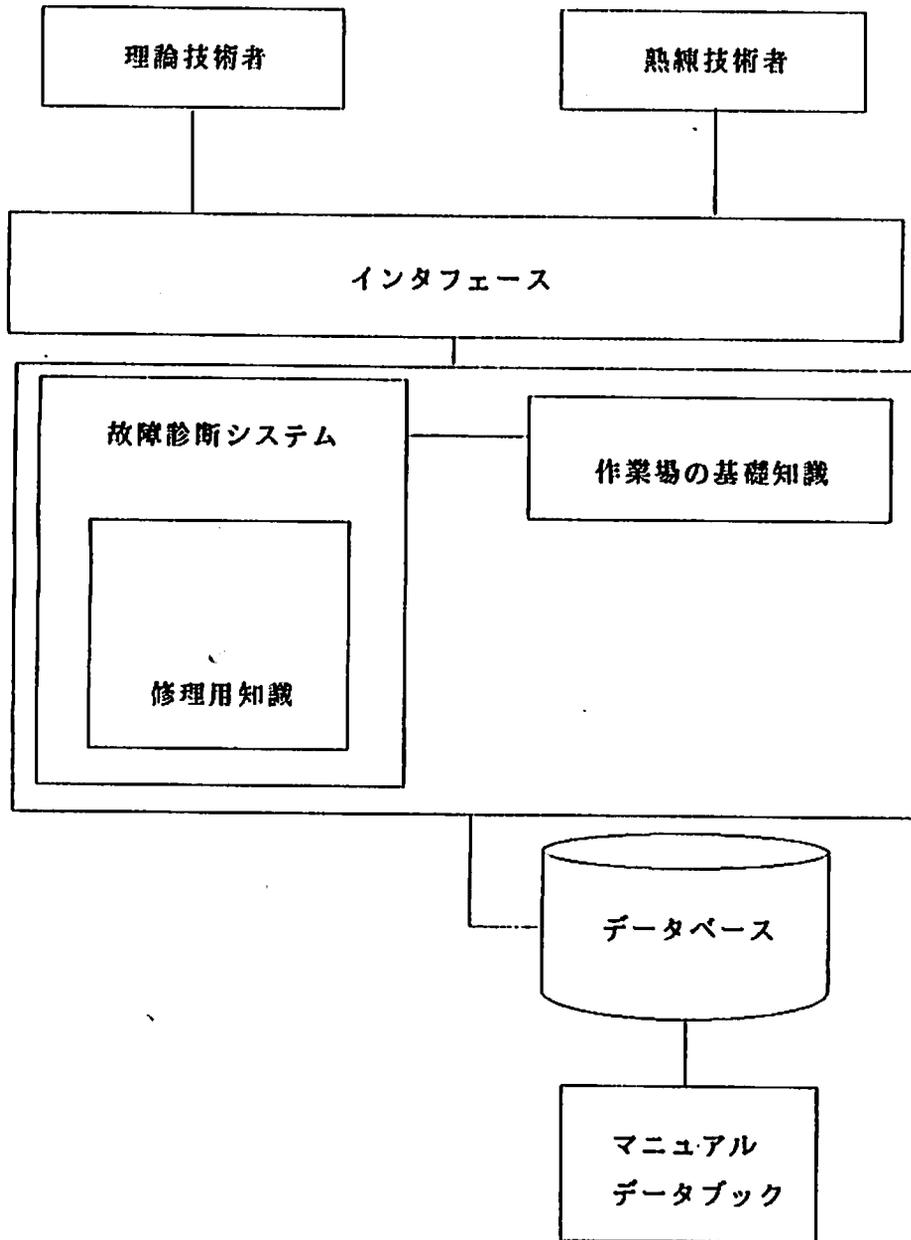


付録 1

故障修理支援システムの構想図



付録3

症状チェックと状態把握の機種ごとのパターン（経験的に得られたもの）
〔チェックリスト〕

1. プリメインアンプ

症状：パイロットランプ、プロテクタ、におい、けむり、音が出るか

状態：入出力特性1（ラインイン—スピーカ間）

入出力特性2（メインイン—スピーカ間）

原因ブロックを仮定する知識

パイロットランプがつかない→電源部

音が出ない、プロテクタ動作不良→プロテクタ

音が出ない、焼けたにおい、入出力特性2不良→メインアンプ

音が出ない、入出力特性2不良、入出力特性1良、パイロットランプ良

プロテクター良→プリアンプ用電源

2. チューナ

症状：パイロットランプ、音が出るか、選局、AM受信、FM受信

状態：局部発振器発振状態

原因ブロックを仮定する知識

パイロットランプがつかない→電源部

音が出ない→電源部、muting回路

選局できない→アンテナ部、局部発振回路部

AM受信できない→AM部

AM受信できない→AM部

局部発振回路不良→局部発振回路部

3. CDプレーヤ

症状：パイロットランプ、リードインするか、音が出るか、音飛び、サーチ、レーザ出ているか

状態：マイコンクロック発振状態、マイコン用電源+5V

原因ブロックを仮定する知識

パイロットランプがつかない→電源部

リードインしない、レーザ出てる、マイコンクロック不良→マイコン部

リードインしない、レーザ出ない→レーザ駆動部

リードインしない、レーザ出てる、マイコンクロック不良、5V電源不良→電源部

音が飛ぶ→サーボ回路、トラッキング・フォーカス調整不良

サーチ不良→サーボ回路、トラッキング・フォーカス調整不良

音が出ない→アナログ回路部、デジタルフィルタ部

4. カセットデッキ

症状：パイロットランプ、音が出るか、デッキのオペレーション

状態：録音再生特性、ダビング状態

原因ブロックを仮定する知識

音が出ない→電源部、muting回路

デッキのオペレーション不良→マイコン部

パイロットランプがつかない→電源部

再生不良、録音良→ヘッドアンプ部

再生良、録音不良→録音アンプ部、バイアス回路部

録音・再生不良、デッキのオペレーション良→アナログ用電源部、muting回路

録音・再生不良、ダビング良→アナログスイッチ、マイコン部

ダビング不良→マイコン部

録音・再生特性不良（高音出ず）→バイアス回路

付録4

故障修理の修理メモ (抜粋)

1. CD

パターン切れる (プリント基板割れ)

アースパターン浮きうき→アナログEとデジタルEの分離→正常な電圧がアナログ系に (デジタルフィルタ、LPF等に) 供給されない。→歪み発生

トラッキングOK、すべての制御OK、→DA変換部か→左右同じ症状→左右分離前→デジタルフィルタ周辺→電圧チェック、+Vcc、-Vccが異常 (+6.3V、-3.7V) ただし合計して10Vなので±5Vの0電位 (アースレベル) が違うと予想できる。→2つのアース間の導通チェック→導通無し→パターンチェック→基板割れ発見

2. アンプ

ツェナーショート→FET破壊 (電源部) アンプ不動作、音出ず→コントロールアンプに電源が供給されず。

パワー段OK→コントロール部電源チェック→-側電圧こず。→イコライザ部はどうか→来てない。→ブリ電源不良→-側平滑回路の電解コンデンサまでOK→定電圧回路のツェナーDiチェック、ショートしている。

3. アンプ

プロテクタ入る。→バイアス制御用のTrのC-E間ショートで片方のChのコールド側のパワーTrすべてoff→DCバランスくずれる。

DCバランスチェックOK (DCもれではない。) 負荷無し時プロテクタ入らず。その状態で波形チェック。片ChOK→もう一方のCh不良とわかる。→Hot側波形。リップルのっているがOK。コールド側 方形波入力で異常 (出力) →パワーTrバイアス抵抗両端電圧測定→cold側 0V、パワーTr offとわかる。→Tr単体でチェック OK→バイアスかかっていない→バイアス制御用のTrのC-E間ショートとわかる。

4. カーオーディオ レシーバ

FMはいらす

シンセサイザ入力OSCのハンダ不良

5. カーオーディオ

muteかかりっぱなし CD入力ジャック 不良

6. カーオーディオRE-5

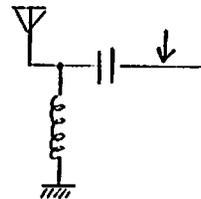
パワーアンプ 片Ch不良、入力追いかけてパワーアンプまで正常。パワーアンプ用の muteかかっている。mute用Tr不良。C-E間ショート。muteピンアースされる。

7. AM感度不足

入力のチョークコイルショート。(約1Ω) 3.3mH

正常の場合、数Ωを示す。調べてわかった。

ANTを ↓ に接続するとOK。



8. CD

リードインせず。フォーカスコイル用バッファアンプ

Q12 2SC2001、 Q13 2SA952 とともに不良。

9. CD

不動作。CXD1125Q、 プロセッサIC 不良。

マイコンの動作あたりから調べてみてわかった。

10. CD

不動作。まわりださず。

レーザー弱い(直接見て、ほんとはしてはいけない。)

① fQ1 不良(2SA952) 交換、→レーザーOK

② CXA1082A 不良

11. ampAu-α607i

ヒューズとび。電源電解C12.000μFショート。

12. カーオーディオ RE-5

FM 不動作

AMはOK、FM時「ザー」ノイズあり。

VT 変化せず。シンセサイザにOSC入力なし。→ハンダ接触不良。

13. カーオーディオ RE-5

カセットに切り変わらず。Tr905 不良。
Tape用B+がかかってもRadio +Bがきれず。
なぜか、D901  を通して

R+B control してRadio+B出てしまう。Tape muteしてしま
う。

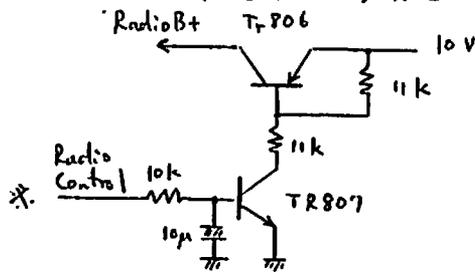
14. カーオーディオ RE-5

カセット切り変え時にもRadio+Bでっぱなし。
Tape Tapeにならず。メカ動く。
Radio+B common (AM FM共通) +供給されている。
D901  あたりから調べる。

RadioをONにするには2通り

- ①メカのマイクロSWがTurn onする。
- ②PLL ICの#30 pin PeeがLow (T-monの時一時的5秒間L)
これはOKだった。]

Tr807 (チップTr) がC-E間導通、Tr806をONしてしまう。



15. カーオーディオ RE-5

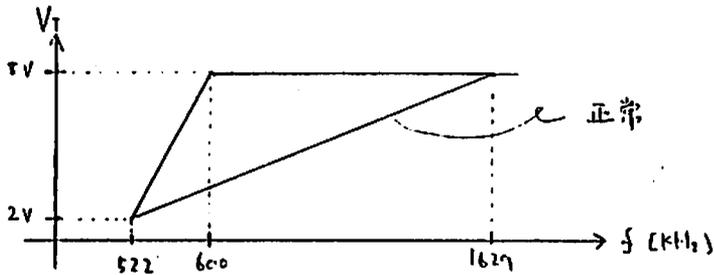
LCh出ず。パワーアンプ用mute Tr Tr701A 不良。
C-E間ショートでmuteがこなくても、#10 pinをアースしてしまい、パワー
アンプmute。

16. カーオーディオ RE-5 (ほかに2件)

AM不良、

PLLシンセサイザ 途中まで 522kHz~600kHz ぐらいまで動く。

V_T (チューニング電圧) の変化が大き過ぎる。



LC回路のCが大き過ぎると予想する。OSC周辺を調べた。C405、510pFの不良5kΩを示す。(無限大∞のはず。)

17. カーオーディオ RE-5

AM不動作、

T405 2次側断線 (IFT)

キャリアをおいかけてわかった。

18. チューナ T-E910

16と同じ症状。

VT用B電圧供給用Tr不良

VTは1~22Vまで変化すべき。1~9Vでストップしていた。

19. カーオーディオ RE-5

Tape→Radio OK

Radio→Tape 切り替え時、LChのみ音が残る。パワーamp用プロテクタIC (mute)、IC652、TA7324Pの不良。

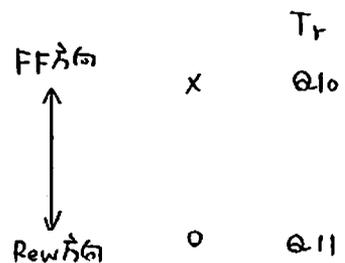
20. CD CD-α607i 不動作

①アースパターン浮き

②Q10 不良

Q10はSled motor用 (+側) Amp

ヘッドベース動かさずでわかった。



21. CD CD-α607i

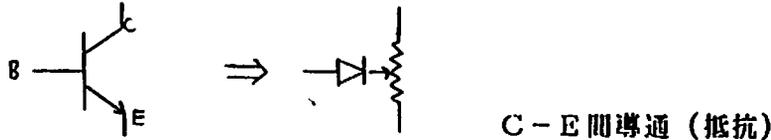
不動作

レーザDへいくパターン ぎれ。

レーザ信号ONでTr (fQ1) もon、コレクター無負荷のように見えたので、パターンチェック fL1 10μHはOK、

22. CD CD-α607i

トラッキング用 Tr Q15 2SC952 不良



トラッキング不良 IC2 CXA1082よりの信号は正常 ±対称→トラッキング
 コイル±不対称

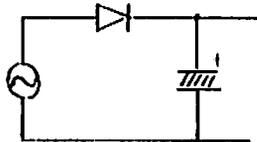
23. amp Au-α507i

1Q1L 不良→プロテクタ

24. イコライザ SE-80

プロテクタ 1 動作→ミューティング。

プロテクタ回路内のEC 不良、容量抜けて。



平滑回路後にリプル出でいて、DC成分30Vのところ
 が17Vぐらいしかこず。

ECの外見、パンクしかけていて中のペーパーがはみだしていた。

25. アンプ

ツェナーショート→FET破壊 (電源部) アンプ不動作、音出ず→コントロールアンプ
 に電源が供給されず。

パワー段OK→コントロール部電源チェック→一側電圧こず。→イコライザ部はどうか→
 来てない。→プリ電源不良→一側平滑回路の電解コンデンサまでOK→定電圧回路のツェ
 ナーDiチェック、ショートしている。

ここまではよくある電源部の故障パターン。ツェナーDiを交換→SW ON, 依然不
 良→ツェナー端子電圧測定→0V→ツェナーに電流れていない→FET不良の疑い (Op
 en) →観察→FET破壊していた→FET交換→OK

故障の原因 ツェナーショート→FETにVかかる。破壊

付録5

修理メモの例（No. 2）から浮かび上がる修理の詳しい様子

計測の状況だけでなく、結果から何を判断しているかや『作業場の基礎知識』に属するものも混在して書かれている。

ボンネットをあける。上部ヒートシンク、基板の部品面、パターンが見えるように底板をはずす。しばらく観察。電源プラグを差す。パワーSWを入れる。パイロットランプ（プロテクタ動作中インジケータを兼ねている）が点滅しているのを確認。その他のインジケータランプ（LED）が点灯しているのを確認。（LED用電源はOK）パワートランジスタを中心に付近の部品を見る、指で触る、温度を調べる。あまり熱くない。約10数秒後プロテクタ解除されるリレーの音（カチン）を確認。

発振器の用意をする。約1kHz、200mV、正弦波、をLINE-IN入力に入れる。スピーカ端子に8Ωのダミー抵抗とオシロスコープを両チャンネルに接続する。徐々にボリュームをあげていく。オシロに波形が現われない。オシロの入力をDCにして直流分を見る準備をする。スピーカスイッチをON・OFFしてスピーカ端子の直流洩れの量を調べる。ごくわずかの直流洩れなので異常なしと判断。

電源回路に着目。電源基板に入る前のトランス2次側のACを電圧計で測る。回路図の規定値通り出ている。メインアンプ用電源部のブリッジダイオード、平滑コンデンサ、と順に±の電圧を測定。規定値出ている。メインアンプ基板のパワートランジスタのコレクタに正規の±電圧がきているかを測定。出ている。ここでパワートランジスタに触る。かすかに暖かい程度である。普通の状態である。最終段のバイアス電流がだいたいまともであろうとわかる。

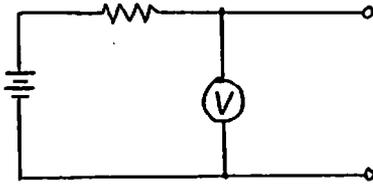
プリアンプ用電源回路、定電圧回路に着目。プリアンプ基板の±電圧を測定。+側は正常だが-側に電圧が来っていない。そのためプリアンプ部が動作していなかったのだとわかる。試しに約1kHz、300mV、正弦波、をMAIN-IN入力に入れてみる。スピーカ端子につないであるオシロに波形が観測される。発振器の出力を方形波にして同様に観測する。正常なきれいな波形を観測。発振器の出力電圧を変化させてメインアンプのリニアリティを調べる。良好な結果が観測される。メインアンプ部は正常であるとほぼ断定する。

再びプリアンプ用電源回路に戻る。電源基板のプリアンプ用リップルフィルタの-側、コレクタまで直流電圧が来ている。エミッタは0V。リップルフィルタ不動作とわかる。

ツェナーダイオード両端の電圧測定。0V。ツェナーダイオードがショートしている疑いがある。電源を切断してインサーキットでツェナーダイオードの導通を調べる。念のためはずして導通検査、抵抗値を調べる。ほぼ0Ω。ツェナーダイオードの不良と断定した。

リップルフィルタのトランジスタが壊れている可能性があるのでトランジスタの検査を行う。B、C、Eのハンダを吸い取りパターンから浮かせる。 V_{BE} 、 V_{BC} をテストに内蔵の治具で測定。

サンワのテスターには



こういう導通計がついている。テスト端子解放で電池電圧Eが読み取れる。テスト端子にダイオードなどをつなぐと順方向電圧が測定できる。順方向電圧が0.8Vをわると内蔵のブザーも鳴る。

V_{BE} 、 V_{BC} 共に0.65Vで正常。逆方向は導通なし、 $\infty \Omega$ 。コレクター-エミッタ間も $\infty \Omega$ 。トランジスタは壊れていないと判断する。ハンダをつけてもとの状態にする。

コンデンサがショートしていないかを一応チェックする。テストの針の戻り具合からショートも容量抜けもないと判断する。

ツェナーダイオードを新品に交換する。

電源を投入して電源回路の電圧を測定する。規定値になっているのを確認する。約1kHz、200mV、正弦波、をLINE-IN入力に入れる。スピーカ端子に8 Ω のダミー抵抗とオシロスコープを両チャンネルに接続する。徐々にボリュームをあげていく。波形を観測する。しばらくエージングして再び動作を確認。

ボンネット・底板などをとりつける。修理完了。

付録 6

計測カードのフォーマット

計測カード				No.	
種類		事例名		計測番号	
名前				対象物	
計測内容					
前処理					結果
手順番号	内容	状態	計測	No	
結果の解釈、判断に使う知識番号					
わかったこと					
モデルを解釈する知識	利用する計測の番号				
わかったこと					
後処理					
計画	チェックリスト/モデル				
備考					

計測カードの記入例 1

計測カード

No.

種類	アンプ	事例名	アンプ修理 2	計測番号	6
名前	電圧測定		対象物	Tr Q1	
計測内容	トランジスタの各電極の電圧を測定する				
前処理					結果
手順番号	内容	状態/計測	計測 No		
1.	Q1 の コレクタ電圧	計		4.2 [V]	
2.	エミタ "	計		0 [V]	
3.	ベース "	計		0 [V]	
結果の解釈、判断に使う知識 番号					
1, 2	コレクタ電圧あり、エミタ電圧なしの場合、リプルフィルタ と見えず不動作				
3	ベース電圧 0[V] とき 定電圧源不良				
わかったこと					
モデルを解釈する知識	利用する計測の番号				
わかったこと					
後処理					
計画	チェックリスト/ <u>モデル</u>				
	定電圧源の状態				
備考					

計測カードの記入例 2

計測カード

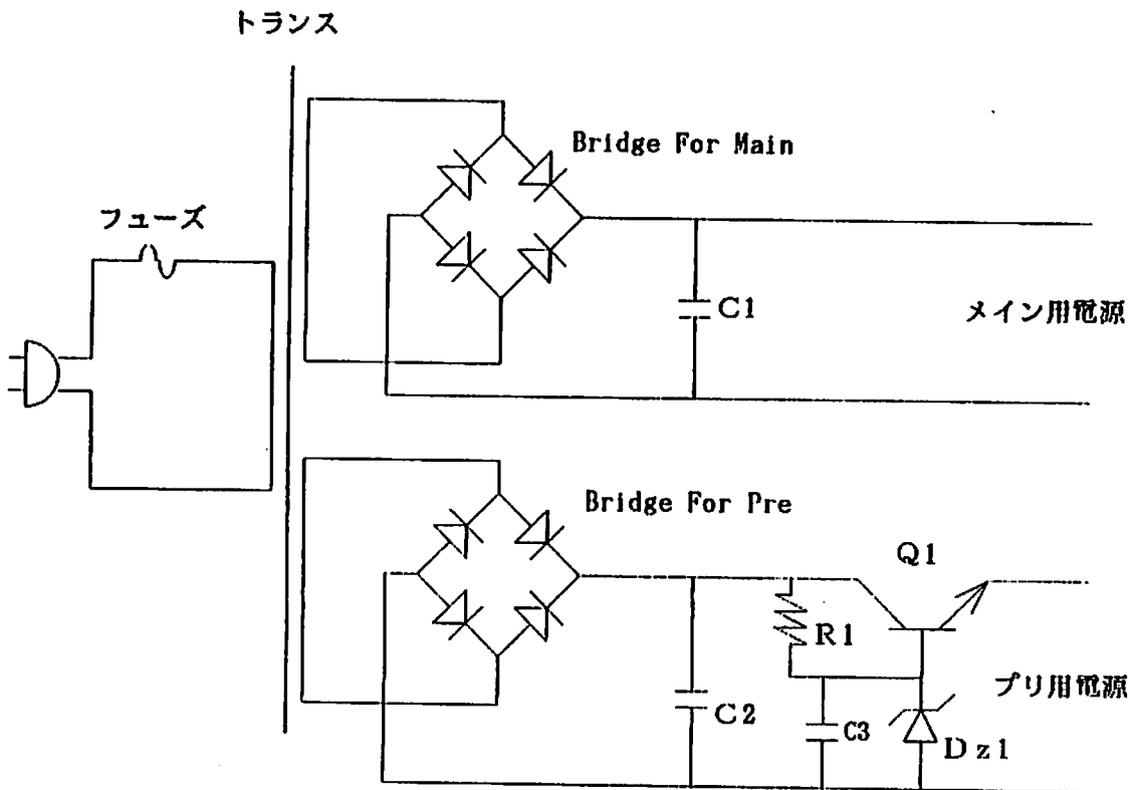
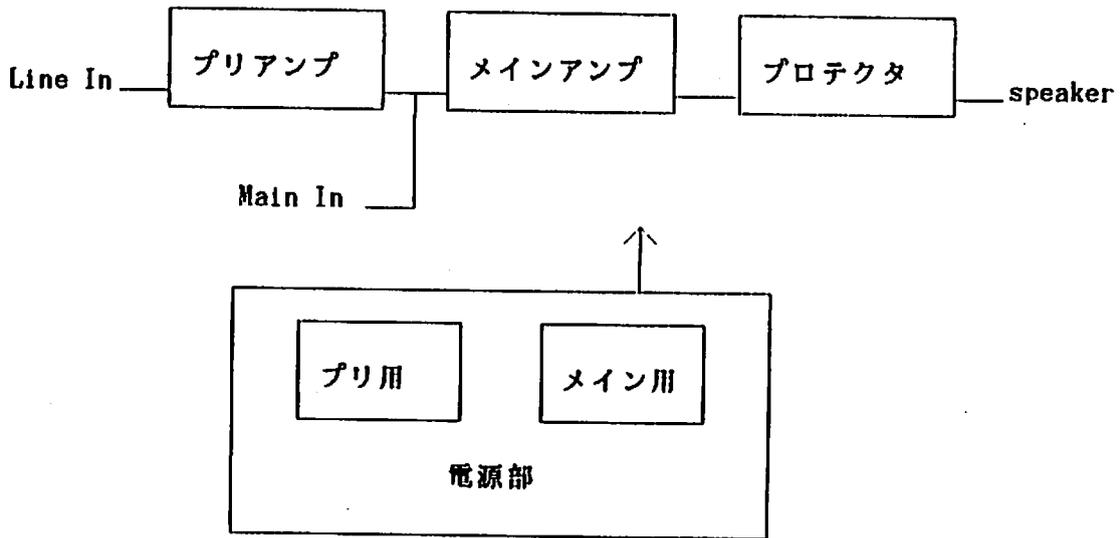
No.

種類	アンプ	事例名	アンプ修理	計測番号	7
名前	定電圧源の状態		対象物	定電圧回路	
計測内容	定電圧源不良の状態を確かめ調べる				
前処理					結果
手順番号	内容	状態/計測	計測 No		
1.	"Zナダイオード" D21 の検査	計		不良, ショット	
2.	リカルフィルタ C1 の検査	計		良	
結果の解釈、判断に使う知識番号					
	"Zナダイオード" の不良または C1 の不良により定電圧源不良となる。 "Zナ" ショットは 定電圧源を 0V にする				
わかったこと	"Zナ" ダイオード不良が原因で 定電圧源不良となる。				
モデルを解釈する知識	利用する計測の番号				
わかったこと					
後処理					
計画	チェックリスト / <u>モデル</u>				
	Q2 の不良の可能性があるので Q2 を検査する				
備考	故障連鎖の疑い。				

付録7

ツールを用いた知識ベースのプロトタイピングと実行例

アンプ修理におけるブロック図と電源部の回路図例



ツール上の知識記述のソースリスト

I t e m

I n q u i r y

R u l e B a s e (大局的診断相のもの)

C a u s e

R u l e B a s e (局所的診断相のもの)

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 9 January 1990 at 2:52:52 pm!'

ItemDescription at: 'PilotLamp' asSymbol
definitionString: 'PilotLamp
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()
note: " "" !

ItemDescription at: 'Protector' asSymbol
definitionString: 'Protector
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()
note: " "" !

ItemDescription at: 'ResponseLineSp' asSymbol
definitionString: 'ResponseLineSp
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()
note: " "" !

ItemDescription at: 'ResponseMainSp' asSymbol
definitionString: 'ResponseMainSp
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()
note: " "" !

ItemDescription at: 'Smell' asSymbol
definitionString: 'Smell
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()
note: " "" !

ItemDescription at: 'Smoke' asSymbol
definitionString: 'Smoke
valueType: Boolean
valueCheck: nil
requisite: false
ifNeeded: #inquire withArguments: #()
ifAdded: nil withArguments: #()

note: " " !

付録7 4

ItemDescription at: 'Sound' asSymbol

definitionString: 'Sound

valueType: Boolean

valueCheck: nil

requisite: false

ifNeeded: #inquire withArguments: #()

ifAdded: nil withArguments: #()

note: " " !

File: InquiryDescriptionFile
Sat Feb 10 18:43:41 1990

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 9 January 1990
at 2:51:51 pm!'

InquiryDescription at: 'Amp-acForMain' asSymbol
definitionString: 'Amp-acForMain
"Yes or No"
message: "The transformer that supply AC for Main works properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-acForPre' asSymbol
definitionString: 'Amp-acForPre
"Yes or No"
message: "The transformer that supply AC for Pre works properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-bridgeMain' asSymbol
definitionString: 'Amp-bridgeMain
"Yes or No"
message: "Does the bridge diode in Main work properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-bridgePre' asSymbol
definitionString: 'Amp-bridgePre
"Yes or No"
message: "Does the bridge diode in Pre work properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-cl' asSymbol
definitionString: 'Amp-cl
"Yes or No"
message: "Does the part cl work properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-c2' asSymbol
definitionString: 'Amp-c2
"Yes or No"
message: "Does the part c2 work properly ?"
duration: #temporary
asAssertion: false
note: " "" !

InquiryDescription at: 'Amp-c3' asSymbol
definitionString: 'Amp-c3
"Yes or No"
message: "Does the part c3 work properly ?"

duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-dz1' asSymbol
definitionString: 'Amp-dz1
"Yes or No"
message: "Does the part dz1 work properly ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-ql' asSymbol
definitionString: 'Amp-ql
"Yes or No"
message: "Does the part ql work properly ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-r1' asSymbol
definitionString: 'Amp-r1
"Yes or No"
message: "Does the part r1 work properly ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-shortc1' asSymbol
definitionString: 'Amp-shortc1
"Yes or No"
message: "Is c1 short ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-shortc2' asSymbol
definitionString: 'Amp-shortc2
"Yes or No"
message: "Is c2 short ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-shortc3' asSymbol
definitionString: 'Amp-shortc3
"Yes or No"
message: "Is c3 short ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-shortdz1' asSymbol
definitionString: 'Amp-shortdz1
"Yes or No"
message: "Is dz1 short ?"
duration: #temporary
asAssertion: false
note: " " !

InquiryDescription at: 'Amp-vACForMain' asSymbol
definitionString: 'Amp-vACForMain

"From the key board"
 message: "Voltage AC Trans For Main ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vACForPre' asSymbol
 definitionString: 'Amp-vACForPre
 "From the key board"
 message: "Voltage AC Trans For Pre ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vc1' asSymbol
 definitionString: 'Amp-vc1
 "From the key board"
 message: "Voltage vc1 ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vc2' asSymbol
 definitionString: 'Amp-vc2
 "From the key board"
 message: "Voltage vc2 ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vc3' asSymbol
 definitionString: 'Amp-vc3
 "From the key board"
 message: "Voltage vc3 ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vdzl' asSymbol
 definitionString: 'Amp-vdzl
 "From the key board"
 message: "Voltage vdzl ?"
 duration: #temporary
 conversion: #asNumber
 initialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vPowerForMainOut' asSymbol
 definitionString: 'Amp-vPowerForMainOut
 "From the key board"
 message: "Voltage vPowerForMainOut ?"

duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vPowerForPreOut' asSymbol
 definitionString: 'Amp-vPowerForPreOut
 "From the key board"
 message: "Voltage vPowerForPreOut ?"
 duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vq1b' asSymbol
 definitionString: 'Amp-vq1b
 "From the key board"
 message: "Voltage vq1b ?"
 duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vq1c' asSymbol
 definitionString: 'Amp-vq1c
 "From the key board"
 message: "Voltage vq1c ?"
 duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vq1e' asSymbol
 definitionString: 'Amp-vq1e
 "From the key board"
 message: "Voltage vq1e ?"
 duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Amp-vr1' asSymbol
 definitionString: 'Amp-vr1
 "From the key board"
 message: "Voltage vr1 ?"
 duration: #temporary
 conversion: #asNumber
 InitialAnswer: ""
 valueCheck: nil
 note: " "" !

InquiryDescription at: 'Item-PilotLamp' asSymbol
 definitionString: 'Item-PilotLamp
 "Yes or No"
 message: "Is the pilot lamp on?"
 duration: #temporary
 asAssertion: false

note: " "" !

InquiryDescription at: 'Item-Protector' asSymbol

definitionString: 'Item-Protector

"Yes or No"

message: "Is the protector working well?"

duration: #temporary

asAssertion: false

note: " "" !

InquiryDescription at: 'Item-ResponseLineSp' asSymbol

definitionString: 'Item-ResponseLineSp

"Yes or No"

message: "Do you recognize a proper signal coming out of the terminal of the speakers when you put a signal into the Line-Input ?"

duration: #temporary

asAssertion: false

note: " "" !

InquiryDescription at: 'Item-ResponseMainSp' asSymbol

definitionString: 'Item-ResponseMainSp

"Yes or No"

message: "Do you recognize a proper signal coming out of the terminal of the speakers when you put a signal into the Main-Input ?"

duration: #temporary

asAssertion: false

note: " "" !

InquiryDescription at: 'Item-Smell' asSymbol

definitionString: 'Item-Smell

"Yes or No"

message: "Do you smell something burning ?"

duration: #temporary

asAssertion: false

note: " "" !

InquiryDescription at: 'Item-Smoke' asSymbol

definitionString: 'Item-Smoke

"Yes or No"

message: "Is there any smoke?"

duration: #temporary

asAssertion: false

note: " "" !

InquiryDescription at: 'Item-Sound' asSymbol

definitionString: 'Item-Sound

"Yes or No"

message: "Do you hear the sound when you operate properly ?"

duration: #temporary

asAssertion: false

note: " "" !

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 8 February 1990
at 12:18:12 am!'

(RuleBaseDescriber

name: #FirstConsultationOfAmp

definition: ' workingMemory: SystemWorkingMemory

note: " ""

rules: #('rule1

if: #("@(Item #Sound false)" "@(Item #ResponseLineSp false)")

"@(Item #ResponseMainSp true)" "@(Item #PilotLamp true)")

"@(Item #Protector true)")

then: #("@(Possible #FaultInPowerForPre)")

"@(Possible #FaultInPreBlock)")

explain: "音が出ない かつ

LineIn-Speaker間に応答がない かつ

MainIn-Speaker間に応答がある かつ

パイロットランプ点灯 かつ

プロテクタ正常動作 ならば

リアンプ用電源に故障の原因がある可能性がある または

リアンプに故障の原因がある可能性がある ""

'rule2

if: #("@(Item #Sound false)" "@(Item #Smell true)")

"@(Item #ResponseMainSp false)")

then: #("@(Possible #FaultInMainBlock)")

explain: "音が出ない かつ

異常なおいがする かつ

MainIn-Speaker間に応答がない ならば

メインアンプに故障の原因がある可能性がある ""

'rule3

if: #("@(Item #Sound false)" "@(Item #Protector false)")

then: #("@(Possible #FaultInProtector)")

"@(Possible #FaultInMainBlock)")

explain: "音が出ない かつ

プロテクタ動作異常 ならば

メインアンプに故障の原因がある可能性がある ""

'rule4

if: #("@(Item #PilotLamp false)")

then: #("@(Possible #FaultInPowerBlock)")

explain: "パイロットランプ不点灯 ならば

電源部に故障の原因がある可能性がある ""

))

File: CauseDescriptionFile
Sat Feb 10 18:37:03 1990

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 9 January 1990
at 2:53:06 pm'

CauseDescription at: 'BrokenBridgeMain' asSymbol
definitionString: 'BrokenBridgeMain
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Bridge Rectifier for Main is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenBridgePre' asSymbol
definitionString: 'BrokenBridgePre
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Bridge Rectifier for Pre is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenC1' asSymbol
definitionString: 'BrokenC1
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "C1 is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenC2' asSymbol
definitionString: 'BrokenC2
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "C2 is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenC3' asSymbol
definitionString: 'BrokenC3
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "C3 is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenDZ1' asSymbol
definitionString: 'BrokenDZ1
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "Zener Diode DZ1 is wrong."
needOfChecking: true
note: " "" !

CauseDescription at: 'BrokenQ1' asSymbol
definitionString: 'BrokenQ1
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil

message: "Transister Q1 is wrong."
needOfChecking: true
note: " " !

付録7 12

CauseDescription at: 'BrokenTransACMain' asSymbol
definitionString: 'BrokenTransACMain
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "A part of the transformer for Main is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'BrokenTransACPre' asSymbol
definitionString: 'BrokenTransACPre
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "A part of the transformer for Pre is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'FaultInMainBlock' asSymbol
definitionString: 'FaultInMainBlock
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Main-Block is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'FaultInPowerBlock' asSymbol
definitionString: 'FaultInPowerBlock
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Power supply unit is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'FaultInPowerForMain' asSymbol
definitionString: 'FaultInPowerForMain
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Power supply unit for Main-block is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'FaultInPowerForPre' asSymbol
definitionString: 'FaultInPowerForPre
ifPossible: #backwardInference:tactics:assumption:
withArguments: #(PowerBlockForPreConsultation FirstMatch "@(Cause 1)"
ifAffirmed: nil withArguments: nil
message: "The Power supply unit for Pre-block is wrong."
needOfChecking: false
note: " " !

CauseDescription at: 'FaultInPreBlock' asSymbol
definitionString: 'FaultInPreBlock
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Pre-Block is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'FaultInProtector' asSymbol

definitionString: 'FaultInProtector
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "The Protector is wrong."
needOfChecking: true
note: " " !

CauseDescription at: 'jhkdjg' asSymbol
definitionString: 'jhkdjg
ifPossible: nil withArguments: nil
ifAffirmed: nil withArguments: nil
message: "原因を示す出力文"
needOfChecking: false
note: "コメント文" !

File: PowerBlockForPreConsultation
Sat Feb 10 18:51:37 1990

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 8 February 1990
at 12:18:15 am!'

(RuleBaseDescriber

name: #PowerBlockForPreConsultation

definition: ' workingMemory: SystemWorkingMemory

note: "This is a set of rules for counsulting about an amplifier."

rules: #('rule1

if: #("@(AccessValue #Amp #vq1e (less 5))")

then: #("@(Symptom #VQ1eIsWrong)")

explain: "Q1のエミッタ電圧が5より小 ならば
エミッタ電圧不良である"

'rule2

if: #("@(Symptom #VQ1eIsWrong)")

"@(AccessValue #Amp #vc2 (greater 40))")

"@(AccessValue #Amp #q1 true)")

then: #("@(Possibility #VSourceWrong)")

explain: "Q1のエミッタ電圧不良 かつ
C2の電圧が40より大 かつ
Q1が良 ならば
定電圧源不良の可能性はある"

'rule3

if: #("@(AccessValue #Amp #vdz1 (less 5))")

then: #("@(Possibility #VSourceWrong)")

explain: "ツェナーダイオードDz1の電圧が5より小 ならば
定電圧源不良の可能性はある"

'DiodeDZ1

if: #("@(AccessValue #Amp #shortdz1 true)")

"@(Possibility #VSourceWrong)")

"@(AccessValue #Amp #c3 true)")

then: #("@(Cause #BrokenDZ1)")

explain: "ツェナーダイオードDz1がショートしている かつ
定電圧源不良の可能性はある かつ
C3は良 ならば
原因はツェナーダイオードDz1である "

'C3short

if: #("@(AccessValue #Amp #shortc3 true)")

"@(Possibility #VSourceWrong)")

"@(AccessValue #Amp #dz1 true)")

then: #("@(Cause #BrokenC3)")

explain: " C3がショートしている かつ
定電圧源不良の可能性はある かつ
ツェナーダイオードDz1は良 ならば
原因はC3である "

'TrQ1

if: #("@(Symptom #VQ1eIsWrong)")

"@(AccessValue #Amp #vdz1 (between 33 37))")

"@(AccessValue #Amp #vq1c (between 38 45))")

then: #("@(Cause #BrokenQ1)")

explain: "Q1のエミッタ電圧不良 かつ
ツェナーダイオードDz1の電圧が33と37の間 かつ
Q1のコレクタ電圧が38と45の間 ならば
原因はQ1である"

'rule4

if: #("@(Symptom #VC2IsWrong)")

then: #("@(Possibility #RoughDCSourceWrong)")

explain: "C2の電圧不良 ならば
直流不良の可能性がある"

'TransPre

```
if: #( "@( Possibility #RoughDCSourceWrong )"
        "@( Symptom #VACForPreIsWrong )"
        "@( AccessValue #Amp #bridgePre true )" )
then: #( "@( Cause #BrokenTransACPre )" )
explain: "直流不良の可能性ある かつ
          プリ用交流不良 かつ
          プリ用ブリッジダイオード良 ならば
          原因はプリ用トランス巻き線である ""
```

'BridgeDiodePre

```
if: #( "@( Possibility #RoughDCSourceWrong )"
        "@( AccessValue #Amp #vACForPre (greater 35) )" )
then: #( "@( Cause #BrokenBridgePre )" )
explain: "直流不良の可能性ある かつ
          プリ用交流電圧が35より大 ならば
          原因はプリ用ブリッジダイオードである""
```

'rule5

```
if: #( "@( AccessValue #Amp #vACForPre (less 35) )" )
then: #( "@( Symptom #VACForPreIsWrong )" )
explain: "プリ用交流電圧が35より小 ならば
          プリ用交流不良""
```

'rule6

```
if: #( "@( Symptom #VC2IsWrong )"
        "@( AccessValue #Amp #bridgePre true )" )
then: #( "@( Symptom #VACForPreIsWrong )" )
explain: "C2の電圧不良 かつ
          プリ用ブリッジダイオード良 ならば
          プリ用交流不良""
```

'rule7

```
if: #( "@( AccessValue #Amp #vc2 (less 40) )" )
then: #( "@( Symptom #VC2IsWrong )" )
explain: "C2の電圧40より小 ならば
          C2の電圧不良""
```

))

File: AmpConsultation
 Sat Feb 10 18:35:25 1990

'From Tektronix Smalltalk-80 version T2.2.0c, of December 10, 1986 on 9 January 1990
 at 2:52:43 pm'

```
( ConsultationControllerDescriber
  name: #AmpConsultation
  consultationName: 'Demonstration of consulting'
  strategy: #breadthFirst
  ruleBase: #FirstConsultationOfAmp
  items: #(PilotLamp Protector ResponseLineSp ResponseMainSp Smell Smoke Sound )
  causes: #(BrokenBridgeMain BrokenBridgePre BrokenC1 BrokenC2 BrokenC3 BrokenC4
1 BrokenQ1 BrokenTransACMain BrokenTransACPre FaultInPowerBlock FaultInPowerForBlock
In FaultInPowerForPre FaultInPreBlock FaultInProtector )
  requisites: #(PilotLamp Protector ResponseLineSp ResponseMainSp Smell Smoke Sound )
  note: ' ' )
```

ブラウザで見た知識の記述および
実行結果の画面のハードコピー

Deflnition of AmpConsultation

<p>----- definition Item list Cause list -----</p>	<p>----- PilotLamp Protector ResponseLineSp ResponseMainSp Smell Smoke Sound -----</p>
---	---

ResponseMainSp

valueType: Boolean
 valueCheck: nil
 requisite: false
 IfNeeded: #Inquire withArguments: #()
 IfAdded: nil withArguments: #()
 note: "

Inquiry Browser

Amp-acForMain
Amp-acForPre
~~Amp-bridgeMain~~
Amp-bridgePre
Amp-c1
Amp-c2
Amp-c3
Amp-dz1
Amp-q1
Amp-r1
Amp-shortc1
Amp-shortc2
Amp-shortc3
Amp-shortdz1
Amp-vACForMain
Amp-vACForPre
Amp-vc1
Amp-vc2
Amp-vc3
Amp-vdz1

Amp-bridgeMain

"Yes or No"
message: 'Does the bridge diode in Main work properly ?'
duration: #temporary
asAssertion: false
note: ''

Definition of AmpConsultation

 definition
 Item List
 Cause List

 BrokenBridgeMain
 BrokenBridgePre
 BrokenC1
 BrokenC2
 BrokenC3
 BrokenDZ1
 BrokenQ1
 BrokenTransACMain
 BrokenTransACPre
 FaultInPowerBlock
 FaultInPowerForMain
 FaultInPowerForPre
 FaultInPreBlock
 FaultInProtector

BrokenTransACMain

IfPossible: nil withArguments: nil
 IfAffirmed: nil withArguments: nil
 message: 'A part of the transformer for Main is wrong.'
 needOfChecking: true
 note: ''

Definition of PowerBlockForPreConsultation

definition of RuleBase
Rules

PowerBlockForPreConsultation

workingMemory: SystemWorkingMemory

note: 'This is a set of rules for counsulting about an amplfier.'

Definition of PowerBlockForPreConsultation

 definition of RuleBase

Rules

 rule1

rule2

rule3

DiodeDZ1

C3short

TrQ1

rule4

TransPre

BridgeDiodePre

rule5

rule2

if: #('@(Symptom #VQ1elsWrong)'
 '@(AccessValue #Amp #vc2 (greater 40))'
 '@(AccessValue #Amp #q1 true)')

then: #('@(Possibility #VSourceWrong)')

explain: 'Q1のエミッタ電圧不良 かつ
 C2の電圧が40より大 かつ
 Q1が良 ならば
 定電圧源不良の可能性がある'

Output Window

Voltage vc2 ?

42

Does the part q1 work properly ?

Y e s

Does the part dz1 work properly ?

Inference Process Window

<rule2> Is selected

self prove FirstMatch - @(Symptom #VQ1eIsWrong)

<rule1>

<rule1> Is selected

<rule1> Is affirmed

...termination

<rule2> Is affirmed

termination

Explain Window

rule2

Q1のエミッタ電圧不良 かつ

C2の電圧が40より大 かつ

Q1が良 ならば

定電圧源不良の可能性がある