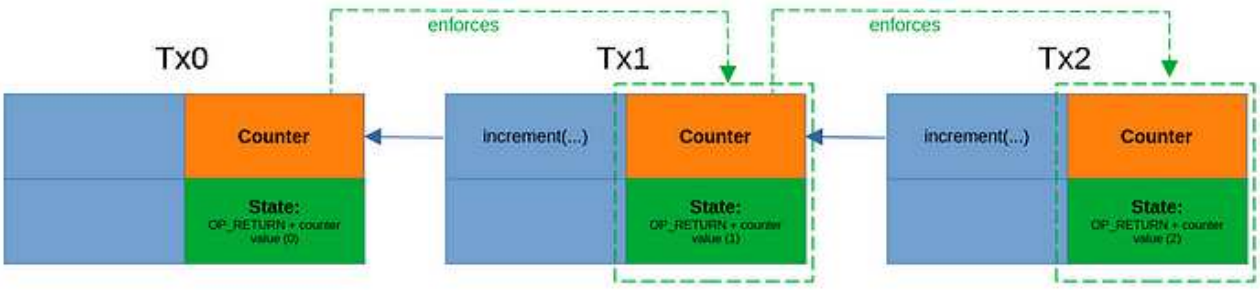
Bitcoin Covenants

`OP_CAT` だけでは無理そうな気もするが、サンプル実装もあるのでそちらの挙動を確認する。
まだ mainnet/testnet を含めて `OP_CAT` は動かないのでsignetでやっているようだ。独自にsignetを立てているのかと思ったがmempool.space の signet でも見えるようだ。

sCrypt 版は実行結果があるのでスクリプトを確認したが、巨大だし機械的に作ってあるのかごまかしのコードなどもあるのか見ても分からない。
元が JavaScript 風の記述になっているのでそちらから読み解いた方がよさそうだが、それはもはや検証ではないと思う。

再帰的コベナントの例

Bitcoin OP_CAT Use Cases Series #4: Recursive Covenants - Implementation - Delving Bitcoin
状態有りのコベナント



カウンターのサンプルで、vout#1 のデータが 1ずつインクリメントされている。
sCrypt-Inc/scrypt-btc-counter: A simple contract to increment a counter using recursive covenants

- state#0 tx
  - vout#1: `OP_RETURN + 0100`
- state#1 tx state#0 の vout#0
  - vout#1: `OP_RETURN + 0101`
- state#2 tx state#1 の vout#0
  - vout#1: `OP_RETURN + 0x02`

state#0 は最初の送金なのでスクリプトは関係ない。
state#1 は state#0 から支払っているのでスクリプトが見える。

非常に長いが全部載せる。

`OP_NOP` などもあるので不要な命令もありそうだが、それはハッシュ値を整えるために挟んだのだろうか。

ともかく、頭で考えつくようなスクリプトでないことは分かった。

2行目の `79be....1798` はいわゆるベースポイント `G` のX座標である。4行目にも2組含まれている。

1行目の `6a` はもしかしたら `OP_RETURN` か？ これはよくわからない。

```
script
   OP_PUSHBYTES_1 6a
   OP_PUSHBYTES_32 79be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f81798
   OP_PUSHDATA1
   7bb52d7a9fef58323eb1bf7a407db382d2f3f2d81bb1224f49fe518f6d48d37c7bb52d7a9fef58323eb
   1bf7a407db382d2f3f2d81bb1224f49fe518f6d48d37c79be667ef9dcbbac55a06295ce870b07029bfc
   db2dce28d959f2815b16f8179879be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b1
   6f81798
   OP_PUSHBYTES_66
   f40a48df4b2a70c8b4924bf2654661ed3d95fd66a313eb87237597c628e4a031f40a48df4b2a70c8b49
   24bf2654661ed3d95fd66a313eb87237597c628e4a0310000
   OP_PUSHBYTES_8 0000000000000000
   OP_NOP
   OP_NOP
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
   OP_PICK
   OP_PUSHBYTES_1 1b
```

```
OP_PICK
OP_PUSHBYTES_1 1b
OP_PICK
OP_PUSHBYTES_1 1b
OP_PICK
OP_PUSHBYTES_1 1b
OP_PICK
OP_PUSHBYTES_1 1b
OP_PICK
OP_PUSHBYTES_1 1b
OP_PICK
OP_NOP
OP_PUSHNUM_16
OP_PICK
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_SHA256
OP_0
OP_PICK
OP_PUSHNUM_2
OP_PICK
OP_PUSHNUM_1
OP_CAT
OP_EQUAL
OP_VERIFY
OP_PUSHBYTES_1 12
OP_PICK
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_2
OP_CAT
OP_PUSHBYTES_1 11
OP_PICK
OP_PUSHNUM_16
OP_PICK
OP_CAT
OP_PUSHNUM_15
```

```
OP_PICK
OP_CAT
OP_PUSHNUM_14
OP_PICK
OP_CAT
OP_PUSHNUM_13
OP_PICK
OP_CAT
OP_PUSHNUM_12
OP_PICK
OP_CAT
OP_PUSHNUM_11
OP_PICK
OP_CAT
OP_PUSHNUM_10
OP_PICK
OP_CAT
OP_PUSHNUM_9
OP_PICK
OP_CAT
OP_PUSHNUM_8
OP_PICK
OP_CAT
OP_PUSHNUM_7
OP_PICK
OP_CAT
OP_PUSHNUM_6
OP_PICK
OP_CAT
OP_PUSHNUM_5
OP_PICK
OP_CAT
OP_SHA256
OP_0
OP_PICK
OP_PUSHNUM_5
OP_PICK
OP_EQUAL
OP_VERIFY
```

```
OP_PUSHNUM_1
OP_PICK
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
```

```
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_NOP
OP_0
OP_PICK
OP_PUSHNUM_5
OP_PICK
OP_CHECKSIG
OP_VERIFY
OP_PUSHNUM_5
OP_PICK
OP_PUSHNUM_7
OP_PICK
OP_NOP
OP_0
OP_PICK
OP_SIZE
OP_NIP
OP_0
OP_PUSHNUM_2
OP_PICK
OP_0
OP_EQUAL
OP_IF
OP_PUSHBYTES_2 0100
OP_PUSHNUM_1
OP_ROLL
```

```
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_1 4c
OP_LESSTHAN
OP_IF
OP_PUSHNUM_1
OP_PICK
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_2 0001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4c
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_3 000001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4d
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
```

```
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_5 0000000001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4e
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_0
OP_VERIFY
OP_ENDIF
OP_ENDIF
OP_ENDIF
OP_ENDIF
OP_ENDIF
OP_0
OP_PICK
OP_PUSHNUM_3
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_NOP
OP_CAT
OP_PUSHNUM_2
OP_PICK
OP_PUSHNUM_1
```

```
OP_PICK
OP_SIZE
OP_NIP
OP_CAT
OP_PUSHNUM_1
OP_PICK
OP_CAT
OP_PUSHNUM_16
OP_PICK
OP_PUSHNUM_15
OP_PICK
OP_CAT
OP_PUSHNUM_14
OP_PICK
OP_CAT
OP_PUSHNUM_2
OP_CAT
OP_PUSHNUM_11
OP_PICK
OP_CAT
OP_PUSHNUM_12
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_PICK
OP_CAT
OP_PUSHNUM_16
OP_PICK
OP_CAT
OP_HASH256
OP_0
OP_PICK
OP_PUSHBYTES_4 00000000
OP_CAT
OP_PUSHNUM_14
OP_PICK
OP_CAT
OP_SHA256
OP_0
```

```
OP_PICK
OP_PUSHBYTES_1 1f
OP_PICK
OP_EQUAL
OP_VERIFY
OP_PUSHBYTES_1 18
OP_PICK
OP_PUSHBYTES_4 00000000
OP_EQUAL
OP_VERIFY
OP_PUSHNUM_10
OP_PICK
OP_PUSHNUM_1
OP_ADD
OP_PUSHNUM_10
OP_PICK
OP_PUSHNUM_1
OP_PICK
OP_NOP
OP_0
OP_PICK
OP_SIZE
OP_NIP
OP_0
OP_PUSHNUM_2
OP_PICK
OP_0
OP_EQUAL
OP_IF
OP_PUSHBYTES_2 0100
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_1 4c
OP_LESSTHAN
OP_IF
```

```
OP_PUSHNUM_1
OP_PICK
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_2 0001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4c
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_3 000001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4d
OP_PUSHNUM_2
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_ROLL
OP_DROP
OP_ELSE
OP_PUSHNUM_1
OP_PICK
OP_PUSHBYTES_5 0000000001
OP_LESSTHAN
OP_IF
OP_PUSHBYTES_1 4e
OP_PUSHNUM_2
```

OP_PICK

OP_CAT

OP_PUSHNUM_1

OP_ROLL

OP_DROP

OP_ELSE

OP_0

OP_VERIFY

OP_ENDIF

OP_ENDIF

OP_ENDIF

OP_ENDIF

OP_ENDIF

OP_0

OP_PICK

OP_PUSHNUM_3

OP_PICK

OP_CAT

OP_PUSHNUM_1

OP_ROLL

OP_DROP

OP_PUSHNUM_1

OP_ROLL

OP_DROP

OP_PUSHNUM_1

OP_ROLL

OP_DROP

OP_NOP

OP_CAT

OP_PUSHNUM_7

OP_PICK

OP_PUSHNUM_1

OP_PICK

OP_SIZE

OP_NIP

OP_CAT

OP_PUSHNUM_1

OP_PICK

OP_CAT

```
OP_PUSHNUM_14
OP_PICK
OP_PUSHBYTES_1 11
OP_PICK
OP_CAT
OP_PUSHNUM_1
OP_PICK
OP_CAT
OP_SHA256
OP_0
OP_PICK
OP_PUSHBYTES_1 1f
OP_PICK
OP_EQUAL
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
OP_NIP
```

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

OP_NIP

INPUTになるwitnessも長い。
最後の2スタックはスクリプト本体とcontrol blockなので行を空けた。

先頭から `ffffffff` まではP2TR script pathでの `SigMsg` と同じ構造である。
それ以降は TXID 計算に使うデータだろうか。下の方に出てくる `512056c7082d30169983d87dea6339bdcd447fb969f9d381e8770993fdd7f0d60f00` は `scriptPubKey`

sigHashの計算をしていて、これが `<e>` になる。
INPUTのTXID計算は locking scriptにTXIDを埋め込めなかったため計算しているのだろう。これは sigHash の計算に使われている。
つまり locking script に `<e>` を埋め込まない代わりに sigHash を計算し、それを witness で与えられたデータと比較するのか。

witness

02000000

f9000000

0ef02497827f4c686b63e4750034de48fcdec56f1752799b9b05cb99f306a088

ab9074d2a7947d23c4bafcd17be2abb5889b04928bf0aeb3aedc3fefcdd9b503

9f5d3feafd3be0850a97a13d090aebe0be33f075b7fd686aed533f906aeb1958

12a3ae445661ce5dee78d0650d33362dec29c4f82af05e7e57fb595bbbacf0ca

4fb999e9bc6d8c276e0822aa24dfa27de9aaf5bd4148fa60dd1812f510ad4701

02

00000000

```
6ede1f343a53fd8c7232a9201490dd3290cd1d0a73680454c0513809e5a75d64
00
ffffffff
331b923a37560aab49cd598262b6a57b4e688f9162385ae40b11ef7844283dff
6a316f92f5e2d075e39866fd9e3e814f008e6c7225a9f6edde8fdffebf2186
02000000
00000000
02f66c3f23e877c419dbe39a2e1d4215c459bfc52f4f060d8efd762ce8fb1fdb020000000000ffffffff
f66c3f23e877c419dbe39a2e1d4215c459bfc52f4f060d8efd762ce8fb1fdb020100000000ffffffff
f66c3f23e877c419dbe39a2e1d4215c459bfc52f4f060d8efd762ce8fb1fdb0202000000
22512056c7082d30169983d87dea6339bdcd447fb969f9d381e8770993fdd7f0d60f00
2202000000000000
2202000000000000
<empty>

016a2079be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f817984c807bb52d7a9
fef58323eb1bf7a407db382d2f3f2d81bb1224f49fe518f6d48d37c7bb52d7a9fef58323eb1bf7a407d
b382d2f3f2d81bb1224f49fe518f6d48d37c79be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d
959f2815b16f8179879be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f8179842
f40a48df4b2a70c8b4924bf2654661ed3d95fd66a313eb87237597c628e4a031f40a48df4b2a70c8b49
24bf2654661ed3d95fd66a313eb87237597c628e4a03100000800000000000000006161011b79011b79
011b79011b79011b79011b79011b79011b79011b79011b79011b79011b7961607952797
ea800795279517e876901127952797e527e01117960797e5f797e5e797e5d797e5c797e5b797e5a797e
59797e58797e57797e56797e55797ea8007955798769517 9517a75517a75517a75517a75517a75517a7
5517a75517a75517a75517a75517a75517a75517a75517a75517a75517a756100795579ac6955
795779610079827700527900876302010 0517a75675179014c9f635179517a756751790200019f63014
c52797e517a75675179030000019f63014d52797e517a756751790500000000019f63014e52797e517a
756700696868686868007953797e517a75517a75517a75617e5279517982777e51797e60795f797e5e7
97e527e5b797e5c797e51797e60797eaa007904000000007e5e797ea80079011f798769011879040000
000087695a7951935a795179610079827700527900876302010 0517a75675179014c9f635179517a756
751790200019f63014c52797e517a75675179030000019f63014d52797e517a756751790500000000019
9f63014e52797e517a756700696868686868007953797e517a75517a75517a75617e5779517982777e5
1797e5e790111797e51797ea80079011f798777777777777777777777777777777777777777777777
77777777777777777777777777
c09868f80087096e0b607955933752428f4e697c23db0412d698a5c01918ad6328
```

シュノア署名のトリックを使っているため sigHashUtils.ts で固定値になっているところが多い。

GitHub - Bitcoin-Wildlife-Sanctuary/covenants-examples: Examples of covenants in Bitcoin script