1 start NOOP ;starting the program

2 INIT #0;

3 LDR R9 #65528;

4 LDR R10 #65535;

5 LDR R7 #0;

6 LDR R0 #0;

7 SSVOP R0;

8 ;Initialising all the interface,declared signals and lock statuses into memory

9 ; I am cosidering that the DM is 16bits wide

10 ; I am not setting the statuses of any input signals since these are going to be set by the env

11 ; Input signals are always stored in the $0000 position

12 LDR R1 #0 ; Setting all the output signals statuses to zero

13 STR R1 $1 ;Storing the statuses of all the output signals in to the dedicated mem space for outputsignals

14 LDR R0 #0 ;loading zeros

15 STR R0 $2;

16 LDR R6 #2

17 LDR R0 #0 ;loading zeros

18 STR R0 $3;

19 LDR R0 #0 ;loading zeros

20 STR R0 $4;

21 LDR R0 #0 ;loading zeros

22 STR R0 $5;

23 LDR R0 #0 ;loading zeros

24 STR R0 $6;

25 LDR R0 #0 ;loading zeros

26 STR R0 $7;

27 LDR R0 #0 ;loading zeros

28 STR R0 $8;

29 LDR R0 #0 ;loading zeros

30 STR R0 $9;

31 LDR R6 #3

32 LDR R1 #0 ;I am setting the statuses of all the internal signals everywhere in the CD

33 STR R1 $a ;Storing them into the mem

34 LDR R0 #0

35 STR R0 $b;

36 LDR R0 #0

37 STR R0 $c; stored

38 LDR R0 #1 ;

39 STR R0 $d;

40 LDR R0 #0 ;

41 STR R0 $e;

42 LDR R0 #0 ;

43 STR R0 $f;

44 LDR R0 #1 ;

45 STR R0 $10;

46 LDR R0 #0 ;

47 STR R0 $11;

48 LDR R0 #0 ;

49 STR R0 $12;

50 LDR R0 #0 ;

51 STR R0 $13;

52 LDR R0 #1 ;

53 STR R0 $14;

54 LDR R0 #0 ;

55 STR R0 $15;

56 LDR R0 #0 ;

57 STR R0 $16;

58 LDR R0 #0 ;

59 STR R0 $17;

60 LDR R0 #0 ;

61 STR R0 $18;

62 LDR R0 #0 ;

63 STR R0 $19;

64 LDR R0 #0 ;

65 STR R0 $1a;

66 LDR R0 #0 ;

67 STR R0 $1b;

68 LDR R0 #0 ;

69 STR R0 $1c;

70 LDR R0 #1 ;

71 STR R0 $1d;

72 LDR R0 #0 ;

73 STR R0 $1e;

74 LDR R0 #0 ;

75 STR R0 $1f;

76 LDR R0 #0 ;

77 STR R0 $20;

78 LDR R0 #0 ;

79 STR R0 $21;

80 LDR R0 #0 ;

81 STR R0 $22;

82 LDR R0 #0 ;

83 STR R0 $23;

84 LDR R0 #1 ;

85 STR R0 $24;

86 LDR R0 #0 ;

87 STR R0 $25;

88 LDR R0 #0 ;

89 STR R0 $26;

90 LDR R0 #0 ;

91 STR R0 $27;

92 LDR R0 #0 ;

93 STR R0 $28;

94 LDR R0 #1 ;

95 STR R0 $29;

96 LDR R0 #0 ;

97 STR R0 $2a;

98 LDR R0 #0 ;

99 STR R0 $2b;

100 LDR R0 #0 ;

101 STR R0 $2c;

102 LDR R0 #0 ;

103 STR R0 $2d;

104 LDR R0 #1 ;

105 STR R0 $2e;

106 LDR R0 #0 ;

107 STR R0 $2f;

108 LDR R0 #0 ;

109 STR R0 $30;

110 LDR R0 #0 ;

111 STR R0 $31;

112 LDR R0 #0 ;

113 STR R0 $32;

114 LDR R0 #0 ;

115 STR R0 $33;

116 LDR R0 #0 ;

117 STR R0 $34;

118 LDR R0 #0 ;

119 STR R0 $35;

120 LDR R0 #1 ;

121 STR R0 $36;

122 LDR R0 #0 ;

123 STR R0 $37;

124 LDR R0 #0 ;

125 STR R0 $38;

126 LDR R0 #0 ;

127 STR R0 $39;

128 LDR R0 #0 ;

129 STR R0 $3a;

130 LDR R0 #0 ;

131 STR R0 $3b;

132 LDR R0 #0 ;

133 STR R0 $3c;

134 LDR R0 #0 ;

135 STR R0 $3d;

136 LDR R0 #0 ;

137 STR R0 $3e;

138 LDR R0 #1 ;

139 STR R0 $3f;

140 LDR R0 #0 ;

141 STR R0 $40;

142 LDR R0 #0 ;

143 STR R0 $41;

144 LDR R0 #0 ;

145 STR R0 $42;

146 LDR R0 #0 ;

147 STR R0 $43;

148 LDR R0 #0 ;

149 STR R0 $44;

150 LDR R0 #1 ;

151 STR R0 $45;

152 LDR R0 #0 ;

153 STR R0 $46;

154 LDR R0 #0 ;

155 STR R0 $47;

156 LDR R0 #0 ;

157 STR R0 $48;

158 LDR R0 #0 ;

159 STR R0 $49;

160 LDR R0 #1 ;

161 STR R0 $4a;

162 LDR R0 #0 ;

163 STR R0 $4b;

164 LDR R0 #0 ;

165 STR R0 $4c;

166 LDR R0 #0 ;

167 STR R0 $4d;

168 LDR R0 #0 ;

169 STR R0 $4e;

170 LDR R0 #0 ;

171 STR R0 $4f;

172 LDR R0 #0 ;

173 STR R0 $50;

174 LDR R0 #0 ;

175 STR R0 $51;

176 LDR R0 #0 ;

177 STR R0 $52;

178 LDR R0 #0 ;

179 STR R0 $53;

180 LDR R0 #0 ;

181 STR R0 $54;

182 LDR R0 #1 ;

183 STR R0 $55;

184 LDR R0 #0 ;

185 STR R0 $56;

186 LDR R0 #0 ;

187 STR R0 $57;

188 LDR R0 #0 ;

189 STR R0 $58;

190 LDR R0 #0 ;

191 STR R0 $59;

192 LDR R0 #1 ;

193 STR R0 $5a;

194 LDR R0 #0 ;

195 STR R0 $5b;

196 LDR R0 #0 ;

197 STR R0 $5c;

198 LDR R0 #0 ;

199 STR R0 $5d;

200 LDR R0 #0 ;

201 STR R0 $5e;

202 LDR R0 #0 ;

203 STR R0 $5f;

204 LDR R0 #0 ;

205 STR R0 $60;

206 LDR R0 #0 ;

207 STR R0 $61;

208 LDR R0 #0 ;

209 STR R0 $62;

210 LDR R0 #0 ;

211 STR R0 $63;

212 LDR R0 #0 ;

213 STR R0 $64;

214 LDR R0 #0 ;

215 STR R0 $65;

216 LDR R0 #0 ;

217 STR R0 $66;

218 LDR R0 #0 ;

219 STR R0 $67;

220 LDR R0 #0 ;

221 STR R0 $68;

222 LDR R0 #0 ;

223 STR R0 $69;

224 LDR R0 #0 ;

225 STR R0 $6a;

226 LDR R0 #0 ;

227 STR R0 $6b;

228 LDR R0 #0 ;

229 STR R0 $6c;

230 LDR R0 #0 ;

231 STR R0 $6d;

232 LDR R0 #0 ;

233 STR R0 $6e;

234 LDR R0 #0 ;

235 STR R0 $6f;

236 LDR R0 #0 ;

237 STR R0 $70;

238 LDR R0 #1 ;

239 STR R0 $71;

240 LDR R0 #0 ;

241 STR R0 $72;

242 LDR R0 #0 ;

243 STR R0 $73;

244 LDR R0 #0 ;

245 STR R0 $74;

246 LDR R0 #0 ;

247 STR R0 $75;

248 LDR R0 #1 ;

249 STR R0 $76;

250 LDR R0 #0 ;

251 STR R0 $77;

252 LDR R0 #0 ;

253 STR R0 $78;

254 LDR R0 #0 ;

255 STR R0 $79;

256 LDR R0 #0 ;

257 STR R0 $7a;

258 LDR R0 #0 ;

259 STR R0 $7b;

260 LDR R0 #0 ;

261 STR R0 $7c;

262 LDR R0 #0 ;

263 STR R0 $7d;

264 LDR R0 #0 ;

265 STR R0 $7e;

266 LDR R0 #1 ;

267 STR R0 $7f;

268 LDR R0 #0 ;

269 STR R0 $80;

270 LDR R0 #0 ;

271 STR R0 $81;

272 LDR R0 #0 ;

273 STR R0 $82;

274 LDR R0 #0 ;

275 STR R0 $83;

276 LDR R0 #1 ;

277 STR R0 $84;

278 LDR R0 #0 ;

279 STR R0 $85;

280 LDR R0 #0 ;

281 STR R0 $86;

282 LDR R0 #0 ;

283 STR R0 $87;

284 LDR R0 #0 ;

285 STR R0 $88;

286 LDR R0 #1 ;

287 STR R0 $89;

288 LDR R0 #0 ;

289 STR R0 $8a;

290 LDR R0 #0 ;

291 STR R0 $8b;

292 LDR R0 #0 ;

293 STR R0 $8c;

294 LDR R0 #0 ;

295 STR R0 $8d;

296 LDR R0 #1 ;

297 STR R0 $8e;

298 LDR R0 #0 ;

299 STR R0 $8f;

300 LDR R0 #0 ;

301 STR R0 $90;

302 LDR R0 #0 ;

303 STR R0 $91;

304 LDR R0 #0 ;

305 STR R0 $92;

306 LDR R0 #0 ;

307 STR R0 $93;

308 LDR R0 #0 ;

309 STR R0 $94;

310 LDR R0 #1 ;

311 STR R0 $95;

312 LDR R0 #0 ;

313 STR R0 $96;

314 LDR R0 #0 ;

315 STR R0 $97;

316 LDR R0 #0 ;

317 STR R0 $98;

318 LDR R0 #0 ;

319 STR R0 $99;

320 LDR R0 #1 ;

321 STR R0 $9a;

322 LDR R0 #0 ;

323 STR R0 $9b;

324 LDR R0 #0 ;

325 STR R0 $9c;

326 LDR R0 #1 ;

327 STR R0 $9d;

328 LDR R0 #0 ;

329 STR R0 $9e;

330 LDR R0 #0 ;

331 STR R0 $9f;

332 LDR R0 #0 ;

333 STR R0 $a0;

334 LDR R0 #0 ;

335 STR R0 $a1;

336 LDR R0 #1 ;

337 STR R0 $a2;

338 LDR R0 #0 ;

339 STR R0 $a3 ;This is the starting point of all all the Jnodes

340 LDR R0 #0 ;

341 STR R0 $a4;

342 LDR R0 #0 ;

343 STR R0 $a5;

344 LDR R0 #0 ;

345 STR R0 $a6;

346 LDR R0 #0 ;

347 STR R0 $a7;

348 LDR R0 #0 ;

349 STR R0 $a8;

350 LDR R0 #0 ;

351 STR R0 $a9;

352 LDR R0 #0 ;

353 STR R0 $aa;

354 LDR R0 #0 ;

355 STR R0 $ab;

356 LDR R0 #0 ;

357 STR R0 $ac;

358 LDR R0 #0 ;

359 STR R0 $ad;

360 LDR R0 #0 ;

361 STR R0 $ae;

362 LDR R0 #0 ;

363 STR R0 $af;

364 LDR R0 #0 ;

365 STR R0 $b0;

366 LDR R0 #0 ;

367 STR R0 $b1;

368 LDR R0 #0 ;

369 STR R0 $b2;

370 LDR R0 #0 ;

371 STR R0 $b3;

372 LDR R0 #0 ;

373 STR R0 $b4 ;This is the starting point of all all the Jnodes

374 LDR R0 #0 ;

375 STR R0 $b5;

376 LDR R0 #0 ;

377 STR R0 $b6;

378 LDR R0 #0 ;

379 STR R0 $b7;

380 LDR R0 #0 ;

381 STR R0 $b8;

382 LDR R0 #0 ;

383 STR R0 $b9;

384 LDR R0 #0 ;

385 STR R0 $ba;

386 LDR R0 #0 ;

387 STR R0 $bb;

388 LDR R0 #0 ;

389 STR R0 $bc;

390 LDR R0 #0 ;

391 STR R0 $bd;

392 LDR R0 #0 ;

393 STR R0 $be;

394 LDR R0 #0 ;

395 STR R0 $bf;

396 LDR R0 #0 ;

397 STR R0 $c0;

398 LDR R0 #0 ;

399 STR R0 $c1;

400 LDR R0 #0 ;

401 STR R0 $c2;

402 LDR R0 #0 ;

403 STR R0 $c3;

404 LDR R0 #0 ;

405 STR R0 $c4;

406 LDR R0 #0 ;

407 STR R0 $c5 ;This is the starting point of all all the Jnodes

408 LDR R0 #0 ;

409 STR R0 $c6;

410 LDR R0 #0 ;

411 STR R0 $c7;

412 LDR R0 #0 ;

413 STR R0 $c8;

414 LDR R0 #0 ;

415 STR R0 $c9;

416 LDR R0 #0 ;

417 STR R0 $ca;

418 LDR R0 #0 ;

419 STR R0 $cb;

420 LDR R0 #0 ;

421 STR R0 $cc;

422 LDR R0 #0 ;

423 STR R0 $cd;

424 LDR R0 #0 ;

425 STR R0 $ce;

426 LDR R0 #0 ;

427 STR R0 $cf;

428 LDR R0 #0 ;

429 STR R0 $d0;

430 LDR R0 #0 ;

431 STR R0 $d1;

432 LDR R0 #0 ;

433 STR R0 $d2;

434 LDR R0 #0 ;

435 STR R0 $d3;

436 LDR R0 #0 ;

437 STR R0 $d4;

438 LDR R0 #0 ;

439 STR R0 $d5;

440 LDR R0 #0 ;

441 STR R0 $d6 ;This is the starting point of all all the Jnodes

442 LDR R0 #0 ;

443 STR R0 $d7;

444 LDR R0 #0 ;

445 STR R0 $d8;

446 LDR R0 #0 ;

447 STR R0 $d9;

448 LDR R0 #0 ;

449 STR R0 $da;

450 LDR R0 #0 ;

451 STR R0 $db;

452 LDR R0 #0 ;

453 STR R0 $dc;

454 LDR R0 #0 ;

455 STR R0 $dd;

456 LDR R0 #0 ;

457 STR R0 $de;

458 LDR R0 #0 ;

459 STR R0 $df;

460 LDR R0 #0 ;

461 STR R0 $e0;

462 LDR R0 #0 ;

463 STR R0 $e1;

464 LDR R0 #0 ;

465 STR R0 $e2;

466 LDR R0 #0 ;

467 STR R0 $e3;

468 LDR R0 #0 ;

469 STR R0 $e4;

470 LDR R0 #0 ;

471 STR R0 $e5;

472 LDR R0 #0 ;

473 STR R0 $e6;

474 LDR R1 #0 ;I am setting the statuses of all the internal signals everywhere in the CD

475 STR R1 $e7 ;Storing them into the mem

476 LDR R0 #0 ;Initalising all the locks to zero

477 STR R0 $e8; Storing the zeros into memory

478 LDR R0 #0 ;Initalising all the locks to zero

479 STR R0 $e9; Storing the zeros into memory

480 LDR R0 #0 ;Initalising all the locks to zero

481 STR R0 $ea; Storing the zeros into memory

482 LDR R0 #0 ;Initalising all the locks to zero

483 STR R0 $eb; Storing the zeros into memory

484 LDR R0 #0 ;Initalising all the locks to zero

485 STR R0 $ec; Storing the zeros into memory

486 LDR R0 #0 ;Initalising all the locks to zero

487 STR R0 $ed; Storing the zeros into memory

488 LDR R0 #0

489 STR R0 $ee;

490 LDR R0 #0

491 STR R0 $ef;

492 LDR R0 #0

493 STR R0 $f0;

494 LDR R0 #0

495 STR R0 $f1;

496 LDR R0 #0

497 STR R0 $f2;

498 LDR R0 #0

499 STR R0 $f3;

500 LDR R0 #0

501 STR R0 $f4;

502 LDR R0 #0

503 STR R0 $f5; stored

504 LDR R0 #0

505 STR R0 $f6; stored

506 LDR R0 #1 ;

507 STR R0 $f7;

508 LDR R0 #0 ;

509 STR R0 $f8;

510 LDR R0 #0 ;

511 STR R0 $f9;

512 LDR R0 #1 ;

513 STR R0 $fa;

514 LDR R0 #0 ;

515 STR R0 $fb;

516 LDR R0 #0 ;

517 STR R0 $fc;

518 LDR R0 #0 ;

519 STR R0 $fd;

520 LDR R0 #1 ;

521 STR R0 $fe;

522 LDR R0 #0 ;

523 STR R0 $ff;

524 LDR R0 #0 ;

525 STR R0 $100;

526 LDR R0 #0 ;

527 STR R0 $101;

528 LDR R0 #0 ;

529 STR R0 $102;

530 LDR R0 #0 ;

531 STR R0 $103;

532 LDR R0 #0 ;

533 STR R0 $104;

534 LDR R0 #0 ;

535 STR R0 $105;

536 LDR R0 #0 ;

537 STR R0 $106;

538 LDR R0 #1 ;

539 STR R0 $107;

540 LDR R0 #0 ;

541 STR R0 $108;

542 LDR R0 #0 ;

543 STR R0 $109;

544 LDR R0 #0 ;

545 STR R0 $10a;

546 LDR R0 #0 ;

547 STR R0 $10b;

548 LDR R0 #0 ;

549 STR R0 $10c;

550 LDR R0 #0 ;

551 STR R0 $10d;

552 LDR R0 #1 ;

553 STR R0 $10e;

554 LDR R0 #0 ;

555 STR R0 $10f;

556 LDR R0 #0 ;

557 STR R0 $110;

558 LDR R0 #0 ;

559 STR R0 $111;

560 LDR R0 #0 ;

561 STR R0 $112;

562 LDR R0 #1 ;

563 STR R0 $113;

564 LDR R0 #0 ;

565 STR R0 $114;

566 LDR R0 #0 ;

567 STR R0 $115;

568 LDR R0 #0 ;

569 STR R0 $116;

570 LDR R0 #0 ;

571 STR R0 $117;

572 LDR R0 #1 ;

573 STR R0 $118;

574 LDR R0 #0 ;

575 STR R0 $119;

576 LDR R0 #0 ;

577 STR R0 $11a;

578 LDR R0 #0 ;

579 STR R0 $11b;

580 LDR R0 #0 ;

581 STR R0 $11c;

582 LDR R0 #0 ;

583 STR R0 $11d;

584 LDR R0 #0 ;

585 STR R0 $11e;

586 LDR R0 #0 ;

587 STR R0 $11f;

588 LDR R0 #1 ;

589 STR R0 $120;

590 LDR R0 #0 ;

591 STR R0 $121;

592 LDR R0 #0 ;

593 STR R0 $122;

594 LDR R0 #0 ;

595 STR R0 $123;

596 LDR R0 #0 ;

597 STR R0 $124;

598 LDR R0 #0 ;

599 STR R0 $125;

600 LDR R0 #0 ;

601 STR R0 $126;

602 LDR R0 #0 ;

603 STR R0 $127;

604 LDR R0 #0 ;

605 STR R0 $128;

606 LDR R0 #1 ;

607 STR R0 $129;

608 LDR R0 #0 ;

609 STR R0 $12a;

610 LDR R0 #0 ;

611 STR R0 $12b;

612 LDR R0 #0 ;

613 STR R0 $12c;

614 LDR R0 #0 ;

615 STR R0 $12d;

616 LDR R0 #0 ;

617 STR R0 $12e;

618 LDR R0 #1 ;

619 STR R0 $12f;

620 LDR R0 #0 ;

621 STR R0 $130;

622 LDR R0 #0 ;

623 STR R0 $131;

624 LDR R0 #0 ;

625 STR R0 $132;

626 LDR R0 #0 ;

627 STR R0 $133;

628 LDR R0 #1 ;

629 STR R0 $134;

630 LDR R0 #0 ;

631 STR R0 $135;

632 LDR R0 #0 ;

633 STR R0 $136;

634 LDR R0 #0 ;

635 STR R0 $137;

636 LDR R0 #0 ;

637 STR R0 $138;

638 LDR R0 #0 ;

639 STR R0 $139;

640 LDR R0 #0 ;

641 STR R0 $13a;

642 LDR R0 #0 ;

643 STR R0 $13b;

644 LDR R0 #0 ;

645 STR R0 $13c;

646 LDR R0 #0 ;

647 STR R0 $13d;

648 LDR R0 #0 ;

649 STR R0 $13e;

650 LDR R0 #1 ;

651 STR R0 $13f;

652 LDR R0 #0 ;

653 STR R0 $140;

654 LDR R0 #0 ;

655 STR R0 $141;

656 LDR R0 #0 ;

657 STR R0 $142;

658 LDR R0 #0 ;

659 STR R0 $143;

660 LDR R0 #1 ;

661 STR R0 $144;

662 LDR R0 #0 ;

663 STR R0 $145;

664 LDR R0 #0 ;

665 STR R0 $146;

666 LDR R0 #0 ;

667 STR R0 $147;

668 LDR R0 #0 ;

669 STR R0 $148;

670 LDR R0 #0 ;

671 STR R0 $149;

672 LDR R0 #0 ;

673 STR R0 $14a;

674 LDR R0 #0 ;

675 STR R0 $14b;

676 LDR R0 #0 ;

677 STR R0 $14c;

678 LDR R0 #0 ;

679 STR R0 $14d;

680 LDR R0 #0 ;

681 STR R0 $14e;

682 LDR R0 #0 ;

683 STR R0 $14f;

684 LDR R0 #0 ;

685 STR R0 $150;

686 LDR R0 #0 ;

687 STR R0 $151;

688 LDR R0 #0 ;

689 STR R0 $152;

690 LDR R0 #0 ;

691 STR R0 $153;

692 LDR R0 #0 ;

693 STR R0 $154;

694 LDR R0 #0 ;

695 STR R0 $155;

696 LDR R0 #0 ;

697 STR R0 $156;

698 LDR R0 #0 ;

699 STR R0 $157;

700 LDR R0 #0 ;

701 STR R0 $158;

702 LDR R0 #0 ;

703 STR R0 $159;

704 LDR R0 #0 ;

705 STR R0 $15a;

706 LDR R0 #1 ;

707 STR R0 $15b;

708 LDR R0 #0 ;

709 STR R0 $15c;

710 LDR R0 #0 ;

711 STR R0 $15d;

712 LDR R0 #0 ;

713 STR R0 $15e;

714 LDR R0 #0 ;

715 STR R0 $15f;

716 LDR R0 #1 ;

717 STR R0 $160;

718 LDR R0 #0 ;

719 STR R0 $161;

720 LDR R0 #0 ;

721 STR R0 $162;

722 LDR R0 #0 ;

723 STR R0 $163;

724 LDR R0 #0 ;

725 STR R0 $164;

726 LDR R0 #0 ;

727 STR R0 $165;

728 LDR R0 #0 ;

729 STR R0 $166;

730 LDR R0 #0 ;

731 STR R0 $167;

732 LDR R0 #0 ;

733 STR R0 $168;

734 LDR R0 #1 ;

735 STR R0 $169;

736 LDR R0 #0 ;

737 STR R0 $16a;

738 LDR R0 #0 ;

739 STR R0 $16b;

740 LDR R0 #0 ;

741 STR R0 $16c;

742 LDR R0 #0 ;

743 STR R0 $16d;

744 LDR R0 #1 ;

745 STR R0 $16e;

746 LDR R0 #0 ;

747 STR R0 $16f;

748 LDR R0 #0 ;

749 STR R0 $170;

750 LDR R0 #0 ;

751 STR R0 $171;

752 LDR R0 #0 ;

753 STR R0 $172;

754 LDR R0 #1 ;

755 STR R0 $173;

756 LDR R0 #0 ;

757 STR R0 $174;

758 LDR R0 #0 ;

759 STR R0 $175;

760 LDR R0 #0 ;

761 STR R0 $176;

762 LDR R0 #0 ;

763 STR R0 $177;

764 LDR R0 #1 ;

765 STR R0 $178;

766 LDR R0 #0 ;

767 STR R0 $179;

768 LDR R0 #0 ;

769 STR R0 $17a;

770 LDR R0 #0 ;

771 STR R0 $17b;

772 LDR R0 #0 ;

773 STR R0 $17c;

774 LDR R0 #0 ;

775 STR R0 $17d;

776 LDR R0 #0 ;

777 STR R0 $17e;

778 LDR R0 #1 ;

779 STR R0 $17f;

780 LDR R0 #0 ;

781 STR R0 $180;

782 LDR R0 #0 ;

783 STR R0 $181;

784 LDR R0 #0 ;

785 STR R0 $182;

786 LDR R0 #0 ;

787 STR R0 $183;

788 LDR R0 #1 ;

789 STR R0 $184;

790 LDR R0 #0 ;

791 STR R0 $185;

792 LDR R0 #0 ;

793 STR R0 $186;

794 LDR R0 #1 ;

795 STR R0 $187;

796 LDR R0 #0 ;

797 STR R0 $188;

798 LDR R0 #0 ;

799 STR R0 $189;

800 LDR R0 #0 ;

801 STR R0 $18a;

802 LDR R0 #0 ;

803 STR R0 $18b;

804 LDR R0 #1 ;

805 STR R0 $18c;

806 LDR R0 #0 ;

807 STR R0 $18d ;This is the starting point of all all the Jnodes

808 LDR R0 #0 ;

809 STR R0 $18e;

810 LDR R0 #0 ;

811 STR R0 $18f;

812 LDR R0 #0 ;

813 STR R0 $190;

814 LDR R0 #0 ;

815 STR R0 $191;

816 LDR R0 #0 ;

817 STR R0 $192;

818 LDR R0 #0 ;

819 STR R0 $193;

820 LDR R0 #0 ;

821 STR R0 $194;

822 LDR R0 #0 ;

823 STR R0 $195;

824 LDR R0 #0 ;

825 STR R0 $196;

826 LDR R0 #0 ;

827 STR R0 $197;

828 LDR R0 #0 ;

829 STR R0 $198;

830 LDR R0 #0 ;

831 STR R0 $199;

832 LDR R0 #0 ;

833 STR R0 $19a;

834 LDR R0 #0 ;

835 STR R0 $19b;

836 LDR R0 #0 ;

837 STR R0 $19c;

838 LDR R0 #0 ;

839 STR R0 $19d;

840 LDR R0 #0 ;

841 STR R0 $19e ;This is the starting point of all all the Jnodes

842 LDR R0 #0 ;

843 STR R0 $19f;

844 LDR R0 #0 ;

845 STR R0 $1a0;

846 LDR R0 #0 ;

847 STR R0 $1a1;

848 LDR R0 #0 ;

849 STR R0 $1a2;

850 LDR R0 #0 ;

851 STR R0 $1a3;

852 LDR R0 #0 ;

853 STR R0 $1a4;

854 LDR R0 #0 ;

855 STR R0 $1a5;

856 LDR R0 #0 ;

857 STR R0 $1a6;

858 LDR R0 #0 ;

859 STR R0 $1a7;

860 LDR R0 #0 ;

861 STR R0 $1a8;

862 LDR R0 #0 ;

863 STR R0 $1a9;

864 LDR R0 #0 ;

865 STR R0 $1aa;

866 LDR R0 #0 ;

867 STR R0 $1ab;

868 LDR R0 #0 ;

869 STR R0 $1ac;

870 LDR R0 #0 ;

871 STR R0 $1ad;

872 LDR R0 #0 ;

873 STR R0 $1ae;

874 LDR R0 #0 ;

875 STR R0 $1af ;This is the starting point of all all the Jnodes

876 LDR R0 #0 ;

877 STR R0 $1b0;

878 LDR R0 #0 ;

879 STR R0 $1b1;

880 LDR R0 #0 ;

881 STR R0 $1b2;

882 LDR R0 #0 ;

883 STR R0 $1b3;

884 LDR R0 #0 ;

885 STR R0 $1b4;

886 LDR R0 #0 ;

887 STR R0 $1b5;

888 LDR R0 #0 ;

889 STR R0 $1b6;

890 LDR R0 #0 ;

891 STR R0 $1b7;

892 LDR R0 #0 ;

893 STR R0 $1b8;

894 LDR R0 #0 ;

895 STR R0 $1b9;

896 LDR R0 #0 ;

897 STR R0 $1ba;

898 LDR R0 #0 ;

899 STR R0 $1bb;

900 LDR R0 #0 ;

901 STR R0 $1bc;

902 LDR R0 #0 ;

903 STR R0 $1bd;

904 LDR R0 #0 ;

905 STR R0 $1be;

906 LDR R0 #0 ;

907 STR R0 $1bf;

908 LDR R0 #0 ;

909 STR R0 $1c0 ;This is the starting point of all all the Jnodes

910 LDR R0 #0 ;

911 STR R0 $1c1;

912 LDR R0 #0 ;

913 STR R0 $1c2;

914 LDR R0 #0 ;

915 STR R0 $1c3;

916 LDR R0 #0 ;

917 STR R0 $1c4;

918 LDR R0 #0 ;

919 STR R0 $1c5;

920 LDR R0 #0 ;

921 STR R0 $1c6;

922 LDR R0 #0 ;

923 STR R0 $1c7;

924 LDR R0 #0 ;

925 STR R0 $1c8;

926 LDR R0 #0 ;

927 STR R0 $1c9;

928 LDR R0 #0 ;

929 STR R0 $1ca;

930 LDR R0 #0 ;

931 STR R0 $1cb;

932 LDR R0 #0 ;

933 STR R0 $1cc;

934 LDR R0 #0 ;

935 STR R0 $1cd;

936 LDR R0 #0 ;

937 STR R0 $1ce;

938 LDR R0 #0 ;

939 STR R0 $1cf;

940 LDR R0 #0 ;

941 STR R0 $1d0;

942

943 BEGIN0 NOOP; loading the num which have to be init

944 INIT #0;

945 LDR R7 #0;

946 LDR R8 #1;previous clock-domain num

947 LDR R6 #2

948 ;Setting the declared signals, signal locks, data-locks, and pc's to 0

949 LDR R0 #0 ;blocked it cool

950 STR R0 $a ;loading the values

951 LDR R0 #0 ;blocked it cool

952 STR R0 $b ;loading the values

953 LDR R0 #0 ;blocked it cool

954 STR R0 $c ;loading the values

955 SEOT1310866724403 CLFZ;

956 LDR R0 #0;clearing

957 LDR R1 #0;clearing

958 LDR R11 #0;clearing

959 LERR1310866724403 LER R0;loading the EREADY bit which is set from ENV

960 PRESENT R0 FER1310866724403;

961 JMP LERR1310866724403;

962 FER1310866724403 SEOT; This is basically the SEOT tick

963 CER;clear the EREADY bit

964 LDR R0 $0001; loading the output signals

965 AND R0 R0 #$0;

966 SSOP R0;throwing output signals to env

967 LSIP R0;getting input signals from SIP

968 AND R0 R0 #$0;

969 LDR R1 $0000;

970 AND R1 R1 #$ffff;

971 OR R0 R0 R1;

972 STR R0 $0000;storing SIP signals in mem

973 LDR R0 #2

974 ADD R1 R6 #0

975 STR R1 #0;

976 SENDATA R0;

977 ADD R4 R4 #1;

978 LOAD01310866724403 LDR R0 R1;

979 CLFZ

980 SUBV R0 R0 #0;

981 SZ EL01310866724403;

982 JMP OEL01310866724403;

983 EL01310866724403 CLFZ;

984 JMP LOAD01310866724403;

985 OEL01310866724403 NOOP;

986 INIT #0;

987 CEOT;now start processing

988 LDR R14 #33; the ESL line numbers start-offset

989 RUN0 NOOP; the locks need to be inside the memory since if they are here then I am just eating up logic

990 LDR R7 #0;

991 LDR R8 #1;previous clock-domain num

992 LDR R6 #2

993 ESL #18

994 LDR R0 $b;

995 SUB R0 #0 ;checking if the value here is actually zero

996 SZ CONT896284316 ;Just jump to the value pointed to by this register

997 JMP R0

998 CONT896284316 NOOP

999 LDR R0 #case902440299 ;

1000 STR R0 $11 ;

1001 LDR R0 #case1513835930 ;

1002 STR R0 $12 ;

1003 LDR R0 #case427427792 ;

1004 STR R0 $13 ;

1005 LDR R0 #16

1006 SWITCH R1 R0 ;

1007case902440299 NOOP;

1008 LDR R0 #0;

1009 STR R0 $10;

1010 LDR R0 #$0;

1011 STR R0 $b;

1012 JMP END0;

1013 JMP ENDS5988805580

1014case1513835930 NOOP;

1015 LDR R0 #2;

1016 STR R0 $10;

1017 LDR R0 #2;

1018 STR R0 $10;

1019 LDR R0 #0;

1020 STR R0 $14;

1021 LDR R0 #0;

1022 STR R0 $19;

1023 LDR R0 #258

1024 ADD R1 R6 #0

1025 STR R1 #0;

1026 SENDATA R0;

1027 SUB R4 #65535;

1028 SZ TOQSTART957074642;

1029 ADD R4 R4 #1;

1030 JMP QS957074642;

1031TOQSTART957074642 LDR R4 #65528;

1032QS957074642 CLFZ; data man man

1033 LDR R0 #TEN963230625

1034 STR R0 $b

1035TEN963230625 NOOP;

1036 LDR R0 $2

1037 SUB R0 #0

1038 SZ EL603194796

1039IF603194796 CLFZ; titi man man

1040 LDR R0 $2

1041 AND R0 R0 #$0002

1042 PRESENT R0 else958998387

1043 LDR R0 #1538;

1044 ADD R1 R6 #0

1045 STR R1 #0;

1046 SENDATA R0;

1047 SUB R4 #65535;

1048 SZ TOQSTART940530440;

1049 ADD R4 R4 #1;

1050 JMP QS940530440;

1051TOQSTART940530440 LDR R4 #65528;

1052QS940530440 CLFZ;

1053 LDR R0 #TEN947071171

1054 STR R0 $b

1055TEN947071171 NOOP;

1056 LDR R0 $2

1057 SUB R0 #0

1058 SZ EL1888906730

1059IF1888906730 CLFZ

1060 ESL #20

1061 LDR R0 #1;

1062 STR R0 $14;

1063 LDR R0 #4096;

1064 LDR R1 $c

1065 AND R1 R1 #$0FFF

1066 OR R1 R1 R0

1067 STR R1 $c; 1

1068 LDR R0 #$0;

1069 STR R0 $b;

1070 JMP END0;

1071 JMP OVERELSE1857007793

1072EL1888906730 CLFZ;

1073 LDR R0 #61440;

1074 LDR R1 $c

1075 AND R1 R1 #$0FFF

1076 OR R1 R1 R0

1077 STR R1 $c; 1

1078 JMP END0;

1079OVERELSE1857007793 NOOP;

1080 JMP OVERELSE428623244

1081else958998387 NOOP

1082 LDR R0 #0;

1083 STR R0 $1b;

1084 LDR R0 #0;

1085 STR R0 $1d;

1086 LDR R0 #514

1087 ADD R1 R6 #0

1088 STR R1 #0;

1089 SENDATA R0;

1090 SUB R4 #65535;

1091 SZ TOQSTART934759206;

1092 ADD R4 R4 #1;

1093 JMP QS934759206;

1094TOQSTART934759206 LDR R4 #65528;

1095QS934759206 CLFZ; data man man

1096 LDR R0 #TEN1013632731

1097 STR R0 $b

1098TEN1013632731 NOOP;

1099 LDR R0 $2

1100 SUB R0 #0

1101 SZ EL441429295

1102IF441429295 CLFZ; titi man man

1103 LDR R0 $2

1104 AND R0 R0 #$0002

1105 PRESENT R0 else1019788713

1106 LDR R0 #770;

1107 ADD R1 R6 #0

1108 STR R1 #0;

1109 SENDATA R0;

1110 SUB R4 #65535;

1111 SZ TOQSTART1001320766;

1112 ADD R4 R4 #1;

1113 JMP QS1001320766;

1114TOQSTART1001320766 LDR R4 #65528;

1115QS1001320766 CLFZ;

1116 LDR R0 #TEN1007861497

1117 STR R0 $b

1118TEN1007861497 NOOP;

1119 LDR R0 $2

1120 SUB R0 #0

1121 SZ EL289029839

1122IF289029839 CLFZ

1123 LDR R0 #1;

1124 STR R0 $1d;

1125 LDR R0 #1026

1126 ADD R1 R6 #0

1127 STR R1 #0;

1128 SENDATA R0;

1129 SUB R4 #65535;

1130 SZ TOQSTART989393550;

1131 ADD R4 R4 #1;

1132 JMP QS989393550;

1133TOQSTART989393550 LDR R4 #65528;

1134QS989393550 CLFZ; data man man

1135 LDR R0 #TEN997473277

1136 STR R0 $b

1137TEN997473277 NOOP;

1138 LDR R0 $2

1139 SUB R0 #0

1140 SZ EL844282640

1141IF844282640 CLFZ; titi man man

1142 LDR R0 $2

1143 AND R0 R0 #$0002

1144 PRESENT R0 else979005330

1145 LDR R0 #1282;

1146 ADD R1 R6 #0

1147 STR R1 #0;

1148 SENDATA R0;

1149 SUB R4 #65535;

1150 SZ TOQSTART985161312;

1151 ADD R4 R4 #1;

1152 JMP QS985161312;

1153TOQSTART985161312 LDR R4 #65528;

1154QS985161312 CLFZ;

1155 LDR R0 #TEN1065573832

1156 STR R0 $b

1157TEN1065573832 NOOP;

1158 LDR R0 $2

1159 SUB R0 #0

1160 SZ EL1813024393

1161IF1813024393 CLFZ

1162 ESL #20

1163 LDR R0 #1;

1164 STR R0 $14;

1165 LDR R0 #4096;

1166 LDR R1 $c

1167 AND R1 R1 #$0FFF

1168 OR R1 R1 R0

1169 STR R1 $c; 1

1170 LDR R0 #$0;

1171 STR R0 $b;

1172 JMP END0;

1173 JMP OVERELSE913514918

1174EL1813024393 CLFZ;

1175 LDR R0 #61440;

1176 LDR R1 $c

1177 AND R1 R1 #$0FFF

1178 OR R1 R1 R0

1179 STR R1 $c; 1

1180 JMP END0;

1181OVERELSE913514918 NOOP;

1182 JMP OVERELSE472967438

1183else979005330 NOOP

1184 LDR R0 #0;

1185 STR R0 $22;

1186 LDR R0 #4096;

1187 LDR R1 $c

1188 AND R1 R1 #$0FFF

1189 OR R1 R1 R0

1190 STR R1 $c; 1

1191 LDR R0 #$0;

1192 STR R0 $b;

1193 JMP END0;

1194OVERELSE472967438 NOOP;

1195 JMP OVERELSE953707828

1196EL844282640 CLFZ; tutu man man

1197 LDR R0 #61440;

1198 LDR R1 $c

1199 AND R1 R1 #$0FFF

1200 OR R1 R1 R0

1201 STR R1 $c; 1

1202 JMP END0;

1203OVERELSE953707828 NOOP;

1204 JMP OVERELSE1527489193

1205EL289029839 CLFZ;

1206 LDR R0 #61440;

1207 LDR R1 $c

1208 AND R1 R1 #$0FFF

1209 OR R1 R1 R0

1210 STR R1 $c; 1

1211 JMP END0;

1212OVERELSE1527489193 NOOP;

1213 JMP OVERELSE1749313326

1214else1019788713 NOOP

1215 LDR R0 #0;

1216 STR R0 $20;

1217 LDR R0 #4096;

1218 LDR R1 $c

1219 AND R1 R1 #$0FFF

1220 OR R1 R1 R0

1221 STR R1 $c; 1

1222 LDR R0 #$0;

1223 STR R0 $b;

1224 JMP END0;

1225OVERELSE1749313326 NOOP;

1226 JMP OVERELSE434598136

1227EL441429295 CLFZ; tutu man man

1228 LDR R0 #61440;

1229 LDR R1 $c

1230 AND R1 R1 #$0FFF

1231 OR R1 R1 R0

1232 STR R1 $c; 1

1233 JMP END0;

1234OVERELSE434598136 NOOP;

1235OVERELSE428623244 NOOP;

1236 JMP OVERELSE1337898102

1237EL603194796 CLFZ; tutu man man

1238 LDR R0 #61440;

1239 LDR R1 $c

1240 AND R1 R1 #$0FFF

1241 OR R1 R1 R0

1242 STR R1 $c; 1

1243 JMP END0;

1244OVERELSE1337898102 NOOP;

1245 JMP ENDS5988805581

1246case427427792 NOOP;

1247 LDR R0 #case1039795656 ;

1248 STR R0 $15 ;

1249 LDR R0 #case1704969132 ;

1250 STR R0 $16 ;

1251 LDR R0 #case1837520939 ;

1252 STR R0 $17 ;

1253 LDR R0 #case634345685 ;

1254 STR R0 $18 ;

1255 LDR R0 #20

1256 SWITCH R1 R0 ;

1257case1039795656 NOOP;

1258 LDR R0 #case1990886754 ;

1259 STR R0 $1a ;

1260 LDR R0 #25

1261 SWITCH R1 R0 ;

1262case1990886754 NOOP;

1263 LDR R0 #case2002813970 ;

1264 STR R0 $1c ;

1265 LDR R0 #27

1266 SWITCH R1 R0 ;

1267case2002813970 NOOP;

1268 LDR R0 #1794

1269 ADD R1 R6 #0

1270 STR R1 #0;

1271 SENDATA R0;

1272 SUB R4 #65535;

1273 SZ TOQSTART2016280182;

1274 ADD R4 R4 #1;

1275 JMP QS2016280182;

1276TOQSTART2016280182 LDR R4 #65528;

1277QS2016280182 CLFZ; data man man

1278 LDR R0 #TEN2034748129

1279 STR R0 $b

1280TEN2034748129 NOOP;

1281 LDR R0 $2

1282 SUB R0 #0

1283 SZ EL1000026709

1284IF1000026709 CLFZ; titi man man

1285 LDR R0 $2

1286 AND R0 R0 #$0002

1287 PRESENT R0 else2028592146

1288 LDR R0 #3586;

1289 ADD R1 R6 #0

1290 STR R1 #0;

1291 SENDATA R0;

1292 SUB R4 #65535;

1293 SZ TOQSTART1948179626;

1294 ADD R4 R4 #1;

1295 JMP QS1948179626;

1296TOQSTART1948179626 LDR R4 #65528;

1297QS1948179626 CLFZ;

1298 LDR R0 #TEN1942023644

1299 STR R0 $b

1300TEN1942023644 NOOP;

1301 LDR R0 $2

1302 SUB R0 #0

1303 SZ EL1701118143

1304IF1701118143 CLFZ

1305 ESL #20

1306 LDR R0 #1;

1307 STR R0 $14;

1308 LDR R0 #4096;

1309 LDR R1 $c

1310 AND R1 R1 #$0FFF

1311 OR R1 R1 R0

1312 STR R1 $c; 1

1313 LDR R0 #$0;

1314 STR R0 $b;

1315 JMP END0;

1316 JMP OVERELSE355862427

1317EL1701118143 CLFZ;

1318 LDR R0 #61440;

1319 LDR R1 $c

1320 AND R1 R1 #$0FFF

1321 OR R1 R1 R0

1322 STR R1 $c; 1

1323 JMP END0;

1324OVERELSE355862427 NOOP;

1325 JMP OVERELSE307857322

1326else2028592146 NOOP

1327 LDR R0 #case1952411864 ;

1328 STR R0 $1e ;

1329 LDR R0 #case1460536674 ;

1330 STR R0 $1f ;

1331 LDR R0 #29

1332 SWITCH R1 R0 ;

1333case1952411864 NOOP;

1334 LDR R0 #case1982807027 ;

1335 STR R0 $21 ;

1336 LDR R0 #32

1337 SWITCH R1 R0 ;

1338case1982807027 NOOP;

1339 LDR R0 #2050

1340 ADD R1 R6 #0

1341 STR R1 #0;

1342 SENDATA R0;

1343 SUB R4 #65535;

1344 SZ TOQSTART1897777520;

1345 ADD R4 R4 #1;

1346 JMP QS1897777520;

1347TOQSTART1897777520 LDR R4 #65528;

1348QS1897777520 CLFZ; data man man

1349 LDR R0 #TEN1891621538

1350 STR R0 $b

1351TEN1891621538 NOOP;

1352 LDR R0 $2

1353 SUB R0 #0

1354 SZ EL656494052

1355IF656494052 CLFZ; titi man man

1356 LDR R0 $2

1357 AND R0 R0 #$0002

1358 PRESENT R0 else1910089485

1359 LDR R0 #2306;

1360 ADD R1 R6 #0

1361 STR R1 #0;

1362 SENDATA R0;

1363 SUB R4 #65535;

1364 SZ TOQSTART1903548754;

1365 ADD R4 R4 #1;

1366 JMP QS1903548754;

1367TOQSTART1903548754 LDR R4 #65528;

1368QS1903548754 CLFZ;

1369 LDR R0 #TEN1922016701

1370 STR R0 $b

1371TEN1922016701 NOOP;

1372 LDR R0 $2

1373 SUB R0 #0

1374 SZ EL1745462337

1375IF1745462337 CLFZ

1376 LDR R0 #1;

1377 STR R0 $1d;

1378 LDR R0 #2562

1379 ADD R1 R6 #0

1380 STR R1 #0;

1381 SENDATA R0;

1382 SUB R4 #65535;

1383 SZ TOQSTART1926248939;

1384 ADD R4 R4 #1;

1385 JMP QS1926248939;

1386TOQSTART1926248939 LDR R4 #65528;

1387QS1926248939 CLFZ; data man man

1388 LDR R0 #TEN1920092957

1389 STR R0 $b

1390TEN1920092957 NOOP;

1391 LDR R0 $2

1392 SUB R0 #0

1393 SZ EL263006753

1394IF263006753 CLFZ; titi man man

1395 LDR R0 $2

1396 AND R0 R0 #$0002

1397 PRESENT R0 else1938560904

1398 LDR R0 #2818;

1399 ADD R1 R6 #0

1400 STR R1 #0;

1401 SENDATA R0;

1402 SUB R4 #65535;

1403 SZ TOQSTART1932020173;

1404 ADD R4 R4 #1;

1405 JMP QS1932020173;

1406TOQSTART1932020173 LDR R4 #65528;

1407QS1932020173 CLFZ;

1408 LDR R0 #TEN1851992401

1409 STR R0 $b

1410TEN1851992401 NOOP;

1411 LDR R0 $2

1412 SUB R0 #0

1413 SZ EL1900667016

1414IF1900667016 CLFZ

1415 ESL #20

1416 LDR R0 #1;

1417 STR R0 $14;

1418 LDR R0 #4096;

1419 LDR R1 $c

1420 AND R1 R1 #$0FFF

1421 OR R1 R1 R0

1422 STR R1 $c; 1

1423 LDR R0 #$0;

1424 STR R0 $b;

1425 JMP END0;

1426 JMP OVERELSE1046922301

1427EL1900667016 CLFZ;

1428 LDR R0 #61440;

1429 LDR R1 $c

1430 AND R1 R1 #$0FFF

1431 OR R1 R1 R0

1432 STR R1 $c; 1

1433 JMP END0;

1434OVERELSE1046922301 NOOP;

1435 JMP OVERELSE108308449

1436else1938560904 NOOP

1437 LDR R0 #0;

1438 STR R0 $22;

1439 LDR R0 #4096;

1440 LDR R1 $c

1441 AND R1 R1 #$0FFF

1442 OR R1 R1 R0

1443 STR R1 $c; 1

1444 LDR R0 #$0;

1445 STR R0 $b;

1446 JMP END0;

1447OVERELSE108308449 NOOP;

1448 JMP OVERELSE1087115211

1449EL263006753 CLFZ; tutu man man

1450 LDR R0 #61440;

1451 LDR R1 $c

1452 AND R1 R1 #$0FFF

1453 OR R1 R1 R0

1454 STR R1 $c; 1

1455 JMP END0;

1456OVERELSE1087115211 NOOP;

1457 JMP OVERELSE433086101

1458EL1745462337 CLFZ;

1459 LDR R0 #61440;

1460 LDR R1 $c

1461 AND R1 R1 #$0FFF

1462 OR R1 R1 R0

1463 STR R1 $c; 1

1464 JMP END0;

1465OVERELSE433086101 NOOP;

1466 JMP OVERELSE285178850

1467else1910089485 NOOP

1468 LDR R0 #4096;

1469 LDR R1 $c

1470 AND R1 R1 #$0FFF

1471 OR R1 R1 R0

1472 STR R1 $c; 1

1473 LDR R0 #$0;

1474 STR R0 $b;

1475 JMP END0;

1476OVERELSE285178850 NOOP;

1477 JMP OVERELSE1259162392

1478EL656494052 CLFZ; tutu man man

1479 LDR R0 #61440;

1480 LDR R1 $c

1481 AND R1 R1 #$0FFF

1482 OR R1 R1 R0

1483 STR R1 $c; 1

1484 JMP END0;

1485OVERELSE1259162392 NOOP;

1486 JMP ENDS17866004130

1487ENDS17866004130 NOOP

1488 JMP ENDS17804444300

1489case1460536674 NOOP;

1490 LDR R0 #case1877385829 ;

1491 STR R0 $23 ;

1492 LDR R0 #34

1493 SWITCH R1 R0 ;

1494case1877385829 NOOP;

1495 LDR R0 #3074

1496 ADD R1 R6 #0

1497 STR R1 #0;

1498 SENDATA R0;

1499 SUB R4 #65535;

1500 SZ TOQSTART1889697794;

1501 ADD R4 R4 #1;

1502 JMP QS1889697794;

1503TOQSTART1889697794 LDR R4 #65528;

1504QS1889697794 CLFZ; data man man

1505 LDR R0 #TEN1881618067

1506 STR R0 $b

1507TEN1881618067 NOOP;

1508 LDR R0 $2

1509 SUB R0 #0

1510 SZ EL678666149

1511IF678666149 CLFZ; titi man man

1512 LDR R0 $2

1513 AND R0 R0 #$0002

1514 PRESENT R0 else2099394127

1515 LDR R0 #3330;

1516 ADD R1 R6 #0

1517 STR R1 #0;

1518 SENDATA R0;

1519 SUB R4 #65535;

1520 SZ TOQSTART2105550109;

1521 ADD R4 R4 #1;

1522 JMP QS2105550109;

1523TOQSTART2105550109 LDR R4 #65528;

1524QS2105550109 CLFZ;

1525 LDR R0 #TEN2087466911

1526 STR R0 $b

1527TEN2087466911 NOOP;

1528 LDR R0 $2

1529 SUB R0 #0

1530 SZ EL121558499

1531IF121558499 CLFZ

1532 ESL #20

1533 LDR R0 #1;

1534 STR R0 $14;

1535 LDR R0 #4096;

1536 LDR R1 $c

1537 AND R1 R1 #$0FFF

1538 OR R1 R1 R0

1539 STR R1 $c; 1

1540 LDR R0 #$0;

1541 STR R0 $b;

1542 JMP END0;

1543 JMP OVERELSE417566369

1544EL121558499 CLFZ;

1545 LDR R0 #61440;

1546 LDR R1 $c

1547 AND R1 R1 #$0FFF

1548 OR R1 R1 R0

1549 STR R1 $c; 1

1550 JMP END0;

1551OVERELSE417566369 NOOP;

1552 JMP OVERELSE1338724988

1553else2099394127 NOOP

1554 LDR R0 #4096;

1555 LDR R1 $c

1556 AND R1 R1 #$0FFF

1557 OR R1 R1 R0

1558 STR R1 $c; 1

1559 LDR R0 #$0;

1560 STR R0 $b;

1561 JMP END0;

1562OVERELSE1338724988 NOOP;

1563 JMP OVERELSE1282795520

1564EL678666149 CLFZ; tutu man man

1565 LDR R0 #61440;

1566 LDR R1 $c

1567 AND R1 R1 #$0FFF

1568 OR R1 R1 R0

1569 STR R1 $c; 1

1570 JMP END0;

1571OVERELSE1282795520 NOOP;

1572 JMP ENDS17985276280

1573ENDS17985276280 NOOP

1574 JMP ENDS17804444301

1575ENDS17804444300 NOOP

1576ENDS17804444301 NOOP

1577OVERELSE307857322 NOOP;

1578 JMP OVERELSE956300990

1579EL1000026709 CLFZ; tutu man man

1580 LDR R0 #61440;

1581 LDR R1 $c

1582 AND R1 R1 #$0FFF

1583 OR R1 R1 R0

1584 STR R1 $c; 1

1585 JMP END0;

1586OVERELSE956300990 NOOP;

1587 JMP ENDS17042641480

1588ENDS17042641480 NOOP

1589 JMP ENDS17227320950

1590ENDS17227320950 NOOP

1591 JMP ENDS17165761130

1592case1704969132 NOOP;

1593 ESL #20

1594 LDR R0 #1;

1595 STR R0 $14;

1596 LDR R0 #3842;

1597 ADD R1 R6 #0

1598 STR R1 #0;

1599 SENDATA R0;

1600 SUB R4 #65535;

1601 SZ TOQSTART2080156682;

1602 ADD R4 R4 #1;

1603 JMP QS2080156682;

1604TOQSTART2080156682 LDR R4 #65528;

1605QS2080156682 CLFZ;

1606 LDR R0 #TEN2061688735

1607 STR R0 $b

1608TEN2061688735 NOOP;

1609 LDR R0 $2

1610 SUB R0 #0

1611 SZ EL229613760

1612IF229613760 CLFZ

1613 ESL #25

1614 LDR R0 #2;

1615 STR R0 $14;

1616 LDR R0 #0;

1617 STR R0 $24;

1618 ESL #26

1619 LDR R0 #4098;

1620 ADD R1 R6 #0

1621 STR R1 #0;

1622 SENDATA R0;

1623 SUB R4 #65535;

1624 SZ TOQSTART2068229466;

1625 ADD R4 R4 #1;

1626 JMP QS2068229466;

1627TOQSTART2068229466 LDR R4 #65528;

1628QS2068229466 CLFZ;

1629 LDR R0 #TEN2146710059

1630 STR R0 $b

1631TEN2146710059 NOOP;

1632 LDR R0 $2

1633 SUB R0 #0

1634 SZ EL2056378071

1635IF2056378071 CLFZ

1636 ESL #28

1637 LDR R0 #0;

1638 STR R0 $27;

1639 LDR R0 #4354

1640 ADD R1 R6 #0

1641 STR R1 #0;

1642 SENDATA R0;

1643 SUB R4 #65535;

1644 SZ TOQSTART2140554076;

1645 ADD R4 R4 #1;

1646 JMP QS2140554076;

1647TOQSTART2140554076 LDR R4 #65528;

1648QS2140554076 CLFZ; data man man

1649 LDR R0 #TEN2137869017

1650 STR R0 $b

1651TEN2137869017 NOOP;

1652 LDR R0 $2

1653 SUB R0 #0

1654 SZ EL923065592

1655IF923065592 CLFZ; titi man man

1656 LDR R0 $2

1657 AND R0 R0 #$0002

1658 PRESENT R0 else2144024999

1659 LDR R0 #0;

1660 STR R0 $29;

1661 ESL #29

1662 LDR R0 #8192;

1663 LDR R1 $c

1664 AND R1 R1 #$0FFF

1665 OR R1 R1 R0

1666 STR R1 $c; 1

1667 LDR R0 #1;

1668 STR R0 $24;

1669 LDR R0 #4610;

1670 ADD R1 R6 #0

1671 STR R1 #0;

1672 SENDATA R0;

1673 SUB R4 #65535;

1674 SZ TOQSTART2125941801;

1675 ADD R4 R4 #1;

1676 JMP QS2125941801;

1677TOQSTART2125941801 LDR R4 #65528;

1678QS2125941801 CLFZ;

1679 LDR R0 #TEN2132097783

1680 STR R0 $b

1681TEN2132097783 NOOP;

1682 LDR R0 $2

1683 SUB R0 #0

1684 SZ EL770666136

1685IF770666136 CLFZ

1686 ESL #46

1687 LDR R0 #0;

1688 STR R0 $3d;

1689 LDR R0 #4096;

1690 LDR R1 $c

1691 AND R1 R1 #$0FFF

1692 OR R1 R1 R0

1693 STR R1 $c; 1

1694 LDR R0 #$0;

1695 STR R0 $b;

1696 JMP END0;

1697 JMP OVERELSE1012013632

1698EL770666136 CLFZ;

1699 LDR R0 #61440;

1700 LDR R1 $c

1701 AND R1 R1 #$0FFF

1702 OR R1 R1 R0

1703 STR R1 $c; 1

1704 JMP END0;

1705OVERELSE1012013632 NOOP;

1706 JMP OVERELSE2064017673

1707else2144024999 NOOP

1708 LDR R0 #1;

1709 STR R0 $29;

1710 LDR R0 #4866;

1711 ADD R1 R6 #0

1712 STR R1 #0;

1713 SENDATA R0;

1714 SUB R4 #65535;

1715 SZ TOQSTART2118631572;

1716 ADD R4 R4 #1;

1717 JMP QS2118631572;

1718TOQSTART2118631572 LDR R4 #65528;

1719QS2118631572 CLFZ;

1720 LDR R0 #TEN2096307953

1721 STR R0 $b

1722TEN2096307953 NOOP;

1723 LDR R0 $2

1724 SUB R0 #0

1725 SZ EL1193965135

1726IF1193965135 CLFZ

1727 LDR R0 #0;

1728 STR R0 $2e;

1729 LDR R0 #0;

1730 STR R0 $32;

1731 LDR R0 #5122

1732 ADD R1 R6 #0

1733 STR R1 #0;

1734 SENDATA R0;

1735 SUB R4 #65535;

1736 SZ TOQSTART2090151971;

1737 ADD R4 R4 #1;

1738 JMP QS2090151971;

1739TOQSTART2090151971 LDR R4 #65528;

1740QS2090151971 CLFZ; data man man

1741 LDR R0 #TEN2108235169

1742 STR R0 $b

1743TEN2108235169 NOOP;

1744 LDR R0 $2

1745 SUB R0 #0

1746 SZ EL1640718675

1747IF1640718675 CLFZ; titi man man

1748 LDR R0 $2

1749 AND R0 R0 #$0002

1750 PRESENT R0 else2102079186

1751 LDR R0 #6402;

1752 ADD R1 R6 #0

1753 STR R1 #0;

1754 SENDATA R0;

1755 SUB R4 #65535;

1756 SZ TOQSTART2118623389;

1757 ADD R4 R4 #1;

1758 JMP QS2118623389;

1759TOQSTART2118623389 LDR R4 #65528;

1760QS2118623389 CLFZ;

1761 LDR R0 #TEN2112467407

1762 STR R0 $b

1763TEN2112467407 NOOP;

1764 LDR R0 $2

1765 SUB R0 #0

1766 SZ EL1815290227

1767IF1815290227 CLFZ

1768 ESL #39

1769 LDR R0 #1;

1770 STR R0 $2e;

1771 LDR R0 #4096;

1772 LDR R1 $c

1773 AND R1 R1 #$0FFF

1774 OR R1 R1 R0

1775 STR R1 $c; 1

1776 LDR R0 #$0;

1777 STR R0 $b;

1778 JMP END0;

1779 JMP OVERELSE108713667

1780EL1815290227 CLFZ;

1781 LDR R0 #61440;

1782 LDR R1 $c

1783 AND R1 R1 #$0FFF

1784 OR R1 R1 R0

1785 STR R1 $c; 1

1786 JMP END0;

1787OVERELSE108713667 NOOP;

1788 JMP OVERELSE1346364590

1789else2102079186 NOOP

1790 LDR R0 #0;

1791 STR R0 $34;

1792 LDR R0 #0;

1793 STR R0 $36;

1794 LDR R0 #5378

1795 ADD R1 R6 #0

1796 STR R1 #0;

1797 SENDATA R0;

1798 SUB R4 #65535;

1799 SZ TOQSTART2124394623;

1800 ADD R4 R4 #1;

1801 JMP QS2124394623;

1802TOQSTART2124394623 LDR R4 #65528;

1803QS2124394623 CLFZ; data man man

1804 LDR R0 #TEN2044366851

1805 STR R0 $b

1806TEN2044366851 NOOP;

1807 LDR R0 $2

1808 SUB R0 #0

1809 SZ EL177629964

1810IF177629964 CLFZ; titi man man

1811 LDR R0 $2

1812 AND R0 R0 #$0002

1813 PRESENT R0 else2038210869

1814 LDR R0 #5634;

1815 ADD R1 R6 #0

1816 STR R1 #0;

1817 SENDATA R0;

1818 SUB R4 #65535;

1819 SZ TOQSTART2057833063;

1820 ADD R4 R4 #1;

1821 JMP QS2057833063;

1822TOQSTART2057833063 LDR R4 #65528;

1823QS2057833063 CLFZ;

1824 LDR R0 #TEN2051677081

1825 STR R0 $b

1826TEN2051677081 NOOP;

1827 LDR R0 $2

1828 SUB R0 #0

1829 SZ EL301740499

1830IF301740499 CLFZ

1831 LDR R0 #1;

1832 STR R0 $36;

1833 LDR R0 #5890

1834 ADD R1 R6 #0

1835 STR R1 #0;

1836 SENDATA R0;

1837 SUB R4 #65535;

1838 SZ TOQSTART2069760279;

1839 ADD R4 R4 #1;

1840 JMP QS2069760279;

1841TOQSTART2069760279 LDR R4 #65528;

1842QS2069760279 CLFZ; data man man

1843 LDR R0 #TEN2063604296

1844 STR R0 $b

1845TEN2063604296 NOOP;

1846 LDR R0 $2

1847 SUB R0 #0

1848 SZ EL1762023986

1849IF1762023986 CLFZ; titi man man

1850 LDR R0 $2

1851 AND R0 R0 #$0002

1852 PRESENT R0 else2082072244

1853 LDR R0 #6146;

1854 ADD R1 R6 #0

1855 STR R1 #0;

1856 SENDATA R0;

1857 SUB R4 #65535;

1858 SZ TOQSTART2086304482;

1859 ADD R4 R4 #1;

1860 JMP QS2086304482;

1861TOQSTART2086304482 LDR R4 #65528;

1862QS2086304482 CLFZ;

1863 LDR R0 #TEN2080148499

1864 STR R0 $b

1865TEN2080148499 NOOP;

1866 LDR R0 $2

1867 SUB R0 #0

1868 SZ EL91746800

1869IF91746800 CLFZ

1870 ESL #39

1871 LDR R0 #1;

1872 STR R0 $2e;

1873 LDR R0 #4096;

1874 LDR R1 $c

1875 AND R1 R1 #$0FFF

1876 OR R1 R1 R0

1877 STR R1 $c; 1

1878 LDR R0 #$0;

1879 STR R0 $b;

1880 JMP END0;

1881 JMP OVERELSE929505717

1882EL91746800 CLFZ;

1883 LDR R0 #61440;

1884 LDR R1 $c

1885 AND R1 R1 #$0FFF

1886 OR R1 R1 R0

1887 STR R1 $c; 1

1888 JMP END0;

1889OVERELSE929505717 NOOP;

1890 JMP OVERELSE1390708784

1891else2082072244 NOOP

1892 LDR R0 #0;

1893 STR R0 $3b;

1894 LDR R0 #4096;

1895 LDR R1 $c

1896 AND R1 R1 #$0FFF

1897 OR R1 R1 R0

1898 STR R1 $c; 1

1899 LDR R0 #$0;

1900 STR R0 $b;

1901 JMP END0;

1902OVERELSE1390708784 NOOP;

1903 JMP OVERELSE2118164250

1904EL1762023986 CLFZ; tutu man man

1905 LDR R0 #61440;

1906 LDR R1 $c

1907 AND R1 R1 #$0FFF

1908 OR R1 R1 R0

1909 STR R1 $c; 1

1910 JMP END0;

1911OVERELSE2118164250 NOOP;

1912 JMP OVERELSE1019183976

1913EL301740499 CLFZ;

1914 LDR R0 #61440;

1915 LDR R1 $c

1916 AND R1 R1 #$0FFF

1917 OR R1 R1 R0

1918 STR R1 $c; 1

1919 JMP END0;

1920OVERELSE1019183976 NOOP;

1921 JMP OVERELSE1485513995

1922else2038210869 NOOP

1923 LDR R0 #0;

1924 STR R0 $39;

1925 LDR R0 #4096;

1926 LDR R1 $c

1927 AND R1 R1 #$0FFF

1928 OR R1 R1 R0

1929 STR R1 $c; 1

1930 LDR R0 #$0;

1931 STR R0 $b;

1932 JMP END0;

1933OVERELSE1485513995 NOOP;

1934 JMP OVERELSE68520757

1935EL177629964 CLFZ; tutu man man

1936 LDR R0 #61440;

1937 LDR R1 $c

1938 AND R1 R1 #$0FFF

1939 OR R1 R1 R0

1940 STR R1 $c; 1

1941 JMP END0;

1942OVERELSE68520757 NOOP;

1943OVERELSE1346364590 NOOP;

1944 JMP OVERELSE688584249

1945EL1640718675 CLFZ; tutu man man

1946 LDR R0 #61440;

1947 LDR R1 $c

1948 AND R1 R1 #$0FFF

1949 OR R1 R1 R0

1950 STR R1 $c; 1

1951 JMP END0;

1952OVERELSE688584249 NOOP;

1953 JMP OVERELSE410396025

1954EL1193965135 CLFZ;

1955 LDR R0 #61440;

1956 LDR R1 $c

1957 AND R1 R1 #$0FFF

1958 OR R1 R1 R0

1959 STR R1 $c; 1

1960 JMP END0;

1961OVERELSE410396025 NOOP;

1962OVERELSE2064017673 NOOP;

1963 JMP OVERELSE1320866335

1964EL923065592 CLFZ; tutu man man

1965 LDR R0 #61440;

1966 LDR R1 $c

1967 AND R1 R1 #$0FFF

1968 OR R1 R1 R0

1969 STR R1 $c; 1

1970 JMP END0;

1971OVERELSE1320866335 NOOP;

1972 JMP OVERELSE492903940

1973EL2056378071 CLFZ;

1974 LDR R0 #61440;

1975 LDR R1 $c

1976 AND R1 R1 #$0FFF

1977 OR R1 R1 R0

1978 STR R1 $c; 1

1979 JMP END0;

1980OVERELSE492903940 NOOP;

1981 JMP OVERELSE1126365069

1982EL229613760 CLFZ;

1983 LDR R0 #61440;

1984 LDR R1 $c

1985 AND R1 R1 #$0FFF

1986 OR R1 R1 R0

1987 STR R1 $c; 1

1988 JMP END0;

1989OVERELSE1126365069 NOOP;

1990 JMP ENDS17165761131

1991case1837520939 NOOP;

1992 ESL #25

1993 LDR R0 #case1893938214 ;

1994 STR R0 $25 ;

1995 LDR R0 #case1509715246 ;

1996 STR R0 $26 ;

1997 LDR R0 #36

1998 SWITCH R1 R0 ;

1999case1893938214 NOOP;

2000 ESL #26

2001 LDR R0 #case1863543051 ;

2002 STR R0 $28 ;

2003 LDR R0 #39

2004 SWITCH R1 R0 ;

2005case1863543051 NOOP;

2006 ESL #28

2007 LDR R0 #case1949726805 ;

2008 STR R0 $2a ;

2009 LDR R0 #case1468176276 ;

2010 STR R0 $2b ;

2011 LDR R0 #41

2012 SWITCH R1 R0 ;

2013case1949726805 NOOP;

2014 LDR R0 #case1945879316 ;

2015 STR R0 $2d ;

2016 LDR R0 #44

2017 SWITCH R1 R0 ;

2018case1945879316 NOOP;

2019 ESL #30

2020 LDR R0 #0;

2021 STR R0 $2c;

2022 LDR R0 #6658;

2023 ADD R1 R6 #0

2024 STR R1 #0;

2025 SENDATA R0;

2026 SUB R4 #65535;

2027 SZ TOQSTART1933567351;

2028 ADD R4 R4 #1;

2029 JMP QS1933567351;

2030TOQSTART1933567351 LDR R4 #65528;

2031QS1933567351 CLFZ;

2032 LDR R0 #TEN1913945157

2033 STR R0 $b

2034TEN1913945157 NOOP;

2035 LDR R0 $2

2036 SUB R0 #0

2037 SZ EL1554059441

2038IF1554059441 CLFZ

2039 ESL #28

2040 LDR R0 #0;

2041 STR R0 $27;

2042 LDR R0 #6914

2043 ADD R1 R6 #0

2044 STR R1 #0;

2045 SENDATA R0;

2046 SUB R4 #65535;

2047 SZ TOQSTART1920101139;

2048 ADD R4 R4 #1;

2049 JMP QS1920101139;

2050TOQSTART1920101139 LDR R4 #65528;

2051QS1920101139 CLFZ; data man man

2052 LDR R0 #TEN2000128911

2053 STR R0 $b

2054TEN2000128911 NOOP;

2055 LDR R0 $2

2056 SUB R0 #0

2057 SZ EL423552186

2058IF423552186 CLFZ; titi man man

2059 LDR R0 $2

2060 AND R0 R0 #$0002

2061 PRESENT R0 else2006669642

2062 LDR R0 #0;

2063 STR R0 $29;

2064 ESL #29

2065 LDR R0 #8192;

2066 LDR R1 $c

2067 AND R1 R1 #$0FFF

2068 OR R1 R1 R0

2069 STR R1 $c; 1

2070 LDR R0 #1;

2071 STR R0 $24;

2072 LDR R0 #7170;

2073 ADD R1 R6 #0

2074 STR R1 #0;

2075 SENDATA R0;

2076 SUB R4 #65535;

2077 SZ TOQSTART1988201695;

2078 ADD R4 R4 #1;

2079 JMP QS1988201695;

2080TOQSTART1988201695 LDR R4 #65528;

2081QS1988201695 CLFZ;

2082 LDR R0 #TEN1996281422

2083 STR R0 $b

2084TEN1996281422 NOOP;

2085 LDR R0 $2

2086 SUB R0 #0

2087 SZ EL906503943

2088IF906503943 CLFZ

2089 ESL #46

2090 LDR R0 #0;

2091 STR R0 $3d;

2092 LDR R0 #4096;

2093 LDR R1 $c

2094 AND R1 R1 #$0FFF

2095 OR R1 R1 R0

2096 STR R1 $c; 1

2097 LDR R0 #$0;

2098 STR R0 $b;

2099 JMP END0;

2100 JMP OVERELSE364211815

2101EL906503943 CLFZ;

2102 LDR R0 #61440;

2103 LDR R1 $c

2104 AND R1 R1 #$0FFF

2105 OR R1 R1 R0

2106 STR R1 $c; 1

2107 JMP END0;

2108OVERELSE364211815 NOOP;

2109 JMP OVERELSE2039816422

2110else2006669642 NOOP

2111 LDR R0 #1;

2112 STR R0 $29;

2113 LDR R0 #7426;

2114 ADD R1 R6 #0

2115 STR R1 #0;

2116 SENDATA R0;

2117 SUB R4 #65535;

2118 SZ TOQSTART1983969457;

2119 ADD R4 R4 #1;

2120 JMP QS1983969457;

2121TOQSTART1983969457 LDR R4 #65528;

2122QS1983969457 CLFZ;

2123 LDR R0 #TEN1965886259

2124 STR R0 $b

2125TEN1965886259 NOOP;

2126 LDR R0 $2

2127 SUB R0 #0

2128 SZ EL182464342

2129IF182464342 CLFZ

2130 LDR R0 #0;

2131 STR R0 $2e;

2132 LDR R0 #0;

2133 STR R0 $32;

2134 LDR R0 #7682

2135 ADD R1 R6 #0

2136 STR R1 #0;

2137 SENDATA R0;

2138 SUB R4 #65535;

2139 SZ TOQSTART1972042241;

2140 ADD R4 R4 #1;

2141 JMP QS1972042241;

2142TOQSTART1972042241 LDR R4 #65528;

2143QS1972042241 CLFZ; data man man

2144 LDR R0 #TEN2052070012

2145 STR R0 $b

2146TEN2052070012 NOOP;

2147 LDR R0 $2

2148 SUB R0 #0

2149 SZ EL948042913

2150IF948042913 CLFZ; titi man man

2151 LDR R0 $2

2152 AND R0 R0 #$0002

2153 PRESENT R0 else2057071748

2154 LDR R0 #8962;

2155 ADD R1 R6 #0

2156 STR R1 #0;

2157 SENDATA R0;

2158 SUB R4 #65535;

2159 SZ TOQSTART2038603801;

2160 ADD R4 R4 #1;

2161 JMP QS2038603801;

2162TOQSTART2038603801 LDR R4 #65528;

2163QS2038603801 CLFZ;

2164 LDR R0 #TEN2045144532

2165 STR R0 $b

2166TEN2045144532 NOOP;

2167 LDR R0 $2

2168 SUB R0 #0

2169 SZ EL1624157026

2170IF1624157026 CLFZ

2171 ESL #39

2172 LDR R0 #1;

2173 STR R0 $2e;

2174 LDR R0 #4096;

2175 LDR R1 $c

2176 AND R1 R1 #$0FFF

2177 OR R1 R1 R0

2178 STR R1 $c; 1

2179 LDR R0 #$0;

2180 STR R0 $b;

2181 JMP END0;

2182 JMP OVERELSE1921304898

2183EL1624157026 CLFZ;

2184 LDR R0 #61440;

2185 LDR R1 $c

2186 AND R1 R1 #$0FFF

2187 OR R1 R1 R0

2188 STR R1 $c; 1

2189 JMP END0;

2190OVERELSE1921304898 NOOP;

2191 JMP OVERELSE1210526784

2192else2057071748 NOOP

2193 LDR R0 #0;

2194 STR R0 $34;

2195 LDR R0 #0;

2196 STR R0 $36;

2197 LDR R0 #7938

2198 ADD R1 R6 #0

2199 STR R1 #0;

2200 SENDATA R0;

2201 SUB R4 #65535;

2202 SZ TOQSTART2032832567;

2203 ADD R4 R4 #1;

2204 JMP QS2032832567;

2205TOQSTART2032832567 LDR R4 #65528;

2206QS2032832567 CLFZ; data man man

2207 LDR R0 #TEN2016288364

2208 STR R0 $b

2209TEN2016288364 NOOP;

2210 LDR R0 $2

2211 SUB R0 #0

2212 SZ EL862159749

2213IF862159749 CLFZ; titi man man

2214 LDR R0 $2

2215 AND R0 R0 #$0002

2216 PRESENT R0 else2022444347

2217 LDR R0 #8194;

2218 ADD R1 R6 #0

2219 STR R1 #0;

2220 SENDATA R0;

2221 SUB R4 #65535;

2222 SZ TOQSTART1708489244;

2223 ADD R4 R4 #1;

2224 JMP QS1708489244;

2225TOQSTART1708489244 LDR R4 #65528;

2226QS1708489244 CLFZ;

2227 LDR R0 #TEN1715029976

2228 STR R0 $b

2229TEN1715029976 NOOP;

2230 LDR R0 $2

2231 SUB R0 #0

2232 SZ EL1939131068

2233IF1939131068 CLFZ

2234 LDR R0 #1;

2235 STR R0 $36;

2236 LDR R0 #8450

2237 ADD R1 R6 #0

2238 STR R1 #0;

2239 SENDATA R0;

2240 SUB R4 #65535;

2241 SZ TOQSTART1696562028;

2242 ADD R4 R4 #1;

2243 JMP QS1696562028;

2244TOQSTART1696562028 LDR R4 #65528;

2245QS1696562028 CLFZ; data man man

2246 LDR R0 #TEN1701563764

2247 STR R0 $b

2248TEN1701563764 NOOP;

2249 LDR R0 $2

2250 SUB R0 #0

2251 SZ EL568581734

2252IF568581734 CLFZ; titi man man

2253 LDR R0 $2

2254 AND R0 R0 #$0002

2255 PRESENT R0 else1683095817

2256 LDR R0 #8706;

2257 ADD R1 R6 #0

2258 STR R1 #0;

2259 SENDATA R0;

2260 SUB R4 #65535;

2261 SZ TOQSTART1689251799;

2262 ADD R4 R4 #1;

2263 JMP QS1689251799;

2264TOQSTART1689251799 LDR R4 #65528;

2265QS1689251799 CLFZ;

2266 LDR R0 #TEN1671168601

2267 STR R0 $b

2268TEN1671168601 NOOP;

2269 LDR R0 $2

2270 SUB R0 #0

2271 SZ EL520386551

2272IF520386551 CLFZ

2273 ESL #39

2274 LDR R0 #1;

2275 STR R0 $2e;

2276 LDR R0 #4096;

2277 LDR R1 $c

2278 AND R1 R1 #$0FFF

2279 OR R1 R1 R0

2280 STR R1 $c; 1

2281 LDR R0 #$0;

2282 STR R0 $b;

2283 JMP END0;

2284 JMP OVERELSE1502441947

2285EL520386551 CLFZ;

2286 LDR R0 #61440;

2287 LDR R1 $c

2288 AND R1 R1 #$0FFF

2289 OR R1 R1 R0

2290 STR R1 $c; 1

2291 JMP END0;

2292OVERELSE1502441947 NOOP;

2293 JMP OVERELSE939896936

2294else1683095817 NOOP

2295 LDR R0 #0;

2296 STR R0 $3b;

2297 LDR R0 #4096;

2298 LDR R1 $c

2299 AND R1 R1 #$0FFF

2300 OR R1 R1 R0

2301 STR R1 $c; 1

2302 LDR R0 #$0;

2303 STR R0 $b;

2304 JMP END0;

2305OVERELSE939896936 NOOP;

2306 JMP OVERELSE189806546

2307EL568581734 CLFZ; tutu man man

2308 LDR R0 #61440;

2309 LDR R1 $c

2310 AND R1 R1 #$0FFF

2311 OR R1 R1 R0

2312 STR R1 $c; 1

2313 JMP END0;

2314OVERELSE189806546 NOOP;

2315 JMP OVERELSE93429927

2316EL1939131068 CLFZ;

2317 LDR R0 #61440;

2318 LDR R1 $c

2319 AND R1 R1 #$0FFF

2320 OR R1 R1 R0

2321 STR R1 $c; 1

2322 JMP END0;

2323OVERELSE93429927 NOOP;

2324 JMP OVERELSE2124923516

2325else2022444347 NOOP

2326 LDR R0 #0;

2327 STR R0 $39;

2328 LDR R0 #4096;

2329 LDR R1 $c

2330 AND R1 R1 #$0FFF

2331 OR R1 R1 R0

2332 STR R1 $c; 1

2333 LDR R0 #$0;

2334 STR R0 $b;

2335 JMP END0;

2336OVERELSE2124923516 NOOP;

2337 JMP OVERELSE1153160343

2338EL862159749 CLFZ; tutu man man

2339 LDR R0 #61440;

2340 LDR R1 $c

2341 AND R1 R1 #$0FFF

2342 OR R1 R1 R0

2343 STR R1 $c; 1

2344 JMP END0;

2345OVERELSE1153160343 NOOP;

2346OVERELSE1210526784 NOOP;

2347 JMP OVERELSE2143780480

2348EL948042913 CLFZ; tutu man man

2349 LDR R0 #61440;

2350 LDR R1 $c

2351 AND R1 R1 #$0FFF

2352 OR R1 R1 R0

2353 STR R1 $c; 1

2354 JMP END0;

2355OVERELSE2143780480 NOOP;

2356 JMP OVERELSE2056460308

2357EL182464342 CLFZ;

2358 LDR R0 #61440;

2359 LDR R1 $c

2360 AND R1 R1 #$0FFF

2361 OR R1 R1 R0

2362 STR R1 $c; 1

2363 JMP END0;

2364OVERELSE2056460308 NOOP;

2365OVERELSE2039816422 NOOP;

2366 JMP OVERELSE1672270034

2367EL423552186 CLFZ; tutu man man

2368 LDR R0 #61440;

2369 LDR R1 $c

2370 AND R1 R1 #$0FFF

2371 OR R1 R1 R0

2372 STR R1 $c; 1

2373 JMP END0;

2374OVERELSE1672270034 NOOP;

2375 JMP OVERELSE1577543526

2376EL1554059441 CLFZ;

2377 LDR R0 #61440;

2378 LDR R1 $c

2379 AND R1 R1 #$0FFF

2380 OR R1 R1 R0

2381 STR R1 $c; 1

2382 JMP END0;

2383OVERELSE1577543526 NOOP;

2384 JMP ENDS17481255230

2385ENDS17481255230 NOOP

2386 JMP ENDS16457823150

2387case1468176276 NOOP;

2388 LDR R0 #case1759276099 ;

2389 STR R0 $2f ;

2390 LDR R0 #case2092576288 ;

2391 STR R0 $30 ;

2392 LDR R0 #case2043769731 ;

2393 STR R0 $31 ;

2394 LDR R0 #46

2395 SWITCH R1 R0 ;

2396case1759276099 NOOP;

2397 LDR R0 #case1733882672 ;

2398 STR R0 $33 ;

2399 LDR R0 #50

2400 SWITCH R1 R0 ;

2401case1733882672 NOOP;

2402 LDR R0 #case1721570707 ;

2403 STR R0 $35 ;

2404 LDR R0 #52

2405 SWITCH R1 R0 ;

2406case1721570707 NOOP;

2407 LDR R0 #9218

2408 ADD R1 R6 #0

2409 STR R1 #0;

2410 SENDATA R0;

2411 SUB R4 #65535;

2412 SZ TOQSTART1808139209;

2413 ADD R4 R4 #1;

2414 JMP QS1808139209;

2415TOQSTART1808139209 LDR R4 #65528;

2416QS1808139209 CLFZ; data man man

2417 LDR R0 #TEN1791595007

2418 STR R0 $b

2419TEN1791595007 NOOP;

2420 LDR R0 $2

2421 SUB R0 #0

2422 SZ EL369032860

2423IF369032860 CLFZ; titi man man

2424 LDR R0 $2

2425 AND R0 R0 #$0002

2426 PRESENT R0 else1797750989

2427 LDR R0 #11010;

2428 ADD R1 R6 #0

2429 STR R1 #0;

2430 SENDATA R0;

2431 SUB R4 #65535;

2432 SZ TOQSTART1779283042;

2433 ADD R4 R4 #1;

2434 JMP QS1779283042;

2435TOQSTART1779283042 LDR R4 #65528;

2436QS1779283042 CLFZ;

2437 LDR R0 #TEN1785823773

2438 STR R0 $b

2439TEN1785823773 NOOP;

2440 LDR R0 $2

2441 SUB R0 #0

2442 SZ EL216633405

2443IF216633405 CLFZ

2444 ESL #39

2445 LDR R0 #1;

2446 STR R0 $2e;

2447 LDR R0 #4096;

2448 LDR R1 $c

2449 AND R1 R1 #$0FFF

2450 OR R1 R1 R0

2451 STR R1 $c; 1

2452 LDR R0 #$0;

2453 STR R0 $b;

2454 JMP END0;

2455 JMP OVERELSE749109148

2456EL216633405 CLFZ;

2457 LDR R0 #61440;

2458 LDR R1 $c

2459 AND R1 R1 #$0FFF

2460 OR R1 R1 R0

2461 STR R1 $c; 1

2462 JMP END0;

2463OVERELSE749109148 NOOP;

2464 JMP OVERELSE1676916892

2465else1797750989 NOOP

2466 LDR R0 #case1773511808 ;

2467 STR R0 $37 ;

2468 LDR R0 #case1895832638 ;

2469 STR R0 $38 ;

2470 LDR R0 #54

2471 SWITCH R1 R0 ;

2472case1773511808 NOOP;

2473 LDR R0 #case1840458117 ;

2474 STR R0 $3a ;

2475 LDR R0 #57

2476 SWITCH R1 R0 ;

2477case1840458117 NOOP;

2478 LDR R0 #9474

2479 ADD R1 R6 #0

2480 STR R1 #0;

2481 SENDATA R0;

2482 SUB R4 #65535;

2483 SZ TOQSTART1828146152;

2484 ADD R4 R4 #1;

2485 JMP QS1828146152;

2486TOQSTART1828146152 LDR R4 #65528;

2487QS1828146152 CLFZ; data man man

2488 LDR R0 #TEN1836225879

2489 STR R0 $b

2490TEN1836225879 NOOP;

2491 LDR R0 $2

2492 SUB R0 #0

2493 SZ EL1261257496

2494IF1261257496 CLFZ; titi man man

2495 LDR R0 $2

2496 AND R0 R0 #$0002

2497 PRESENT R0 else1817757932

2498 LDR R0 #9730;

2499 ADD R1 R6 #0

2500 STR R1 #0;

2501 SENDATA R0;

2502 SUB R4 #65535;

2503 SZ TOQSTART1823913914;

2504 ADD R4 R4 #1;

2505 JMP QS1823913914;

2506TOQSTART1823913914 LDR R4 #65528;

2507QS1823913914 CLFZ;

2508 LDR R0 #TEN1077516372

2509 STR R0 $b

2510TEN1077516372 NOOP;

2511 LDR R0 $2

2512 SUB R0 #0

2513 SZ EL1909190962

2514IF1909190962 CLFZ

2515 LDR R0 #1;

2516 STR R0 $36;

2517 LDR R0 #9986

2518 ADD R1 R6 #0

2519 STR R1 #0;

2520 SENDATA R0;

2521 SUB R4 #65535;

2522 SZ TOQSTART1083672355;

2523 ADD R4 R4 #1;

2524 JMP QS1083672355;

2525TOQSTART1083672355 LDR R4 #65528;

2526QS1083672355 CLFZ; data man man

2527 LDR R0 #TEN1064050161

2528 STR R0 $b

2529TEN1064050161 NOOP;

2530 LDR R0 $2

2531 SUB R0 #0

2532 SZ EL598521839

2533IF598521839 CLFZ; titi man man

2534 LDR R0 $2

2535 AND R0 R0 #$0002

2536 PRESENT R0 else1070206143

2537 LDR R0 #10242;

2538 ADD R1 R6 #0

2539 STR R1 #0;

2540 SENDATA R0;

2541 SUB R4 #65535;

2542 SZ TOQSTART1051738196;

2543 ADD R4 R4 #1;

2544 JMP QS1051738196;

2545TOQSTART1051738196 LDR R4 #65528;

2546QS1051738196 CLFZ;

2547 LDR R0 #TEN1058278927

2548 STR R0 $b

2549TEN1058278927 NOOP;

2550 LDR R0 $2

2551 SUB R0 #0

2552 SZ EL446122384

2553IF446122384 CLFZ

2554 ESL #39

2555 LDR R0 #1;

2556 STR R0 $2e;

2557 LDR R0 #4096;

2558 LDR R1 $c

2559 AND R1 R1 #$0FFF

2560 OR R1 R1 R0

2561 STR R1 $c; 1

2562 LDR R0 #$0;

2563 STR R0 $b;

2564 JMP END0;

2565 JMP OVERELSE249681226

2566EL446122384 CLFZ;

2567 LDR R0 #61440;

2568 LDR R1 $c

2569 AND R1 R1 #$0FFF

2570 OR R1 R1 R0

2571 STR R1 $c; 1

2572 JMP END0;

2573OVERELSE249681226 NOOP;

2574 JMP OVERELSE1906405871

2575else1070206143 NOOP

2576 LDR R0 #0;

2577 STR R0 $3b;

2578 LDR R0 #4096;

2579 LDR R1 $c

2580 AND R1 R1 #$0FFF

2581 OR R1 R1 R0

2582 STR R1 $c; 1

2583 LDR R0 #$0;

2584 STR R0 $b;

2585 JMP END0;

2586OVERELSE1906405871 NOOP;

2587 JMP OVERELSE2083198741

2588EL598521839 CLFZ; tutu man man

2589 LDR R0 #61440;

2590 LDR R1 $c

2591 AND R1 R1 #$0FFF

2592 OR R1 R1 R0

2593 STR R1 $c; 1

2594 JMP END0;

2595OVERELSE2083198741 NOOP;

2596 JMP OVERELSE1799962268

2597EL1909190962 CLFZ;

2598 LDR R0 #61440;

2599 LDR R1 $c

2600 AND R1 R1 #$0FFF

2601 OR R1 R1 R0

2602 STR R1 $c; 1

2603 JMP END0;

2604OVERELSE1799962268 NOOP;

2605 JMP OVERELSE1632572698

2606else1817757932 NOOP

2607 LDR R0 #4096;

2608 LDR R1 $c

2609 AND R1 R1 #$0FFF

2610 OR R1 R1 R0

2611 STR R1 $c; 1

2612 LDR R0 #$0;

2613 STR R0 $b;

2614 JMP END0;

2615OVERELSE1632572698 NOOP;

2616 JMP OVERELSE1652409113

2617EL1261257496 CLFZ; tutu man man

2618 LDR R0 #61440;

2619 LDR R1 $c

2620 AND R1 R1 #$0FFF

2621 OR R1 R1 R0

2622 STR R1 $c; 1

2623 JMP END0;

2624OVERELSE1652409113 NOOP;

2625 JMP ENDS16554010380

2626ENDS16554010380 NOOP

2627 JMP ENDS16738689850

2628case1895832638 NOOP;

2629 LDR R0 #case1134074461 ;

2630 STR R0 $3c ;

2631 LDR R0 #59

2632 SWITCH R1 R0 ;

2633case1134074461 NOOP;

2634 LDR R0 #10498

2635 ADD R1 R6 #0

2636 STR R1 #0;

2637 SENDATA R0;

2638 SUB R4 #65535;

2639 SZ TOQSTART1122147245;

2640 ADD R4 R4 #1;

2641 JMP QS1122147245;

2642TOQSTART1122147245 LDR R4 #65528;

2643QS1122147245 CLFZ; data man man

2644 LDR R0 #TEN1103679298

2645 STR R0 $b

2646TEN1103679298 NOOP;

2647 LDR R0 $2

2648 SUB R0 #0

2649 SZ EL645651125

2650IF645651125 CLFZ; titi man man

2651 LDR R0 $2

2652 AND R0 R0 #$0002

2653 PRESENT R0 else1108681033

2654 LDR R0 #10754;

2655 ADD R1 R6 #0

2656 STR R1 #0;

2657 SENDATA R0;

2658 SUB R4 #65535;

2659 SZ TOQSTART1090213086;

2660 ADD R4 R4 #1;

2661 JMP QS1090213086;

2662TOQSTART1090213086 LDR R4 #65528;

2663QS1090213086 CLFZ;

2664 LDR R0 #TEN1096369068

2665 STR R0 $b

2666TEN1096369068 NOOP;

2667 LDR R0 $2

2668 SUB R0 #0

2669 SZ EL1125021588

2670IF1125021588 CLFZ

2671 ESL #39

2672 LDR R0 #1;

2673 STR R0 $2e;

2674 LDR R0 #4096;

2675 LDR R1 $c

2676 AND R1 R1 #$0FFF

2677 OR R1 R1 R0

2678 STR R1 $c; 1

2679 LDR R0 #$0;

2680 STR R0 $b;

2681 JMP END0;

2682 JMP OVERELSE1173549172

2683EL1125021588 CLFZ;

2684 LDR R0 #61440;

2685 LDR R1 $c

2686 AND R1 R1 #$0FFF

2687 OR R1 R1 R0

2688 STR R1 $c; 1

2689 JMP END0;

2690OVERELSE1173549172 NOOP;

2691 JMP OVERELSE1490746474

2692else1108681033 NOOP

2693 LDR R0 #4096;

2694 LDR R1 $c

2695 AND R1 R1 #$0FFF

2696 OR R1 R1 R0

2697 STR R1 $c; 1

2698 LDR R0 #$0;

2699 STR R0 $b;

2700 JMP END0;

2701OVERELSE1490746474 NOOP;

2702 JMP OVERELSE222885953

2703EL645651125 CLFZ; tutu man man

2704 LDR R0 #61440;

2705 LDR R1 $c

2706 AND R1 R1 #$0FFF

2707 OR R1 R1 R0

2708 STR R1 $c; 1

2709 JMP END0;

2710OVERELSE222885953 NOOP;

2711 JMP ENDS16677130020

2712ENDS16677130020 NOOP

2713 JMP ENDS16738689851

2714ENDS16738689850 NOOP

2715ENDS16738689851 NOOP

2716OVERELSE1676916892 NOOP;

2717 JMP OVERELSE1212978182

2718EL369032860 CLFZ; tutu man man

2719 LDR R0 #61440;

2720 LDR R1 $c

2721 AND R1 R1 #$0FFF

2722 OR R1 R1 R0

2723 STR R1 $c; 1

2724 JMP END0;

2725OVERELSE1212978182 NOOP;

2726 JMP ENDS16838724560

2727ENDS16838724560 NOOP

2728 JMP ENDS17023404030

2729ENDS17023404030 NOOP

2730 JMP ENDS17419695400

2731case2092576288 NOOP;

2732 ESL #39

2733 LDR R0 #1;

2734 STR R0 $2e;

2735 ESL #40

2736 LDR R0 #2;

2737 STR R0 $2e;

2738 LDR R0 #4096;

2739 LDR R1 $c

2740 AND R1 R1 #$0FFF

2741 OR R1 R1 R0

2742 STR R1 $c; 1

2743 LDR R0 #$0;

2744 STR R0 $b;

2745 JMP END0;

2746 JMP ENDS17419695401

2747case2043769731 NOOP;

2748 ESL #40

2749 LDR R0 #2;

2750 STR R0 $2e;

2751 LDR R0 #11266;

2752 ADD R1 R6 #0

2753 STR R1 #0;

2754 SENDATA R0;

2755 SUB R4 #65535;

2756 SZ TOQSTART1160237386;

2757 ADD R4 R4 #1;

2758 JMP QS1160237386;

2759TOQSTART1160237386 LDR R4 #65528;

2760QS1160237386 CLFZ;

2761 LDR R0 #TEN1166393368

2762 STR R0 $b

2763TEN1166393368 NOOP;

2764 LDR R0 $2

2765 SUB R0 #0

2766 SZ EL1280226268

2767IF1280226268 CLFZ

2768 ESL #28

2769 LDR R0 #0;

2770 STR R0 $27;

2771 LDR R0 #11522

2772 ADD R1 R6 #0

2773 STR R1 #0;

2774 SENDATA R0;

2775 SUB R4 #65535;

2776 SZ TOQSTART1148310170;

2777 ADD R4 R4 #1;

2778 JMP QS1148310170;

2779TOQSTART1148310170 LDR R4 #65528;

2780QS1148310170 CLFZ; data man man

2781 LDR R0 #TEN1154466152

2782 STR R0 $b

2783TEN1154466152 NOOP;

2784 LDR R0 $2

2785 SUB R0 #0

2786 SZ EL1554457542

2787IF1554457542 CLFZ; titi man man

2788 LDR R0 $2

2789 AND R0 R0 #$0002

2790 PRESENT R0 else1135998205

2791 LDR R0 #0;

2792 STR R0 $29;

2793 ESL #29

2794 LDR R0 #8192;

2795 LDR R1 $c

2796 AND R1 R1 #$0FFF

2797 OR R1 R1 R0

2798 STR R1 $c; 1

2799 LDR R0 #1;

2800 STR R0 $24;

2801 LDR R0 #7170;

2802 ADD R1 R6 #0

2803 STR R1 #0;

2804 SENDATA R0;

2805 SUB R4 #65535;

2806 SZ TOQSTART1140999941;

2807 ADD R4 R4 #1;

2808 JMP QS1140999941;

2809TOQSTART1140999941 LDR R4 #65528;

2810QS1140999941 CLFZ;

2811 LDR R0 #TEN1221027712

2812 STR R0 $b

2813TEN1221027712 NOOP;

2814 LDR R0 $2

2815 SUB R0 #0

2816 SZ EL410173729

2817IF410173729 CLFZ

2818 ESL #46

2819 LDR R0 #0;

2820 STR R0 $3d;

2821 LDR R0 #4096;

2822 LDR R1 $c

2823 AND R1 R1 #$0FFF

2824 OR R1 R1 R0

2825 STR R1 $c; 1

2826 LDR R0 #$0;

2827 STR R0 $b;

2828 JMP END0;

2829 JMP OVERELSE1463955989

2830EL410173729 CLFZ;

2831 LDR R0 #61440;

2832 LDR R1 $c

2833 AND R1 R1 #$0FFF

2834 OR R1 R1 R0

2835 STR R1 $c; 1

2836 JMP END0;

2837OVERELSE1463955989 NOOP;

2838 JMP OVERELSE1925772744

2839else1135998205 NOOP

2840 LDR R0 #1;

2841 STR R0 $29;

2842 LDR R0 #11778;

2843 ADD R1 R6 #0

2844 STR R1 #0;

2845 SENDATA R0;

2846 SUB R4 #65535;

2847 SZ TOQSTART1209100496;

2848 ADD R4 R4 #1;

2849 JMP QS1209100496;

2850TOQSTART1209100496 LDR R4 #65528;

2851QS1209100496 CLFZ;

2852 LDR R0 #TEN1215256479

2853 STR R0 $b

2854TEN1215256479 NOOP;

2855 LDR R0 $2

2856 SUB R0 #0

2857 SZ EL562573185

2858IF562573185 CLFZ

2859 LDR R0 #0;

2860 STR R0 $2e;

2861 LDR R0 #0;

2862 STR R0 $32;

2863 LDR R0 #12034

2864 ADD R1 R6 #0

2865 STR R1 #0;

2866 SENDATA R0;

2867 SUB R4 #65535;

2868 SZ TOQSTART1198712276;

2869 ADD R4 R4 #1;

2870 JMP QS1198712276;

2871TOQSTART1198712276 LDR R4 #65528;

2872QS1198712276 CLFZ; data man man

2873 LDR R0 #TEN1204868258

2874 STR R0 $b

2875TEN1204868258 NOOP;

2876 LDR R0 $2

2877 SUB R0 #0

2878 SZ EL1695885664

2879IF1695885664 CLFZ; titi man man

2880 LDR R0 $2

2881 AND R0 R0 #$0002

2882 PRESENT R0 else1186400311

2883 LDR R0 #13314;

2884 ADD R1 R6 #0

2885 STR R1 #0;

2886 SENDATA R0;

2887 SUB R4 #65535;

2888 SZ TOQSTART1192941042;

2889 ADD R4 R4 #1;

2890 JMP QS1192941042;

2891TOQSTART1192941042 LDR R4 #65528;

2892QS1192941042 CLFZ;

2893 LDR R0 #TEN878985940

2894 STR R0 $b

2895TEN878985940 NOOP;

2896 LDR R0 $2

2897 SUB R0 #0

2898 SZ EL1138778013

2899IF1138778013 CLFZ

2900 ESL #39

2901 LDR R0 #1;

2902 STR R0 $2e;

2903 LDR R0 #4096;

2904 LDR R1 $c

2905 AND R1 R1 #$0FFF

2906 OR R1 R1 R0

2907 STR R1 $c; 1

2908 LDR R0 #$0;

2909 STR R0 $b;

2910 JMP END0;

2911 JMP OVERELSE282703792

2912EL1138778013 CLFZ;

2913 LDR R0 #61440;

2914 LDR R1 $c

2915 AND R1 R1 #$0FFF

2916 OR R1 R1 R0

2917 STR R1 $c; 1

2918 JMP END0;

2919OVERELSE282703792 NOOP;

2920 JMP OVERELSE1324570462

2921else1186400311 NOOP

2922 LDR R0 #0;

2923 STR R0 $34;

2924 LDR R0 #0;

2925 STR R0 $36;

2926 LDR R0 #12290

2927 ADD R1 R6 #0

2928 STR R1 #0;

2929 SENDATA R0;

2930 SUB R4 #65535;

2931 SZ TOQSTART865519728;

2932 ADD R4 R4 #1;

2933 JMP QS865519728;

2934TOQSTART865519728 LDR R4 #65528;

2935QS865519728 CLFZ; data man man

2936 LDR R0 #TEN871675711

2937 STR R0 $b

2938TEN871675711 NOOP;

2939 LDR R0 $2

2940 SUB R0 #0

2941 SZ EL1618148476

2942IF1618148476 CLFZ; titi man man

2943 LDR R0 $2

2944 AND R0 R0 #$0002

2945 PRESENT R0 else853592513

2946 LDR R0 #12546;

2947 ADD R1 R6 #0

2948 STR R1 #0;

2949 SENDATA R0;

2950 SUB R4 #65535;

2951 SZ TOQSTART859748495;

2952 ADD R4 R4 #1;

2953 JMP QS859748495;

2954TOQSTART859748495 LDR R4 #65528;

2955QS859748495 CLFZ;

2956 LDR R0 #TEN841280548

2957 STR R0 $b

2958TEN841280548 NOOP;

2959 LDR R0 $2

2960 SUB R0 #0

2961 SZ EL1587850535

2962IF1587850535 CLFZ

2963 LDR R0 #1;

2964 STR R0 $36;

2965 LDR R0 #12802

2966 ADD R1 R6 #0

2967 STR R1 #0;

2968 SENDATA R0;

2969 SUB R4 #65535;

2970 SZ TOQSTART849360275;

2971 ADD R4 R4 #1;

2972 JMP QS849360275;

2973TOQSTART849360275 LDR R4 #65528;

2974QS849360275 CLFZ; data man man

2975 LDR R0 #TEN929388046

2976 STR R0 $b

2977TEN929388046 NOOP;

2978 LDR R0 $2

2979 SUB R0 #0

2980 SZ EL94153923

2981IF94153923 CLFZ; titi man man

2982 LDR R0 $2

2983 AND R0 R0 #$0002

2984 PRESENT R0 else935928777

2985 LDR R0 #13058;

2986 ADD R1 R6 #0

2987 STR R1 #0;

2988 SENDATA R0;

2989 SUB R4 #65535;

2990 SZ TOQSTART917460830;

2991 ADD R4 R4 #1;

2992 JMP QS917460830;

2993TOQSTART917460830 LDR R4 #65528;

2994QS917460830 CLFZ;

2995 LDR R0 #TEN923616812

2996 STR R0 $b

2997TEN923616812 NOOP;

2998 LDR R0 $2

2999 SUB R0 #0

3000 SZ EL246553378

3001IF246553378 CLFZ

3002 ESL #39

3003 LDR R0 #1;

3004 STR R0 $2e;

3005 LDR R0 #4096;

3006 LDR R1 $c

3007 AND R1 R1 #$0FFF

3008 OR R1 R1 R0

3009 STR R1 $c; 1

3010 LDR R0 #$0;

3011 STR R0 $b;

3012 JMP END0;

3013 JMP OVERELSE1712283793

3014EL246553378 CLFZ;

3015 LDR R0 #61440;

3016 LDR R1 $c

3017 AND R1 R1 #$0FFF

3018 OR R1 R1 R0

3019 STR R1 $c; 1

3020 JMP END0;

3021OVERELSE1712283793 NOOP;

3022 JMP OVERELSE1925752611

3023else935928777 NOOP

3024 LDR R0 #0;

3025 STR R0 $3b;

3026 LDR R0 #4096;

3027 LDR R1 $c

3028 AND R1 R1 #$0FFF

3029 OR R1 R1 R0

3030 STR R1 $c; 1

3031 LDR R0 #$0;

3032 STR R0 $b;

3033 JMP END0;

3034OVERELSE1925752611 NOOP;

3035 JMP OVERELSE620596174

3036EL94153923 CLFZ; tutu man man

3037 LDR R0 #61440;

3038 LDR R1 $c

3039 AND R1 R1 #$0FFF

3040 OR R1 R1 R0

3041 STR R1 $c; 1

3042 JMP END0;

3043OVERELSE620596174 NOOP;

3044 JMP OVERELSE1369351791

3045EL1587850535 CLFZ;

3046 LDR R0 #61440;

3047 LDR R1 $c

3048 AND R1 R1 #$0FFF

3049 OR R1 R1 R0

3050 STR R1 $c; 1

3051 JMP END0;

3052OVERELSE1369351791 NOOP;

3053 JMP OVERELSE91348699

3054else853592513 NOOP

3055 LDR R0 #0;

3056 STR R0 $39;

3057 LDR R0 #4096;

3058 LDR R1 $c

3059 AND R1 R1 #$0FFF

3060 OR R1 R1 R0

3061 STR R1 $c; 1

3062 LDR R0 #$0;

3063 STR R0 $b;

3064 JMP END0;

3065OVERELSE91348699 NOOP;

3066 JMP OVERELSE1233367011

3067EL1618148476 CLFZ; tutu man man

3068 LDR R0 #61440;

3069 LDR R1 $c

3070 AND R1 R1 #$0FFF

3071 OR R1 R1 R0

3072 STR R1 $c; 1

3073 JMP END0;

3074OVERELSE1233367011 NOOP;

3075OVERELSE1324570462 NOOP;

3076 JMP OVERELSE1983065681

3077EL1695885664 CLFZ; tutu man man

3078 LDR R0 #61440;

3079 LDR R1 $c

3080 AND R1 R1 #$0FFF

3081 OR R1 R1 R0

3082 STR R1 $c; 1

3083 JMP END0;

3084OVERELSE1983065681 NOOP;

3085 JMP OVERELSE498131341

3086EL562573185 CLFZ;

3087 LDR R0 #61440;

3088 LDR R1 $c

3089 AND R1 R1 #$0FFF

3090 OR R1 R1 R0

3091 STR R1 $c; 1

3092 JMP END0;

3093OVERELSE498131341 NOOP;

3094OVERELSE1925772744 NOOP;

3095 JMP OVERELSE1408601650

3096EL1554457542 CLFZ; tutu man man

3097 LDR R0 #61440;

3098 LDR R1 $c

3099 AND R1 R1 #$0FFF

3100 OR R1 R1 R0

3101 STR R1 $c; 1

3102 JMP END0;

3103OVERELSE1408601650 NOOP;

3104 JMP OVERELSE1787385372

3105EL1280226268 CLFZ;

3106 LDR R0 #61440;

3107 LDR R1 $c

3108 AND R1 R1 #$0FFF

3109 OR R1 R1 R0

3110 STR R1 $c; 1

3111 JMP END0;

3112OVERELSE1787385372 NOOP;

3113 JMP ENDS17419695402

3114ENDS17419695400 NOOP

3115ENDS17419695401 NOOP

3116ENDS17419695402 NOOP

3117 JMP ENDS16457823151

3118ENDS16457823150 NOOP

3119ENDS16457823151 NOOP

3120 JMP ENDS17285033290

3121ENDS17285033290 NOOP

3122 JMP ENDS11144440840

3123case1509715246 NOOP;

3124 LDR R0 #case898223385 ;

3125 STR R0 $3e ;

3126 LDR R0 #61

3127 SWITCH R1 R0 ;

3128case898223385 NOOP;

3129 ESL #46

3130 LDR R0 #0;

3131 STR R0 $3d;

3132 ESL #18

3133 LDR R0 #0;

3134 STR R0 $10;

3135 LDR R0 #0;

3136 LDR R1 $c

3137 AND R1 R1 #$0FFF

3138 OR R1 R1 R0

3139 STR R1 $c; 1

3140 LDR R0 #0;

3141 STR R0 $10;

3142 LDR R0 #$0;

3143 STR R0 $b;

3144 JMP END0;

3145 JMP ENDS12010125870

3146ENDS12010125870 NOOP

3147 JMP ENDS11144440841

3148ENDS11144440840 NOOP

3149ENDS11144440841 NOOP

3150 JMP ENDS17165761132

3151case634345685 NOOP;

3152 ESL #46

3153 LDR R0 #3;

3154 STR R0 $14;

3155 ESL #18

3156 LDR R0 #0;

3157 STR R0 $10;

3158 LDR R0 #0;

3159 LDR R1 $c

3160 AND R1 R1 #$0FFF

3161 OR R1 R1 R0

3162 STR R1 $c; 1

3163 LDR R0 #0;

3164 STR R0 $10;

3165 LDR R0 #$0;

3166 STR R0 $b;

3167 JMP END0;

3168 JMP ENDS17165761133

3169ENDS17165761130 NOOP

3170ENDS17165761131 NOOP

3171ENDS17165761132 NOOP

3172ENDS17165761133 NOOP

3173 JMP ENDS5988805582

3174ENDS5988805580 NOOP

3175ENDS5988805581 NOOP

3176ENDS5988805582 NOOP

3177END0 LDR R0 $ee;

3178 SUB R0 #$0;

3179 SZ BEGIN1;

3180 JMP RUN1;

3181BEGIN1 NOOP; loading the num which have to be init

3182 INIT #0;

3183 LDR R7 #1;

3184 LDR R8 #0;previous clock-domain num

3185 LDR R6 #3

3186;Setting the declared signals, signal locks, data-locks, and pc's to 0

3187 LDR R0 #0 ;blocked it cool

3188 STR R0 $e7 ;loading the values

3189 LDR R0 #0 ;blocked it cool

3190 STR R0 $e8 ;loading the values

3191 LDR R0 #0 ;blocked it cool

3192 STR R0 $e9 ;loading the values

3193 LDR R0 #0 ;blocked it cool

3194 STR R0 $ea ;loading the values

3195 LDR R0 #0 ;blocked it cool

3196 STR R0 $eb ;loading the values

3197 LDR R0 #0 ;blocked it cool

3198 STR R0 $ec ;loading the values

3199 LDR R0 #0 ;blocked it cool

3200 STR R0 $ed ;loading the values

3201 LDR R0 #0 ;blocked it cool

3202 STR R0 $ee ;loading the values

3203 LDR R0 #0 ;blocked it cool

3204 STR R0 $ef ;loading the values

3205 LDR R0 #0 ;blocked it cool

3206 STR R0 $f0 ;loading the values

3207 LDR R0 #0 ;blocked it cool

3208 STR R0 $f1 ;loading the values

3209 LDR R0 #0 ;blocked it cool

3210 STR R0 $f2 ;loading the values

3211 LDR R0 #0 ;blocked it cool

3212 STR R0 $f3 ;loading the values

3213 LDR R0 #0 ;blocked it cool

3214 STR R0 $f4 ;loading the values

3215 LDR R0 #0 ;blocked it cool

3216 STR R0 $f5 ;loading the values

3217 LDR R0 #0 ;blocked it cool

3218 STR R0 $f6 ;loading the values

3219SEOT1310866727731 CLFZ;

3220 LDR R0 #0;clearing

3221 LDR R1 #0;clearing

3222 LDR R11 #0;clearing

3223LERR1310866727731 LER R0;loading the EREADY bit which is set from ENV

3224 PRESENT R0 FER1310866727731;

3225 JMP LERR1310866727731;

3226FER1310866727731 SEOT; This is basically the SEOT tick

3227 CER;clear the EREADY bit

3228 LDR R0 $0001; loading the output signals

3229 AND R0 R0 #$0;

3230 SSOP R0;throwing output signals to env

3231 LSIP R0;getting input signals from SIP

3232 AND R0 R0 #$0;

3233 LDR R1 $0000;

3234 AND R1 R1 #$ffff;

3235 OR R0 R0 R1;

3236 STR R0 $0000;storing SIP signals in mem

3237 LDR R0 #13571

3238 ADD R1 R6 #0

3239 STR R1 #0;

3240 SENDATA R0;

3241 ADD R4 R4 #1;

3242LOAD01310866727731 LDR R0 R1;

3243 CLFZ

3244 SUBV R0 R0 #0;

3245 SZ EL01310866727731;

3246 JMP OEL01310866727731;

3247EL01310866727731 CLFZ;

3248 JMP LOAD01310866727731;

3249OEL01310866727731 NOOP;

3250 INIT #0;

3251 CEOT;now start processing

3252 LDR R14 #34; the ESL line numbers start-offset

3253RUN1 NOOP; the locks need to be inside the memory since if they are here then I am just eating up logic

3254 LDR R7 #1;

3255 LDR R8 #0;previous clock-domain num

3256 LDR R6 #3

3257 ESL #18

3258 LDR R0 $ee;

3259 SUB R0 #0 ;checking if the value here is actually zero

3260 SZ CONT955935720 ;Just jump to the value pointed to by this register

3261 JMP R0

3262CONT955935720 NOOP

3263 LDR R0 #case962091702 ;

3264 STR R0 $12a ;

3265 LDR R0 #case662212774 ;

3266 STR R0 $12b ;

3267 LDR R0 #case1907964102 ;

3268 STR R0 $12c ;

3269 LDR R0 #297

3270 SWITCH R1 R0 ;

3271case962091702 NOOP;

3272 LDR R0 #0;

3273 STR R0 $129;

3274 LDR R0 #$0;

3275 STR R0 $ee;

3276 JMP END1;

3277 JMP ENDS13822221770

3278case662212774 NOOP;

3279 LDR R0 #2;

3280 STR R0 $129;

3281 ESL #52

3282 ESL #52

3283 ESL #53

3284 ESL #53

3285 ESL #53

3286 ESL #54

3287 LDR R0 #13827;

3288 ADD R1 R6 #0

3289 STR R1 #0;

3290 SENDATA R0;

3291 SUB R4 #65535;

3292 SZ TOQSTART1016726046;

3293 ADD R4 R4 #1;

3294 JMP QS1016726046;

3295TOQSTART1016726046 LDR R4 #65528;

3296QS1016726046 CLFZ;

3297 LDR R0 #TEN1022882029

3298 STR R0 $ee

3299TEN1022882029 NOOP;

3300 LDR R0 $3

3301 SUB R0 #0

3302 SZ EL1515723796

3303IF1515723796 CLFZ

3304 ESL #56

3305 LDR R0 #0;

3306 STR R0 $12d;

3307PNODE1018649791 NOOP

3308 LDR R0 #PNODE1018649791; loading the case address into register

3309 STR R0 $ee; storing the address in the memory

3310 LDR R14 #34;

3311 LDR R0 $ef; loading the address from the mem

3312 SUB R0 #0; checking if this is actually set

3313 SZ CONT1000181843

3314 JMP R0

3315CONT1000181843 LDR R0 #1;

3316 STR R0 $12f;

3317 ESL #57

3318 LDR R0 #0;

3319 STR R0 $132;

3320 LDR R0 #0;

3321 STR R0 $134;

3322 LDR R0 #0;

3323 STR R0 $137;

3324 LDR R0 #14084

3325 ADD R1 R6 #1

3326 STR R1 #0;

3327 SENDATA R0;

3328 SUB R4 #65535;

3329 SZ TOQSTART1006337826;

3330 ADD R4 R4 #1;

3331 JMP QS1006337826;

3332TOQSTART1006337826 LDR R4 #65528;

3333QS1006337826 CLFZ; data man man

3334 LDR R0 #TEN988254628

3335 STR R0 $ef

3336TEN988254628 NOOP;

3337 LDR R0 $4

3338 SUB R0 #0

3339 SZ EL601327063

3340IF601327063 CLFZ; titi man man

3341 LDR R0 $4

3342 AND R0 R0 #$0002

3343 PRESENT R0 else994410610

3344 LDR R0 #15108;

3345 ADD R1 R6 #1

3346 STR R1 #0;

3347 SENDATA R0;

3348 SUB R4 #65535;

3349 SZ TOQSTART679301261;

3350 ADD R4 R4 #1;

3351 JMP QS679301261;

3352TOQSTART679301261 LDR R4 #65528;

3353QS679301261 CLFZ;

3354 LDR R0 #TEN685457243

3355 STR R0 $ef

3356TEN685457243 NOOP;

3357 LDR R0 $4

3358 SUB R0 #0

3359 SZ EL1768032535

3360IF1768032535 CLFZ

3361 ESL #61

3362 LDR R0 #15364

3363 ADD R1 R6 #1

3364 STR R1 #0;

3365 SENDATA R0;

3366 SUB R4 #65535;

3367 SZ TOQSTART666989296;

3368 ADD R4 R4 #1;

3369 JMP QS666989296;

3370TOQSTART666989296 LDR R4 #65528;

3371QS666989296 CLFZ; data man man

3372 LDR R0 #TEN673530027

3373 STR R0 $ef

3374TEN673530027 NOOP;

3375 LDR R0 $4

3376 SUB R0 #0

3377 SZ EL307749048

3378IF307749048 CLFZ; titi man man

3379 LDR R0 $4

3380 AND R0 R0 #$0002

3381 PRESENT R0 else655062080

3382 ESL #62

3383 LDR R0 $e7 ; loading from mem

3384 OR R0 R0 #8192 ;loading the emit signal in

3385 STR R0 $e7; emitted signal res1\_\_1418003825 in mem

3386 ESL #63

3387 LDR R0 $e7 ; loading from mem

3388 OR R0 R0 #4096 ;loading the emit signal in

3389 STR R0 $e7; emitted signal res2\_\_1331820071 in mem

3390 ESL #64

3391 LDR R0 $e7 ; loading from mem

3392 OR R0 R0 #2048 ;loading the emit signal in

3393 STR R0 $e7; emitted signal res3\_\_1343747287 in mem

3394 ESL #66

3395 LDR R0 #1;

3396 STR R0 $134;

3397 LDR R0 #TI661218062; loaded the case label into the regiser

3398 STR R0 $ef; stored in memory

3399TI661218062 NOOP

3400 LDR R0 #256;

3401 LDR R1 $f5

3402 AND R1 R1 #$F0FF

3403 OR R1 R1 R0

3404 STR R1 $f5; 2

3405 JMP OVERELSE679064250

3406else655062080 NOOP

3407 ESL #66

3408 LDR R0 #1;

3409 STR R0 $134;

3410 LDR R0 #TI644673860; loaded the case label into the regiser

3411 STR R0 $ef; stored in memory

3412TI644673860 NOOP

3413 LDR R0 #256;

3414 LDR R1 $f5

3415 AND R1 R1 #$F0FF

3416 OR R1 R1 R0

3417 STR R1 $f5; 2

3418OVERELSE679064250 NOOP;

3419 JMP OVERELSE1707121862

3420EL307749048 CLFZ; tutu man man

3421 LDR R0 #3840;

3422 LDR R1 $f5

3423 AND R1 R1 #$F0FF

3424 OR R1 R1 R0

3425 STR R1 $f5; 2

3426OVERELSE1707121862 NOOP;

3427 JMP OVERELSE1488865159

3428EL1768032535 CLFZ;

3429 LDR R0 #3840;

3430 LDR R1 $f5

3431 AND R1 R1 #$F0FF

3432 OR R1 R1 R0

3433 STR R1 $f5; 2

3434OVERELSE1488865159 NOOP;

3435 JMP OVERELSE1909211095

3436else994410610 NOOP

3437 LDR R0 #0;

3438 STR R0 $139;

3439 LDR R0 #0;

3440 STR R0 $13b;

3441 LDR R0 #14340

3442 ADD R1 R6 #1

3443 STR R1 #0;

3444 SENDATA R0;

3445 SUB R4 #65535;

3446 SZ TOQSTART650829842;

3447 ADD R4 R4 #1;

3448 JMP QS650829842;

3449TOQSTART650829842 LDR R4 #65528;

3450QS650829842 CLFZ; data man man

3451 LDR R0 #TEN731242362

3452 STR R0 $ef

3453TEN731242362 NOOP;

3454 LDR R0 $4

3455 SUB R0 #0

3456 SZ EL1831743602

3457IF1831743602 CLFZ; titi man man

3458 LDR R0 $4

3459 AND R0 R0 #$0002

3460 PRESENT R0 else737398345

3461 LDR R0 #14596;

3462 ADD R1 R6 #1

3463 STR R1 #0;

3464 SENDATA R0;

3465 SUB R4 #65535;

3466 SZ TOQSTART718930397;

3467 ADD R4 R4 #1;

3468 JMP QS718930397;

3469TOQSTART718930397 LDR R4 #65528;

3470QS718930397 CLFZ;

3471 LDR R0 #TEN723932133

3472 STR R0 $ef

3473TEN723932133 NOOP;

3474 LDR R0 $4

3475 SUB R0 #0

3476 SZ EL1352373139

3477IF1352373139 CLFZ

3478 ESL #61

3479 LDR R0 #14852

3480 ADD R1 R6 #1

3481 STR R1 #0;

3482 SENDATA R0;

3483 SUB R4 #65535;

3484 SZ TOQSTART705464186;

3485 ADD R4 R4 #1;

3486 JMP QS705464186;

3487TOQSTART705464186 LDR R4 #65528;

3488QS705464186 CLFZ; data man man

3489 LDR R0 #TEN711620168

3490 STR R0 $ef

3491TEN711620168 NOOP;

3492 LDR R0 $4

3493 SUB R0 #0

3494 SZ EL1263394924

3495IF1263394924 CLFZ; titi man man

3496 LDR R0 $4

3497 AND R0 R0 #$0002

3498 PRESENT R0 else693536970

3499 ESL #62

3500 LDR R0 $e7 ; loading from mem

3501 OR R0 R0 #8192 ;loading the emit signal in

3502 STR R0 $e7; emitted signal res1\_\_1418003825 in mem

3503 ESL #63

3504 LDR R0 $e7 ; loading from mem

3505 OR R0 R0 #4096 ;loading the emit signal in

3506 STR R0 $e7; emitted signal res2\_\_1331820071 in mem

3507 ESL #64

3508 LDR R0 $e7 ; loading from mem

3509 OR R0 R0 #2048 ;loading the emit signal in

3510 STR R0 $e7; emitted signal res3\_\_1343747287 in mem

3511 ESL #66

3512 LDR R0 #1;

3513 STR R0 $134;

3514 LDR R0 #TI699692952; loaded the case label into the regiser

3515 STR R0 $ef; stored in memory

3516TI699692952 NOOP

3517 LDR R0 #256;

3518 LDR R1 $f5

3519 AND R1 R1 #$F0FF

3520 OR R1 R1 R0

3521 STR R1 $f5; 2

3522 JMP OVERELSE263404854

3523else693536970 NOOP

3524 ESL #66

3525 LDR R0 #1;

3526 STR R0 $134;

3527 LDR R0 #TI781644468; loaded the case label into the regiser

3528 STR R0 $ef; stored in memory

3529TI781644468 NOOP

3530 LDR R0 #256;

3531 LDR R1 $f5

3532 AND R1 R1 #$F0FF

3533 OR R1 R1 R0

3534 STR R1 $f5; 2

3535OVERELSE263404854 NOOP;

3536 JMP OVERELSE783253916

3537EL1263394924 CLFZ; tutu man man

3538 LDR R0 #3840;

3539 LDR R1 $f5

3540 AND R1 R1 #$F0FF

3541 OR R1 R1 R0

3542 STR R1 $f5; 2

3543OVERELSE783253916 NOOP;

3544 JMP OVERELSE1684545468

3545EL1352373139 CLFZ;

3546 LDR R0 #3840;

3547 LDR R1 $f5

3548 AND R1 R1 #$F0FF

3549 OR R1 R1 R0

3550 STR R1 $f5; 2

3551OVERELSE1684545468 NOOP;

3552 JMP OVERELSE1155339662

3553else737398345 NOOP

3554 LDR R0 #0;

3555 STR R0 $13d;

3556 LDR R0 #TI787800451; loaded the case label into the regiser

3557 STR R0 $ef; stored in memory

3558TI787800451 NOOP

3559 LDR R0 #256;

3560 LDR R1 $f5

3561 AND R1 R1 #$F0FF

3562 OR R1 R1 R0

3563 STR R1 $f5; 2

3564OVERELSE1155339662 NOOP;

3565 JMP OVERELSE733882249

3566EL1831743602 CLFZ; tutu man man

3567 LDR R0 #3840;

3568 LDR R1 $f5

3569 AND R1 R1 #$F0FF

3570 OR R1 R1 R0

3571 STR R1 $f5; 2

3572OVERELSE733882249 NOOP;

3573OVERELSE1909211095 NOOP;

3574 JMP OVERELSE364154974

3575EL601327063 CLFZ; tutu man man

3576 LDR R0 #3840;

3577 LDR R1 $f5

3578 AND R1 R1 #$F0FF

3579 OR R1 R1 R0

3580 STR R1 $f5; 2

3581OVERELSE364154974 NOOP;

3582 LDR R14 #35;

3583 LDR R0 $f0; loading the address from the mem

3584 SUB R0 #0; checking if this is actually set

3585 SZ CONT787800451

3586 JMP R0

3587CONT787800451 LDR R0 #1;

3588 STR R0 $13f;

3589 LDR R0 #15621;

3590 ADD R1 R6 #2

3591 STR R1 #0;

3592 SENDATA R0;

3593 SUB R4 #65535;

3594 SZ TOQSTART769332503;

3595 ADD R4 R4 #1;

3596 JMP QS769332503;

3597TOQSTART769332503 LDR R4 #65528;

3598QS769332503 CLFZ;

3599 LDR R0 #TEN775873235

3600 STR R0 $f0

3601TEN775873235 NOOP;

3602 LDR R0 $5

3603 SUB R0 #0

3604 SZ EL1570999058

3605IF1570999058 CLFZ

3606 ESL #74

3607 LDR R0 #0;

3608 STR R0 $142;

3609 ESL #75

3610 LDR R0 #0;

3611 STR R0 $144;

3612 LDR R0 #0;

3613 STR R0 $147;

3614 LDR R0 #15877;

3615 ADD R1 R6 #2

3616 STR R1 #0;

3617 SENDATA R0;

3618 SUB R4 #65535;

3619 SZ TOQSTART757405287;

3620 ADD R4 R4 #1;

3621 JMP QS757405287;

3622TOQSTART757405287 LDR R4 #65528;

3623QS757405287 CLFZ;

3624 LDR R0 #TEN763561270

3625 STR R0 $f0

3626TEN763561270 NOOP;

3627 LDR R0 $5

3628 SUB R0 #0

3629 SZ EL108200175

3630IF108200175 CLFZ

3631 ESL #77

3632 LDR R0 #0;

3633 STR R0 $149;

3634 LDR R0 #0;

3635 STR R0 $14b;

3636 LDR R0 #16133;

3637 ADD R1 R6 #2

3638 STR R1 #0;

3639 SENDATA R0;

3640 SUB R4 #65535;

3641 SZ TOQSTART743939076;

3642 ADD R4 R4 #1;

3643 JMP QS743939076;

3644TOQSTART743939076 LDR R4 #65528;

3645QS743939076 CLFZ;

3646 LDR R0 #TEN750095058

3647 STR R0 $f0

3648TEN750095058 NOOP;

3649 LDR R0 $5

3650 SUB R0 #0

3651 SZ EL1679054320

3652IF1679054320 CLFZ

3653 ESL #79

3654 LDR R0 #0;

3655 STR R0 $14d;

3656 LDR R0 #16389

3657 ADD R1 R6 #2

3658 STR R1 #0;

3659 SENDATA R0;

3660 SUB R4 #65535;

3661 SZ TOQSTART830507579;

3662 ADD R4 R4 #1;

3663 JMP QS830507579;

3664TOQSTART830507579 LDR R4 #65528;

3665QS830507579 CLFZ; data man man

3666 LDR R0 #TEN836663561

3667 STR R0 $f0

3668TEN836663561 NOOP;

3669 LDR R0 $5

3670 SUB R0 #0

3671 SZ EL606937511

3672IF606937511 CLFZ; titi man man

3673 LDR R0 $5

3674 AND R0 R0 #$0002

3675 PRESENT R0 else818195614

3676 ESL #80

3677 LDR R0 #32;

3678 LDR R1 $f5

3679 AND R1 R1 #$FF0F

3680 OR R1 R1 R0

3681 STR R1 $f5; 5

3682 ESL #89

3683 LDR R0 $e7 ; loading from mem

3684 OR R0 R0 #32768 ;loading the emit signal in

3685 STR R0 $e7; emitted signal packet\_\_1392610397 in mem

3686 ESL #89

3687 LDR R0 $e7 ; loading from mem

3688 OR R0 R0 #32768 ;loading the emit signal in

3689 STR R0 $e7; emitted signal packet\_\_1392610397 in mem

3690 ESL #91

3691 LDR R0 #1;

3692 STR R0 $144;

3693 LDR R0 #TI826275341; loaded the case label into the regiser

3694 STR R0 $f0; stored in memory

3695TI826275341 NOOP

3696 LDR R0 #16;

3697 LDR R1 $f5

3698 AND R1 R1 #$FF0F

3699 OR R1 R1 R0

3700 STR R1 $f5; 5

3701 JMP OVERELSE978252713

3702else818195614 NOOP

3703 LDR R0 #0;

3704 STR R0 $14f;

3705 LDR R0 #0;

3706 STR R0 $151;

3707 LDR R0 #0;

3708 STR R0 $153;

3709 LDR R0 #16645

3710 ADD R1 R6 #2

3711 STR R1 #0;

3712 SENDATA R0;

3713 SUB R4 #65535;

3714 SZ TOQSTART807807393;

3715 ADD R4 R4 #1;

3716 JMP QS807807393;

3717TOQSTART807807393 LDR R4 #65528;

3718QS807807393 CLFZ; data man man

3719 LDR R0 #TEN813963376

3720 STR R0 $f0

3721TEN813963376 NOOP;

3722 LDR R0 $5

3723 SUB R0 #0

3724 SZ EL1152824266

3725IF1152824266 CLFZ; titi man man

3726 LDR R0 $5

3727 AND R0 R0 #$0002

3728 PRESENT R0 else795880177

3729 LDR R0 #17413;

3730 ADD R1 R6 #2

3731 STR R1 #0;

3732 SENDATA R0;

3733 SUB R4 #65535;

3734 SZ TOQSTART802036160;

3735 ADD R4 R4 #1;

3736 JMP QS802036160;

3737TOQSTART802036160 LDR R4 #65528;

3738QS802036160 CLFZ;

3739 LDR R0 #TEN486926811

3740 STR R0 $f0

3741TEN486926811 NOOP;

3742 LDR R0 $5

3743 SUB R0 #0

3744 SZ EL1756521812

3745IF1756521812 CLFZ

3746 ESL #89

3747 LDR R0 $e7 ; loading from mem

3748 OR R0 R0 #32768 ;loading the emit signal in

3749 STR R0 $e7; emitted signal packet\_\_1392610397 in mem

3750 ESL #91

3751 LDR R0 #1;

3752 STR R0 $144;

3753 LDR R0 #TI493082793; loaded the case label into the regiser

3754 STR R0 $f0; stored in memory

3755TI493082793 NOOP

3756 LDR R0 #16;

3757 LDR R1 $f5

3758 AND R1 R1 #$FF0F

3759 OR R1 R1 R0

3760 STR R1 $f5; 5

3761 JMP OVERELSE723436077

3762EL1756521812 CLFZ;

3763 LDR R0 #240;

3764 LDR R1 $f5

3765 AND R1 R1 #$FF0F

3766 OR R1 R1 R0

3767 STR R1 $f5; 5

3768OVERELSE723436077 NOOP;

3769 JMP OVERELSE1615343252

3770else795880177 NOOP

3771 LDR R0 #0;

3772 STR R0 $155;

3773 LDR R0 #0;

3774 STR R0 $157;

3775 LDR R0 #16901

3776 ADD R1 R6 #2

3777 STR R1 #0;

3778 SENDATA R0;

3779 SUB R4 #65535;

3780 SZ TOQSTART474614846;

3781 ADD R4 R4 #1;

3782 JMP QS474614846;

3783TOQSTART474614846 LDR R4 #65528;

3784QS474614846 CLFZ; data man man

3785 LDR R0 #TEN481155577

3786 STR R0 $f0

3787TEN481155577 NOOP;

3788 LDR R0 $5

3789 SUB R0 #0

3790 SZ EL1908921267

3791IF1908921267 CLFZ; titi man man

3792 LDR R0 $5

3793 AND R0 R0 #$0002

3794 PRESENT R0 else462687630

3795 LDR R0 #17157;

3796 ADD R1 R6 #2

3797 STR R1 #0;

3798 SENDATA R0;

3799 SUB R4 #65535;

3800 SZ TOQSTART468843612;

3801 ADD R4 R4 #1;

3802 JMP QS468843612;

3803TOQSTART468843612 LDR R4 #65528;

3804QS468843612 CLFZ;

3805 LDR R0 #TEN464611375

3806 STR R0 $f0

3807TEN464611375 NOOP;

3808 LDR R0 $5

3809 SUB R0 #0

3810 SZ EL55150481

3811IF55150481 CLFZ

3812 ESL #89

3813 LDR R0 $e7 ; loading from mem

3814 OR R0 R0 #32768 ;loading the emit signal in

3815 STR R0 $e7; emitted signal packet\_\_1392610397 in mem

3816 ESL #91

3817 LDR R0 #1;

3818 STR R0 $144;

3819 LDR R0 #TI446143427; loaded the case label into the regiser

3820 STR R0 $f0; stored in memory

3821TI446143427 NOOP

3822 LDR R0 #16;

3823 LDR R1 $f5

3824 AND R1 R1 #$FF0F

3825 OR R1 R1 R0

3826 STR R1 $f5; 5

3827 JMP OVERELSE1438226078

3828EL55150481 CLFZ;

3829 LDR R0 #240;

3830 LDR R1 $f5

3831 AND R1 R1 #$FF0F

3832 OR R1 R1 R0

3833 STR R1 $f5; 5

3834OVERELSE1438226078 NOOP;

3835 JMP OVERELSE1537606065

3836else462687630 NOOP

3837 LDR R0 #0;

3838 STR R0 $159;

3839 LDR R0 #TI452684159; loaded the case label into the regiser

3840 STR R0 $f0; stored in memory

3841TI452684159 NOOP

3842 LDR R0 #16;

3843 LDR R1 $f5

3844 AND R1 R1 #$FF0F

3845 OR R1 R1 R0

3846 STR R1 $f5; 5

3847OVERELSE1537606065 NOOP;

3848 JMP OVERELSE1609443889

3849EL1908921267 CLFZ; tutu man man

3850 LDR R0 #240;

3851 LDR R1 $f5

3852 AND R1 R1 #$FF0F

3853 OR R1 R1 R0

3854 STR R1 $f5; 5

3855OVERELSE1609443889 NOOP;

3856OVERELSE1615343252 NOOP;

3857 JMP OVERELSE1207637100

3858EL1152824266 CLFZ; tutu man man

3859 LDR R0 #240;

3860 LDR R1 $f5

3861 AND R1 R1 #$FF0F

3862 OR R1 R1 R0

3863 STR R1 $f5; 5

3864OVERELSE1207637100 NOOP;

3865OVERELSE978252713 NOOP;

3866 JMP OVERELSE1221034737

3867EL606937511 CLFZ; tutu man man

3868 LDR R0 #240;

3869 LDR R1 $f5

3870 AND R1 R1 #$FF0F

3871 OR R1 R1 R0

3872 STR R1 $f5; 5

3873OVERELSE1221034737 NOOP;

3874 JMP OVERELSE587573608

3875EL1679054320 CLFZ;

3876 LDR R0 #240;

3877 LDR R1 $f5

3878 AND R1 R1 #$FF0F

3879 OR R1 R1 R0

3880 STR R1 $f5; 5

3881OVERELSE587573608 NOOP;

3882 JMP OVERELSE304337135

3883EL108200175 CLFZ;

3884 LDR R0 #240;

3885 LDR R1 $f5

3886 AND R1 R1 #$FF0F

3887 OR R1 R1 R0

3888 STR R1 $f5; 5

3889OVERELSE304337135 NOOP;

3890 JMP OVERELSE2131505046

3891EL1570999058 CLFZ;

3892 LDR R0 #240;

3893 LDR R1 $f5

3894 AND R1 R1 #$FF0F

3895 OR R1 R1 R0

3896 STR R1 $f5; 5

3897OVERELSE2131505046 NOOP;

3898 LDR R14 #36;

3899 LDR R0 $f1; loading the address from the mem

3900 SUB R0 #0; checking if this is actually set

3901 SZ CONT452684159

3902 JMP R0

3903CONT452684159 LDR R0 #1;

3904 STR R0 $15b;

3905 LDR R0 #17670;

3906 ADD R1 R6 #3

3907 STR R1 #0;

3908 SENDATA R0;

3909 SUB R4 #65535;

3910 SZ TOQSTART532711930;

3911 ADD R4 R4 #1;

3912 JMP QS532711930;

3913TOQSTART532711930 LDR R4 #65528;

3914QS532711930 CLFZ;

3915 LDR R0 #TEN538867912

3916 STR R0 $f1

3917TEN538867912 NOOP;

3918 LDR R0 $6

3919 SUB R0 #0

3920 SZ EL384926713

3921IF384926713 CLFZ

3922 ESL #97

3923 LDR R0 #0;

3924 STR R0 $15e;

3925 ESL #98

3926 LDR R0 #0;

3927 STR R0 $160;

3928 LDR R0 #0;

3929 STR R0 $163;

3930 ESL #98

3931 LDR R0 #0;

3932 STR R0 $165;

3933 LDR R0 #0;

3934 STR R0 $167;

3935 ESL #98

3936 LDR R0 #TI519245718; loaded the case label into the regiser

3937 STR R0 $f1; stored in memory

3938TI519245718 NOOP

3939 LDR R0 #1;

3940 LDR R1 $f5

3941 AND R1 R1 #$FFF0

3942 OR R1 R1 R0

3943 STR R1 $f5; 8

3944 JMP OVERELSE244519295

3945EL384926713 CLFZ;

3946 LDR R0 #15;

3947 LDR R1 $f5

3948 AND R1 R1 #$FFF0

3949 OR R1 R1 R0

3950 STR R1 $f5; 8

3951OVERELSE244519295 NOOP;

3952 LDR R14 #37;

3953 LDR R0 $f2; loading the address from the mem

3954 SUB R0 #0; checking if this is actually set

3955 SZ CONT519245718

3956 JMP R0

3957CONT519245718 LDR R0 #1;

3958 STR R0 $169;

3959 LDR R0 #17927;

3960 ADD R1 R6 #4

3961 STR R1 #0;

3962 SENDATA R0;

3963 SUB R4 #65535;

3964 SZ TOQSTART525401701;

3965 ADD R4 R4 #1;

3966 JMP QS525401701;

3967TOQSTART525401701 LDR R4 #65528;

3968QS525401701 CLFZ;

3969 LDR R0 #TEN506933754

3970 STR R0 $f2

3971TEN506933754 NOOP;

3972 LDR R0 $7

3973 SUB R0 #0

3974 SZ EL1800866006

3975IF1800866006 CLFZ

3976 ESL #114

3977 LDR R0 #0;

3978 STR R0 $16c;

3979 ESL #115

3980 LDR R0 #0;

3981 STR R0 $16e;

3982 LDR R0 #0;

3983 STR R0 $171;

3984 ESL #115

3985 LDR R0 #0;

3986 STR R0 $173;

3987 LDR R0 #0;

3988 STR R0 $176;

3989 ESL #115

3990 LDR R0 #TI513474485; loaded the case label into the regiser

3991 STR R0 $f2; stored in memory

3992TI513474485 NOOP

3993 LDR R0 #4096;

3994 LDR R1 $f6

3995 AND R1 R1 #$0FFF

3996 OR R1 R1 R0

3997 STR R1 $f6; 15

3998 JMP OVERELSE65512451

3999EL1800866006 CLFZ;

4000 LDR R0 #61440;

4001 LDR R1 $f6

4002 AND R1 R1 #$0FFF

4003 OR R1 R1 R0

4004 STR R1 $f6; 15

4005OVERELSE65512451 NOOP;

4006 LDR R1 #0 ;this will hold the terminate nodes

4007 LDR R0 $f5; loading the value from mem pipi

4008 AND R0 R0 #3840; making it right

4009 OR R1 R1 R0 ;getting the terminate node into R1 2

4010 LDR R0 $f5; loading the value from mem pipi

4011 AND R0 R0 #240; making it right

4012 OR R1 R1 R0 ;getting the terminate node into R1 5

4013 LDR R0 $f5; loading the value from mem pipi

4014 AND R0 R0 #15; making it right

4015 OR R1 R1 R0 ;getting the terminate node into R1 8

4016 LDR R0 $f6; loading the value from mem pipi

4017 AND R0 R0 #61440; making it right

4018 OR R1 R1 R0 ;getting the terminate node into R1 15

4019 STR R1 $1d1; loading the memory in the place

4020 LDR R1 #0 ;loading zeros for the next turn

4021 LDR R0 #N169029534811; loading the address into register

4022 STR R0 $18f; loading its mem

4023 LDR R0 #N16902953480123456 ;loading the address in register

4024 STR R0 $19d; loading it in mem

4025 LDR R0 #397 ;loaded the address of the joinnode in register

4026 STR R0 $1d2; loaded it into the endPointer memory

4027 LDR R0 #$1d1

4028 LDR R2 #0;

4029 LDR R1 #0

4030CHKENDLAB1690295348 LDR R0 #$1d1

4031 ADD R0 R0 R1;

4032 LDR R0 R0;

4033 CHKEND R2 R0;

4034 SUB R1 #0

4035 SZ ENDOVER1690295348

4036 ADD R1 R1 #1

4037 JMP CHKENDLAB1690295348

4038ENDOVER1690295348 CLFZ

4039 LDR R0 #$1d1

4040 ADD R1 R1 #1;

4041 ADD R0 R0 R1

4042 LDR R0 R0

4043 ADD R2 R2 #1

4044 ADD R0 R0 R2;

4045 LDR R0 R0

4046 JMP R0; adding stuff

4047N16902953480123456 NOOP

4048 LDR R0 #61440;

4049 LDR R1 $f5

4050 AND R1 R1 #$0FFF

4051 OR R1 R1 R0

4052 STR R1 $f5; 16

4053 JMP END1;

4054 JMP DUMMY1690295348;

4055N169029534811 NOOP

4056 LDR R0 #4096;

4057 LDR R1 $f5

4058 AND R1 R1 #$0FFF

4059 OR R1 R1 R0

4060 STR R1 $f5; 16

4061 LDR R0 #$0;

4062 STR R0 $ee;

4063 JMP END1;

4064 JMP DUMMY1690295348;

4065DUMMY1690295348

4066 JMP OVERELSE1476532885

4067EL1515723796 CLFZ;

4068 LDR R0 #61440;

4069 LDR R1 $f5

4070 AND R1 R1 #$0FFF

4071 OR R1 R1 R0

4072 STR R1 $f5; 16

4073 JMP END1;

4074OVERELSE1476532885 NOOP;

4075 JMP ENDS13822221771

4076case1907964102 NOOP;

4077 ESL #52

4078 ESL #52

4079 ESL #53

4080 ESL #53

4081 ESL #53

4082 ESL #54

4083 LDR R0 #case495006538 ;

4084 STR R0 $12e ;

4085 LDR R0 #301

4086 SWITCH R1 R0 ;

4087case495006538 NOOP;

4088 ESL #56

4089PNODE583114036 NOOP

4090 LDR R0 #PNODE583114036; loading the case address into register

4091 STR R0 $ee; storing the address in the memory

4092 LDR R14 #38;

4093 LDR R0 $ef; loading the address from the mem

4094 SUB R0 #0; checking if this is actually set

4095 SZ CONT589270018

4096 JMP R0

4097CONT589270018 LDR R0 #case571186820 ;

4098 STR R0 $130 ;

4099 LDR R0 #case2108470141 ;

4100 STR R0 $131 ;

4101 LDR R0 #303

4102 SWITCH R1 R0 ;

4103case571186820 NOOP;

4104 LDR R0 $e8; loading the right lock place res1

4105 OR R0 R0 #16384; making the app place high for lock release

4106 STR R0 $e8; putting the thing back im mem

4107 LDR R0 $e9; loading the right lock place res2

4108 OR R0 R0 #16384; making the app place high for lock release

4109 STR R0 $e9; putting the thing back im mem

4110 LDR R0 $ea; loading the right lock place res3

4111 OR R0 R0 #16384; making the app place high for lock release

4112 STR R0 $ea; putting the thing back im mem

4113 LDR R0 #TI563876591; loaded the case label into the regiser

4114 STR R0 $ef; stored in memory

4115TI563876591 NOOP

4116 LDR R0 #0;

4117 LDR R1 $f5

4118 AND R1 R1 #$F0FF

4119 OR R1 R1 R0

4120 STR R1 $f5; 2

4121 JMP ENDS10867268390

4122case2108470141 NOOP;

4123 LDR R0 #case545408644 ;

4124 STR R0 $133 ;

4125 LDR R0 #306

4126 SWITCH R1 R0 ;

4127case545408644 NOOP;

4128 ESL #57

4129 LDR R0 #case631977146 ;

4130 STR R0 $135 ;

4131 LDR R0 #case69466429 ;

4132 STR R0 $136 ;

4133 LDR R0 #308

4134 SWITCH R1 R0 ;

4135case631977146 NOOP;

4136 LDR R0 #case627744908 ;

4137 STR R0 $138 ;

4138 LDR R0 #311

4139 SWITCH R1 R0 ;

4140case627744908 NOOP;

4141 LDR R0 #case615817692 ;

4142 STR R0 $13a ;

4143 LDR R0 #313

4144 SWITCH R1 R0 ;

4145case615817692 NOOP;

4146 LDR R0 #18180

4147 ADD R1 R6 #1

4148 STR R1 #0;

4149 SENDATA R0;

4150 SUB R4 #65535;

4151 SZ TOQSTART602351481;

4152 ADD R4 R4 #1;

4153 JMP QS602351481;

4154TOQSTART602351481 LDR R4 #65528;

4155QS602351481 CLFZ; data man man

4156 LDR R0 #TEN1864327873

4157 STR R0 $ef

4158TEN1864327873 NOOP;

4159 LDR R0 $4

4160 SUB R0 #0

4161 SZ EL1175245925

4162IF1175245925 CLFZ; titi man man

4163 LDR R0 $4

4164 AND R0 R0 #$0002

4165 PRESENT R0 else1870483855

4166 LDR R0 #19204;

4167 ADD R1 R6 #1

4168 STR R1 #0;

4169 SENDATA R0;

4170 SUB R4 #65535;

4171 SZ TOQSTART1852400657;

4172 ADD R4 R4 #1;

4173 JMP QS1852400657;

4174TOQSTART1852400657 LDR R4 #65528;

4175QS1852400657 CLFZ;

4176 LDR R0 #TEN1858556640

4177 STR R0 $ef

4178TEN1858556640 NOOP;

4179 LDR R0 $4

4180 SUB R0 #0

4181 SZ EL1327645381

4182IF1327645381 CLFZ

4183 ESL #61

4184 LDR R0 #19460

4185 ADD R1 R6 #1

4186 STR R1 #0;

4187 SENDATA R0;

4188 SUB R4 #65535;

4189 SZ TOQSTART1840088692;

4190 ADD R4 R4 #1;

4191 JMP QS1840088692;

4192TOQSTART1840088692 LDR R4 #65528;

4193QS1840088692 CLFZ; data man man

4194 LDR R0 #TEN1848168419

4195 STR R0 $ef

4196TEN1848168419 NOOP;

4197 LDR R0 $4

4198 SUB R0 #0

4199 SZ EL1834009436

4200IF1834009436 CLFZ; titi man man

4201 LDR R0 $4

4202 AND R0 R0 #$0002

4203 PRESENT R0 else1829700472

4204 ESL #62

4205 LDR R0 $e7 ; loading from mem

4206 OR R0 R0 #128 ;loading the emit signal in

4207 STR R0 $e7; emitted signal res1\_\_1398381631 in mem

4208 LDR R0 $e8; loading the right lock place res1

4209 OR R0 R0 #16384; making the app place high for lock release

4210 STR R0 $e8; putting the thing back im mem

4211 ESL #63

4212 LDR R0 $e7 ; loading from mem

4213 OR R0 R0 #64 ;loading the emit signal in

4214 STR R0 $e7; emitted signal res2\_\_1411847842 in mem

4215 LDR R0 $e9; loading the right lock place res2

4216 OR R0 R0 #16384; making the app place high for lock release

4217 STR R0 $e9; putting the thing back im mem

4218 ESL #64

4219 LDR R0 $e7 ; loading from mem

4220 OR R0 R0 #32 ;loading the emit signal in

4221 STR R0 $e7; emitted signal res3\_\_1325279340 in mem

4222 LDR R0 $ea; loading the right lock place res3

4223 OR R0 R0 #16384; making the app place high for lock release

4224 STR R0 $ea; putting the thing back im mem

4225 ESL #66

4226 LDR R0 #1;

4227 STR R0 $134;

4228 LDR R0 #TI1836241203; loaded the case label into the regiser

4229 STR R0 $ef; stored in memory

4230TI1836241203 NOOP

4231 LDR R0 #256;

4232 LDR R1 $f5

4233 AND R1 R1 #$F0FF

4234 OR R1 R1 R0

4235 STR R1 $f5; 2

4236 JMP OVERELSE2089642658

4237else1829700472 NOOP

4238 LDR R0 $e8; loading the right lock place res1

4239 OR R0 R0 #16384; making the app place high for lock release

4240 STR R0 $e8; putting the thing back im mem

4241 LDR R0 $e9; loading the right lock place res2

4242 OR R0 R0 #16384; making the app place high for lock release

4243 STR R0 $e9; putting the thing back im mem

4244 LDR R0 $ea; loading the right lock place res3

4245 OR R0 R0 #16384; making the app place high for lock release

4246 STR R0 $ea; putting the thing back im mem

4247 ESL #66

4248 LDR R0 #1;

4249 STR R0 $134;

4250 LDR R0 #TI1916268975; loaded the case label into the regiser

4251 STR R0 $ef; stored in memory

4252TI1916268975 NOOP

4253 LDR R0 #256;

4254 LDR R1 $f5

4255 AND R1 R1 #$F0FF

4256 OR R1 R1 R0

4257 STR R1 $f5; 2

4258OVERELSE2089642658 NOOP;

4259 JMP OVERELSE70919003

4260EL1834009436 CLFZ; tutu man man

4261 LDR R0 #3840;

4262 LDR R1 $f5

4263 AND R1 R1 #$F0FF

4264 OR R1 R1 R0

4265 STR R1 $f5; 2

4266OVERELSE70919003 NOOP;

4267 JMP OVERELSE1742851272

4268EL1327645381 CLFZ;

4269 LDR R0 #3840;

4270 LDR R1 $f5

4271 AND R1 R1 #$F0FF

4272 OR R1 R1 R0

4273 STR R1 $f5; 2

4274OVERELSE1742851272 NOOP;

4275 JMP OVERELSE132638106

4276else1870483855 NOOP

4277 LDR R0 #case1922424957 ;

4278 STR R0 $13c ;

4279 LDR R0 #315

4280 SWITCH R1 R0 ;

4281case1922424957 NOOP;

4282 LDR R0 #case1908958745 ;

4283 STR R0 $13e ;

4284 LDR R0 #317

4285 SWITCH R1 R0 ;

4286case1908958745 NOOP;

4287 LDR R0 #18436

4288 ADD R1 R6 #1

4289 STR R1 #0;

4290 SENDATA R0;

4291 SUB R4 #65535;

4292 SZ TOQSTART1897031530;

4293 ADD R4 R4 #1;

4294 JMP QS1897031530;

4295TOQSTART1897031530 LDR R4 #65528;

4296QS1897031530 CLFZ; data man man

4297 LDR R0 #TEN1878563582

4298 STR R0 $ef

4299TEN1878563582 NOOP;

4300 LDR R0 $4

4301 SUB R0 #0

4302 SZ EL1371989575

4303IF1371989575 CLFZ; titi man man

4304 LDR R0 $4

4305 AND R0 R0 #$0002

4306 PRESENT R0 else1874331344

4307 LDR R0 #18692;

4308 ADD R1 R6 #1

4309 STR R1 #0;

4310 SENDATA R0;

4311 SUB R4 #65535;

4312 SZ TOQSTART1880487327;

4313 ADD R4 R4 #1;

4314 JMP QS1880487327;

4315TOQSTART1880487327 LDR R4 #65528;

4316QS1880487327 CLFZ;

4317 LDR R0 #TEN1960515098

4318 STR R0 $ef

4319TEN1960515098 NOOP;

4320 LDR R0 $4

4321 SUB R0 #0

4322 SZ EL66910768

4323IF66910768 CLFZ

4324 ESL #61

4325 LDR R0 #18948

4326 ADD R1 R6 #1

4327 STR R1 #0;

4328 SENDATA R0;

4329 SUB R4 #65535;

4330 SZ TOQSTART1967055829;

4331 ADD R4 R4 #1;

4332 JMP QS1967055829;

4333TOQSTART1967055829 LDR R4 #65528;

4334QS1967055829 CLFZ; data man man

4335 LDR R0 #TEN1948587882

4336 STR R0 $ef

4337TEN1948587882 NOOP;

4338 LDR R0 $4

4339 SUB R0 #0

4340 SZ EL1527194254

4341IF1527194254 CLFZ; titi man man

4342 LDR R0 $4

4343 AND R0 R0 #$0002

4344 PRESENT R0 else1954743865

4345 ESL #62

4346 LDR R0 $e7 ; loading from mem

4347 OR R0 R0 #128 ;loading the emit signal in

4348 STR R0 $e7; emitted signal res1\_\_1398381631 in mem

4349 LDR R0 $e8; loading the right lock place res1

4350 OR R0 R0 #16384; making the app place high for lock release

4351 STR R0 $e8; putting the thing back im mem

4352 ESL #63

4353 LDR R0 $e7 ; loading from mem

4354 OR R0 R0 #64 ;loading the emit signal in

4355 STR R0 $e7; emitted signal res2\_\_1411847842 in mem

4356 LDR R0 $e9; loading the right lock place res2

4357 OR R0 R0 #16384; making the app place high for lock release

4358 STR R0 $e9; putting the thing back im mem

4359 ESL #64

4360 LDR R0 $e7 ; loading from mem

4361 OR R0 R0 #32 ;loading the emit signal in

4362 STR R0 $e7; emitted signal res3\_\_1325279340 in mem

4363 LDR R0 $ea; loading the right lock place res3

4364 OR R0 R0 #16384; making the app place high for lock release

4365 STR R0 $ea; putting the thing back im mem

4366 ESL #66

4367 LDR R0 #1;

4368 STR R0 $134;

4369 LDR R0 #TI1935121671; loaded the case label into the regiser

4370 STR R0 $ef; stored in memory

4371TI1935121671 NOOP

4372 LDR R0 #256;

4373 LDR R1 $f5

4374 AND R1 R1 #$F0FF

4375 OR R1 R1 R0

4376 STR R1 $f5; 2

4377 JMP OVERELSE219310223

4378else1954743865 NOOP

4379 LDR R0 $e8; loading the right lock place res1

4380 OR R0 R0 #16384; making the app place high for lock release

4381 STR R0 $e8; putting the thing back im mem

4382 LDR R0 $e9; loading the right lock place res2

4383 OR R0 R0 #16384; making the app place high for lock release

4384 STR R0 $e9; putting the thing back im mem

4385 LDR R0 $ea; loading the right lock place res3

4386 OR R0 R0 #16384; making the app place high for lock release

4387 STR R0 $ea; putting the thing back im mem

4388 ESL #66

4389 LDR R0 #1;

4390 STR R0 $134;

4391 LDR R0 #TI1941277653; loaded the case label into the regiser

4392 STR R0 $ef; stored in memory

4393TI1941277653 NOOP

4394 LDR R0 #256;

4395 LDR R1 $f5

4396 AND R1 R1 #$F0FF

4397 OR R1 R1 R0

4398 STR R1 $f5; 2

4399OVERELSE219310223 NOOP;

4400 JMP OVERELSE1149331296

4401EL1527194254 CLFZ; tutu man man

4402 LDR R0 #3840;

4403 LDR R1 $f5

4404 AND R1 R1 #$F0FF

4405 OR R1 R1 R0

4406 STR R1 $f5; 2

4407OVERELSE1149331296 NOOP;

4408 JMP OVERELSE2046655726

4409EL66910768 CLFZ;

4410 LDR R0 #3840;

4411 LDR R1 $f5

4412 AND R1 R1 #$F0FF

4413 OR R1 R1 R0

4414 STR R1 $f5; 2

4415OVERELSE2046655726 NOOP;

4416 JMP OVERELSE1197418022

4417else1874331344 NOOP

4418 LDR R0 $e8; loading the right lock place res1

4419 OR R0 R0 #16384; making the app place high for lock release

4420 STR R0 $e8; putting the thing back im mem

4421 LDR R0 $e9; loading the right lock place res2

4422 OR R0 R0 #16384; making the app place high for lock release

4423 STR R0 $e9; putting the thing back im mem

4424 LDR R0 $ea; loading the right lock place res3

4425 OR R0 R0 #16384; making the app place high for lock release

4426 STR R0 $ea; putting the thing back im mem

4427 LDR R0 #TI1923194455; loaded the case label into the regiser

4428 STR R0 $ef; stored in memory

4429TI1923194455 NOOP

4430 LDR R0 #256;

4431 LDR R1 $f5

4432 AND R1 R1 #$F0FF

4433 OR R1 R1 R0

4434 STR R1 $f5; 2

4435OVERELSE1197418022 NOOP;

4436 JMP OVERELSE1763167496

4437EL1371989575 CLFZ; tutu man man

4438 LDR R0 #3840;

4439 LDR R1 $f5

4440 AND R1 R1 #$F0FF

4441 OR R1 R1 R0

4442 STR R1 $f5; 2

4443OVERELSE1763167496 NOOP;

4444 JMP ENDS5484784520

4445ENDS5484784520 NOOP

4446 JMP ENDS5665616500

4447ENDS5665616500 NOOP

4448OVERELSE132638106 NOOP;

4449 JMP OVERELSE590028694

4450EL1175245925 CLFZ; tutu man man

4451 LDR R0 #3840;

4452 LDR R1 $f5

4453 AND R1 R1 #$F0FF

4454 OR R1 R1 R0

4455 STR R1 $f5; 2

4456OVERELSE590028694 NOOP;

4457 JMP ENDS5604056680

4458ENDS5604056680 NOOP

4459 JMP ENDS4803778970

4460ENDS4803778970 NOOP

4461 JMP ENDS4753761610

4462case69466429 NOOP;

4463 ESL #66

4464 LDR R0 #1;

4465 STR R0 $134;

4466 LDR R0 #0;

4467 STR R0 $134;

4468 LDR R0 #0;

4469 STR R0 $137;

4470 LDR R0 #19716

4471 ADD R1 R6 #1

4472 STR R1 #0;

4473 SENDATA R0;

4474 SUB R4 #65535;

4475 SZ TOQSTART1929350437;

4476 ADD R4 R4 #1;

4477 JMP QS1929350437;

4478TOQSTART1929350437 LDR R4 #65528;

4479QS1929350437 CLFZ; data man man

4480 LDR R0 #TEN2009378208

4481 STR R0 $ef

4482TEN2009378208 NOOP;

4483 LDR R0 $4

4484 SUB R0 #0

4485 SZ EL650742315

4486IF650742315 CLFZ; titi man man

4487 LDR R0 $4

4488 AND R0 R0 #$0002

4489 PRESENT R0 else2017457935

4490 LDR R0 #20740;

4491 ADD R1 R6 #1

4492 STR R1 #0;

4493 SENDATA R0;

4494 SUB R4 #65535;

4495 SZ TOQSTART1998989988;

4496 ADD R4 R4 #1;

4497 JMP QS1998989988;

4498TOQSTART1998989988 LDR R4 #65528;

4499QS1998989988 CLFZ;

4500 LDR R0 #TEN2005145971

4501 STR R0 $ef

4502TEN2005145971 NOOP;

4503 LDR R0 $4

4504 SUB R0 #0

4505 SZ EL825313868

4506IF825313868 CLFZ

4507 ESL #61

4508 LDR R0 #20996

4509 ADD R1 R6 #1

4510 STR R1 #0;

4511 SENDATA R0;

4512 SUB R4 #65535;

4513 SZ TOQSTART1987062772;

4514 ADD R4 R4 #1;

4515 JMP QS1987062772;

4516TOQSTART1987062772 LDR R4 #65528;

4517QS1987062772 CLFZ; data man man

4518 LDR R0 #TEN1993218755

4519 STR R0 $ef

4520TEN1993218755 NOOP;

4521 LDR R0 $4

4522 SUB R0 #0

4523 SZ EL634969619

4524IF634969619 CLFZ; titi man man

4525 LDR R0 $4

4526 AND R0 R0 #$0002

4527 PRESENT R0 else1973596561

4528 ESL #62

4529 LDR R0 $e7 ; loading from mem

4530 OR R0 R0 #128 ;loading the emit signal in

4531 STR R0 $e7; emitted signal res1\_\_1398381631 in mem

4532 LDR R0 $e8; loading the right lock place res1

4533 OR R0 R0 #16384; making the app place high for lock release

4534 STR R0 $e8; putting the thing back im mem

4535 ESL #63

4536 LDR R0 $e7 ; loading from mem

4537 OR R0 R0 #64 ;loading the emit signal in

4538 STR R0 $e7; emitted signal res2\_\_1411847842 in mem

4539 LDR R0 $e9; loading the right lock place res2

4540 OR R0 R0 #16384; making the app place high for lock release

4541 STR R0 $e9; putting the thing back im mem

4542 ESL #64

4543 LDR R0 $e7 ; loading from mem

4544 OR R0 R0 #32 ;loading the emit signal in

4545 STR R0 $e7; emitted signal res3\_\_1325279340 in mem

4546 LDR R0 $ea; loading the right lock place res3

4547 OR R0 R0 #16384; making the app place high for lock release

4548 STR R0 $ea; putting the thing back im mem

4549 ESL #66

4550 LDR R0 #1;

4551 STR R0 $134;

4552 LDR R0 #TI1979752543; loaded the case label into the regiser

4553 STR R0 $ef; stored in memory

4554TI1979752543 NOOP

4555 LDR R0 #256;

4556 LDR R1 $f5

4557 AND R1 R1 #$F0FF

4558 OR R1 R1 R0

4559 STR R1 $f5; 2

4560 JMP OVERELSE564859151

4561else1973596561 NOOP

4562 LDR R0 $e8; loading the right lock place res1

4563 OR R0 R0 #16384; making the app place high for lock release

4564 STR R0 $e8; putting the thing back im mem

4565 LDR R0 $e9; loading the right lock place res2

4566 OR R0 R0 #16384; making the app place high for lock release

4567 STR R0 $e9; putting the thing back im mem

4568 LDR R0 $ea; loading the right lock place res3

4569 OR R0 R0 #16384; making the app place high for lock release

4570 STR R0 $ea; putting the thing back im mem

4571 ESL #66

4572 LDR R0 #1;

4573 STR R0 $134;

4574 LDR R0 #TI1665797441; loaded the case label into the regiser

4575 STR R0 $ef; stored in memory

4576TI1665797441 NOOP

4577 LDR R0 #256;

4578 LDR R1 $f5

4579 AND R1 R1 #$F0FF

4580 OR R1 R1 R0

4581 STR R1 $f5; 2

4582OVERELSE564859151 NOOP;

4583 JMP OVERELSE280248705

4584EL634969619 CLFZ; tutu man man

4585 LDR R0 #3840;

4586 LDR R1 $f5

4587 AND R1 R1 #$F0FF

4588 OR R1 R1 R0

4589 STR R1 $f5; 2

4590OVERELSE280248705 NOOP;

4591 JMP OVERELSE818731569

4592EL825313868 CLFZ;

4593 LDR R0 #3840;

4594 LDR R1 $f5

4595 AND R1 R1 #$F0FF

4596 OR R1 R1 R0

4597 STR R1 $f5; 2

4598OVERELSE818731569 NOOP;

4599 JMP OVERELSE853885366

4600else2017457935 NOOP

4601 LDR R0 #0;

4602 STR R0 $139;

4603 LDR R0 #0;

4604 STR R0 $13b;

4605 LDR R0 #19972

4606 ADD R1 R6 #1

4607 STR R1 #0;

4608 SENDATA R0;

4609 SUB R4 #65535;

4610 SZ TOQSTART1672338172;

4611 ADD R4 R4 #1;

4612 JMP QS1672338172;

4613TOQSTART1672338172 LDR R4 #65528;

4614QS1672338172 CLFZ; data man man

4615 LDR R0 #TEN1653870225

4616 STR R0 $ef

4617TEN1653870225 NOOP;

4618 LDR R0 $4

4619 SUB R0 #0

4620 SZ EL1865116463

4621IF1865116463 CLFZ; titi man man

4622 LDR R0 $4

4623 AND R0 R0 #$0002

4624 PRESENT R0 else1660026207

4625 LDR R0 #20228;

4626 ADD R1 R6 #1

4627 STR R1 #0;

4628 SENDATA R0;

4629 SUB R4 #65535;

4630 SZ TOQSTART1643482004;

4631 ADD R4 R4 #1;

4632 JMP QS1643482004;

4633TOQSTART1643482004 LDR R4 #65528;

4634QS1643482004 CLFZ;

4635 LDR R0 #TEN1649637987

4636 STR R0 $ef

4637TEN1649637987 NOOP;

4638 LDR R0 $4

4639 SUB R0 #0

4640 SZ EL1690544911

4641IF1690544911 CLFZ

4642 ESL #61

4643 LDR R0 #20484

4644 ADD R1 R6 #1

4645 STR R1 #0;

4646 SENDATA R0;

4647 SUB R4 #65535;

4648 SZ TOQSTART1631554789;

4649 ADD R4 R4 #1;

4650 JMP QS1631554789;

4651TOQSTART1631554789 LDR R4 #65528;

4652QS1631554789 CLFZ; data man man

4653 LDR R0 #TEN1637710771

4654 STR R0 $ef

4655TEN1637710771 NOOP;

4656 LDR R0 $4

4657 SUB R0 #0

4658 SZ EL1144138899

4659IF1144138899 CLFZ; titi man man

4660 LDR R0 $4

4661 AND R0 R0 #$0002

4662 PRESENT R0 else1717738542

4663 ESL #62

4664 LDR R0 $e7 ; loading from mem

4665 OR R0 R0 #128 ;loading the emit signal in

4666 STR R0 $e7; emitted signal res1\_\_1398381631 in mem

4667 LDR R0 $e8; loading the right lock place res1

4668 OR R0 R0 #16384; making the app place high for lock release

4669 STR R0 $e8; putting the thing back im mem

4670 ESL #63

4671 LDR R0 $e7 ; loading from mem

4672 OR R0 R0 #64 ;loading the emit signal in

4673 STR R0 $e7; emitted signal res2\_\_1411847842 in mem

4674 LDR R0 $e9; loading the right lock place res2

4675 OR R0 R0 #16384; making the app place high for lock release

4676 STR R0 $e9; putting the thing back im mem

4677 ESL #64

4678 LDR R0 $e7 ; loading from mem

4679 OR R0 R0 #32 ;loading the emit signal in

4680 STR R0 $e7; emitted signal res3\_\_1325279340 in mem

4681 LDR R0 $ea; loading the right lock place res3

4682 OR R0 R0 #16384; making the app place high for lock release

4683 STR R0 $ea; putting the thing back im mem

4684 ESL #66

4685 LDR R0 #1;

4686 STR R0 $134;

4687 LDR R0 #TI1722740278; loaded the case label into the regiser

4688 STR R0 $ef; stored in memory

4689TI1722740278 NOOP

4690 LDR R0 #256;

4691 LDR R1 $f5

4692 AND R1 R1 #$F0FF

4693 OR R1 R1 R0

4694 STR R1 $f5; 2

4695 JMP OVERELSE966762122

4696else1717738542 NOOP

4697 LDR R0 $e8; loading the right lock place res1

4698 OR R0 R0 #16384; making the app place high for lock release

4699 STR R0 $e8; putting the thing back im mem

4700 LDR R0 $e9; loading the right lock place res2

4701 OR R0 R0 #16384; making the app place high for lock release

4702 STR R0 $e9; putting the thing back im mem

4703 LDR R0 $ea; loading the right lock place res3

4704 OR R0 R0 #16384; making the app place high for lock release

4705 STR R0 $ea; putting the thing back im mem

4706 ESL #66

4707 LDR R0 #1;

4708 STR R0 $134;

4709 LDR R0 #TI1704272331; loaded the case label into the regiser

4710 STR R0 $ef; stored in memory

4711TI1704272331 NOOP

4712 LDR R0 #256;

4713 LDR R1 $f5

4714 AND R1 R1 #$F0FF

4715 OR R1 R1 R0

4716 STR R1 $f5; 2

4717OVERELSE966762122 NOOP;

4718 JMP OVERELSE1184239965

4719EL1144138899 CLFZ; tutu man man

4720 LDR R0 #3840;

4721 LDR R1 $f5

4722 AND R1 R1 #$F0FF

4723 OR R1 R1 R0

4724 STR R1 $f5; 2

4725OVERELSE1184239965 NOOP;

4726 JMP OVERELSE2011747057

4727EL1690544911 CLFZ;

4728 LDR R0 #3840;

4729 LDR R1 $f5

4730 AND R1 R1 #$F0FF

4731 OR R1 R1 R0

4732 STR R1 $f5; 2

4733OVERELSE2011747057 NOOP;

4734 JMP OVERELSE557232432

4735else1660026207 NOOP

4736 LDR R0 $e8; loading the right lock place res1

4737 OR R0 R0 #16384; making the app place high for lock release

4738 STR R0 $e8; putting the thing back im mem

4739 LDR R0 $e9; loading the right lock place res2

4740 OR R0 R0 #16384; making the app place high for lock release

4741 STR R0 $e9; putting the thing back im mem

4742 LDR R0 $ea; loading the right lock place res3

4743 OR R0 R0 #16384; making the app place high for lock release

4744 STR R0 $ea; putting the thing back im mem

4745 LDR R0 #0;

4746 STR R0 $13d;

4747 LDR R0 #TI1710428313; loaded the case label into the regiser

4748 STR R0 $ef; stored in memory

4749TI1710428313 NOOP

4750 LDR R0 #256;

4751 LDR R1 $f5

4752 AND R1 R1 #$F0FF

4753 OR R1 R1 R0

4754 STR R1 $f5; 2

4755OVERELSE557232432 NOOP;

4756 JMP OVERELSE1703349657

4757EL1865116463 CLFZ; tutu man man

4758 LDR R0 #3840;

4759 LDR R1 $f5

4760 AND R1 R1 #$F0FF

4761 OR R1 R1 R0

4762 STR R1 $f5; 2

4763OVERELSE1703349657 NOOP;

4764OVERELSE853885366 NOOP;

4765 JMP OVERELSE238860987

4766EL650742315 CLFZ; tutu man man

4767 LDR R0 #3840;

4768 LDR R1 $f5

4769 AND R1 R1 #$F0FF

4770 OR R1 R1 R0

4771 STR R1 $f5; 2

4772OVERELSE238860987 NOOP;

4773 JMP ENDS4753761611

4774ENDS4753761610 NOOP

4775ENDS4753761611 NOOP

4776 JMP ENDS10682588920

4777ENDS10682588920 NOOP

4778 JMP ENDS10867268391

4779ENDS10867268390 NOOP

4780ENDS10867268391 NOOP

4781 LDR R14 #39;

4782 LDR R0 $f0; loading the address from the mem

4783 SUB R0 #0; checking if this is actually set

4784 SZ CONT1710428313

4785 JMP R0

4786CONT1710428313 LDR R0 #case1692345115 ;

4787 STR R0 $140 ;

4788 LDR R0 #case2014191437 ;

4789 STR R0 $141 ;

4790 LDR R0 #319

4791 SWITCH R1 R0 ;

4792case1692345115 NOOP;

4793 LDR R0 $eb; loading the right lock place packet

4794 OR R0 R0 #8192; making the app place high for lock release

4795 STR R0 $eb; putting the thing back im mem

4796 LDR R0 #TI1688112877; loaded the case label into the regiser

4797 STR R0 $f0; stored in memory

4798TI1688112877 NOOP

4799 LDR R0 #0;

4800 LDR R1 $f5

4801 AND R1 R1 #$FF0F

4802 OR R1 R1 R0

4803 STR R1 $f5; 5

4804 JMP ENDS9697631730

4805case2014191437 NOOP;

4806 LDR R0 #case1768140648 ;

4807 STR R0 $143 ;

4808 LDR R0 #322

4809 SWITCH R1 R0 ;

4810case1768140648 NOOP;

4811 ESL #74

4812 LDR R0 #case1756213432 ;

4813 STR R0 $145 ;

4814 LDR R0 #case551102726 ;

4815 STR R0 $146 ;

4816 LDR R0 #324

4817 SWITCH R1 R0 ;

4818case1756213432 NOOP;

4819 ESL #75

4820 LDR R0 #case1748903203 ;

4821 STR R0 $148 ;

4822 LDR R0 #327

4823 SWITCH R1 R0 ;

4824case1748903203 NOOP;

4825 LDR R0 #case1736975987; loading the right label into register

4826 STR R0 $f0; loading into mem

4827case1736975987 NOOP

4828 LDR R0 $e8; loading the value into register

4829 AND R0 R0 #16384; getting the right value into R0

4830 SUB R0 #16384;

4831 SZ PRESE78089347;

4832 JMP else1388551224;

4833PRESE78089347 CLFZ;Jump to the right place if this thing is not presentres1\_\_1398381631

4834 LDR R0 $e7 ;Loading the register which has this signal

4835 AND R0 R0 #128 ;Got the exact signal

4836 PRESENT R0 else1817003758 ;checking if the signal is present res1\_\_1398381631

4837 LDR R0 $eb; loading the right lock place packet

4838 OR R0 R0 #8192; making the app place high for lock release

4839 STR R0 $eb; putting the thing back im mem

4840 ESL #91

4841 LDR R0 #1;

4842 STR R0 $144;

4843 LDR R0 #TI1812771520; loaded the case label into the regiser

4844 STR R0 $f0; stored in memory

4845TI1812771520 NOOP

4846 LDR R0 #16;

4847 LDR R1 $f5

4848 AND R1 R1 #$FF0F

4849 OR R1 R1 R0

4850 STR R1 $f5; 5

4851 JMP OVERELSE1565928000

4852else1817003758 NOOP

4853 LDR R0 #case1818927503 ;

4854 STR R0 $14a ;

4855 LDR R0 #329

4856 SWITCH R1 R0 ;

4857case1818927503 NOOP;

4858 ESL #77

4859 LDR R0 #case1807000287 ;

4860 STR R0 $14c ;

4861 LDR R0 #331

4862 SWITCH R1 R0 ;

4863case1807000287 NOOP;

4864 LDR R0 #case1793534075 ;

4865 STR R0 $14e ;

4866 LDR R0 #333

4867 SWITCH R1 R0 ;

4868case1793534075 NOOP;

4869 ESL #79

4870 LDR R0 #case1781222111 ;

4871 STR R0 $150 ;

4872 LDR R0 #335

4873 SWITCH R1 R0 ;

4874case1781222111 NOOP;

4875 LDR R0 #case1473807739 ;

4876 STR R0 $152 ;

4877 LDR R0 #337

4878 SWITCH R1 R0 ;

4879case1473807739 NOOP;

4880 LDR R0 #case1463419519 ;

4881 STR R0 $154 ;

4882 LDR R0 #339

4883 SWITCH R1 R0 ;

4884case1463419519 NOOP;

4885 LDR R0 #case1451492303 ;

4886 STR R0 $156 ;

4887 LDR R0 #341

4888 SWITCH R1 R0 ;

4889case1451492303 NOOP;

4890 LDR R0 #21253

4891 ADD R1 R6 #2

4892 STR R1 #0;

4893 SENDATA R0;

4894 SUB R4 #65535;

4895 SZ TOQSTART1439180338;

4896 ADD R4 R4 #1;

4897 JMP QS1439180338;

4898TOQSTART1439180338 LDR R4 #65528;

4899QS1439180338 CLFZ; data man man

4900 LDR R0 #TEN1518053863

4901 STR R0 $f0

4902TEN1518053863 NOOP;

4903 LDR R0 $5

4904 SUB R0 #0

4905 SZ EL1729278657

4906IF1729278657 CLFZ; titi man man

4907 LDR R0 $5

4908 AND R0 R0 #$0002

4909 PRESENT R0 else1524209845

4910 LDR R0 #22021;

4911 ADD R1 R6 #2

4912 STR R1 #0;

4913 SENDATA R0;

4914 SUB R4 #65535;

4915 SZ TOQSTART1505741898;

4916 ADD R4 R4 #1;

4917 JMP QS1505741898;

4918TOQSTART1505741898 LDR R4 #65528;

4919QS1505741898 CLFZ;

4920 LDR R0 #TEN1512282629

4921 STR R0 $f0

4922TEN1512282629 NOOP;

4923 LDR R0 $5

4924 SUB R0 #0

4925 SZ EL1881678112

4926IF1881678112 CLFZ

4927 ESL #89

4928 LDR R0 $e7 ; loading from mem

4929 OR R0 R0 #512 ;loading the emit signal in

4930 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

4931 LDR R0 $eb; loading the right lock place packet

4932 OR R0 R0 #8192; making the app place high for lock release

4933 STR R0 $eb; putting the thing back im mem

4934 ESL #91

4935 LDR R0 #1;

4936 STR R0 $144;

4937 LDR R0 #TI1493814682; loaded the case label into the regiser

4938 STR R0 $f0; stored in memory

4939TI1493814682 NOOP

4940 LDR R0 #16;

4941 LDR R1 $f5

4942 AND R1 R1 #$FF0F

4943 OR R1 R1 R0

4944 STR R1 $f5; 5

4945 JMP OVERELSE18271507

4946EL1881678112 CLFZ;

4947 LDR R0 #240;

4948 LDR R1 $f5

4949 AND R1 R1 #$FF0F

4950 OR R1 R1 R0

4951 STR R1 $f5; 5

4952OVERELSE18271507 NOOP;

4953 JMP OVERELSE421394625

4954else1524209845 NOOP

4955 LDR R0 #case1501894409 ;

4956 STR R0 $158 ;

4957 LDR R0 #343

4958 SWITCH R1 R0 ;

4959case1501894409 NOOP;

4960 LDR R0 #case1489582444 ;

4961 STR R0 $15a ;

4962 LDR R0 #345

4963 SWITCH R1 R0 ;

4964case1489582444 NOOP;

4965 LDR R0 #21509

4966 ADD R1 R6 #2

4967 STR R1 #0;

4968 SENDATA R0;

4969 SUB R4 #65535;

4970 SZ TOQSTART1576150947;

4971 ADD R4 R4 #1;

4972 JMP QS1576150947;

4973TOQSTART1576150947 LDR R4 #65528;

4974QS1576150947 CLFZ; data man man

4975 LDR R0 #TEN1557683000

4976 STR R0 $f0

4977TEN1557683000 NOOP;

4978 LDR R0 $5

4979 SUB R0 #0

4980 SZ EL1321515675

4981IF1321515675 CLFZ; titi man man

4982 LDR R0 $5

4983 AND R0 R0 #$0002

4984 PRESENT R0 else1562684735

4985 LDR R0 #21765;

4986 ADD R1 R6 #2

4987 STR R1 #0;

4988 SENDATA R0;

4989 SUB R4 #65535;

4990 SZ TOQSTART1544216788;

4991 ADD R4 R4 #1;

4992 JMP QS1544216788;

4993TOQSTART1544216788 LDR R4 #65528;

4994QS1544216788 CLFZ;

4995 LDR R0 #TEN1550757519

4996 STR R0 $f0

4997TEN1550757519 NOOP;

4998 LDR R0 $5

4999 SUB R0 #0

5000 SZ EL1997629788

5001IF1997629788 CLFZ

5002 ESL #89

5003 LDR R0 $e7 ; loading from mem

5004 OR R0 R0 #512 ;loading the emit signal in

5005 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

5006 LDR R0 $eb; loading the right lock place packet

5007 OR R0 R0 #8192; making the app place high for lock release

5008 STR R0 $eb; putting the thing back im mem

5009 ESL #91

5010 LDR R0 #1;

5011 STR R0 $144;

5012 LDR R0 #TI1532289572; loaded the case label into the regiser

5013 STR R0 $f0; stored in memory

5014TI1532289572 NOOP

5015 LDR R0 #16;

5016 LDR R1 $f5

5017 AND R1 R1 #$FF0F

5018 OR R1 R1 R0

5019 STR R1 $f5; 5

5020 JMP OVERELSE177408801

5021EL1997629788 CLFZ;

5022 LDR R0 #240;

5023 LDR R1 $f5

5024 AND R1 R1 #$FF0F

5025 OR R1 R1 R0

5026 STR R1 $f5; 5

5027OVERELSE177408801 NOOP;

5028 JMP OVERELSE837054021

5029else1562684735 NOOP

5030 LDR R0 $eb; loading the right lock place packet

5031 OR R0 R0 #8192; making the app place high for lock release

5032 STR R0 $eb; putting the thing back im mem

5033 LDR R0 #TI1538445555; loaded the case label into the regiser

5034 STR R0 $f0; stored in memory

5035TI1538445555 NOOP

5036 LDR R0 #16;

5037 LDR R1 $f5

5038 AND R1 R1 #$FF0F

5039 OR R1 R1 R0

5040 STR R1 $f5; 5

5041OVERELSE837054021 NOOP;

5042 JMP OVERELSE45066781

5043EL1321515675 CLFZ; tutu man man

5044 LDR R0 #240;

5045 LDR R1 $f5

5046 AND R1 R1 #$FF0F

5047 OR R1 R1 R0

5048 STR R1 $f5; 5

5049OVERELSE45066781 NOOP;

5050 JMP ENDS10924980730

5051ENDS10924980730 NOOP

5052 JMP ENDS14045294310

5053ENDS14045294310 NOOP

5054OVERELSE421394625 NOOP;

5055 JMP OVERELSE1943815822

5056EL1729278657 CLFZ; tutu man man

5057 LDR R0 #240;

5058 LDR R1 $f5

5059 AND R1 R1 #$FF0F

5060 OR R1 R1 R0

5061 STR R1 $f5; 5

5062OVERELSE1943815822 NOOP;

5063 JMP ENDS13983734480

5064ENDS13983734480 NOOP

5065 JMP ENDS14168413950

5066ENDS14168413950 NOOP

5067 JMP ENDS14103006640

5068ENDS14103006640 NOOP

5069 JMP ENDS14287686110

5070ENDS14287686110 NOOP

5071 JMP ENDS13660545410

5072ENDS13660545410 NOOP

5073 JMP ENDS13841377390

5074ENDS13841377390 NOOP

5075 JMP ENDS12787165400

5076ENDS12787165400 NOOP

5077OVERELSE1565928000 NOOP;

5078 JMP OVERELSE786514705

5079else1388551224 NOOP

5080 LDR R0 #240;

5081 LDR R1 $f5

5082 AND R1 R1 #$FF0F

5083 OR R1 R1 R0

5084 STR R1 $f5; 5

5085OVERELSE786514705 NOOP

5086 JMP ENDS12971844870

5087ENDS12971844870 NOOP

5088 JMP ENDS12891047610

5089case551102726 NOOP;

5090 ESL #91

5091 LDR R0 #1;

5092 STR R0 $144;

5093 ESL #75

5094 LDR R0 #0;

5095 STR R0 $144;

5096 LDR R0 #0;

5097 STR R0 $147;

5098 LDR R0 #22277;

5099 ADD R1 R6 #2

5100 STR R1 #0;

5101 SENDATA R0;

5102 SUB R4 #65535;

5103 SZ TOQSTART1626553053;

5104 ADD R4 R4 #1;

5105 JMP QS1626553053;

5106TOQSTART1626553053 LDR R4 #65528;

5107QS1626553053 CLFZ;

5108 LDR R0 #TEN1608085106

5109 STR R0 $f0

5110TEN1608085106 NOOP;

5111 LDR R0 $5

5112 SUB R0 #0

5113 SZ EL1928827530

5114IF1928827530 CLFZ

5115 ESL #77

5116 LDR R0 #0;

5117 STR R0 $149;

5118 LDR R0 #0;

5119 STR R0 $14b;

5120 LDR R0 #22533;

5121 ADD R1 R6 #2

5122 STR R1 #0;

5123 SENDATA R0;

5124 SUB R4 #65535;

5125 SZ TOQSTART1614625837;

5126 ADD R4 R4 #1;

5127 JMP QS1614625837;

5128TOQSTART1614625837 LDR R4 #65528;

5129QS1614625837 CLFZ;

5130 LDR R0 #TEN1596157890

5131 STR R0 $f0

5132TEN1596157890 NOOP;

5133 LDR R0 $5

5134 SUB R0 #0

5135 SZ EL905856279

5136IF905856279 CLFZ

5137 ESL #79

5138 LDR R0 #0;

5139 STR R0 $14d;

5140 LDR R0 #22789

5141 ADD R1 R6 #2

5142 STR R1 #0;

5143 SENDATA R0;

5144 SUB R4 #65535;

5145 SZ TOQSTART1601159625;

5146 ADD R4 R4 #1;

5147 JMP QS1601159625;

5148TOQSTART1601159625 LDR R4 #65528;

5149QS1601159625 CLFZ; data man man

5150 LDR R0 #TEN1582691678

5151 STR R0 $f0

5152TEN1582691678 NOOP;

5153 LDR R0 $5

5154 SUB R0 #0

5155 SZ EL881398216

5156IF881398216 CLFZ; titi man man

5157 LDR R0 $5

5158 AND R0 R0 #$0002

5159 PRESENT R0 else1588847661

5160 ESL #80

5161 LDR R0 #32;

5162 LDR R1 $f5

5163 AND R1 R1 #$FF0F

5164 OR R1 R1 R0

5165 STR R1 $f5; 5

5166 ESL #89

5167 LDR R0 $e7 ; loading from mem

5168 OR R0 R0 #512 ;loading the emit signal in

5169 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

5170 ESL #89

5171 LDR R0 $e7 ; loading from mem

5172 OR R0 R0 #512 ;loading the emit signal in

5173 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

5174 LDR R0 $eb; loading the right lock place packet

5175 OR R0 R0 #8192; making the app place high for lock release

5176 STR R0 $eb; putting the thing back im mem

5177 ESL #91

5178 LDR R0 #1;

5179 STR R0 $144;

5180 LDR R0 #TI1275277307; loaded the case label into the regiser

5181 STR R0 $f0; stored in memory

5182TI1275277307 NOOP

5183 LDR R0 #16;

5184 LDR R1 $f5

5185 AND R1 R1 #$FF0F

5186 OR R1 R1 R0

5187 STR R1 $f5; 5

5188 JMP OVERELSE426485816

5189else1588847661 NOOP

5190 LDR R0 #0;

5191 STR R0 $14f;

5192 LDR R0 #0;

5193 STR R0 $151;

5194 LDR R0 #0;

5195 STR R0 $153;

5196 LDR R0 #23045

5197 ADD R1 R6 #2

5198 STR R1 #0;

5199 SENDATA R0;

5200 SUB R4 #65535;

5201 SZ TOQSTART1281433289;

5202 ADD R4 R4 #1;

5203 JMP QS1281433289;

5204TOQSTART1281433289 LDR R4 #65528;

5205QS1281433289 CLFZ; data man man

5206 LDR R0 #TEN1262965342

5207 STR R0 $f0

5208TEN1262965342 NOOP;

5209 LDR R0 $5

5210 SUB R0 #0

5211 SZ EL983593466

5212IF983593466 CLFZ; titi man man

5213 LDR R0 $5

5214 AND R0 R0 #$0002

5215 PRESENT R0 else1258733104

5216 LDR R0 #23813;

5217 ADD R1 R6 #2

5218 STR R1 #0;

5219 SENDATA R0;

5220 SUB R4 #65535;

5221 SZ TOQSTART1264889087;

5222 ADD R4 R4 #1;

5223 JMP QS1264889087;

5224TOQSTART1264889087 LDR R4 #65528;

5225QS1264889087 CLFZ;

5226 LDR R0 #TEN1246805888

5227 STR R0 $f0

5228TEN1246805888 NOOP;

5229 LDR R0 $5

5230 SUB R0 #0

5231 SZ EL302118468

5232IF302118468 CLFZ

5233 ESL #89

5234 LDR R0 $e7 ; loading from mem

5235 OR R0 R0 #512 ;loading the emit signal in

5236 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

5237 LDR R0 $eb; loading the right lock place packet

5238 OR R0 R0 #8192; making the app place high for lock release

5239 STR R0 $eb; putting the thing back im mem

5240 ESL #91

5241 LDR R0 #1;

5242 STR R0 $144;

5243 LDR R0 #TI1252961871; loaded the case label into the regiser

5244 STR R0 $f0; stored in memory

5245TI1252961871 NOOP

5246 LDR R0 #16;

5247 LDR R1 $f5

5248 AND R1 R1 #$FF0F

5249 OR R1 R1 R0

5250 STR R1 $f5; 5

5251 JMP OVERELSE79975450

5252EL302118468 CLFZ;

5253 LDR R0 #240;

5254 LDR R1 $f5

5255 AND R1 R1 #$FF0F

5256 OR R1 R1 R0

5257 STR R1 $f5; 5

5258OVERELSE79975450 NOOP;

5259 JMP OVERELSE1158165019

5260else1258733104 NOOP

5261 LDR R0 #0;

5262 STR R0 $155;

5263 LDR R0 #0;

5264 STR R0 $157;

5265 LDR R0 #23301

5266 ADD R1 R6 #2

5267 STR R1 #0;

5268 SENDATA R0;

5269 SUB R4 #65535;

5270 SZ TOQSTART1234493924;

5271 ADD R4 R4 #1;

5272 JMP QS1234493924;

5273TOQSTART1234493924 LDR R4 #65528;

5274QS1234493924 CLFZ; data man man

5275 LDR R0 #TEN1239495659

5276 STR R0 $f0

5277TEN1239495659 NOOP;

5278 LDR R0 $5

5279 SUB R0 #0

5280 SZ EL781488931

5281IF781488931 CLFZ; titi man man

5282 LDR R0 $5

5283 AND R0 R0 #$0002

5284 PRESENT R0 else1319523431

5285 LDR R0 #23557;

5286 ADD R1 R6 #2

5287 STR R1 #0;

5288 SENDATA R0;

5289 SUB R4 #65535;

5290 SZ TOQSTART1325679413;

5291 ADD R4 R4 #1;

5292 JMP QS1325679413;

5293TOQSTART1325679413 LDR R4 #65528;

5294QS1325679413 CLFZ;

5295 LDR R0 #TEN1307596215

5296 STR R0 $f0

5297TEN1307596215 NOOP;

5298 LDR R0 $5

5299 SUB R0 #0

5300 SZ EL1875818101

5301IF1875818101 CLFZ

5302 ESL #89

5303 LDR R0 $e7 ; loading from mem

5304 OR R0 R0 #512 ;loading the emit signal in

5305 STR R0 $e7; emitted signal packet\_\_1374142450 in mem

5306 LDR R0 $eb; loading the right lock place packet

5307 OR R0 R0 #8192; making the app place high for lock release

5308 STR R0 $eb; putting the thing back im mem

5309 ESL #91

5310 LDR R0 #1;

5311 STR R0 $144;

5312 LDR R0 #TI1313752197; loaded the case label into the regiser

5313 STR R0 $f0; stored in memory

5314TI1313752197 NOOP

5315 LDR R0 #16;

5316 LDR R1 $f5

5317 AND R1 R1 #$FF0F

5318 OR R1 R1 R0

5319 STR R1 $f5; 5

5320 JMP OVERELSE830494860

5321EL1875818101 CLFZ;

5322 LDR R0 #240;

5323 LDR R1 $f5

5324 AND R1 R1 #$FF0F

5325 OR R1 R1 R0

5326 STR R1 $f5; 5

5327OVERELSE830494860 NOOP;

5328 JMP OVERELSE958865708

5329else1319523431 NOOP

5330 LDR R0 $eb; loading the right lock place packet

5331 OR R0 R0 #8192; making the app place high for lock release

5332 STR R0 $eb; putting the thing back im mem

5333 LDR R0 #0;

5334 STR R0 $159;

5335 LDR R0 #TI1284896030; loaded the case label into the regiser

5336 STR R0 $f0; stored in memory

5337TI1284896030 NOOP

5338 LDR R0 #16;

5339 LDR R1 $f5

5340 AND R1 R1 #$FF0F

5341 OR R1 R1 R0

5342 STR R1 $f5; 5

5343OVERELSE958865708 NOOP;

5344 JMP OVERELSE870687770

5345EL781488931 CLFZ; tutu man man

5346 LDR R0 #240;

5347 LDR R1 $f5

5348 AND R1 R1 #$FF0F

5349 OR R1 R1 R0

5350 STR R1 $f5; 5

5351OVERELSE870687770 NOOP;

5352OVERELSE1158165019 NOOP;

5353 JMP OVERELSE599085141

5354EL983593466 CLFZ; tutu man man

5355 LDR R0 #240;

5356 LDR R1 $f5

5357 AND R1 R1 #$FF0F

5358 OR R1 R1 R0

5359 STR R1 $f5; 5

5360OVERELSE599085141 NOOP;

5361OVERELSE426485816 NOOP;

5362 JMP OVERELSE132622945

5363EL881398216 CLFZ; tutu man man

5364 LDR R0 #240;

5365 LDR R1 $f5

5366 AND R1 R1 #$FF0F

5367 OR R1 R1 R0

5368 STR R1 $f5; 5

5369OVERELSE132622945 NOOP;

5370 JMP OVERELSE150613528

5371EL905856279 CLFZ;

5372 LDR R0 #240;

5373 LDR R1 $f5

5374 AND R1 R1 #$FF0F

5375 OR R1 R1 R0

5376 STR R1 $f5; 5

5377OVERELSE150613528 NOOP;

5378 JMP OVERELSE948366746

5379EL1928827530 CLFZ;

5380 LDR R0 #240;

5381 LDR R1 $f5

5382 AND R1 R1 #$FF0F

5383 OR R1 R1 R0

5384 STR R1 $f5; 5

5385OVERELSE948366746 NOOP;

5386 JMP ENDS12891047611

5387ENDS12891047610 NOOP

5388ENDS12891047611 NOOP

5389 JMP ENDS10801861080

5390ENDS10801861080 NOOP

5391 JMP ENDS9697631731

5392ENDS9697631730 NOOP

5393ENDS9697631731 NOOP

5394 LDR R14 #40;

5395 LDR R0 $f1; loading the address from the mem

5396 SUB R0 #0; checking if this is actually set

5397 SZ CONT1284896030

5398 JMP R0

5399CONT1284896030 LDR R0 #case1291436761 ;

5400 STR R0 $15c ;

5401 LDR R0 #case590106167 ;

5402 STR R0 $15d ;

5403 LDR R0 #347

5404 SWITCH R1 R0 ;

5405case1291436761 NOOP;

5406 LDR R0 $ec; loading the right lock place crc\_ok

5407 OR R0 R0 #4096; making the app place high for lock release

5408 STR R0 $ec; putting the thing back im mem

5409 LDR R0 #TI1357998321; loaded the case label into the regiser

5410 STR R0 $f1; stored in memory

5411TI1357998321 NOOP

5412 LDR R0 #0;

5413 LDR R1 $f5

5414 AND R1 R1 #$FFF0

5415 OR R1 R1 R0

5416 STR R1 $f5; 8

5417 JMP ENDS3387678360

5418case590106167 NOOP;

5419 LDR R0 #case1364154303 ;

5420 STR R0 $15f ;

5421 LDR R0 #350

5422 SWITCH R1 R0 ;

5423case1364154303 NOOP;

5424 ESL #97

5425 LDR R0 #case1352227087 ;

5426 STR R0 $161 ;

5427 LDR R0 #case1526924559 ;

5428 STR R0 $162 ;

5429 LDR R0 #352

5430 SWITCH R1 R0 ;

5431case1352227087 NOOP;

5432 ESL #98

5433 LDR R0 #case1421866638 ;

5434 STR R0 $164 ;

5435 LDR R0 #355

5436 SWITCH R1 R0 ;

5437case1421866638 NOOP;

5438 LDR R0 #case1409939422; loading the right label into register

5439 STR R0 $f1; loading into mem

5440case1409939422 NOOP

5441 LDR R0 $e9; loading the value into register

5442 AND R0 R0 #16384; getting the right value into R0

5443 SUB R0 #16384;

5444 SZ PRESE406111676;

5445 JMP else2930005;

5446PRESE406111676 CLFZ;Jump to the right place if this thing is not presentres2\_\_1411847842

5447 LDR R0 $e7 ;Loading the register which has this signal

5448 AND R0 R0 #64 ;Got the exact signal

5449 PRESENT R0 else1416095404 ;checking if the signal is present res2\_\_1411847842

5450 LDR R0 $ec; loading the right lock place crc\_ok

5451 OR R0 R0 #4096; making the app place high for lock release

5452 STR R0 $ec; putting the thing back im mem

5453 ESL #105

5454 LDR R0 #1;

5455 STR R0 $160;

5456 LDR R0 #TI1396473211; loaded the case label into the regiser

5457 STR R0 $f1; stored in memory

5458TI1396473211 NOOP

5459 LDR R0 #1;

5460 LDR R1 $f5

5461 AND R1 R1 #$FFF0

5462 OR R1 R1 R0

5463 STR R1 $f5; 8

5464 JMP OVERELSE1304954026

5465else1416095404 NOOP

5466 LDR R0 #case1402629193 ;

5467 STR R0 $166 ;

5468 LDR R0 #357

5469 SWITCH R1 R0 ;

5470case1402629193 NOOP;

5471 ESL #98

5472 LDR R0 #case1390701977 ;

5473 STR R0 $168 ;

5474 LDR R0 #359

5475 SWITCH R1 R0 ;

5476case1390701977 NOOP;

5477 LDR R0 #case1636132944; loading the right label into register

5478 STR R0 $f1; loading into mem

5479case1636132944 NOOP

5480 LDR R0 $eb; loading the value into register

5481 AND R0 R0 #8192; getting the right value into R0

5482 SUB R0 #8192;

5483 SZ PRESE1058945978;

5484 JMP else1193554151;

5485PRESE1058945978 CLFZ;Jump to the right place if this thing is not presentpacket\_\_1374142450

5486 LDR R0 $e7 ;Loading the register which has this signal

5487 AND R0 R0 #512 ;Got the exact signal

5488 PRESENT R0 else1652677147 ;checking if the signal is present packet\_\_1374142450

5489 LDR R0 #24070;

5490 ADD R1 R6 #3

5491 STR R1 #0;

5492 SENDATA R0;

5493 SUB R4 #65535;

5494 SZ TOQSTART1646521165;

5495 ADD R4 R4 #1;

5496 JMP QS1646521165;

5497TOQSTART1646521165 LDR R4 #65528;

5498QS1646521165 CLFZ;

5499 LDR R0 #TEN1664604363

5500 STR R0 $f1

5501TEN1664604363 NOOP;

5502 LDR R0 $6

5503 SUB R0 #0

5504 SZ EL1587041450

5505IF1587041450 CLFZ

5506 LDR R0 $e7 ; loading from mem

5507 OR R0 R0 #16 ;loading the emit signal in

5508 STR R0 $e7; emitted signal crc\_ok\_\_1335667560 in mem

5509 LDR R0 #24326;

5510 ADD R1 R6 #3

5511 STR R1 #0;

5512 SENDATA R0;

5513 SUB R4 #65535;

5514 SZ TOQSTART1658448380;

5515 ADD R4 R4 #1;

5516 JMP QS1658448380;

5517TOQSTART1658448380 LDR R4 #65528;

5518QS1658448380 CLFZ;

5519 LDR R0 #TEN1676916328

5520 STR R0 $f1

5521TEN1676916328 NOOP;

5522 LDR R0 $6

5523 SUB R0 #0

5524 SZ EL1028726613

5525IF1028726613 CLFZ

5526 LDR R0 $ec; loading the right lock place crc\_ok

5527 OR R0 R0 #4096; making the app place high for lock release

5528 STR R0 $ec; putting the thing back im mem

5529 ESL #105

5530 LDR R0 #1;

5531 STR R0 $160;

5532 LDR R0 #TI1671914592; loaded the case label into the regiser

5533 STR R0 $f1; stored in memory

5534TI1671914592 NOOP

5535 LDR R0 #1;

5536 LDR R1 $f5

5537 AND R1 R1 #$FFF0

5538 OR R1 R1 R0

5539 STR R1 $f5; 8

5540 JMP OVERELSE1180467759

5541EL1028726613 CLFZ;

5542 LDR R0 #15;

5543 LDR R1 $f5

5544 AND R1 R1 #$FFF0

5545 OR R1 R1 R0

5546 STR R1 $f5; 8

5547OVERELSE1180467759 NOOP;

5548 JMP OVERELSE1287331626

5549EL1587041450 CLFZ;

5550 LDR R0 #15;

5551 LDR R1 $f5

5552 AND R1 R1 #$FFF0

5553 OR R1 R1 R0

5554 STR R1 $f5; 8

5555OVERELSE1287331626 NOOP;

5556 JMP OVERELSE1247642360

5557else1652677147 NOOP

5558 LDR R0 $ec; loading the right lock place crc\_ok

5559 OR R0 R0 #4096; making the app place high for lock release

5560 STR R0 $ec; putting the thing back im mem

5561 ESL #98

5562 LDR R0 #TI1591886821; loaded the case label into the regiser

5563 STR R0 $f1; stored in memory

5564TI1591886821 NOOP

5565 LDR R0 #1;

5566 LDR R1 $f5

5567 AND R1 R1 #$FFF0

5568 OR R1 R1 R0

5569 STR R1 $f5; 8

5570OVERELSE1247642360 NOOP;

5571 JMP OVERELSE1028610469

5572else1193554151 NOOP

5573 LDR R0 #15;

5574 LDR R1 $f5

5575 AND R1 R1 #$FFF0

5576 OR R1 R1 R0

5577 STR R1 $f5; 8

5578OVERELSE1028610469 NOOP

5579 JMP ENDS9882311210

5580ENDS9882311210 NOOP

5581 JMP ENDS9820751380

5582ENDS9820751380 NOOP

5583OVERELSE1304954026 NOOP;

5584 JMP OVERELSE487559613

5585else2930005 NOOP

5586 LDR R0 #15;

5587 LDR R1 $f5

5588 AND R1 R1 #$FFF0

5589 OR R1 R1 R0

5590 STR R1 $f5; 8

5591OVERELSE487559613 NOOP

5592 JMP ENDS10001583370

5593ENDS10001583370 NOOP

5594 JMP ENDS9940023540

5595case1526924559 NOOP;

5596 LDR R0 $ec; loading the right lock place crc\_ok

5597 OR R0 R0 #4096; making the app place high for lock release

5598 STR R0 $ec; putting the thing back im mem

5599 ESL #105

5600 LDR R0 #1;

5601 STR R0 $160;

5602 ESL #98

5603 LDR R0 #0;

5604 STR R0 $160;

5605 LDR R0 #0;

5606 STR R0 $163;

5607 ESL #98

5608 LDR R0 #0;

5609 STR R0 $165;

5610 LDR R0 #0;

5611 STR R0 $167;

5612 ESL #98

5613 LDR R0 #TI1597658054; loaded the case label into the regiser

5614 STR R0 $f1; stored in memory

5615TI1597658054 NOOP

5616 LDR R0 #1;

5617 LDR R1 $f5

5618 AND R1 R1 #$FFF0

5619 OR R1 R1 R0

5620 STR R1 $f5; 8

5621 JMP ENDS9940023541

5622ENDS9940023540 NOOP

5623ENDS9940023541 NOOP

5624 JMP ENDS3572357830

5625ENDS3572357830 NOOP

5626 JMP ENDS3387678361

5627ENDS3387678360 NOOP

5628ENDS3387678361 NOOP

5629 LDR R14 #41;

5630 LDR R0 $f2; loading the address from the mem

5631 SUB R0 #0; checking if this is actually set

5632 SZ CONT1597658054

5633 JMP R0

5634CONT1597658054 LDR R0 #case1601890292 ;

5635 STR R0 $16a ;

5636 LDR R0 #case952466307 ;

5637 STR R0 $16b ;

5638 LDR R0 #361

5639 SWITCH R1 R0 ;

5640case1601890292 NOOP;

5641 LDR R0 #TI1632285455; loaded the case label into the regiser

5642 STR R0 $f2; stored in memory

5643TI1632285455 NOOP

5644 LDR R0 #0;

5645 LDR R1 $f6

5646 AND R1 R1 #$0FFF

5647 OR R1 R1 R0

5648 STR R1 $f6; 15

5649 JMP ENDS12017596190

5650case952466307 NOOP;

5651 LDR R0 #case1626129473 ;

5652 STR R0 $16d ;

5653 LDR R0 #364

5654 SWITCH R1 R0 ;

5655case1626129473 NOOP;

5656 ESL #114

5657 LDR R0 #case1541099966 ;

5658 STR R0 $16f ;

5659 LDR R0 #case1164564419 ;

5660 STR R0 $170 ;

5661 LDR R0 #366

5662 SWITCH R1 R0 ;

5663case1541099966 NOOP;

5664 ESL #115

5665 LDR R0 #case1571495129 ;

5666 STR R0 $172 ;

5667 LDR R0 #369

5668 SWITCH R1 R0 ;

5669case1571495129 NOOP;

5670 LDR R0 #case1581883349; loading the right label into register

5671 STR R0 $f2; loading into mem

5672case1581883349 NOOP

5673 LDR R0 $ea; loading the value into register

5674 AND R0 R0 #16384; getting the right value into R0

5675 SUB R0 #16384;

5676 SZ PRESE654187723;

5677 JMP else908122113;

5678PRESE654187723 CLFZ;Jump to the right place if this thing is not presentres3\_\_1325279340

5679 LDR R0 $e7 ;Loading the register which has this signal

5680 AND R0 R0 #32 ;Got the exact signal

5681 PRESENT R0 else1575727367 ;checking if the signal is present res3\_\_1325279340

5682 ESL #142

5683 LDR R0 #1;

5684 STR R0 $16e;

5685 LDR R0 #TI1495699596; loaded the case label into the regiser

5686 STR R0 $f2; stored in memory

5687TI1495699596 NOOP

5688 LDR R0 #4096;

5689 LDR R1 $f6

5690 AND R1 R1 #$0FFF

5691 OR R1 R1 R0

5692 STR R1 $f6; 15

5693 JMP OVERELSE2078961152

5694else1575727367 NOOP

5695 LDR R0 #case1489158864 ;

5696 STR R0 $174 ;

5697 LDR R0 #case207030679 ;

5698 STR R0 $175 ;

5699 LDR R0 #371

5700 SWITCH R1 R0 ;

5701case1489158864 NOOP;

5702 ESL #115

5703 LDR R0 #case1521093023 ;

5704 STR R0 $177 ;

5705 LDR R0 #374

5706 SWITCH R1 R0 ;

5707case1521093023 NOOP;

5708 LDR R0 #case1533020239; loading the right label into register

5709 STR R0 $f2; loading into mem

5710case1533020239 NOOP

5711 LDR R0 $eb; loading the value into register

5712 AND R0 R0 #8192; getting the right value into R0

5713 SUB R0 #8192;

5714 SZ PRESE1355262860;

5715 JMP else1625775196;

5716PRESE1355262860 CLFZ;Jump to the right place if this thing is not presentpacket\_\_1374142450

5717 LDR R0 $e7 ;Loading the register which has this signal

5718 AND R0 R0 #512 ;Got the exact signal

5719 PRESENT R0 else1526864257 ;checking if the signal is present packet\_\_1374142450

5720 ESL #117

5721 LDR R0 #1;

5722 STR R0 $173;

5723PNODE1840819359 NOOP

5724 LDR R0 #PNODE1840819359; loading the case address into register

5725 STR R0 $f2; storing the address in the memory

5726 LDR R14 #42;

5727 LDR R0 $f3; loading the address from the mem

5728 SUB R0 #0; checking if this is actually set

5729 SZ CONT1832739632

5730 JMP R0

5731CONT1832739632 LDR R0 #1;

5732 STR R0 $178;

5733 ESL #118

5734 LDR R0 #0;

5735 STR R0 $17b;

5736 LDR R0 #0;

5737 STR R0 $17d;

5738 LDR R0 #24584;

5739 ADD R1 R6 #5

5740 STR R1 #0;

5741 SENDATA R0;

5742 SUB R4 #65535;

5743 SZ TOQSTART1851207579;

5744 ADD R4 R4 #1;

5745 JMP QS1851207579;

5746TOQSTART1851207579 LDR R4 #65528;

5747QS1851207579 CLFZ;

5748 LDR R0 #TEN1844666848

5749 STR R0 $f3

5750TEN1844666848 NOOP;

5751 LDR R0 $8

5752 SUB R0 #0

5753 SZ EL1986139197

5754IF1986139197 CLFZ

5755 ESL #121

5756 LDR R0 #0;

5757 STR R0 $17f;

5758 LDR R0 #TI1863134795; loaded the case label into the regiser

5759 STR R0 $f3; stored in memory

5760TI1863134795 NOOP

5761 LDR R0 #256;

5762 LDR R1 $f6

5763 AND R1 R1 #$F0FF

5764 OR R1 R1 R0

5765 STR R1 $f6; 11

5766 JMP OVERELSE1518237830

5767EL1986139197 CLFZ;

5768 LDR R0 #3840;

5769 LDR R1 $f6

5770 AND R1 R1 #$F0FF

5771 OR R1 R1 R0

5772 STR R1 $f6; 11

5773OVERELSE1518237830 NOOP;

5774 LDR R14 #43;

5775 LDR R0 $f4; loading the address from the mem

5776 SUB R0 #0; checking if this is actually set

5777 SZ CONT1863134795

5778 JMP R0

5779CONT1863134795 LDR R0 #1;

5780 STR R0 $184;

5781 ESL #128

5782 LDR R0 #0;

5783 STR R0 $187;

5784 LDR R0 #0;

5785 STR R0 $18a;

5786 ESL #128

5787 LDR R0 #TI1856978813; loaded the case label into the regiser

5788 STR R0 $f4; stored in memory

5789TI1856978813 NOOP

5790 LDR R0 #16;

5791 LDR R1 $f6

5792 AND R1 R1 #$FF0F

5793 OR R1 R1 R0

5794 STR R1 $f6; 12

5795 LDR R1 #0 ;this will hold the terminate nodes

5796 LDR R0 $f6; loading the value from mem pipi

5797 AND R0 R0 #3840; making it right

5798 OR R1 R1 R0 ;getting the terminate node into R1 11

5799 LDR R0 $f6; loading the value from mem pipi

5800 AND R0 R0 #240; making it right

5801 OR R1 R1 R0 ;getting the terminate node into R1 12

5802 STR R1 $1d3; loading the memory in the place

5803 LDR R0 #N65608306801; loading the address into register

5804 STR R0 $1b1; loading its mem

5805 LDR R0 #N6560830681123456 ;loading the address in register

5806 STR R0 $1bf; loading it in mem

5807 LDR R0 #431 ;loaded the address of the joinnode in register

5808 STR R0 $1d4; loaded it into the endPointer memory

5809 LDR R0 #$1d3

5810 LDR R2 #0;

5811 LDR R1 #0

5812CHKENDLAB656083068 LDR R0 #$1d3

5813 ADD R0 R0 R1;

5814 LDR R0 R0;

5815 CHKEND R2 R0;

5816 SUB R1 #0

5817 SZ ENDOVER656083068

5818 ADD R1 R1 #1

5819 JMP CHKENDLAB656083068

5820ENDOVER656083068 CLFZ

5821 LDR R0 #$1d3

5822 ADD R1 R1 #1;

5823 ADD R0 R0 R1

5824 LDR R0 R0

5825 ADD R2 R2 #1

5826 ADD R0 R0 R2;

5827 LDR R0 R0

5828 JMP R0; adding stuff

5829N65608306801 NOOP

5830 LDR R0 #TI1856978813; loaded the case label into the regiser

5831 STR R0 $f2; stored in memory

5832TI1856978813 NOOP

5833 LDR R0 #4096;

5834 LDR R1 $f6

5835 AND R1 R1 #$0FFF

5836 OR R1 R1 R0

5837 STR R1 $f6; 15

5838 JMP DUMMY656083068;

5839N6560830681123456 NOOP

5840 LDR R0 #61440;

5841 LDR R1 $f6

5842 AND R1 R1 #$0FFF

5843 OR R1 R1 R0

5844 STR R1 $f6; 15

5845 JMP DUMMY656083068;

5846DUMMY656083068

5847 JMP OVERELSE1361308069

5848else1526864257 NOOP

5849 ESL #115

5850 LDR R0 #TI1876601007; loaded the case label into the regiser

5851 STR R0 $f2; stored in memory

5852TI1876601007 NOOP

5853 LDR R0 #4096;

5854 LDR R1 $f6

5855 AND R1 R1 #$0FFF

5856 OR R1 R1 R0

5857 STR R1 $f6; 15

5858OVERELSE1361308069 NOOP;

5859 JMP OVERELSE288450766

5860else1625775196 NOOP

5861 LDR R0 #61440;

5862 LDR R1 $f6

5863 AND R1 R1 #$0FFF

5864 OR R1 R1 R0

5865 STR R1 $f6; 15

5866OVERELSE288450766 NOOP

5867 JMP ENDS3468475630

5868ENDS3468475630 NOOP

5869 JMP ENDS5142133340

5870case207030679 NOOP;

5871 ESL #117

5872PNODE1790417253 NOOP

5873 LDR R0 #PNODE1790417253; loading the case address into register

5874 STR R0 $f2; storing the address in the memory

5875 LDR R14 #44;

5876 LDR R0 $f3; loading the address from the mem

5877 SUB R0 #0; checking if this is actually set

5878 SZ CONT1783876522

5879 JMP R0

5880CONT1783876522 LDR R0 #case1802344469 ;

5881 STR R0 $179 ;

5882 LDR R0 #case240423672 ;

5883 STR R0 $17a ;

5884 LDR R0 #376

5885 SWITCH R1 R0 ;

5886case1802344469 NOOP;

5887 LDR R0 #TI1806576707; loaded the case label into the regiser

5888 STR R0 $f3; stored in memory

5889TI1806576707 NOOP

5890 LDR R0 #0;

5891 LDR R1 $f6

5892 AND R1 R1 #$F0FF

5893 OR R1 R1 R0

5894 STR R1 $f6; 11

5895 JMP ENDS4211041000

5896case240423672 NOOP;

5897 LDR R0 #case1824659905 ;

5898 STR R0 $17c ;

5899 LDR R0 #379

5900 SWITCH R1 R0 ;

5901case1824659905 NOOP;

5902 ESL #118

5903 LDR R0 #case1738476152 ;

5904 STR R0 $17e ;

5905 LDR R0 #381

5906 SWITCH R1 R0 ;

5907case1738476152 NOOP;

5908 LDR R0 #case1751942363; loading the right label into register

5909 STR R0 $f3; loading into mem

5910case1751942363 NOOP

5911 LDR R0 $ed; loading the value into register

5912 AND R0 R0 #512; getting the right value into R0

5913 SUB R0 #512;

5914 SZ PRESE917799267;

5915 JMP else1285047763;

5916PRESE917799267 CLFZ;Jump to the right place if this thing is not presentkill\_check\_\_1386069666

5917 LDR R0 $e7 ;Loading the register which has this signal

5918 AND R0 R0 #256 ;Got the exact signal

5919 PRESENT R0 else1745401632 ;checking if the signal is present kill\_check\_\_1386069666

5920 ESL #117

5921 LDR R0 #0;

5922 STR R0 $178;

5923 LDR R0 #TI1763869579; loaded the case label into the regiser

5924 STR R0 $f3; stored in memory

5925TI1763869579 NOOP

5926 LDR R0 #0;

5927 LDR R1 $f6

5928 AND R1 R1 #$F0FF

5929 OR R1 R1 R0

5930 STR R1 $f6; 11

5931 JMP OVERELSE546550926

5932else1745401632 NOOP

5933 LDR R0 #case1757713597 ;

5934 STR R0 $180 ;

5935 LDR R0 #case1132648308 ;

5936 STR R0 $181 ;

5937 LDR R0 #case1415080700 ;

5938 STR R0 $182 ;

5939 LDR R0 #case402027627 ;

5940 STR R0 $183 ;

5941 LDR R0 #383

5942 SWITCH R1 R0 ;

5943case1757713597 NOOP;

5944 ESL #121

5945 LDR R0 #0;

5946 STR R0 $17f;

5947 ESL #122

5948 LDR R0 #1;

5949 STR R0 $17f;

5950 LDR R0 #TI1687689297; loaded the case label into the regiser

5951 STR R0 $f3; stored in memory

5952TI1687689297 NOOP

5953 LDR R0 #256;

5954 LDR R1 $f6

5955 AND R1 R1 #$F0FF

5956 OR R1 R1 R0

5957 STR R1 $f6; 11

5958 JMP ENDS3283796160

5959case1132648308 NOOP;

5960 ESL #122

5961 LDR R0 #1;

5962 STR R0 $17f;

5963 ESL #123

5964 LDR R0 #2;

5965 STR R0 $17f;

5966 LDR R0 #TI1707311491; loaded the case label into the regiser

5967 STR R0 $f3; stored in memory

5968TI1707311491 NOOP

5969 LDR R0 #256;

5970 LDR R1 $f6

5971 AND R1 R1 #$F0FF

5972 OR R1 R1 R0

5973 STR R1 $f6; 11

5974 JMP ENDS3283796161

5975case1415080700 NOOP;

5976 ESL #123

5977 LDR R0 #2;

5978 STR R0 $17f;

5979 ESL #124

5980 LDR R0 #3;

5981 STR R0 $17f;

5982 LDR R0 #TI1701155508; loaded the case label into the regiser

5983 STR R0 $f3; stored in memory

5984TI1701155508 NOOP

5985 LDR R0 #256;

5986 LDR R1 $f6

5987 AND R1 R1 #$F0FF

5988 OR R1 R1 R0

5989 STR R1 $f6; 11

5990 JMP ENDS3283796162

5991case402027627 NOOP;

5992 ESL #124

5993 LDR R0 #3;

5994 STR R0 $17f;

5995 ESL #117

5996 LDR R0 #0;

5997 STR R0 $178;

5998 LDR R0 #TI1719623456; loaded the case label into the regiser

5999 STR R0 $f3; stored in memory

6000TI1719623456 NOOP

6001 LDR R0 #0;

6002 LDR R1 $f6

6003 AND R1 R1 #$F0FF

6004 OR R1 R1 R0

6005 STR R1 $f6; 11

6006 JMP ENDS3283796163

6007ENDS3283796160 NOOP

6008ENDS3283796161 NOOP

6009ENDS3283796162 NOOP

6010ENDS3283796163 NOOP

6011OVERELSE546550926 NOOP;

6012 JMP OVERELSE472545489

6013else1285047763 NOOP

6014 LDR R0 #3840;

6015 LDR R1 $f6

6016 AND R1 R1 #$F0FF

6017 OR R1 R1 R0

6018 STR R1 $f6; 11

6019OVERELSE472545489 NOOP

6020 JMP ENDS3349203470

6021ENDS3349203470 NOOP

6022 JMP ENDS4149481180

6023ENDS4149481180 NOOP

6024 JMP ENDS4211041001

6025ENDS4211041000 NOOP

6026ENDS4211041001 NOOP

6027 LDR R14 #45;

6028 LDR R0 $f4; loading the address from the mem

6029 SUB R0 #0; checking if this is actually set

6030 SZ CONT1713082724

6031 JMP R0

6032CONT1713082724 LDR R0 #case1731550671 ;

6033 STR R0 $185 ;

6034 LDR R0 #case1898779151 ;

6035 STR R0 $186 ;

6036 LDR R0 #388

6037 SWITCH R1 R0 ;

6038case1731550671 NOOP;

6039 LDR R0 $ed; loading the right lock place kill\_check

6040 OR R0 R0 #512; making the app place high for lock release

6041 STR R0 $ed; putting the thing back im mem

6042 LDR R0 #TI2031270065; loaded the case label into the regiser

6043 STR R0 $f4; stored in memory

6044TI2031270065 NOOP

6045 LDR R0 #0;

6046 LDR R1 $f6

6047 AND R1 R1 #$FF0F

6048 OR R1 R1 R0

6049 STR R1 $f6; 12

6050 JMP ENDS3772427260

6051case1898779151 NOOP;

6052 LDR R0 #case2049353263 ;

6053 STR R0 $188 ;

6054 LDR R0 #case1448668114 ;

6055 STR R0 $189 ;

6056 LDR R0 #391

6057 SWITCH R1 R0 ;

6058case2049353263 NOOP;

6059 ESL #128

6060 LDR R0 #case2056663492 ;

6061 STR R0 $18b ;

6062 LDR R0 #394

6063 SWITCH R1 R0 ;

6064case2056663492 NOOP;

6065 LDR R0 #case2068975457; loading the right label into register

6066 STR R0 $f4; loading into mem

6067case2068975457 NOOP

6068 LDR R0 $ec; loading the value into register

6069 AND R0 R0 #4096; getting the right value into R0

6070 SUB R0 #4096;

6071 SZ PRESE2108359668;

6072 JMP else1646470412;

6073PRESE2108359668 CLFZ;Jump to the right place if this thing is not presentcrc\_ok\_\_1335667560

6074 LDR R0 $e7 ;Loading the register which has this signal

6075 AND R0 R0 #16 ;Got the exact signal

6076 PRESENT R0 else1988562937 ;checking if the signal is present crc\_ok\_\_1335667560

6077 LDR R0 #25097;

6078 ADD R1 R6 #6

6079 STR R1 #0;

6080 SENDATA R0;

6081 SUB R4 #65535;

6082 SZ TOQSTART1982406954;

6083 ADD R4 R4 #1;

6084 JMP QS1982406954;

6085TOQSTART1982406954 LDR R4 #65528;

6086QS1982406954 CLFZ;

6087 LDR R0 #TEN1998951157

6088 STR R0 $f4

6089TEN1998951157 NOOP;

6090 LDR R0 $9

6091 SUB R0 #0

6092 SZ EL1801675091

6093IF1801675091 CLFZ

6094 ESL #132

6095 LDR R0 #1;

6096 STR R0 $187;

6097 LDR R0 #25353

6098 ADD R1 R6 #6

6099 STR R1 #0;

6100 SENDATA R0;

6101 SUB R4 #65535;

6102 SZ TOQSTART1992795175;

6103 ADD R4 R4 #1;

6104 JMP QS1992795175;

6105TOQSTART1992795175 LDR R4 #65528;

6106QS1992795175 CLFZ; data man man

6107 LDR R0 #TEN2011263122

6108 STR R0 $f4

6109TEN2011263122 NOOP;

6110 LDR R0 $9

6111 SUB R0 #0

6112 SZ EL122475858

6113IF122475858 CLFZ; titi man man

6114 LDR R0 $9

6115 AND R0 R0 #$0002

6116 PRESENT R0 else2004722391

6117 ESL #133

6118 LDR R0 $e7 ; loading from mem

6119 OR R0 R0 #256 ;loading the emit signal in

6120 STR R0 $e7; emitted signal kill\_check\_\_1386069666 in mem

6121 LDR R0 $ed; loading the right lock place kill\_check

6122 OR R0 R0 #512; making the app place high for lock release

6123 STR R0 $ed; putting the thing back im mem

6124 ESL #128

6125 LDR R0 #0;

6126 STR R0 $184;

6127 LDR R0 #TI2023190338; loaded the case label into the regiser

6128 STR R0 $f4; stored in memory

6129TI2023190338 NOOP

6130 LDR R0 #0;

6131 LDR R1 $f6

6132 AND R1 R1 #$FF0F

6133 OR R1 R1 R0

6134 STR R1 $f6; 12

6135 JMP OVERELSE1954074546

6136else2004722391 NOOP

6137 LDR R0 $ed; loading the right lock place kill\_check

6138 OR R0 R0 #512; making the app place high for lock release

6139 STR R0 $ed; putting the thing back im mem

6140 ESL #128

6141 LDR R0 #0;

6142 STR R0 $184;

6143 LDR R0 #TI2017034355; loaded the case label into the regiser

6144 STR R0 $f4; stored in memory

6145TI2017034355 NOOP

6146 LDR R0 #0;

6147 LDR R1 $f6

6148 AND R1 R1 #$FF0F

6149 OR R1 R1 R0

6150 STR R1 $f6; 12

6151OVERELSE1954074546 NOOP;

6152 JMP OVERELSE254396483

6153EL122475858 CLFZ; tutu man man

6154 LDR R0 #240;

6155 LDR R1 $f6

6156 AND R1 R1 #$FF0F

6157 OR R1 R1 R0

6158 STR R1 $f6; 12

6159OVERELSE254396483 NOOP;

6160 JMP OVERELSE1572771428

6161EL1801675091 CLFZ;

6162 LDR R0 #240;

6163 LDR R1 $f6

6164 AND R1 R1 #$FF0F

6165 OR R1 R1 R0

6166 STR R1 $f6; 12

6167OVERELSE1572771428 NOOP;

6168 JMP OVERELSE668362612

6169else1988562937 NOOP

6170 LDR R0 $ed; loading the right lock place kill\_check

6171 OR R0 R0 #512; making the app place high for lock release

6172 STR R0 $ed; putting the thing back im mem

6173 ESL #128

6174 LDR R0 #TI1938160831; loaded the case label into the regiser

6175 STR R0 $f4; stored in memory

6176TI1938160831 NOOP

6177 LDR R0 #16;

6178 LDR R1 $f6

6179 AND R1 R1 #$FF0F

6180 OR R1 R1 R0

6181 STR R1 $f6; 12

6182OVERELSE668362612 NOOP;

6183 JMP OVERELSE34738083

6184else1646470412 NOOP

6185 LDR R0 #240;

6186 LDR R1 $f6

6187 AND R1 R1 #$FF0F

6188 OR R1 R1 R0

6189 STR R1 $f6; 12

6190OVERELSE34738083 NOOP

6191 JMP ENDS4014819070

6192ENDS4014819070 NOOP

6193 JMP ENDS4076378890

6194case1448668114 NOOP;

6195 LDR R0 $ed; loading the right lock place kill\_check

6196 OR R0 R0 #512; making the app place high for lock release

6197 STR R0 $ed; putting the thing back im mem

6198 ESL #132

6199 LDR R0 #396

6200 SWITCH R1 R0 ;

6201 JMP ENDS4076378891

6202ENDS4076378890 NOOP

6203ENDS4076378891 NOOP

6204 JMP ENDS3772427261

6205ENDS3772427260 NOOP

6206ENDS3772427261 NOOP

6207 LDR R1 #0 ;this will hold the terminate nodes

6208 LDR R0 $f6; loading the value from mem pipi

6209 AND R0 R0 #3840; making it right

6210 OR R1 R1 R0 ;getting the terminate node into R1 11

6211 LDR R0 $f6; loading the value from mem pipi

6212 AND R0 R0 #240; making it right

6213 OR R1 R1 R0 ;getting the terminate node into R1 12

6214 STR R1 $1d5; loading the memory in the place

6215 LDR R0 #N17007071590

6216 STR R0 $1c1; loaded the case in the memory location

6217 LDR R0 #N170070715901; loading the address into register

6218 STR R0 $1c2; loading its mem

6219 LDR R0 #N17007071591123456 ;loading the address in register

6220 STR R0 $1d0; loading it in mem

6221 LDR R0 #448 ;loaded the address of the joinnode in register

6222 STR R0 $1d6; loaded it into the endPointer memory

6223 LDR R0 #$1d5

6224 LDR R2 #0;

6225 LDR R1 #0

6226CHKENDLAB1700707159 LDR R0 #$1d5

6227 ADD R0 R0 R1;

6228 LDR R0 R0;

6229 CHKEND R2 R0;

6230 SUB R1 #0

6231 SZ ENDOVER1700707159

6232 ADD R1 R1 #1

6233 JMP CHKENDLAB1700707159

6234ENDOVER1700707159 CLFZ

6235 LDR R0 #$1d5

6236 ADD R1 R1 #1;

6237 ADD R0 R0 R1

6238 LDR R0 R0

6239 ADD R2 R2 #1

6240 ADD R0 R0 R2;

6241 LDR R0 R0

6242 JMP R0; adding stuff

6243N170070715901 NOOP

6244 LDR R0 #TI1950088047; loaded the case label into the regiser

6245 STR R0 $f2; stored in memory

6246TI1950088047 NOOP

6247 LDR R0 #4096;

6248 LDR R1 $f6

6249 AND R1 R1 #$0FFF

6250 OR R1 R1 R0

6251 STR R1 $f6; 15

6252 JMP DUMMY1700707159;

6253N17007071591123456 NOOP

6254 LDR R0 #61440;

6255 LDR R1 $f6

6256 AND R1 R1 #$0FFF

6257 OR R1 R1 R0

6258 STR R1 $f6; 15

6259 JMP DUMMY1700707159;

6260N17007071590 NOOP

6261 LDR R0 #24839;

6262 ADD R1 R6 #4

6263 STR R1 #0;

6264 SENDATA R0;

6265 SUB R4 #65535;

6266 SZ TOQSTART1950088047;

6267 ADD R4 R4 #1;

6268 JMP QS1950088047;

6269TOQSTART1950088047 LDR R4 #65528;

6270QS1950088047 CLFZ;

6271 LDR R0 #TEN1943932064

6272 STR R0 $f2

6273TEN1943932064 NOOP;

6274 LDR R0 $7

6275 SUB R0 #0

6276 SZ EL223862023

6277IF223862023 CLFZ

6278 ESL #142

6279 LDR R0 #1;

6280 STR R0 $16e;

6281 LDR R0 #TI1962400012; loaded the case label into the regiser

6282 STR R0 $f2; stored in memory

6283TI1962400012 NOOP

6284 LDR R0 #4096;

6285 LDR R1 $f6

6286 AND R1 R1 #$0FFF

6287 OR R1 R1 R0

6288 STR R1 $f6; 15

6289 JMP OVERELSE1670578848

6290EL223862023 CLFZ;

6291 LDR R0 #61440;

6292 LDR R1 $f6

6293 AND R1 R1 #$0FFF

6294 OR R1 R1 R0

6295 STR R1 $f6; 15

6296OVERELSE1670578848 NOOP;

6297 JMP DUMMY1700707159;

6298DUMMY1700707159

6299 JMP ENDS5142133341

6300ENDS5142133340 NOOP

6301ENDS5142133341 NOOP

6302OVERELSE2078961152 NOOP;

6303 JMP OVERELSE832484917

6304else908122113 NOOP

6305 LDR R0 #61440;

6306 LDR R1 $f6

6307 AND R1 R1 #$0FFF

6308 OR R1 R1 R0

6309 STR R1 $f6; 15

6310OVERELSE832484917 NOOP

6311 JMP ENDS5203693170

6312ENDS5203693170 NOOP

6313 JMP ENDS5038251140

6314case1164564419 NOOP;

6315 ESL #142

6316 LDR R0 #1;

6317 STR R0 $16e;

6318 ESL #115

6319 LDR R0 #0;

6320 STR R0 $16e;

6321 LDR R0 #0;

6322 STR R0 $171;

6323 ESL #115

6324 LDR R0 #0;

6325 STR R0 $173;

6326 LDR R0 #0;

6327 STR R0 $176;

6328 ESL #115

6329 LDR R0 #TI1972788232; loaded the case label into the regiser

6330 STR R0 $f2; stored in memory

6331TI1972788232 NOOP

6332 LDR R0 #4096;

6333 LDR R1 $f6

6334 AND R1 R1 #$0FFF

6335 OR R1 R1 R0

6336 STR R1 $f6; 15

6337 JMP ENDS5038251141

6338ENDS5038251140 NOOP

6339ENDS5038251141 NOOP

6340 JMP ENDS11217318480

6341ENDS11217318480 NOOP

6342 JMP ENDS12017596191

6343ENDS12017596190 NOOP

6344ENDS12017596191 NOOP

6345 LDR R1 #0 ;this will hold the terminate nodes

6346 LDR R0 $f5; loading the value from mem pipi

6347 AND R0 R0 #3840; making it right

6348 OR R1 R1 R0 ;getting the terminate node into R1 2

6349 LDR R0 $f5; loading the value from mem pipi

6350 AND R0 R0 #240; making it right

6351 OR R1 R1 R0 ;getting the terminate node into R1 5

6352 LDR R0 $f5; loading the value from mem pipi

6353 AND R0 R0 #15; making it right

6354 OR R1 R1 R0 ;getting the terminate node into R1 8

6355 LDR R0 $f6; loading the value from mem pipi

6356 AND R0 R0 #61440; making it right

6357 OR R1 R1 R0 ;getting the terminate node into R1 15

6358 STR R1 $1d7; loading the memory in the place

6359 LDR R1 #0 ;loading zeros for the next turn

6360 LDR R0 #N6481866540

6361 STR R0 $19f; loaded the case in the memory location

6362 LDR R0 #N64818665401; loading the address into register

6363 STR R0 $1a0; loading its mem

6364 LDR R0 #N6481866541123456 ;loading the address in register

6365 STR R0 $1ae; loading it in mem

6366 LDR R0 #414 ;loaded the address of the joinnode in register

6367 STR R0 $1d8; loaded it into the endPointer memory

6368 LDR R0 #$1d7

6369 LDR R2 #0;

6370 LDR R1 #0

6371CHKENDLAB648186654 LDR R0 #$1d7

6372 ADD R0 R0 R1;

6373 LDR R0 R0;

6374 CHKEND R2 R0;

6375 SUB R1 #0

6376 SZ ENDOVER648186654

6377 ADD R1 R1 #1

6378 JMP CHKENDLAB648186654

6379ENDOVER648186654 CLFZ

6380 LDR R0 #$1d7

6381 ADD R1 R1 #1;

6382 ADD R0 R0 R1

6383 LDR R0 R0

6384 ADD R2 R2 #1

6385 ADD R0 R0 R2;

6386 LDR R0 R0

6387 JMP R0; adding stuff

6388N64818665401 NOOP

6389 LDR R0 #4096;

6390 LDR R1 $f5

6391 AND R1 R1 #$0FFF

6392 OR R1 R1 R0

6393 STR R1 $f5; 16

6394 LDR R0 #$0;

6395 STR R0 $ee;

6396 JMP END1;

6397 JMP DUMMY648186654;

6398N6481866541123456 NOOP

6399 LDR R0 #61440;

6400 LDR R1 $f5

6401 AND R1 R1 #$0FFF

6402 OR R1 R1 R0

6403 STR R1 $f5; 16

6404 JMP END1;

6405 JMP DUMMY648186654;

6406N6481866540 NOOP

6407 ESL #51

6408 LDR R0 #0;

6409 STR R0 $129;

6410 LDR R0 #0;

6411 LDR R1 $f5

6412 AND R1 R1 #$0FFF

6413 OR R1 R1 R0

6414 STR R1 $f5; 16

6415 LDR R0 #0;

6416 STR R0 $129;

6417 LDR R0 #$0;

6418 STR R0 $ee;

6419 JMP END1;

6420 JMP DUMMY648186654;

6421DUMMY648186654

6422 JMP ENDS12798789700

6423ENDS12798789700 NOOP

6424 JMP ENDS13822221772

6425ENDS13822221770 NOOP

6426ENDS13822221771 NOOP

6427ENDS13822221772 NOOP

6428END1 LDR R0 $b;

6429 SUB R0 #$0;

6430 SZ BEGIN0;

6431 JMP RUN0;

6432 ENDPROG

6433