CTI VT2100 Die Retest Phase 2 Scope, Estimates, Schedule

Version: 21 Oct 05
Ken Freed (kxf)
Cypress Semiconductor Texas, Inc.
17 Cypress Blvd
Round Rock, Texas 78664

CONTENTS

Introduction

This document describes the next set of features which will be implemented as part of the "VT2100 Die Retest" project.

Phase 1 (done)

Was to rebuild the VT2100 code (modules dprc.exe and rdpio.exe) under Borland Turbo C (for a debuggable environment), get it running under Desqview, and implement a simple "retest/retouchdown" retest algorithm.

This work has been completed.

Phase 2 (this document)

Is to:

1. Duplicate the plan file entries and algorithms used at CMI on their VT3300 testers.
   - This means shifting the test head so that a defective die is retested with a different site.
2. Implement and collect retest data pay-back information on one to three VT-KLA testers
3. Get the new code qualified on one to three VT-KLA testers

Phase 3+ (future)

Likely items for a follow on phase include:

- Eollout to more (than one to three) VT-KLA testers
- EG prober support and further qualification
- Items listed under the "Limitations" section of this document

Code Requirements

Test Head Shifting for Retesting

These requirements are based on CMI's implementation as described in:

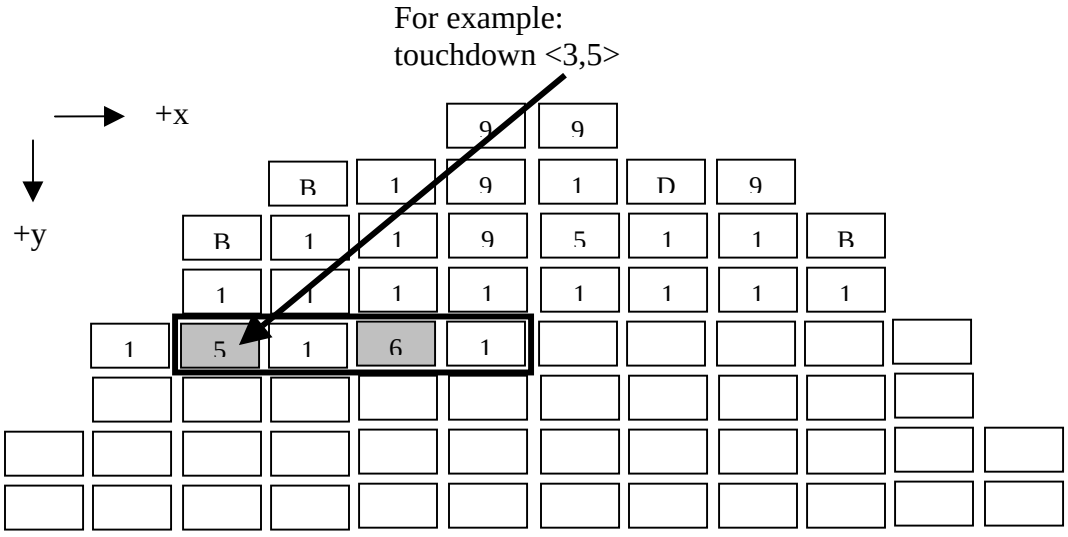
"Cypress Intra-Wafer Retest and Yield Monitor Scope Document Version 1.5", Agilent Technologies, April 9, 2003 by DeWang Li.

and a later enhancement description:

"Enhanced Cypress Intra-Wafer Retest Scope Document Version 1.0", Agilent Technologies, June 4, 2004 by Kerry Schoenecke

When a retest condition is detected, the die needs to be **retested using a DIFFERENT tester site** by moving the chuck up to one die in the x or y direction.

Code internals: Before instructing the KLA prober to "step next" via a BA (Bin to Category) command (telling the KLA what bins these just tested die are automatically initiates a move to the next set of die), the KLA should be instructed to move the chuck in the +/- 1 in the x or y direction (this depends on the probe card geometry) and retest the die in question with a different tester site.



For example - in the above scenario where sites 1 through 4 have just been tested and sites #1 and #3 (in gray) resulted in certain bins which tripped the limit - the prober should then be instructed to move to Touch-down <2,5> so that die <3,5> is then tested by tester site #2 and so that die <5,5> is tested by tester site #4.

The retest condition is controlled via the planfile [test_switches] entry:

RE_TEST <bin> <turn_on_consec> <cum_confirm_turn_off>

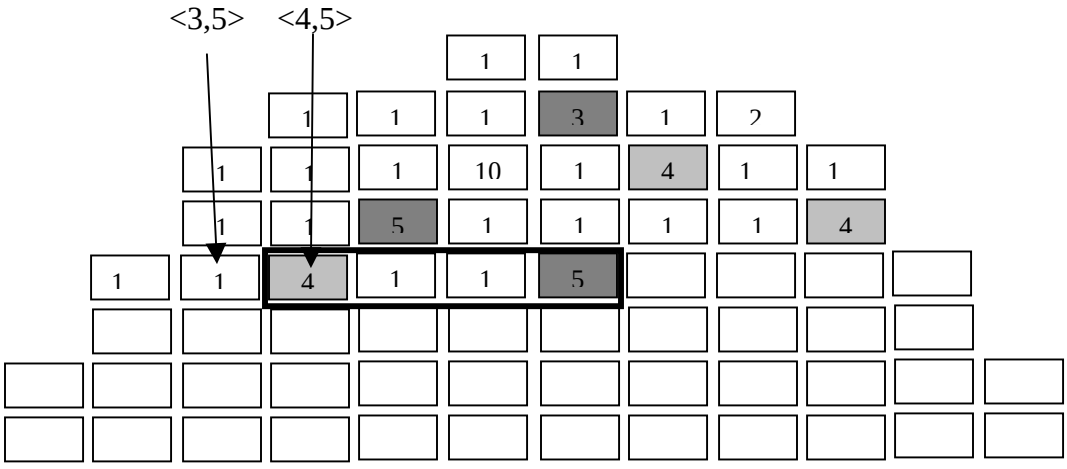
Where:

<bin>	Is the bin number that kicks off a retest
<turn_on_consec>	Is the number of times that this bin has appeared on the same site consecutively. Once this limit is reached, re-test keeps reoccurring.
<cum_confirm_turn_off>	The number of times retesting (with a different tester site) confirms the failing bin has really failed. When this (retest and it really was a failing bin) limit is reached, retesting is turned off for the this bin for the rest of the wafer.

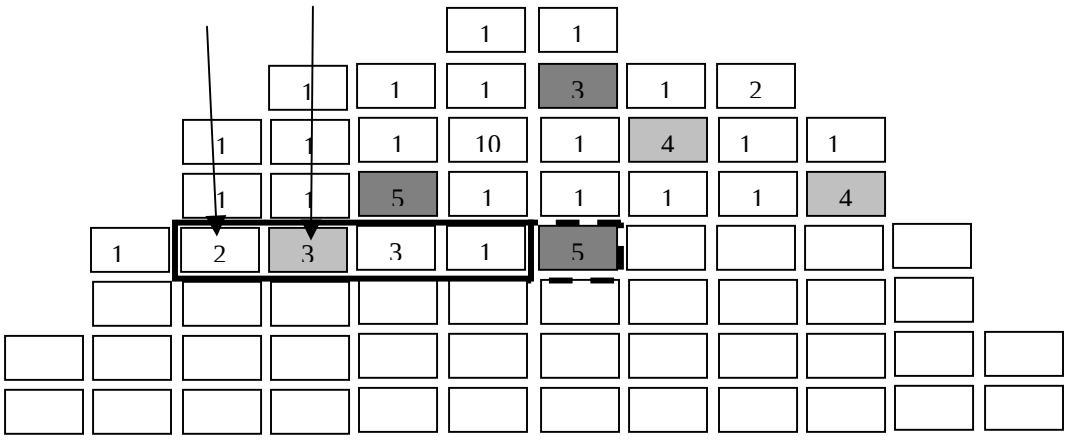
Processing Retest Information

Use the lowest bin number for retested die:

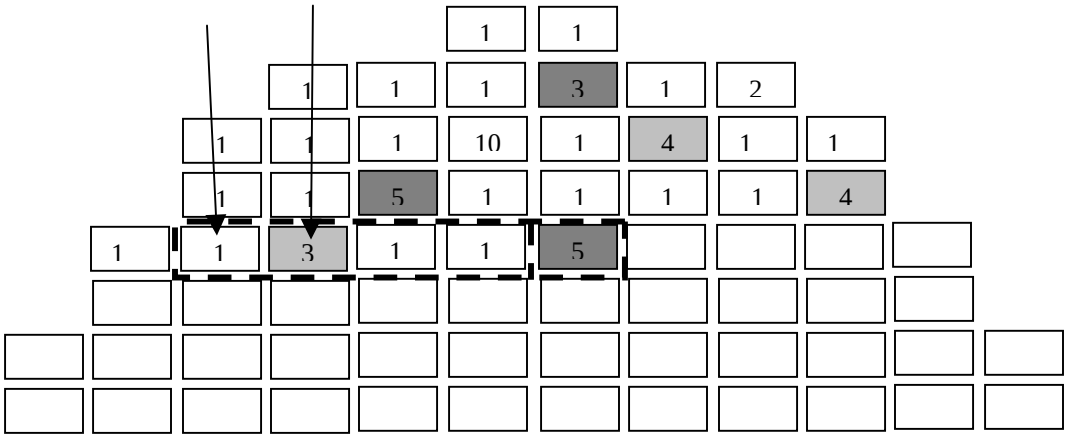
After moving to the new touch-down and retest, a new set of test results will be available. The results having the lower bin numbers will be used for a die. See diagram below:



At touch-down <4,5> needs to be rested using a different tester site. The prober is instructed to move to new touch-down <3,5>. Remember, die <2,5> was tested previously and is bin 1. Upon re touch-down at <2,5> the results are:



The resulting wafer map should be this after taking the lowest numbered bincodes:



Not over turnable die:

The plan file [test_switches] entry:

NOT_OVER_TURNABLE_BINS 5,6,7

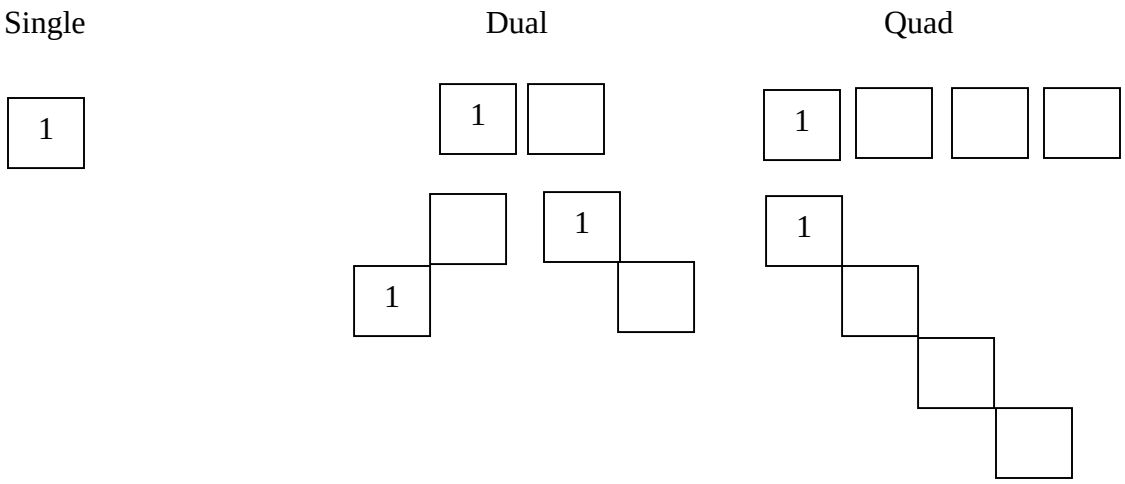
means that anytime these bincodes are detected on a die they stay and are not replaced with any other bincode (even another from this same list, if it comes along later as the result of a retest)

Two Touch-downs Maximum

Due to test head shifting to get a different site over the die to be retested, many die can be subjected to multiple touch-downs. For this reason, a re-test "hole" on the prober's wafermap is acceptable. Two touchdowns on a die are the maximum allowed.

CTI Dual and Quad Probe Card Configurations

Code will be developed to shift the test head to retest the following SIX probe card configurations with another site:



The probe card in use will be inferred from the plan file entries:

xofs_1, yofs_1
xofs_2, yofs_2
xofs_3, yofs_3
xofs_4, yofs_4

Phase 2 Code Limitations

CMI-CTI Die Retest Differences

1. CTI will not use the plan file entry:
   WAFER_EDGE <ROW> <START COLUMN> < END COLUMN>
   which defines Valid Probe Area. CMI uses bigger probe cards which could extend further off the wafer, wasting test time and causing uneven probe tip wear. Since CTI probe cards are smaller, this is not an issue.

2. CTI will not use the plan file entry: RE_TEST_BIN_MIN <BIN VALUE>, which (overrides other settings and) sets the minimum bin value that will be retested

3. CTI will not use plan file entries: ART_PRIMARY_MOVE_X
   ART_PRIMARY_MOVE_Y
   ART_SECONDARY_MOVE_X
   ART_SECONDARY_MOVE_Y

   These were in the June 2004 Agilent proposal are used to describe how the test head will move to locate a different site over the die in question. They include skip row, column information for those types of probe card configuration.

   CTI will hardcode test head movement logic (to put a different site over a die to be retested).

4. The tester produces 6 maps (or bin types) simultaneously:
   - data
   - alt1, alt2, alt3, alt4 (holding intermediate test results)
   - reject

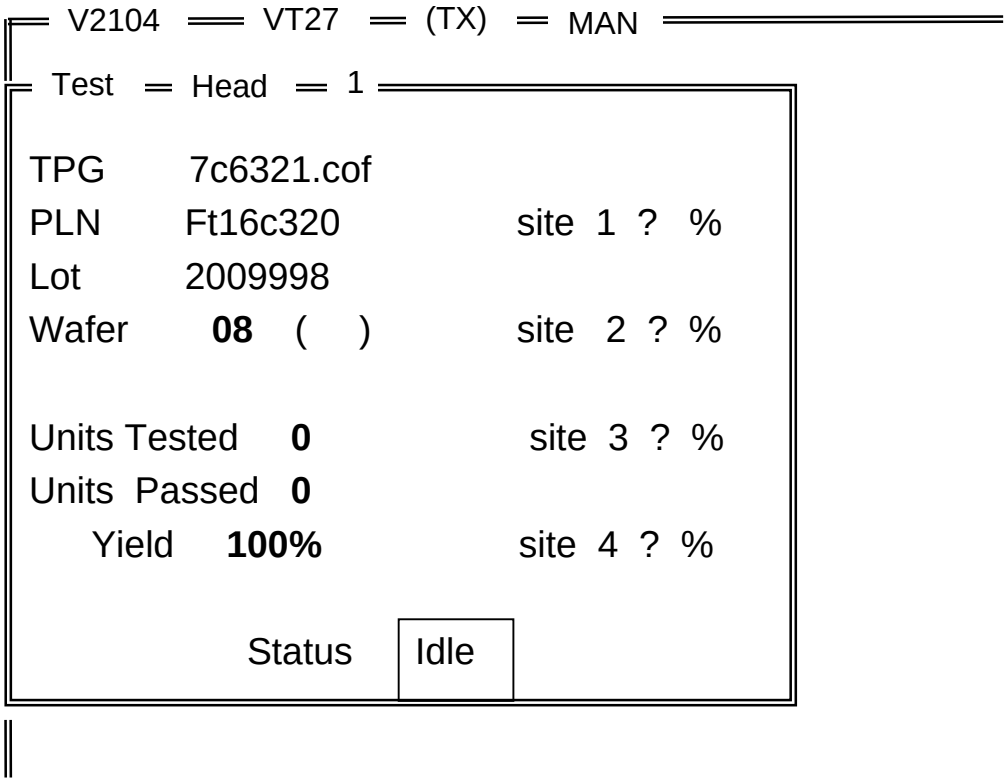
   The CTI will only operate on "data" bins.

   The CMI code operates on the other bintypes as well. It assumes that a bin number (e.g. bincode 90) will be unique to a bin type.

5. The CMI die retest code puts diagnostic information into test counters. The CTI code will not do this at this time.
   - The CTI code creates per-wafer logfiles on the c:\ drive. The (csv formatted) information in these files can be pulled into Excel for analysis of die retest effectiveness.
   - These files will not automatically be transferred to a server as part of this phase.

Other Limitations

1. EG probers will not be supported
2. Development and debug will only be on a limited number (one to three) VTs.
3. The "Test Head Task" display is not updated
   - This task loads information on bootup, or when a new lot is setup.
   - We cannot update this portion of the display as wafers run (since rdpio.exe uses Desqview commands to write directly to it).
     - We could not get Desqview windowing commands to work.
     - We cannot update **Wafer**, Units Tested, Units Passed, Yield
     - We have put this data on the Wafermap display screen



Qualification Runs

Base Code Qualification

To confirm that the Phase 1 code works:

> Correlation data (maps generated with existing vs new code) WITHOUT RETEST turned on, will be run on VT-27 for **5 wafers from each of 5 lots**.

Die Retest Qualification

PCR3 retest data will be gathered from at least **200 wafers** run on different probe card configurations.

Proposed devices are as follows:

	single	**dual1**	**dual2**	**dual3**	**quad1**	**quad2**
7C6370AT						
7C6602AT						
8C25001DT						
8C29000AT						
7C0241CT	sort1					
7C132GT	sort1					
Via Link (7C38*)						

Note: Closer to runtime we will decide which devices get run on which VT-KLA probe card configurations under which sort programs.

> The purpose of this section at this time is to describe enough of what has to be done so that we can estimate and schedule the project.

Efforts and Timespans

Code Development and Debug

	Item	Full time effort (8hr days)	Notes
	Phase 1 Flush-Out, Base Code Qualification - run 5 consecutive wafers from 5 lots. - misc screen and logfile updates - correlate old-new maps for wafers run	3 days	The actual running of these qual wafers is concurrent with new code development
	Trigger retesting - define two new [test_switches] parms, fish out of the plan file - logic to track consecutive failures, successes to turn-on/off die retest *per bin*	 2 days 2 days	
	Mechanically shift test head Decide on which prober commands to use, create testing code. Update production code Figure out which direction for retest for four different prober configurations (considering the wafer edge and two touchdown limits) & code it up Figure out where to shift back to (to pickup and run onward) when done retesting & code it up.	 3-4 days 3 days 2 days	
	Process retested die - use lower bin numbers, no overturns.	2 days	
	Subtotal:	**18 days**	

Timespan: Checkpoint: ready to debug and collect retest data:

18 days / 0.75 availability = 24 days timespan

Debug, Die Retest Qualification

	Item	Full time effort (8hr days)	Notes
	Debug @ 25% of 18 day effort	5 days	hopefully, running the qual wafers will surface bugs, usability items, screen changes, etc.
	Operator training on new software (for 24x5 day running)	3 days	The fewer the number of operators to train, the less this will be
	Subtotal	**8 days**	
	Run qualification wafers, examine data, tweak week1: 33 wafers on 1 tester week2: 67 wafers over 2 testers week3: 100 wafers over 3 testers ------ 200 wafer qualification		figure a ~15 day timespan for this item
	Post Qual Wafer Run Items Create spec for new VT software Write-up PCR3, present it, follow-up actions	 3 days 3 days	

Timespan: Run Qual Wafers

8 days effort / 0.75 avail = 11 days timespan within the 15 days of running 200 wafers

= 15 day timespan

Post Qual Wafer Run Items

6 days effort / 0.75 avail = 8 days timespan

Schedule

Key Checkpoints / Milestones

Start: 10/24/05

Ready for Die Retest Qualification Runs (+24 days): 12/01/05

Run Qual Wafers, Debug , Train (+15 days): 12/23/05

Spec, PCR3 completed, ready for turn-on (+8 days): **01/12/06**

Timeline / Gantt Chart